Volume 43 Number 1 March 2019

ISSN 0350-5596

# Informatica

## An International Journal of Computing and Informatics

Special Issue:

**Middle-European Conference on Applied Theoretical Computer Science (MATCOS)**

Guest Editors:

**Andrej Brodnik**
**Gábor Galambos**

1977

# Editorial Boards

Informatica is a journal primarily covering intelligent systems in the European computer science, informatics and cognitive community; scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor from the Editorial Board can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the list of referees. Each paper bears the name of the editor who appointed the referees. Each editor can propose new members for the Editorial Board or referees. Editors and referees inactive for a longer period can be automatically replaced. Changes in the Editorial Board are confirmed by the Executive Editors.

The coordination necessary is made through the Executive Editors who examine the reviews, sort the accepted articles and maintain appropriate international distribution. The Executive Board is appointed by the Society Informatika. Informatica is partially supported by the Slovenian Ministry of Higher Education, Science and Technology.

Each author is guaranteed to receive the reviews of his article. When accepted, publication in Informatica is guaranteed in less than one year after the Executive Editors receive the corrected version of the article.

# Editors' Introduction to the Special Issue on "MATCOS conference"

Dear Reader,

about ten years ago in the frame of collaboration between the universities in Szeged and in Ljubljana we decided to organize a conference which set an aim to bring the theory and the practice closer to each other. The first conference – "First Mini Conference on Theoretical Computer Science" – was organized in 2008. Feedback was showing that the conference was well accepted and we decided to organize the new conference regularly. In the meantime, the conference's popularity grew and researchers from more and more countries took part in it. Therefore, we decided to "rename" the conference to MATCOS (*__M__iddle-European Conference on __A__pplied __T__heoretical __CO__mputer __S__cience*). Besides the harmony of the theory and the practice we also wanted to highlight the cooperation and dialogue between senior and junior researchers. Consequently, we organized separate sessions for PhD students. It became customary, that the first day student papers and the invited talk were presented, and they were followed by the presentation of regular papers. At the same time the conference also moved to the Coast, to the city of Koper and the University of Primorska.

Among these conferences the last one was organized in 2016, and from the presented papers the programme committee decided to invite nine papers for a special issue of Informatica. All papers were rigorously reviewed. The papers reflect the general spirit of the MATCOS as they sweep from the theoretical aspects to the study of practical problems.

The programme committee of MATCOS 2016 was:

Jacek Blazevicz (Poznan, Poland)
Andrej Brodnik (Koper, Ljubljana, Slovenia) co-chair
Angel Corberan (Valencia, Spain)
Ruben Dorado Vicente (Jaén, Spain)
Elena Fernandez (Barcelona, Spain)
Gábor Galambos (Szeged, Hungary) co-chair
Gabriel Istrate (Timisoara, Romania)
Miklós Krész (Szeged, Hungary)
Silvano Martello (Bologna, Italy)
Benedek Nagy (Famagusta, Cyprus, Turkey)
Gerhard Reinelt (Heidelberg, Germany)
Giovanni Rinaldi (Rome, Italy)
Borut Žalik (Maribor, Slovenia)

Koper, Ljubljana, Szeged, October 2018

Andrej Brodnik
Gábor Galambos
editors

# Implementation and Evaluation of Algorithms with ALGator

Tomaž Dobravec
Faculty of Computer and Information Science
University of Ljubljana, Slovenia
E-mail: tomaz.dobravec@fri.uni-lj.si

**Keywords:** automatic algorithm testing, quality evaluation, empirical analysis

**Received:** October 30, 2018

*In this paper we present an automatic algorithm evaluation system called* ALGATOR*, which was developed to facilitate the algorithm design and evaluation process. The system enables unbiased tests of the correctness of the algorithm's results on given test cases and comparisons of the quality of implemented algorithms for solving various kinds of problems (e.g. sorting data, matrix multiplication, traveler salesman problem, shortest path problem, and the like). Within the* ALGATOR *one can define a problem by specifying the problem descriptors, test sets with corresponding test cases, input parameters and output indicators, algorithm specifications and criteria for measuring the quality of algorithms. When a user of the system submits an algorithm for solving a given problem,* ALGATOR *automatically executes this algorithm on predefined tests, measures the quality indicators and prepares the results to be compared with the results of other algorithms in the system. The* ALGATOR *is meant to be used by algorithm developers to perform independent quality tests for their solutions.*

*Povzetek: V tem delu je predstavljen sistem* ALGATOR*, ki omogoča izvajanje implementiranih algoritmov na vnaprej pripravljenih testnih primerih ter ocenjevanje pravilnosti in kvalitete izvajanja. Da bi omogočili uporabo pri reševanju čim večjega nabora problemov, smo sistem zasnovali tako, da so njegovi gradniki (problem, algoritem, testna množica, testni primer) definirani na abstrakten način. Pred uporabo* ALGATOR*ja za reševanje nalog konkretnega problema, administrator projekta konkretizira abstraktne entitete, kasnejši uporabniki (razvijalci algoritmov) pa konkretizirajo le svoje izdelke (algoritme). Izvedba testiranja poteka povsem samodejno, implementirani algoritmi se poženejo na vseh testnih množicah, rezultati testiranja (časovni in drugi indikatorji) pa se zapišejo v bazo in se kasneje lahko uporabijo za prikaz v obliki grafikonov in tabel ali za analizo in primerjavo z rezultati drugih algoritmov.*

## 1 Introduction

As written by Aho et al. in [1], *"Efforts must be made to ensure that promising algorithms discovered by the theory community are implemented, tested and refined to the point where they can be usefully applied in practice. [...] to increase the impact of theory on key application areas"*. It is very important to develop good algorithms and theoretically prove their efficiency. On the other hand, if these algorithms do not perform well in practice, all the theoretical work is not much more than "art for art's sake".

From this point of view, algorithm evaluation is a very important part of an algorithm design process. To facilitate the implementation, execution and evaluation of the algorithms and make this part of the algorithm design process as simple as possible, the ALGATOR system was developed. It can be used to execute an algorithm implementation on the given predefined sets of test cases and to analyze various indicators of the execution. Within every project of the system user can define the problem to be solved, sets of test cases, parameters of the input and indicators of the output data and the criteria for the algorithm quality evaluation. When a project is defined, any number of algorithm implementations (programs) can be added. When requested, system executes all the implemented algorithms, checks the correctness (on a given set of tests) and compares the quality of the results. Using the ALGATOR user can add additional quality criteria, draw graphs and perform evaluations and comparisons of defined algorithms.

### 1.1 A typical usage of the system

ALGATOR is a complex system in which users can perform various kinds of tasks - from technically demanding (e.g., defining the properties of a project) to rather straightforward and simple tasks (e.g., using charts presented on a web page to compare the quality of two algorithms). To keep the hard job for the experts and to simplify the usage for the others, ALGATOR uses different user roles and privileges, namely, a system administrator, a project administrator, a researcher and a guest. There are many different possible ways of using the ALGATOR system but the main common and the most perspective is the one presented in the following. A system administrator prepares the system by providing the hardware, installing ALGATOR software packages, and publishing the internet address of the installed system. A project administrator adds a new project and defines all the project's properties. When the project is completely defined and declared as public, ALGATOR automatically generates an internet subpage with the project presentation and usage guide sections. The project administrator adds some state of the art algorithms for solving the problem of the project, which will be used as a reference for the evaluation process (i.e. the results of the algorithms added by researchers will be compared with the results of these referential algorithms). According to the rules, presented at the project's website, a researcher adds a new algorithm. ALGATOR will automatically run the new algorithm on predefined tests. The researcher then checks the correctness and compares the results of his algorithm with the results of the other algorithms defined in the project. The researcher can also decide to make the algorithm public (by default, the algorithms are private and can only be seen by the author). A guest of the system lists the results and prints the graphs and other data produced by ALGATOR. A guest can also perform some actions (like customization of the presentation) that do not alter the project configuration. At any time a guest can register to the system and contribute as a project administrator or as a researcher.

## 1.2 Related work

ALGATOR was created to fill the gap in the area of algorithm evaluation process since, as far as we know, there is no similar tool available that would offer all of ALGATOR's capabilities. Even though there are some tools available on the web that allow analysis of program execution, they have been developed with different goals, therefore their support in the area of algorithm analysis is not possible or it is too cumbersome. On the web, we can find the following tools that partly cover the ALGATOR's functionality.

**Code Profilers** (JProfiler [12], JMeter (`jmeter.apache.org`), Netbeans Profiler (`profiler.netbeans.org`) and the like) can follow and record a program execution indicators and they are able to display various information about the usage of computer resources (like time and memory). Using the recorded data, a user can detect possible code errors and memory leaks and perform various code optimization. Although code profilers offer a number of different tests and measurements, their use is not appropriate for accurate analysis of algorithms, as they do not allow to tailor the sets of test cases and to perform analysis based on user-defined measurements and output indicators.

**Algorithms presentation pages** (like [5]) systematically present various computer problems and practical examples, grouped into categories (e.g. searching and sorting problems, mathematical problems, graph problems, string problems, etc.). Besides the presentation of a wider field for each domain, the pages usually present at least one solution (i.e. algorithm) for each problem and offer the possibility to upload user-defined implementations. The emphasis of this kind of pages is mainly on educational asspects and on presentation of problem descriptions therefore they do not allow the independent analysis of the algorithms using the user criteria nor do they offer the tools to display custom-defined graphs or other statistical information.

**Problem-domain dedicated pages** present a specific problem domain (like [2] for mixed integer programming problem and [3] for the traveler selsman problem) and they offer tools to compare different solutions. Although these pages are not configurable (for example, user can use only pre-defined measurements and result indicators), they are ideal for testing a specific solutions. The main drawback comparing to ALGATOR is that each page covers only one problem domain while ALGATOR aims to cover as many areas and problems domains as possible and to present the results in a uniform way for all of them.

**Educational coding platforms** (like Sphere online judge (`www.spoj.com`), CodeChef (`www.codechef.com`), CodeForces (`codeforces.com`), CodeFights (`codefights.com` ) and the like) introduce different problems with exactly defined input and expected output. After a user of such a platform uploads its own solution, the system performs its execution, evaluation and ranking according to regularity and efficiency. The essential difference between ALGATOR and these environments is that they offer only several predefined basic algorithm quality criteria (like, does an algorithm return a correct solution or not, or, how many of the given tests were solved correctly, and the like). In addition, the purpose of these platforms is focused on the process of learning programming skills and not in finding an optimal solutions for specific problems, which results in insufficient result presentation (e.g., it is not possible to compare different solutions using charts, plots or tables). For the purpose, for which these pages exist (i.e. to help their users to learn programming), they serve very good and they can be of a great help, nevertheless for the purpose of comparing different algorithms to solve the same problem they lack some crucial functionalities (like fully configurable input, output and result-presentation layer of the project) which are supported by ALGATOR.

# 2 Project definition

The main task of the ALGATOR's project administrator is to provide the configuration files and to implement corresponding Java or C++ interfaces. Besides the definition of the output format (where the sequence of the parameters and indicators in output file is described), the test cases, the test sets and the algorithm structure has to be defined precisely.

### The test cases and the test sets

A test case in ALGATOR execution environment is defined by a subclass of the `TestCase` class, which contains data structures to hold the input and output (result) data. Since these data structures are project-specific (i.e. each problem needs data of its own type) the project administrator has to implement the `[Project]Input` and the `[Project]Output` classes and prepare the corresponding data structures. For example, in the data-sorting problem, the `SortInput` class could be defined as presented in Listings 1.

Listings 1.
An input of the sorting problem

```
public class SortInput extends AbstractInput{
   // An array of data to be sorted
  public int [] arrayToSort;
}
```

A test set contains one or more test cases and it is a minimal execution unit. A test set is defined by a single text file in which every line defines one test case. The format of these lines is project-specific and it is defined by a project administrator. If required, additional files can be used to specify the test cases. Again, the syntax and the semantics of the content of these files is defined by the project administrator. Listings 2 presents an example of the text file defining five test cases for the data-sorting problem.

Listings 2.
Examples of test cases for the sorting problem

```
test1:10:INLINE:3 5 1 8 6 3 8 9 0 6
test2:10000:RND
test3:20000:SORTED
test4:30000:INVERSE
test5:50000:FILE:numbers.txt:16534
```

To iterate through the text file associated with a given test set, ALGATOR uses the methods of the `DefaultTestSetIterator` class. For each line read from the text file the `getTestCase()` method of the `[Project]TestCase` class is called. This method parses the input line and creates a set of parameters that describe the test case. Using these parameters it calls the `generateTestCase()` method which creates the instance of the test case. Since the representation of test cases is project-specific, the project administrator has to provide the correct implementation of the `getTestCase()` and the `generateTestCase()` methods. All the other methods are general and they can be used without modification. A part of an implementation of the `generateTestCase()` method is presented in Listings 3.

Listings 3.
A part of the `generateTestCase()` method for the sorting problem

```
@Override
public SortingTestCase generateTestCase
                 (Variables inputParameters) {
  int probSize  = inputParameters.
          getVariable("N").getIntValue();
  String group = inputParameters.
          getVariable("Group").getStringValue();

  // prepare an array of integers  ...
  int [] array = new int[probSize];
  // ... and fill table according to group
  switch (group) {
    case "RND":
      Random rnd = new Random();
      for (int i = 0; i < probSize; i++)
        array[i] = Math.abs(rnd.nextInt());
      break;
    case "SORTED":
      for (int i = 0; i < probSize; i++)
        array[i] = i;
      break;
    // ...
  }

  // create a test case ...
  SortingTestCase sortingTestCase =
    new SortingTestCase();
  sortingTestCase.setInput(
    new SortingInput(array));
  int [] expectedResultArray =
    getSortedArray(array);
  sortingTestCase.setExpectedOutput(
    new SortingOutput(expectedResultArray));
  // ... and return
  return sortingTestCase;
}
```

### Algorithms

The “heart” of each project are the implemented algorithms. Each algorithm is represented by a subclass of the `AbsAlgorithm` class with the following methods:

`ErrorStatus init(TestCase test)`. This method takes care of the input of the algorithm; it reads the test case and prepares the data. To enable fast algorithm execution all expensive initial tasks have to be done in this method. When this method is done all the required algorithm's input data has to be prepared in a proper format.

`void run()`. In this method the `execute()` method is called. The parameters of the `execute()` method are project-specific and are provided by the project administrator. ALGATOR takes the time of the execution of the `run()` method as an algorithm execution time therefore nothing else than the `execute()` method call should be placed in the `run()` method (see Listings 4).

`ParameterSet done()` . This method collects all the parameters and indicators of the execution and prepares them in the form suitable to be written into the output file.

The `AbsAlgorithm` class is abstract and the project administrator has to provide the `[Project]AbsAlgorithm` subclass with the above

Listings 4.
A simple implementation of the `run()` method

```
public void run() {
  result = execute(sortTestCase.getInput());
}
```

mentioned methods implemented. Besides he has to declare fields for input data (in these fields the input data obtained from the test case will be stored during the execution of the `init()` method) and the `abstract execute()` method with appropriate number and type of parameters. The task of the researcher is to implement a subclass of `[Project]AbsAlgorithm` and implement the `execute()` method. In other words, all the “dirty job” of preparing data and collecting the results is done by the project administrator. The researcher who wants to provide an algorithm only has to implement one method which returns a proper result. In the case of data-sorting problem, an algorithm only needs to sort the array of data; a very simple (but technically correct) algorithm for sorting data is listed in Listings 5.

Listings 5.
A simple implementation of the algorithm

```
public class JavaSortAlgorithm
            extends SortAbsAlgorithm {

  SortOutput execute(SortInput input) {
    SortOutput result = new SortOutput();
    java.util.Arrays.sort(input.arrayToSort);
    result.sortedArray = input.arrayToSort;
    return result;
  }
}
```

# 3 Indicators of the algorithm

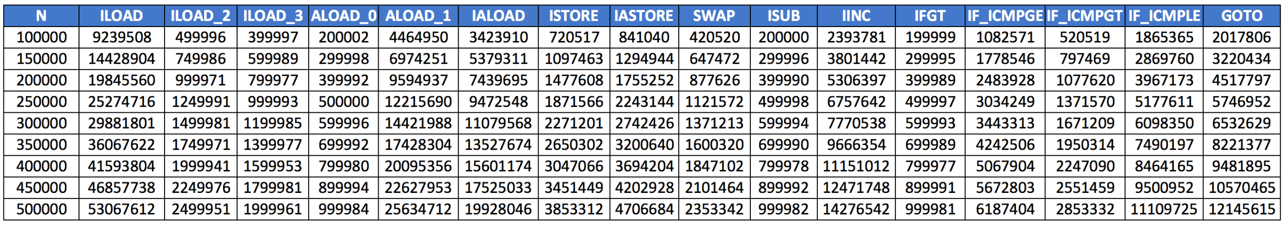
Since ALGATOR was designed to be used for various kinds of problems, the criteria for measuring the quality of algorithms are not defined as a part of the system but they have to be defined by the project administrator. The current version of the system enables measurements of three different kinds of indicators: a) the indicators to measure the speed and the quality of the algorithm (the so called EM indicators), b) the project-specific counters to count the usage of the parts of the algorithm's program code (the so called CNT indicators), and c) the counters of the Java byte code usage (the so called JVM indicators). These indicators are calculated with independent measurements that are performed as separated tasks so they do not interfere with one another. For example: when ALGATOR measures time, the CNT and JVM indicators are disabled. To perform the JVM measurements a dedicated Java virtual machine is used.

**The EM measurements.**
These measurements are used to measure the time and other project-specific indicators. All measurements of the time are performed automatically. To provide as accurate time indicators as possible ALGATOR tries to reduce the influence of the uncontrolled computer activities (e.g. sudden increase of a system resource usage) by running each algorithm several times. The system measures the first, the best, the worst and the average time of the execution. The project administrator only needs to specify the phases of algorithm execution (e.g. the pre-processing phase, the main phase, the post-processing phase, ...) and to select which of the time indicators are to be presented as the result of execution.

The project-specific indicators are defined by the project administrator. They can be presented as a string or as a number. For example, for exact algorithms, the value of an indicator could be "OK" (is the algorithm produced the correct result) or "NOK" (if the result of the algorithm is not correct). For approximation algorithms the value of an indicator could be the quality of the result (i.e. the quotient of the correct result and the result of the algorithm).

ALGATOR produces the values of the EM indicators for each `(algorithm, test_case)` pair by performing the following steps: a) load the test case and create its project-specific representation, b) load the algorithm (by using the Java reflection), c) read the values of the test case specific parameters, d) run the algorithm and measure its time consumption, e) read the values of the time indicators, f) determine the values of the project-specific indicators, g) writes all the parameters and the indicators of the execution to the output file. Since the time indicators are natively plugged into the ALGATOR system, the step f) of the above procedure is the only step that has to be configured by the project administrator (everything else is done automatically by ALGATOR). To configure a project-specific indicator the administrator has to provide its description in a configuration file (defining indicators type and possible values) and a program code to determine its value using the test case parameters and the algorithm's result. For example, in the Sorting problem the administrator provides a correctness indicator by a code as presented in Listings 6. In this code (which is a part of a `done()` method that is invoked just after the execution of the algorithm) the `isArraySorted()` method returns `true` if the input array is sorted. The name (`"Correctness"`) and the type (`String`) of the indicator has to be defined in a configuration file.

**The CNT measurements.**
The CNT measurements are used to count the usage of the parts of the program code. This option is used to analyze the usage of a certain system resource or to count the usage of the selected type of commands on the programming language level. Using this one can, for example, measure how many times the memory allocation functions were executed during the algorithm execution and the amount of



| N | ILOAD | ILOAD_2 | ILOAD_3 | ALOAD_0 | ALOAD_1 | IALOAD | ISTORE | IASTORE | SWAP | ISUB | IINC | IFGT | IF_ICMPGE | IF_ICMPGT | IF_ICMPLE | GOTO |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 100000 | 9239508 | 499996 | 399997 | 200002 | 4464950 | 3423910 | 720517 | 841040 | 420520 | 200000 | 2393781 | 199999 | 1082571 | 520519 | 1865365 | 2017806 |
| 150000 | 14428904 | 749986 | 599989 | 299998 | 6974251 | 5379311 | 1097463 | 1294944 | 647472 | 299996 | 3801442 | 299995 | 1778546 | 797469 | 2869760 | 3220434 |
| 200000 | 19845560 | 999971 | 799977 | 399992 | 9594937 | 7439695 | 1477608 | 1755252 | 877626 | 399990 | 5306397 | 399989 | 2483928 | 1077620 | 3967173 | 4517797 |
| 250000 | 25274716 | 1249991 | 999993 | 500000 | 12215690 | 9472548 | 1871566 | 2243144 | 1121572 | 499998 | 6757642 | 499997 | 3034249 | 1371570 | 5177611 | 5746952 |
| 300000 | 29881801 | 1499981 | 1199985 | 599996 | 14421988 | 11079568 | 2271201 | 2742426 | 1371213 | 599994 | 7770538 | 599993 | 3443313 | 1671209 | 6098350 | 6532629 |
| 350000 | 36067622 | 1749971 | 1399977 | 699992 | 17428304 | 13527674 | 2650302 | 3200640 | 1600320 | 699990 | 9666354 | 699989 | 4242506 | 1950314 | 7490197 | 8221377 |
| 400000 | 41593804 | 1999941 | 1599953 | 799980 | 20095356 | 15601174 | 3047066 | 3694204 | 1847102 | 799978 | 11151012 | 799977 | 5067904 | 2247090 | 8464165 | 9481895 |
| 450000 | 46857738 | 2249976 | 1799981 | 899994 | 22627953 | 17525033 | 3451449 | 4202928 | 2101464 | 899992 | 12471748 | 899991 | 5672803 | 2551459 | 9500952 | 10570465 |
| 500000 | 53067612 | 2499951 | 1999961 | 999984 | 25634712 | 19928046 | 3853312 | 4706684 | 2353342 | 999982 | 14276542 | 999981 | 6187404 | 2853332 | 11109725 | 12145615 |

Figure 1: The number of Java bytecode instructions used by Hoare's Quicksort algorithm while sorting integer arrays with $\mathbf{N = 100.000 \ldots 500.000}$ elements.

Listings 6.
Checking for the correctness of the algorithm

```
boolean checkOK = isArraySorted(testArray);
EIndicator checkPar = new EIndicator(
  "Correctness",  checkOK ? "OK" : "NOK");
indicators.addIndicator(checkPar);
```

the memory allocated by these calls. One can also use CNT measurements to detect which part of the algorithm is most frequently used. For example, if the problem in concern would be data-sorting, using the CNT measurements one could count the number of comparisons, the number of swaps of elements and the number of recursive function calls (which are the measures that can predict the algorithm execution behavior [11]). To facilitate the CNT measurement in the project, the project administrator has to define the names and the meaning of the counters and the researchers have to tag the appropriate places in their code. Everything else is done automatically by ALGATOR.
For example, if we want to count the number of swaps and the number of comparisons performed by a sorting algorithm, we would define two counters (namely, the `"CMP"` and the `"SWAP"` counter) in a configuration of a sorting problem. Additionally, to ensure a correct value of these counters, the source code of algorithms should be tagged, so that every code line in which a comparison of two elements appears would be accompanied by at tag line `//@count{CMP,1}`, and a code line in which a swap of two elements is invoked by `//@count{SWAP,1}`. An example of tagged code for the BubbleSort algorithm is presented in Listings 7.

When ALGATOR is asked to provide the counter values, it replaces all the tags with a Java code (e.g., it replaces a tag `//@count{CMP,1}` with a Java code `Counters.add("CMP", 1);`), recompiles the source and runs the algorithm. When the algorithm stops, ALGATOR collects the values of the counters and writes them to an appropriate output file.

**The JVM measurements.**
An algorithm written in the Java programming language compiles into Java byte code. An interesting option offered by ALGATOR is the ability to count how many times each byte code instruction was used during the execution of an algorithm on a given test case. To facilitate this option ALGATOR uses a dedicated Java virtual machine `VMEP` which was developed as a part of ALGATOR project [9]. This virtual machine [10] extends an open source virtual machine JamVM [8] and supports counting of the usage of each bytecode instruction used during the algorithm execution. When ALGATOR is asked to provide JVM statistics, it executes the algorithm in the VMEP and stores the bytecode–usage counters that it returns. In [7] Lambert and Power indicated that the frequency of the usage of each byte code instruction can be used to predict the execution time. Even though ALGATOR's ability to count the byte code instructions usage is quite young, we expect that the data produced by the JVM measurements could be useful not only for the quantitative but also the substantive analysis of the algorithms.

Listings 7.
A tagged source code of the BubbleSort algorithm used to count the number of comparisons and swaps.

```
public void execute(int[] data) {
  for (int i=0; i<data.length; i++)
    for (int j=0; j<data.length-1; j++) {
      //@COUNT{CMP, 1}
      if (data[j] > data[j+1]) {
        //@COUNT{SWAP, 1}
        swap(data, i, j);
      }
    }
  }
}
```

As an usage example of the VMEP's indicators let us consider the Hoare's implementation of a sorting algorithm (i.e. a Quicksort algorithm in which the partitioning phase uses one pivot to split a given array into two subarrays [6]). It is known that this algorithm performs $\mathcal{O}(n \ \log n)$ steps on average to sort a given array of $n$ elements. Running this algorithm in ALGator leads to interesting conclusions as presented in the following. To sort an array of integers the Hoare's algorithm uses only 16 (out of 202 possible) Java bytecode instructions, namely, `ILOAD`, `ILOAD_2`, `ILOAD_3`, `ALOAD_0`, `ALOAD_1`, `IALOAD`, `ISTORE`, `IASTORE`, `SWAP`, `ISUB`, `IINC`, `IFGT`, `IF_ICMPGE`, `IF_ICMPGT`, `IF_ICMPLE`, and `GOTO` (see Figure 1).

Among these instructions the most common used are the ILOAD (32%), ALOAD_1 (15%), IALOAD (12%), IINC (8%) and GOTO (6%) instructions (an average usage of the other instructions is less than 5%). The overall number of all instructions used is $57,8 * n \log n$, with a relative error (for $n = 100.000 \ldots 1.000.000$) less than 3,5%. Which means, for example, that for n=100.000, the algorithm will perform approximately 29 millions of java bytecode instructions to sort an array. As a consequence, knowing only the size of the input array, one can predict the number of required instructions very accurately. On the other hand, this shallow analysis can not be used to predict the execution time of the algorithm due to a weak relation between the number of used instructions and the time consumption. Using the results obtained on an Intel(R) Core(TM) i7-6700 CPU computer running at 3.40GHz an average quotient between the number of instructions and the execution time (in microseconds) is 3949 (with a relative error 17.3% for $n = 100.000 \ldots 1.000.000$), which means that on average the Java virtual machine performs around 4000 instructions per microsecond (i.e. 0.25 nano second per instruction). For more accurate analysis of the relation between the number of instructions and the execution time, we would need to distinguish slow and fast instructions [7] with a special attention being paid to instructions that can fetch the data from a non-cached memory.

Listings 8.
An example of the ALGATOR's query

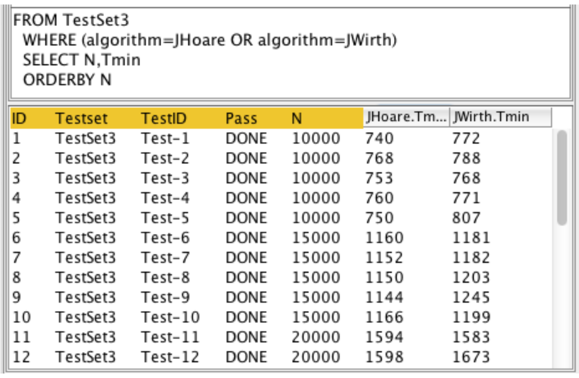
```
queryF1C1 = FROM TestSet0
  WHERE (algorithm=*) AND ComputerID=F1.C1
  SELECT Tmin AS A1;
queryF2C1 = FROM TestSet0
  WHERE (algorithm=*) AND ComputerID=F2.C1
  SELECT Tmin AS A2;
FROM queryF1C1, queryF2C1
  WHERE (algorithm=JHoare)
  SELECT N, A1/A2 AS Q
```

# 4 Analyzing the results

As a result of the algorithm execution ALGATOR produces text output files. For each tuple (algorithm, test set, measurement) one file is created; each line in this file contains parameters and indicators of one test case.

The data in the output line is separated by semicolons (CSV format). For efficient work with this data ALGATOR provides the analyzer with its own query language and with the visualization module for presenting data as graphs. For example, to get the minimal execution times for algorithms named JHoare and JWirth on the test set called TestSet3, a user can run query as depicted in Figure 2.

ALGATOR query language is a powerful tool that enables all sorts of data manipulation. An example of a complex query to calculate the quotient of minimal times for the JHoare algorithm running on two different computers (F1.C1 and F2.C1) is presented in Listings 8. Note that the ALGATOR system might contain several computers that are able to execute the algorithms (such a computer in the system is called an execution engine). Each execution engine has its name, which comprises of a name of a computer family and a unique name of the computer inside this family (e.g. in the name F1.C1 the F1 represents the family and C1 the computer name). One family contains computers with equal hardware configuration. To provide comparable results, the algorithms of a given problem are usually run on the same computer (or at least on computers of the same family). Nevertheless, a researcher might additionally run algorithms on other computers and compare the execution results as presented in the query in Listings 8. If the computers used in this test have different hardware configurations, the results of such a comparison might reveal the influence of the particular hardware to the algorithms' behaviour.



```
FROM TestSet3
  WHERE (algorithm=JHoare OR algorithm=JWirth)
  SELECT N,Tmin
  ORDERBY N
```

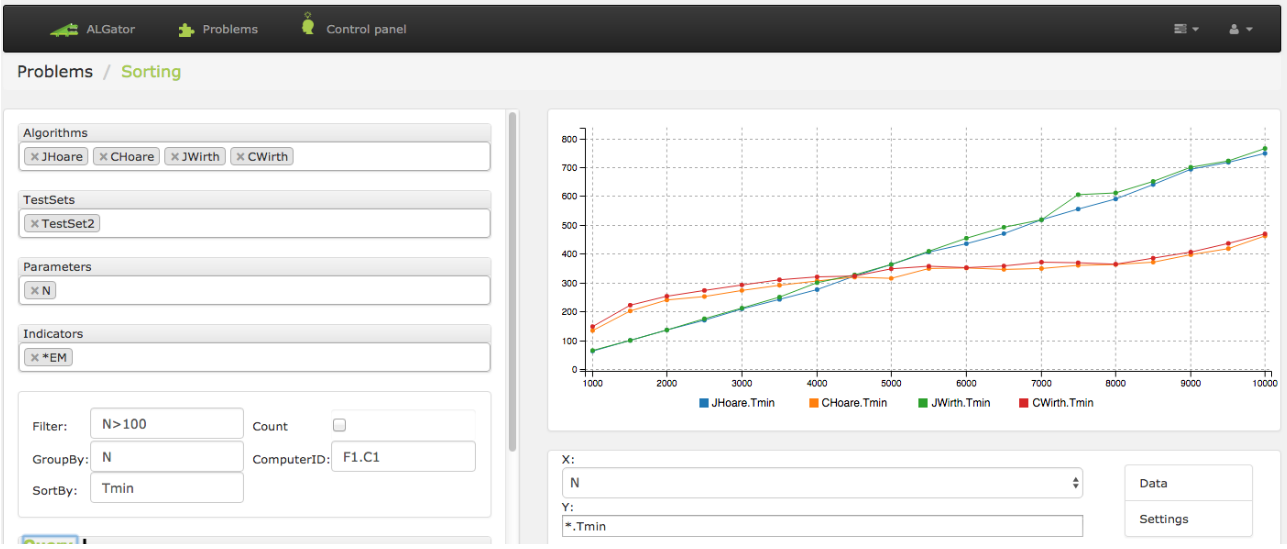
| ID | Testset | TestID | Pass | N | JHoare.Tm... | JWirth.Tmin |
|---|---|---|---|---|---|---|
| 1 | TestSet3 | Test-1 | DONE | 10000 | 740 | 772 |
| 2 | TestSet3 | Test-2 | DONE | 10000 | 768 | 788 |
| 3 | TestSet3 | Test-3 | DONE | 10000 | 753 | 768 |
| 4 | TestSet3 | Test-4 | DONE | 10000 | 760 | 771 |
| 5 | TestSet3 | Test-5 | DONE | 10000 | 750 | 807 |
| 6 | TestSet3 | Test-6 | DONE | 15000 | 1160 | 1181 |
| 7 | TestSet3 | Test-7 | DONE | 15000 | 1152 | 1182 |
| 8 | TestSet3 | Test-8 | DONE | 15000 | 1150 | 1203 |
| 9 | TestSet3 | Test-9 | DONE | 15000 | 1144 | 1245 |
| 10 | TestSet3 | Test-10 | DONE | 15000 | 1166 | 1199 |
| 11 | TestSet3 | Test-11 | DONE | 20000 | 1594 | 1583 |
| 12 | TestSet3 | Test-12 | DONE | 20000 | 1598 | 1673 |

Figure 2: An example of data query with result.

The results of the execution can be analysed in one of the ALGATOR's visualization modules (one is implemented as a web and the other as a standalone application). In these modules a user can design queries (to produce arrays of numerical results) and draw charts as depicted in Figure 3.

# 5 Conclusion

The execution part of ALGATOR [4] was developed in both Java and C++ programming languages, therefore the algorithms to be tested could be implemented in one of these two languages. Measuring the exact execution time of the algorithms written in Java is a challenging task since the system can only measure real time and because there is no way to eliminate the side effects of the Java virtual machine's background tasks (e.g. garbage collection). To overcome this problem, ALGATOR executes each algorithm several times and reports the first, the minimal, the maximal and the average time of execution. Comparing and analyzing these times one can detect the influence of



Algorithms: × JHoare × CHoare × JWirth × CWirth

TestSets: × TestSet2

Parameters: × N

Indicators: × *EM

Filter: N>100

Count □

GroupBy: N

ComputerID: F1.C1

SortBy: Tmin

X: N

Y: *.Tmin

Figure 3: The visualization module of ALGATOR.

the execution environment to the overall execution time. In many cases this influence is negligible. Having the Java implementation of the algorithm also has some benefits. Namely, ALGATOR counts and generates the statistics of the usage of the Java byte code instructions. As stated in [7] these statistics provide enough information to be used for the platform independent timing of the algorithms. Our preliminary tests indicate a high correlation between the number of used Java byte code instructions (multiplied by the corresponding weight depending on the type of instruction) and the execution time. The ability to implement the algorithms in both (Java and C++) programming languages, enables the researchers to compare the execution time of both and to estimate the impact of the programming language. ALGATOR is a testing environment, which aims to make the testing process as easy as possible for both, the project administrators and for the researchers. We tired to minimize the effort that has to be used to prepare the project and the algorithm and we think that this goal was achieved. The biggest challenge for the project administrator is to prepare adequate test cases and to write several lines of Java of C++ code (in an average case not more that about 100 lines of code), while the researcher has to write only a few lines of code to call the existing Java or C++ implementation of the algorithm. All the other tasks needed to execute the algorithm and to produce the desired indicators are performed automatically by ALGATOR, therefore the researchers can focus on the analyses of the results. Furthermore, ALGATOR uses the same test cases for all the algorithms of the project, therefore the researchers can not tailor the tests to be optimal for their implementations, which makes the results of the evaluation fair and reliable.

# References

[1] A. V. Aho, D. S. Johnson, R. M. Karp, S. R. Kosaraju, C. C. McGeoch, C. H. Papadimitriou, and P. Pevzner. Emerging opportunities for theoretical computer science. *SIGACT News*, 28(3):65–74, 1997.

[2] D. Applegate, W. Cook, S. Dash, and M. Mevenkamp. QSopt linear programming solver. www.math.uwaterloo.ca/ bico/qsopt, 2011.

[3] W. Cook. Concorde TSP solver. www.math.uwaterloo.ca/tsp/concorde, 2015.

[4] T. Dobravec. ALGator - an open source automatic algorithm evaluation system. https://github.com/ALGatorDevel/Algator, 2018.

[5] G. F. Geeks. A computer science portal for geeks. www.geeksforgeeks.org/fundamentals-of-algorithms, 2018.

[6] C. A. R. Hoare. Quicksort. *Comput. J.*, 5:10–15, 1962.

[7] J. M. Lambert and J. F. Power. Platform independent timing of Java virtual machine bytecode instructions. *Electronic Notes in Theoretical Computer Science*, 220:79–113, 2008.

[8] R. Lougher. JamVM - an open source Java virtual machine. jamvm.sourceforge.net, 2014.

[9] J. Nikolaj. Java virtual machine for counting the Java bytecode usage (original title: Predelava javanskega navideznega stroja za štetje ukazov zložne kode, language: Slovene). *University of Ljubljana, Faculty of Computer and Information Science*, 2014.

[10] J. Nikolaj. A source code of VMEP virtual machine. github.com/nikolai5slo/jamvm, 2014.

[11] R. Segedwick. The analysis of Quicksort programs. *Acta Informatica*, 7:327–355, 1977.

[12] E. Technologies. The definitive guide to JProfiler (ebook). resources.ej-technologies.com/jprofiler/help/doc/JProfiler.pdf, 2018.

# Packing Tree Degree Sequences

Kristóf Bérczi
Department of Operations Research and MTA-ELTE Egerváry Research Group
Eötvös Loránd University
Pázmány Péter sétány 1/c, 1117 Budapest, Hungary
E-mail: berkri@cs.elte.hu

Zoltán Király
Department of Computer Science and MTA-ELTE Egerváry Research Group
Eötvös Loránd University
Pázmány Péter sétány 1/c, 1117 Budapest, Hungary
E-mail: kiraly@cs.elte.hu

Changshuo Liu
Budapest Semesters in Mathematics
Bethlen Gábor tér 2, 1071 Budapest, Hungary
E-mail: cl20@princeton.edu

István Miklós
Rényi Institute
Reáltanoda u. 13-15, 1053 Budapest, Hungary
E-mail: miklos.istvan@renyi.mta.hu

**Keywords:** edge-disjoint realizations, packing spanning trees, degree sequences, uniform sampling

**Received:** October 30, 2018

*Special cases of the edge disjoint realizations of two tree degree sequences are considered in this paper. We show that if there is no node which have degree one in both degree sequences, then they always have edge-disjoint caterpillar realizations. By using a probabilistic method, we prove that two tree degree sequences always have edge-disjoint realizations if each vertex is a leaf in at least one of the trees. We also show that the edge-disjoint realization problem is in* P *for an arbitrary number of tree sequences with the property that each vertex is a non-leaf in at most one of the trees.*
*On the other hand, we show that the following problem is already* NP*-complete: given two graphical degree sequences $D_1$ and $D_2$ such that $D_2$ is a tree degree sequence, decide if there exist edge-disjoint realizations of $D_1$ and $D_2$ where the realization of $D_2$ does not need to be a tree.*
*Finally, we show that efficient approximations for the number of solutions as well as an almost uniform sampler exist for two tree degree sequences if each vertex is a leaf in at least one of the trees.*

*Povzetek: V članku so obravnavani posebni primeri povezavno-disjunktnih realizaciji dveh zaporedji stopenj dreves. Pokazano je, da, če ni vozlišča stopnje ena v obeh zaporedjih, potem imata zaporedji vedno povezavno- disjunktni gosenični realizaciji. Pokazan je tudi primer, ko je problem $NP$-poln.*

## 1 Introduction

Packing degree sequences is related to discrete tomography. The central problem of tomography is to reconstruct spatial objects from lower dimensional projections. The discrete 2D version is to reconstruct a colored grid from vertical and horizontal projections. In the simplest version, this problem is to reconstruct the coloring of an $n \times m$ grid with the requirement that each row and column has a specific number of entries for each color. Such colored matrix can be considered as a factorization of the complete bipartite graph $K_{n,m}$. Indeed, for each color $c_i$, the 0-1 matrix of size $n \times m$ obtained by replacing $c_i$ by 1 and all other colors by 0 is an adjacency matrix of a simple bipartite graph such that the disjoint union of these simple graphs is $K_{n,m}$. The prescribed number of entries for each color are the degrees of the simple bipartite graphs. Therefore, an equivalent problem is to give a factorization of the complete bipartite graph into subgraphs with prescribed degree sequences.

It is also possible to consider the non-bipartite version of the edge-disjoint realization problem above. Obviously, the sum of the degrees for each vertex must be $n-1$ when the complete graph $K_n$ is factorized. Therefore, if there are $k$ degree sequences, the last degree sequence is uniquely determined by the first $k-1$ degree sequences. When $k = 2$, the problem is reduced to the degree sequence problem, and can be solved in polynomial time [3, 4]. Unfortunately the problem becomes NP-complete already for $k = 3$ [1]. However, special cases are polynomially solvable. Such a special case is when one of the degree sequences is almost regular, that is, any two degrees differ by at most 1 [5].

In this paper we consider the case when $k = 3$, and two of the degree sequences are tree degree sequences. It was already known that this case is tractable [6]. Here we present a new result considering special, caterpillar realizations. Another alternative proof is given for a special subclass of pairs of tree degree sequences that can be extended to an arbitrary number of sequences. The size of the solution space and sampling from it is also discussed. As a negative result, we show that deciding the existence of edge-disjoint realizations for two degree sequences $D_1$ and $D_2$ is NP-complete even if $D_2$ is a tree degree sequence (but its realization does not have to be a tree).

## 2 Preliminaries

In this section we give the definitions and lemmas needed to state the theorems. The central subject of study in this paper is the edge-disjoint realization problem.

**Definition 1.** *A* ***degree sequence*** *$D = (d_1, \ldots, d_n)$ is a series of non-negative integers. A degree sequence is* ***graphical*** *if there is a vertex labeled simple graph $G = (V, E)$ with $V = \{v_1, \ldots, v_n\}$ in which the degree of vertex $v_i$ is exactly $d_i$ for $i = 1, \ldots, n$. Such graph $G$ is called a* ***realization*** *of $D$. The edge-disjoint realization problem is the following: given a $c \times n$ degree matrix $D = \{(d_{1,1}, \ldots, d_{1,n}), (d_{2,1}, \ldots, d_{2,n}), \ldots, (d_{c,1}, \ldots, d_{c,n})\}$, in which each row of the matrix is a degree sequence, decide if there is an ensemble of edge-disjoint realizations of the degree sequences. Such a set of edge-disjoint graphs is called a* ***realization*** *of the degree matrix. Given two degree sequences $D = (d_1, \ldots, d_n)$ and $F = (f_1, \ldots, f_n)$, their* ***sum*** *is defined as $D + F = (d_1 + f_1, \ldots, d_n + f_n)$.*

For sake of completeness, we define tree degree sequences, path sequences and caterpillars.

**Definition 2.** *Let $D = (d_1, \ldots, d_n)$ be a degree sequence. Then $D$ is called a* ***tree sequence*** *if $\sum_{i=1}^n d_i = 2n - 2$ and each degree is positive. If all of the degrees are 2 except two of them which are 1, then $D$ is called a* ***path sequence****. A tree is called a* ***caterpillar*** *if its non-leaf vertices span a path.*

We will use the following complexity classes later on.

**Definition 3.** *A decision problem is in* NP *if a non-deterministic Turing Machine can solve it in polynomial time. An equivalent definition is that a witness proving the "yes" answer to the question can be verified in polynomial time. A counting problem is in* #P *if it asks for the number of witnesses of a problem in* NP. *A counting problem in* #P *is in* FP *if there is a polynomial running time algorithm which gives the solution. It is* #P*-complete if any problem in* #P *can be reduced to it by a polynomial-time counting reduction.*

**Definition 4.** *A counting problem in* #P *is in* FPRAS *(Fully Polynomial Randomized Approximation Scheme) if there exists a randomized algorithm such that for any problem instance $x$ and $\epsilon, \delta > 0$, it generates an approximation $\hat{f}$ for the solution $f$, satisfying*

$$P\left(\frac{f}{1+\epsilon} \leq \hat{f} \leq (1+\epsilon) \cdot f\right) \geq 1 - \delta,$$

*and the algorithm has a time complexity bounded by a polynomial of $|x|$, $1/\epsilon$ and $-\log(\delta)$.*

The total variational distance $d_{TV}(p, \pi)$ between two discrete distributions P and $\pi$ over the set $X$ is defined as

$$d_{TV}(p, \pi) := \frac{1}{2} \sum_{x \in X} |p(x) - \pi(x)|$$

**Definition 5.** *A counting problem in* #P *is in* FPAUS *(Fully Polynomial Almost Uniform Sampler) if there exists a randomized algorithm such that for any instance $x$ and $\epsilon > 0$, it generates a random element of the solution space following a distribution* P *satisfying*

$$d_{TV}(p, U) \leq \epsilon,$$

*where $U$ is the uniform distribution over the solution space, and the algorithm has a time complexity bounded by a polynomial of $|x|$ and $-\log(\epsilon)$.*

The following technical lemma will be used later for constructing edge-disjoint caterpillar realizations.

**Lemma 1.** *For $n \geq 4$, there are two edge-disjoint Hamiltonian paths in the complete graph $K_n$ whose ends are pairwise different.*

*Proof.* Let $V = \{v_1, \ldots, v_n\}$, and let the first Hamiltonian path be $v_1, v_2, v_3 \ldots, v_n$. We are going to show by induction that there is a second Hamiltonian path $H$ starting at $v_2$, ending at $v_3$ and using no edge between consecutive integers. For $n = 4$, the path $H = v_2, v_4, v_1, v_3$ does the job. Suppose that $n > 4$ and that we have a path $H'$ on vertices $v_1, \ldots, v_{n-1}$ between $v_2$ and $v_3$. Since the path has at least three edges, there is an edge $v_i v_j$ for which $i, j < n-1$. Replace this edge by edges $v_i v_n$ and $v_n v_j$ for getting the desired path $H$. □

# 3 Packing trees

First we consider the problem of edge-disjoint realization of two tree degree sequences without common leaves.

**Theorem 2.** *Let $D = (d_1, \dots, d_n)$ and $F = (f_1, \dots, f_n)$ be two tree degree sequences such that $\min_i\{d_i + f_i\} \geq 3$. Then $D$ and $F$ have edge-disjoint caterpillar realizations.*

*Proof.* The proof goes by induction on $n$. Observe that the smallest possible $n$ is 4 to accommodate at least $4 = 2 \cdot 2$ leaves (note that each tree has at least two leaves). For $n = 4$, the only possible pair of degree sequences is $(2, 2, 1, 1)$ and $(1, 1, 2, 2)$. By Lemma 1, these sequences have edge-disjoint realizations.

If $n > 4$ and both $D$ and $F$ are path sequences, then there exists edge-disjoint Hamiltonian paths, according to Lemma 1.

So we may suppose that at least one of the degree sequences is not a path sequence. As the sum of the degrees in $D + F$ is $4n - 4$, there are at least four indices where $d_j + f_j = 3$. It is not difficult to see that we can select indices $i$ and $j$ such that, possibly after interchanging the roles of $D$ and $F$, we have $d_i \geq 3$, $d_j = 1$ and $f_j = 2$.

Modify $D$ and $F$ by removing $d_j$ and $f_j$ and decreasing $d_i$ by 1. This modified $D'$ and $F'$ are tree degree sequences without common leaves on $n - 1$ vertices, therefore, by induction, $D'$ and $F'$ have edge-disjoint caterpillar realizations $T_1'$ and $T_2'$, respectively. Modify $T_1'$ and $T_2'$ as follows. Add back vertex $v_j$ and connect it to vertex $v_i$ in $T_1'$. The tree $T_1$ thus arising is a realization of $D$. Take a path P in $T_2'$ containing all non-leaf vertices and two leaves. Observe that P has at least 3 edges as otherwise $F$ has $n - 2$ leaves, so $D$ has only two, contradicting to $d_i \geq 3$. Hence P has an edge $v_k v_\ell$ such that $k \neq i$ and $\ell \neq i$. For constructing $T_2$, replace edge $v_k v_\ell$ of $T_2'$ by two edges, $v_k v_j$ and $v_j v_\ell$. The tree $T_2$ thus obtained is a caterpillar, edge-disjoint from $T_1$ and is a realization of $F$. □

The theorem implicitly states that if two tree degree sequences do not share common leaves then their sum is graphical. If the two trees have common leaves, their sum is not necessarily graphical as shown by the example $D = (2, 1, 1)$ and $F = (2, 1, 1)$. Observe that the largest degree in $D + F$ is 4, and there are only 3 vertices.

On the other hand, if the sum of the two sequences happens to be graphical, then they also have edge-disjoint realizations, as was shown by Kundu in [6].

**Theorem 3** ([6]). *Let $D = (d_1, \dots, d_n)$ and $F = (f_1, \dots, f_n)$ be two tree degree sequences. Then there exist edge-disjoint tree realizations of $D$ and $F$ if and only if $D + F$ is graphical.*

However, there are tree degree sequences that have edge-disjoint tree realizations but do not have edge-disjoint caterpillar realizations. For example, consider the tree degree sequences

$$D = (5, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1)$$

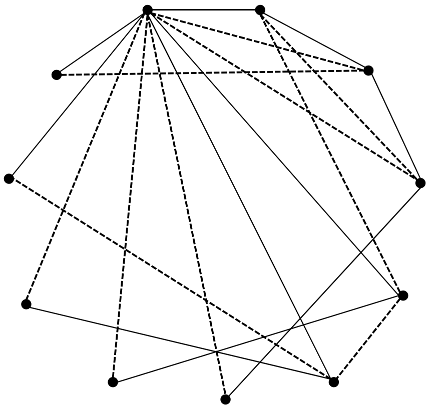


Figure 1: Edge-disjoint realization of two degree sequences, both of them are $(5, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1)$

.

and

$$F = (5, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1).$$

According to Theorem 3, these sequences have edge-disjoint realizations as their sum is graphical (see Fig. 1). We claim that they do not have edge-disjoint caterpillar realizations. To see this, observe that in any caterpillar realization, the degree 5 vertices must be connected to at least 3 leaves. However, there are only 5 vertices that are leaves in any of the trees, showing that any pair of caterpillar realizations will share at least one edge.

Theorem 2 considered the case when the leaf vertices of the degree sequences do not coincide. Now we turn to the opposite end, namely when each vertex is a leaf in at least one of the sequences.

**Theorem 4.** *Let $D = (d_1, \dots, d_n)$ and $F = (f_1, \dots, f_n)$ be tree degree sequences such that $\min(d_i, f_i) = 1$ for all $i$. Let $T_1$ and $T_2$ be random realizations of $D$ and $F$ uniformly distributed. Then the expected number of common edges of $T_1$ and $T_2$ is 1.*

*Proof.* The proof is based on the following lemma.

**Lemma 5.** *Let $T$ be a random realization of the tree degree sequence $D = (d_1, \dots, d_n)$ (where $n \geq 3$). Then the probability of having an edge between $v_i$ and $v_j$ is*

$$\frac{d_i + d_j - 2}{n - 2}.$$

*Proof.* It is well known that the number of trees with a given degree sequence is

$$\frac{(n-2)!}{\prod_{k=1}^{n}(d_k - 1)!}. \tag{1}$$

Let $T'$ denote those trees in which $v_i$ and $v_j$ are adjacent. Let $f$ be a mapping from $T'$ to the set of trees with degree sequence

$$(d_1, \dots, d_{i-1}, d_{i+1}, \dots, d_{j-1}, d_{j+1}, \dots, d_n, d_i + d_j - 2)$$

obtained by joining $v_i$ and $v_j$ to a common vertex. The function $f$ is surjective, and each tree is an image $\binom{d_i+d_j-2}{d_i-1}$ times. Therefore the number of trees in which $v_i$ is adjacent to $v_j$ is

$$\frac{(n-3)!}{(d_i+d_j-3)!\prod_{k\neq i,j}(d_k-1)!}\frac{(d_i+d_j-2)!}{(d_i-1)!(d_j-1)!} = \frac{(d_i+d_j-2)(n-3)!}{\prod_{k=1}^{n}(d_k-1)!}. \qquad (2)$$

The probability that $v_i$ is adjacent to $v_j$ is the ratio of (2) and (1), which is indeed

$$\frac{d_i+d_j-2}{n-2},$$

thus concluding the proof of the lemma. □

Now we turn to the proof of the theorem. Let $D$ and $F$ be degree sequences satisfying that each vertex is a leaf in at least one of the trees. Define

$$\begin{aligned}\mathcal{A} &:= \{i \mid d_i > 1 \wedge f_i = 1\}, and\\ \mathcal{B} &:= \{i \mid d_i = 1 \wedge f_i > 1\}.\end{aligned}$$

Note that there might be parallel edges in the two trees only between these two sets. The expected number of parallel edges is then

$$\sum_{i\in\mathcal{A}}\sum_{j\in\mathcal{B}}\frac{(d_i-1)(f_j-1)}{(n-2)^2} = \sum_{i\in\mathcal{A}}\frac{d_i-1}{n-2}\sum_{j\in\mathcal{B}}\frac{f_j-1}{n-2} = \sum_{i=1}^{n}\frac{d_i-1}{n-2}\sum_{j=1}^{n}\frac{f_j-1}{n-2} = 1,$$

since $d_i = 1$ for all $i \in \bar{\mathcal{A}}$, $f_j = 1$ for all $j \in \bar{\mathcal{B}}$, and the sum of the degrees decreased by 1 is $n-2$ for any tree degree sequence. This finishes the proof of the theorem. □ □

Theorem 4 implies a characterization of realizability for a subclass of tree degree sequences.

**Corollary 6.** *Let $D = (d_1, \dots, d_n)$ and $F = (f_1, \dots, f_n)$ be tree degree sequences such that each vertex is a leaf in at least one of them. Then $D$ and $F$ have edge-disjoint tree realizations if and only if $d_i < n-1$ and $f_i < n-1$ for all $i$.*

*Proof.* If $\max_i\{d_i\} = n-1$ or $\max_i\{f_i\} = n-1$ then $D+F$ is not graphical. On the other hand, if none of the trees is a star, then there are four distinct indices $i_1, i_2, j_1$ and $j_2$ such that $i_1, i_2 \in \mathcal{A}$ and $j_1, j_2 \in \mathcal{B}$. Then there exists a pair of trees $T_1$ and $T_2$ such that both trees contain edges $(v_{i_1}, v_{j_1})$ and $(v_{i_2}, v_{j_2})$ and $T_1$ realizes $D$ while $T_2$ realizes $F$. Indeed, the degree 1 vertices can be connected to any of the non-leaf vertices. This means that the two sequences have realizations having at least 2 common edges, which is above the expected value. Hence there must also exist a realization with less common edges than the expected number, which is 1. That is, there exists an edge-disjoint realization of the two sequences, thus concluding the proof. □

This theorem will be useful also at generating random realizations, see the next section.

Similar theorem holds for more tree sequences as well. Let us fix again $V = \{v_1, \dots, v_n\}$. We need the following technical preliminary lemma.

**Lemma 7.** *Let $D = (d_1, \dots, d_n)$ be a tree degree sequence, $n > m > 2$ and $U = \{v_i \mid d_i > 1\}$. Suppose $V_1, \dots, V_{m-1}$ are pairwise disjoint sets in $L = V \setminus U$. Suppose further that $|U| > 1, |V_1| > 1, \dots, |V_{m-1}| > 1$ and $d_i \le n-m$ for all $i$. Then there is a tree $T$ realizing $D$ such that its restriction to $U \cup V_j$ is a non-star tree for all $j$.*

*Proof.* For any tree realization $T$, its restriction to $U \cup V_j$ is a tree because outside $U$ there are only leaves. In the case $|U| \ge 4$ we claim that there is a tree realization $T$ such that its restriction to $U$ is not a star. Indeed, if $T$ restricted to $U$ is a star centered at $u \in U$, then, by the degree bound, there is a leaf $w \in L$ not connected to $u$. Let $u_1 \in U$ denote the unique neighbor of $w$ in the tree, and let $u_2$ be a third vertex of $U$. Replacing edges $uu_2$ and $u_1w$ by edges $u_1u_2$ and $uw$ gives another tree realization $T$ whose restriction to $U$ is not a star.

For the case $|U| = 2$, let $U = \{v_i, v_j\}$. Connect first $v_i$ to $v_j$. Now $d_i + d_j = n$, so $d_i \ge m$ and $d_j \ge m$. For each $k \le m-1$, connect one vertex of $V_k$ to $v_i$ and another one to $v_j$. The remaining leaves in $L$ can be distributed easily: connect any $d_i - m$ of them to $v_i$ and the remainder to $v_j$, giving the aimed tree realization.

Finally suppose $U = \{v_i, v_j, v_k\}$. We may also suppose that $2 \le d_i \le d_j \le d_k$. Connect first $v_i$ and $v_j$ to $v_k$. It is easy to connect vertices of $L$ to $U$ in such a way that for each $\ell \le m-1$, at least one vertex of $V_\ell$ is connected to either $v_j$ or $v_k$. □

**Theorem 8.** *Let $D_1, D_2, \dots, D_m$ be tree degree sequences with $D_i = d_{i,1}, d_{i,2}, \dots, d_{i,n}$ such that each vertex is a leaf in all except at most one of them. Then $D_1, D_2, \dots, D_m$ have edge-disjoint realizations if and only if $\max_{i,j}\{d_{i,j}\} \le n-m$.*

*Proof.* Necessity is clear as $D_1 + D_2 + \cdots + D_m$ is not graphical if $\max_{i,j}\{d_{i,j}\} > n-m$.

The statement is trivial when $m = 1$. If $m = 2$ then it is equivalent to Corollary 6, so we may suppose $m > 2$.

We give a constructive proof for the other direction. First a trial solution is built which might contain parallel edges, then these parallel edges are eliminated to get an edge-disjoint realization.

Let $V_i$ denote the subset of vertices on which the degrees in $D_i$ are larger than 1. Note that $\{V_1, V_2, \dots, V_m\}$ forms a subpartition of $V$ and $|V_i| \ge 2$ for each $i = 1, \dots, m$. For a degree sequence $D_i$, construct a trial tree $\tilde{T}_i$ by using Lemma 7, which ensures that the subtree on vertices $V_i \cup V_k$ is a non-star tree for any $k \neq i$.

From the trial solution, which might contain several parallel edges, a final solution is built in the following way. While there exists a pair of indexes $(i, k)$ such that there is one or more parallel edges between $V_i$ and $V_k$, do the following. Let $\tilde{T}_{i,k}$ denote the subtree of the tree $\tilde{T}_i$ on vertices $V_i \cup V_k$ and let $\tilde{D}_{i,k}$ denote its degree sequence. By Corollary 6, $\tilde{D}_{i,k}$ and $\tilde{D}_{k,i}$ have edge-disjoint tree realizations. Replace $\tilde{T}_{i,k}$ and $\tilde{T}_{k,i}$ by such realizations. This removes all parallel edges between $V_i$ and $V_k$ because $\tilde{T}_j$ has no edge between these sets if $j \neq i,\ j \neq k$. □

# 4 Counting and sampling realizations

Since typically there are more than one realizations when a realization exists, and typically the number of realizations might grow exponentially, it is also a computational challenge to estimate their number and/or sample almost uniformly a solution. Here we prove the following theorem.

**Theorem 9.** *Let $D = (d_1, \ldots, d_n)$ and $F = (f_1, \ldots, f_n)$ be two tree degree sequences such that each vertex is a leaf in at least one of the trees. Furthermore, assume that none of the trees is a star. Then there is an FPRAS for estimating the number of disjoint realizations and there is an FPAUS for almost uniformly sampling realizations.*

*Proof.* This theorem is based on Theorem 4. As we discussed, there are random trees with at least two parallel edges. The number of pair of trees containing parallel edges $(v_{i_1}, v_{j_1})$ and $(v_{i_2}, v_{j_2})$ such that $d_{i_1}, d_{i_2} > 1$ and $f_{j_1}, f_{j_2} > 1$ is

$$\frac{(n-4)!}{(d_{i_1}-2)!(d_{i_2}-2)!\prod_{k\neq i_1,i_2}(d_k-1)!} \cdot \frac{(n-4)!}{(d_{j_1}-2)!(d_{j_2}-2)!\prod_{k\neq j_1,j_2}(f_k-1)!} . \qquad (3)$$

Therefore, at least the same number of pair of trees have no parallel edges (that is, are edge-disjoint realizations of the degree sequences) to get the expectation 1 for the number of parallel edges. Therefore, the probability that two random trees will be edge-disjoint is at least

$$\frac{(d_{i_1}-1)(d_{i_2}-1)(f_{j_1}-1)(f_{j_2}-1)}{(n-2)^2(n-3)^2}.$$

It follows from basic probabilistic arguments that an FPRAS algorithm can be designed based on this property. Indeed, let $\xi$ be the indicator variable that a random pair of trees are edge-disjoint realizations. Then the number of edge-disjoint realizations is

$$E[\xi]\frac{(n-2)!}{\prod_{k=1}^n (d_i-1)!}\frac{(n-2)!}{\prod_{k=1}^n (f_i-1)!}.$$

Furthermore, we know that

$$E[\xi] \geq \frac{(d_{i_1}-1)(d_{i_2}-1)(f_{j_1}-1)(f_{j_2}-1)}{(n-2)^2(n-3)^2}.$$

Uniformly distributed random trees with a prescribed degree sequence can be generated in polynomial time based on the fact that the probability that a given leaf is connected to a vertex with degree $d_i$ is

$$\frac{d_i-1}{n-2}.$$

A uniformly distributed tree can be generated by randomly selecting a neighbor of a given leaf, then generating a random tree for the remaining degree sequence. Equivalently, the trees with a prescribed degree sequence can be encoded by the Prüffer codes in which the index $i$ appears exactly $d_i - 1$ times. Uniformly generating such Prüffer codes is an elementary computational task.

Therefore, random pair of trees can be generated in polynomial time, and it is easy to check whether or not they are edge-disjoint realizations. Such sampling of random trees provide an unbiased estimation for the expectation of the indicator variable $\xi$. Indeed, if $X_i$ is 1 if the $i^{\text{th}}$ pair of random trees are edge-disjoint and 0 otherwise, then the random variable

$$Y_m := \sum_{i=1}^m X_i$$

follows a binomial distribution with parameter $p = E[\xi]$ and expectation $mE[\xi]$. The tails of the binomial distributions can be bounded by the Chernoff's inequality:

$$P(Y_m \leq mp(1-\epsilon)) \leq exp\left(-\frac{1}{2p}\frac{(mp-mp(1-\epsilon))^2}{m}\right).$$

This should be bounded by $\frac{\delta}{2}$ (the other half $\delta$ error will go to the other tail)

$$exp\left(-\frac{1}{2p}\frac{(mp-mp(1-\epsilon))^2}{m}\right) \leq \frac{\delta}{2}. \qquad (4)$$

Solving Equation 4, we get

$$m \geq \frac{-2\log\left(\frac{\delta}{2}\right)}{p\epsilon^2}.$$

For the upper tail, we can also use the Chernoff's inequality, just replacing P with $1-p$ and the upper threshold $mp(1+\epsilon)$ with $m - mp(1+\epsilon)$:

$$P(Y_m \geq mp(1+\epsilon)) \leq$$
$$exp\left(-\frac{(m(1-p)-(m-mp(1+\epsilon)))^2}{2(1-p)m}\right).$$

Upper bounding this with $\frac{\delta}{2}$ and solving the inequality, we get that

$$m \geq \frac{-2(1-p)\log\left(\frac{\delta}{2}\right)}{p^2\epsilon^2}.$$

Since $\frac{1}{p} = O(n^4)$, the necessary number of samples is indeed polynomial with the size of the problem, $\frac{1}{e}$ and $-\log(\delta)$. Furthermore, one sample can be generated in polynomial time, therefore this algorithm is indeed an FPRAS.

It is also well known that an FPAUS algorithm can be designed in this case. The FPAUS algorithm generate $\frac{-\log(\epsilon)}{p}$ pair of random trees. If any of them is an edge-disjoint realization, then the algorithm returns with it. Otherwise it generates an arbitrary realization and returns with it.

This is indeed an FPAUS algorithm, since any random pair of trees which are edge-disjoint come from sharp the uniform distribution of the solutions. The probability that there will be no edge-disjoint pair of trees in $m$ number of samples is

$$(1-p)^m.$$

This probability is not larger than $\epsilon$. Indeed,

$$(1-p)^{\frac{-\log(\epsilon)}{p}} \leq \epsilon,$$

since

$$\frac{-\log(\epsilon)}{p}\log(1-p) \leq \log(\epsilon)$$

because

$$-\log(1-p) \geq p.$$

Namely, the algorithm generates realizations from a distribution which is the convex combination $(1-\epsilon')U + \epsilon'\pi$, where $\epsilon' \leq \epsilon$, $U$ is the uniform distribution and $\pi$ is an arbitrary distribution. However, the variational distance of this distribution from the uniform one is

$$d_{TV}(U,(1-\epsilon')U+\epsilon'\pi) =$$
$$\tfrac{1}{2}\sum_x |U(x) - ((1-\epsilon')U(x) + \epsilon'\pi(x)| =$$
$$\epsilon'\tfrac{1}{2}\sum_x |U(x) - \pi(x)| \leq \epsilon' \leq \epsilon.$$

Since one sample can be generated in polynomial time, and the total number of samples is polynomial with the size of the problem and $-\log(\epsilon)$, this algorithm is indeed and FPAUS. □

It remains an open question whether or not similar theorems exist for the case when the tree degree sequences have common high degrees. Also it is open if exact counting of the edge-disjoint solutions is possible in polynomial time, although the natural conjecture is that this counting problem is #P-complete.

# 5 A complexity result

What can we say when only one of the two degree sequences is a tree degree sequence and the other is arbitrary? Unfortunately, we have a negative result here.

**Theorem 10.** *It is* NP*-complete to decide if a tree degree sequence and an arbitrary degree sequence have an edge-disjoint realization (in which the tree degree sequence does not necessarily have to be represented by a tree).*

*Proof.* By [1], it is NP-complete to decide if two bipartite degree sequences has edge-disjoint realizations. We have the following observations.

**Claim 1.** *A bipartite degree sequence pair*

$$D = (d_{1,1},\ldots,d_{1,n_1}),(d_{2,1},\ldots,d_{2,n_2})$$

*and*

$$F = (f_{1,1},\ldots,f_{1,n_1}),(f_{2,1},\ldots,f_{2,n_2})$$

*has an edge disjoint realization if and only if the simple degree sequence pair*

$$D' = (d_{1,1}+n_1-1,\ldots,d_{1,n_1}+n_1-1,d_{2,1},\ldots,d_{2,n_2})$$

*and*

$$F' = (f_{1,1},\ldots,f_{1,n_1},f_{2,1}+n_2-1,\ldots,f_{2,n_2}+n_2-1)$$

*has an edge-disjoint realization.*

*Proof.* If an edge-disjoint bipartite realization of $D$ and $F$ is given, then the complete graph on the first vertex class can be added to the first realization and the complete graph on the second vertex class can be added to the second realization to get a (now non-bipartite) realization of $D'$ and $F'$. On the other hand, it is easy to see that any realization of $D'$ contains $K_{n_1}$ on the first $n_1$ vertices, and any realization of $F'$ contains $K_{n_2}$ on the last $n_2$ vertices. Given an edge-disjoint realization of $D'$ and $F'$, deleting $K_{n_1}$ from $D'$ and $K_{n_2}$ from $F'$ yields an edge-disjoint realization of $D$ and $F$. □

**Claim 2.** *The degree sequence pair $D = (d_1,\ldots,d_n)$ and $F = (f_1,\ldots,f_n)$ has an edge-disjoint realization if and only if the degree sequence pair $D' = (d_1+1,\ldots,d_n+1,n)$ and $F' = (f_1,\ldots,f_n,0)$ has an edge-disjoint realization.*

*Proof.* Let $G_1$ and $G_2$ be an edge-disjoint realization of $D$ and $F$. Then add a vertex $v_{n+1}$ to $G_1$, and connect it to all the other vertices to get a realization of $D'$. Add an isolated vertex $v_{n+1}$ to $G_2$ to get a realization of $F'$. These realizations of $D'$ and $F'$ are edge-disjoint. On the other hand, in any realization of $D'$, $v_{n+1}$ is connected to all the other vertices. If edge-disjoint realizations of $D'$ and $F'$ are given, delete $v_{n+1}$ from both realizations to get edge-disjoint realizations of $D$ and $F$. □

**Claim 3.** *The degree sequence pair $D = (d_1,\ldots,d_n)$ and $F = (f_1,\ldots,f_n)$ has an edge-disjoint realization if and only if the degree sequence pair $D' = (d_1,\ldots,d_n,1,1)$ and $F' = (f_1+1,\ldots,f_n+1,n,0)$ has an edge-disjoint realization.*

*Proof.* Any edge-disjoint realization $G_1$ and $G_2$ of $D$ and $F$ can be extended to an edge-disjoint realization of $D'$ and $F'$ by adding two vertices $v_{n+1}$ and $v_{n+2}$, and then connecting $v_{n+1}$ to all $v_1,\ldots,v_n$ in $G_2$ and connecting $v_{n+1}$ and $v_{n+2}$ in $G_1$. On the other hand, in any edge-disjoint realizations $G'_1$ and $G'_2$ of $D'$ and $F'$, $v_{n+1}$ is connected to all $v_1,\ldots,v_n$ in $G'_2$, therefore, $v_{n+1}$ must be connected to $v_{n+2}$ in $G'_1$. Therefore deleting $v_{n+1}$ and $v_{n+2}$ yields an edge-disjoint realization of $D$ and $F$. □

We can use Claim 1 to prove that it is NP-complete to decide if two simple degree sequences have edge-disjoint realizations. Claim 2 shows that it is NP-complete to decide if two degree sequences have edge-disjoint realizations such that one of the degree sequences does not have 0 degrees. Finally, Claim 3 can be used to iteratively transform any $D$ degree sequence (that already does not have a 0 degree) into a tree degree sequence. Indeed, in each step, we add two vertices to $D$ and extend the sum of the degrees only by 2. Therefore in a polynomial number of steps, we get a degree sequence $D'$ in which the sum of the degrees is exactly twice the number of vertices minus 2. Therefore it follows that given any bipartite degree sequences $D$ and $F$, we can construct in polynomial time two simple degree sequences $D'$ and $F'$ such that $D$ and $F$ have edge-disjoint realizations if and only if $D'$ and $F'$ have edge-disjoint realizations, furthermore, $D'$ is a tree degree sequence. □

## 6 Discussion and conclusions

In this paper, we considered packing tree degree sequences. When there are no common leaves, there are always edge-disjoint caterpillar realizations. On the other hand, there might not be edge-disjoint caterpillar realizations when there are common leaves, even if otherwise there are edge-disjoint tree realizations.

When there are no common high degree vertices, there are edge-disjoint tree realizations if and only if none of the degree sequences is a degree sequence of a star. Similar theorem exists for arbitrary number of trees, and it is easy to decide if arbitrary number of tree degree sequences without common high degrees have edge-disjoint realizations.

It is also known [5] that a degree sequence and an almost regular degree sequence have an edge-disjoint realization if and only if their sum is graphical. This raises the natural question if a degree sequence and a tree sequence have edge-disjoint realizations if and only if their sum is graphical. We showed that the answer is no to this question, and actually it is NP-complete to decide if an arbitrary degree sequence and a tree degree sequence have edge-disjoint realizations.

We also showed hot to approximately count and sample edge-disjoint tree realizations with prescribed degrees. We proved that this is possible if there are no common high degree vertices. However, when the two degree sequences have common high degree vertices then the problem remains open.

## References

[1] Dürr, C., Guinez, F., Matamala, M.: Reconstructing 3-colored grids from horizontal and vertical projections is NP-hard. European Symposium on Algorithms, 776–787 (2009)

[2] Erdős, P.; Gallai, T. : Graphs with vertices of prescribed degrees (in Hungarian) Matematikai Lapok, 11: 264–274. (1960)

[3] S.L. Hakimi: On the degrees of the vertices of a directed graph. J. Franklin Institute, 279(4):290–308. (1965)

[4] V. Havel: A remark on the existence of finite graphs. (Czech), Časopis Pěst. Mat. 80:477–480. (1955)

[5] Kundu, S.: The k-factor conjecture is true. Discrete Mathematics, 6(4):367–376. (1973)

[6] Kundu, S.: Disjoint Representation of Tree Realizable Sequences. SIAM Journal on Applied Mathematics, 26(1):103–107. (1974)

# Construction of Orthogonal CC-sets

Andrej Brodnik, Marko Palangetić and Daniel Silađi
University of Primorska, Koper, Slovenia
E-mail: andrej.brodnik@upr.si, palangeticmarko95@hotmail.com, szilagyi.d@gmail.com

Vladan Jovičić
École Normale Supérieure, Lyon, France
E-mail: vladan94.jovicic@gmail.com

**Keywords:** algorithms, NP-hardness, modeling

**Received:** October 30, 2018

*In this paper we present a graph-theoretical method for computing the maximum orthogonal subset of a set of coiled-coil peptides. In chemistry, an orthogonal set of peptides is defined as a set of pairs of peptides, where the paired peptides interact only mutually and not with any other peptide from any other pair. The main method used is a reduction to the maximum independent set problem. Then we use a relatively well-known maximum independent set solving algorithm which turned out to be the best suited for our problem. We obtained an orthogonal set consisting of 29 peptides (homodimeric and heterodimeric) from initial 5-heptade set. If we allow only heterodimeric interactions we obtain an orthogonal set of 26 peptides.*

*Povzetek: V članku je predstavljen izračun največje ortogonalne množice peptidov z uporabo metod teorije grafov. Za elemente ortogonalne množice velja, da, če dva elementa vzajemno delujeta, potem ne delujeta z nobenim drugim elementom množice. Algoritem, ki uporablja prevedbo na problem največje neodvisne množice, je bil uporabljena v praksi.*

## 1 Motivation

In the last 30 years, impressive 3D structures have been built using DNA, in a field called DNA origami [5]. Complex structures built from proteins would have many advantages, since amino acids provide much more functionality. The main problem is that the simple Watson-Crick paring rules present in DNA have no simple analogue for proteins. Using a special class of polypeptides, called coiled-coil polypeptides, the orthogonal binding rules of DNA can be emulated [2, 4]. By specifying only the primary structure of those polypeptides (the order of amino acids), complex 3D structures can be built, such as the recent protein tetrahedron [3]. More specifically, that structure is determined by taking the wireframe of the desired object, doubling every edge, and performing an Euler traversal of the obtained graph. Then, we know that the peptides associated with edges that were initially parallel must bind, and all others must not.

Essential for such designs is that each pair of peptides interacts only mutually, and not with any other pair. Thus, the notion of an *orthogonal set* is introduced. Obviously, the greater our orthogonal set is, the more complex are the structures we can create. Currently the limiting factor in designing larger structures is the small set of available peptides.

In this paper, we describe a method for determining an orthogonal set of maximum size, from a given set of admissible peptides. Also, in section 6 we present a possible approach for extending a given orthogonal set.

## 2 Problem description

As input we are given a set of peptides $P = p_1, p_2, \dots p_n$ (their primary structures – given as strings of fixed length) and interaction matrix $I$. If $I_{i,j} = 1$, then $p_i$ interacts with $p_j$ and if it is 0 they do not interact. We have to construct a set of pairs $S$, where $(p_i, p_j) \in S$, iff $I_{i,j} = 1$ and for all other $p_k$ that are in any pair of $S$ $I_{i,k} = 0$. Moreover, if $i = j$ in $(p_i, p_j)$ we are talking of *homodimer* and otherwise of *heterodimer*.

We can model this problem as a graph-theoretical one: First, an undirected graph $G = (V, E)$ where $V$ is the set of peptides $P$, and the edge set $E$ contains an edge $p_ip_j$ (or a loop at $p_i$, denoted by $p_ip_i$) if and only if $p_i$ and $p_j$ interact. Given that graph, we want to find a subset of non-adjacent edges whose vertices are also non-adjacent. More formally, and more conveniently for later consideration, our problem can be defined as follows:

**Definition** (Maximum Independent Set of Pairs (MISP))**.** Let $G = (V, E)$ be an undirected graph and let $k$ be a positive integer. Does there exist set a $S \subseteq E$ such that for any $u_1v_1, u_2v_2 \in S$

$$\{u_1, v_1\} \cap \{u_2, v_2\} = \emptyset,$$

$$\{\{u_1, u_2\}, \{u_1, v_2\}, \{v_1, u_2\}, \{v_1, v_2\}\} \cap E = \emptyset$$

and $|S| > k$?

# 3 Hardness of the problem

In order to determine the best possible solution of our problem, in this section we will prove that MISP is NP-complete.

**Theorem 1.** *Maximum independent set of pairs is NP-complete.*

*Proof.*

**Algorithm 1** NP certifier

```
$S \leftarrow$ given set of pairs
if $|S| < k$ then
    return No
for $u_1v_1 \in S$ do
    for $u_2v_2 \in S - u_1v_1$ do
        if $u_1u_2 \in E \vee u_1v_2 \in E \vee v_1u_2 \in E \vee v_1v_2 \in E \vee u_1v_1 \notin E$ then
            return No
return Yes
```

The problem is easily seen to be in NP, and Algorithm 1 is its polynomial certifier.

Now we will reduce the *independent set* problem to MISP in order to show that MISP is NP-hard.

Let $G = (V, E)$ be a graph, for which we want to check if there exists an independent set of size greater than $k$. Define a new graph $G' = (V', E')$ as follows. Initially, let $V' = V$ and $E' = E$. Then, for each vertex $v \in V$ add another vertex $v'$ (a copy of $v$) to $V'$ and add the edge $vv'$ to $E'$.

**Lemma 1.** *Every maximal independent set of pairs in $G'$ consists only of the edges of the form $vv'$.*

*Proof.* Let $S$ be a MISP in $G'$. Suppose the contrary, i.e., there is a pair $uv \in S$ which is not of the form $ww'$ for $w \in V$. Then, for all $u_1v_1 \in S$ we have $u_1u \notin E'$, $u_1v \notin E'$, $v_1u \notin E'$, $v_1v \notin E'$. Then we can delete the pair $uv$ from $S$ and add pairs $uu'$ and $vv'$ where $u'$ and $v'$ are copies of $u$ and $v$, respectively. We can do this since the only neighbors of $u'$ and $v'$ are $u$ and $v$, respectively. We obtained an independent set of pairs, with more more than $|S|$ elements, a contradiction. □

Now we will prove that there is an independent set $|S| \geq k$ in $G$ if and only if there is an independent set of pairs $|S_P| \geq k$ in $G'$.

($\Rightarrow$) Suppose that $S$ is an independent set in $G$ with $|S| \geq k$. Then, define the independent set of pairs $S_P$ in $G'$ on the following way:

$$S_P = \{vv' \mid v \in S\}.$$

It is easy to verify that this is independent set of pairs according to the definition above. Then $|S_P| = |S| \geq k$.

($\Leftarrow$) For the other direction, suppose that $S_P$ is an independent set of pairs in $G'$ with $|S_P| \geq k$. Then, by the previous lemma, we can define the following independent set $S$ in $G$:

$$S = \{v \in V \mid vv' \in S_P\}.$$

By the construction of graph $G'$ and by the lemma, one can show that $S$ is an independent set of $G$. Then $|S| = |S_P| \geq k$ which completes proof that MISP is NP-hard.

Combining the NP-hardness with the earlier fact that MISP is in NP, we conclude that MISP is NP-complete. □

# 4 Reducing MISP to the maximum independent set

Now that we know that MISP is NP-complete, we can use one of the vast number of algorithms already developed for solving various NP-hard problems, once we reduce MISP to that problem. The most natural choice is the maximum independent set problem [1, 6, 7].

Based on the MISP graph $G = (V, E)$, we construct a new graph $G^* = (V^*, E^*)$, where $V^* = E$, and two vertices are connected (in $G^*$) if and only if their corresponding edges in $G$ share a common vertex or have two of their vertices connected by an edge. It is easy to see that finding an independent set in $G^*$ will give us an independent set of pairs, according to the definition in section 2. Moreover, due to our construction, an independent set of pairs in $G$ also gives us an unique independent set in $G^*$.

Thus, we have obtained a bijection between the independent sets of $G^*$ and the independent sets of pairs of $G$.

# 5 Results

We use results from the previous section to solve the MISP of the input graph $G$ which is constructed from the input set of peptides $P = p1, p_2, \ldots, p_n$ in several steps.

1. Based on previous work by [4], we calculate the interaction scores $s_{ij}$ for each pair of peptides $p_ip_j$ (including homodimers $p_ip_i$), and store that matrix for the following steps.

2. Choose a threshold $t$ based on which we decide whether peptides $p_i$ and $p_j$ with interaction score $s_{ij}$ will interact. If $s_{ij} < t$, we declare that $p_i$ and $p_j$ are not interacting (or, more precisely, interacting in a negligibly small proportion), and likewise, if $s_{ij} \geq t$, $p_i$ and $p_j$ interact. For practical purposes, we might want to introduce two thresholds $t_s$ (strong interaction threshold) and $t_w$ (weak interaction threshold) such that $t_s - t_w$ is a positive “safety margin” accounting for the inexactness of the scoring function from the previous step. Then, the vertices of $G^*$ would be just the pairs that interact strongly, but they will be connected even if they interact weakly. However, such considerations are outside the scope of this paper.

3. Construct the graph $G$ on the set of peptides by connecting the interacting ones, as described in section 2.

4. Reduce $G$ to $G^*$, suitable for computing the independent set, as described in section 4.

5. Find the maximum independent set in $G^*$. As shown before, it corresponds to the MISP (or, orthogonal set) in $G$. We use the (exact) maximum clique solving algorithm presented in [1], which is based on greedy graph colorings – i.e. if we can color a particular subgraph with $k$ colors, we know that that the maximum clique in that subgraph has size at most $k$.

In order to test our algorithm, we generated synthetic initial sets of peptides, based on two observations: Firstly, the interaction scoring function is designed to consider only 4 positions in each heptad. Secondly, using electrostatic arguments about individual amino acids and their positions in the coiled-coil, we reduced the variation even further, by allowing only 2 different amino acids on 3 of those 4 positions, and completely fixing the remaining amino acid. Thus, we obtain 8 essentially different heptads, which we use to build up larger peptides. Our main result is the calculation of a 29-peptide orthogonal subset of the 5-heptad initial set ($2^{15}$) peptides (generated as described above), as well as a 26-peptide purely heterodimeric orthogonal subset of the same initial set. The interaction score heatmap can be seen on Figures 1 and 2.

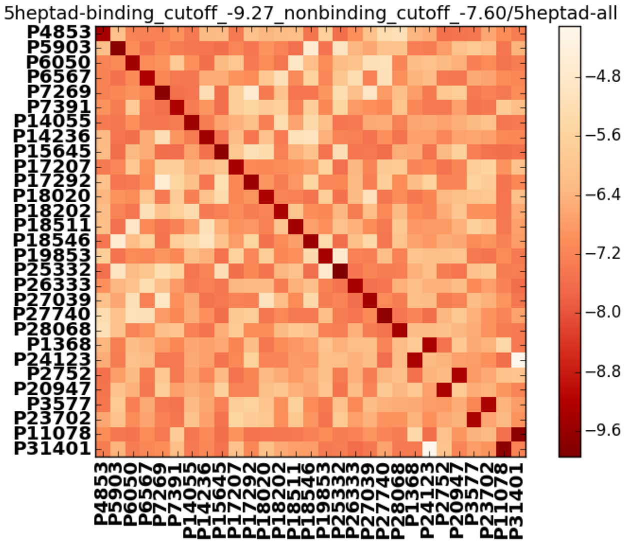


Figure 1: 5-heptad orthogonal set, no restriction

The peptidets which belong to orthogonal set are in both figures colored in dark red.

## 6 Future work

Up to now, we have only considered orthogonal sets derived from synthetically generated peptides, as described in the previous section. To actually use such an orthogonal set, we had to manually synthesize all of those peptides.

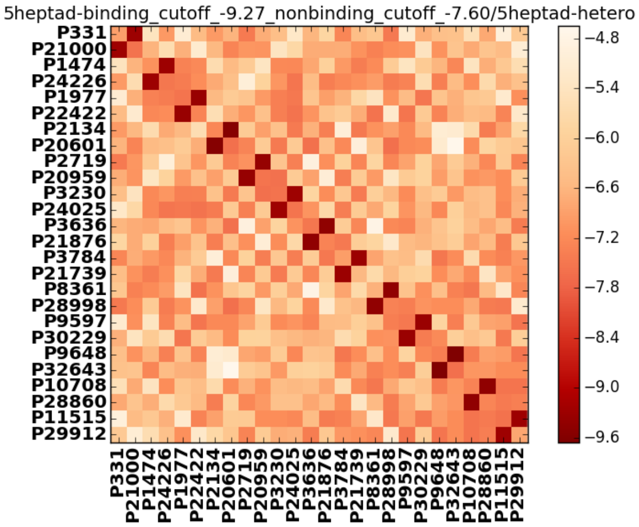


Figure 2: 5-heptad orthogonal set, heterodimers only

An alternative would be to construct a maximal orthogonal set from the set of all natural tetraheptads (coiled-coils where each of the 4 heptads occurs naturally). Since there are 1171 known natural heptads, we can combine them to get $1171^4 = 1\,880\,301\,880\,081$ possible tetraheptads. Finding a maximal orthogonal subset of this set would require finding the maximum independent set of a graph with more than $10^{12}$ vertices – a task clearly impossible to do in a reasonable amount of time.

Either way, we generate a large number of "useless" peptides, that will be discarded later, and not used in the (much smaller) orthogonal set.

Our idea is to use a heuristic to reduce the initial set to a more manageable size: since it is possible to calculate the interaction matrix for single natural heptads, we can approximate scores for tetraheptads as shown at Figure 3. More specifically, we will add up the precalculated scores between (adjacent) heptads which are connected as on figure 3. Of course, some interactions will be left unaccounted for in the final score, for example the last amino acid in heptad 1 on 3 may interact with first amino acid of heptad 7 which is not added to the final score.

This observation enables us to construct more meaningful initial peptide sets consisting of longer peptides, based on the already-calculated orthogonal sets of shorter peptides.

## 7 Concluding remarks

In this paper, we presented an exact method for determining an orthogonal set of coiled-coil polypeptides, if we are given a numeric measure of their interaction strength. Our approach has been demonstrated to be successful for moderately large initial peptide sets (tens of thousands), and has

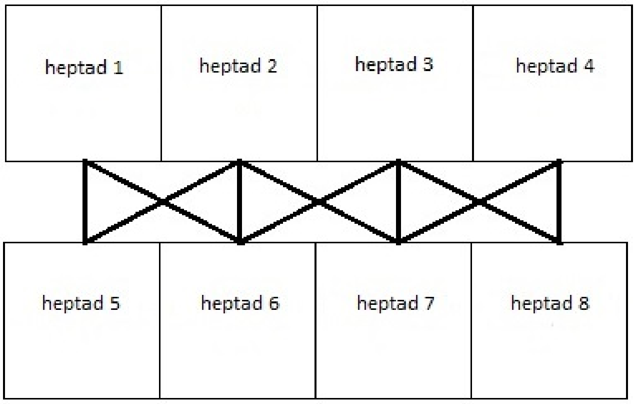


Figure 3: Proposed way of scoring

given us optimal orthogonal sets that could not have been calculated by hand.

Unfortunately, for even larger initial sets, the maximum-clique solver becomes an apparent bottleneck, as it has to operate on graphs of size $O(n^4)$, where $n$ is the size of the initial set. In that case, we suggest investigating a bottom-up method described in the section 6.

## References

[1] M. Depolli, J. Konc, K. Rozman, R. Trobec, and D. Janezic. Exact parallel maximum clique algorithm for general and protein graphs. *Journal of chemical information and modeling*, 53(9):2217–2228, 2013. https://doi.org/10.1021/ci4002525.

[2] J. H. Fong, A. E. Keating, and M. Singh. Predicting specificity in bZIP coiled-coil protein interactions. *Genome biology*, 5(2):R11, 2004. https://doi.org/10.1186/gb-2004-5-2-r11.

[3] H. Gradišar, S. Božič, T. Doles, D. Vengust, I. Hafner-Bratkovič, A. Mertelj, B. Webb, A. Šali, S. Klavžar, and R. Jerala. Design of a single-chain polypeptide tetrahedron assembled from coiled-coil segments. *Nature chemical biology*, 9(6):362–366, 2013. https://doi.org/10.1038/nchembio.1248.

[4] V. Potapov, J. B. Kaplan, and A. E. Keating. Data-driven prediction and design of bzip coiled-coil interactions. *PLoS Comput Biol*, 11(2):1–28, 02 2015. https://doi.org/10.1371/journal.pcbi.1004046.

[5] P. W. Rothemund. Folding DNA to create nanoscale shapes and patterns. *Nature*, 440(7082):297–302, 2006. https://doi.org/10.1038/nature04586.

[6] P. San Segundo, F. Matia, D. Rodriguez-Losada, and M. Hernando. An improved bit parallel exact maximum clique algorithm. *Optimization Letters*, pages 1–13, 2013. https://doi.org/10.1007/s11590-011-0431-y.

[7] E. Tomita, Y. Sutani, T. Higashi, S. Takahashi, and M. Wakatsuki. A simple and faster branch-and-bound algorithm for finding a maximum clique. In *International Workshop on Algorithms and Computation*, pages 191–203. Springer, 2010. https://doi.org/10.1007/978-3-642-11440-3_18.

# On Embedding Degree Sequences

Béla Csaba
Bolyai Institute, Interdisciplinary Excellence Centre, University of Szeged, Hungary
E-mail: csaba@math.u-szeged.hu

Bálint Vásárhelyi
University of Szeged, Hungary
E-mail: mesti@math.u-szeged.hu

**Keywords:** graph theory, degree sequence, embedding

**Received:** October 30, 2018

*Assume that we are given two graphic sequences, $\pi_1$ and $\pi_2$. We consider conditions for $\pi_1$ and $\pi_2$ which guarantee that there exists a simple graph $G_2$ realizing $\pi_2$ such that $G_2$ is the subgraph of any simple graph $G_1$ that realizes $\pi_1$.*

*Povzetek: Recimo, da imamo grafni zaporedji $\pi_1$ in $\pi_2$. V prispevku preučujemo pogoje za zaporedji, ki zagotavljajo, da obstaja preprost graf $G_2$, ki je realizacija $\pi_2$ in je podgraf grafa $G_1$, ki je realizacija zaporedja $\pi_2$.*

## 1 Introduction

All graphs considered in this paper are simple. We use standard graph theory notation, see for example [16]. Let us provide a short list of a few perhaps not so common notions, notations. Given a bipartite graph $G(A, B)$ we call it *balanced* if $|A| = |B|$. This notion naturally generalizes for $r$-partite graphs with $r \in \mathbb{N}$, $r \geq 2$.

If $S \subset V$ for some graph $G = (V, E)$ then the subgraph spanned by $S$ is denoted by $G[S]$. Moreover, let $Q \subset V$ so that $S \cap Q = \emptyset$, then $G[S, Q]$ denotes the bipartite subgraph of $G$ on vertex classes $S$ and $Q$, having every edge of $G$ that connects a vertex of $S$ with a vertex of $Q$. The number of vertices in $G$ is denoted by $v(G)$, the number of its edges is denoted by $e(G)$. The degree of a vertex $x \in V(G)$ is denoted by $deg_G(x)$, or if $G$ is clear from the context, by $deg(x)$. The number of neighbors of $x$ in a subset $S \subset V(G)$ is denoted by $deg_G(x, S)$, and $\delta(G)$ and $\Delta(G)$ denote the minimum and maximum degree of $G$, respectively. The complete graph on $n$ vertices is denoted by $K_n$, the complete bipartite graph with vertex class sizes $n$ and $m$ is denoted by $K_{n,m}$.

A finite sequence of natural numbers $\pi = (d_1, \ldots, d_n)$ is a *graphic sequence* or *degree sequence* if there exists a graph $G$ such that $\pi$ is the (not necessarily) monotone degree sequence of $G$. Such a graph $G$ *realizes* $\pi$. For example, the degree sequence $\pi = (2, 2, \ldots, 2)$ can be realized only by vertex-disjoint union of cycles.

The largest value of $\pi$ is denoted by $\Delta(\pi)$. Often the positions of $\pi$ will be identified with the elements of a vertex set $V$. In this case, we write $\pi(v)$ $(v \in V)$ for the corresponding component of $\pi$.

The degree sequence $\pi = (a_1, \ldots, a_k; b_1, \ldots, b_l)$ is a *bigraphic* sequence if there exists a simple bipartite graph $G = G(A, B)$ with $|A| = k$, $|B| = l$ realizing $\pi$ such that the degrees of vertices in $A$ are $a_1, \ldots, a_k$, and the degrees of the vertices of $B$ are $b_1, \ldots, b_l$.

Let $G$ and $H$ be two graphs on $n$ vertices. We say that $H$ is a subgraph of $G$, if we can delete edges from $G$ so that we obtain an isomorphic copy of $H$. We denote this relation by $H \subset G$. In the literature the equivalent complementary formulation can be found as well: we say that $H$ and $\overline{G}$ *pack* if there exist edge-disjoint copies of $H$ and $\overline{G}$ in $K_n$. Here $\overline{G}$ denotes the *complement* of $G$.

It is an old an well-understood problem in graph theory to tell whether a given sequence of natural numbers is a degree sequence or not. We consider a generalization of it, which is remotely related to the so-called discrete tomography[1] (or degree sequence packing) problem (see e.g. [5]) as well. The question whether a sequence of $n$ numbers $\pi$ is a degree sequence can also be formulated as follows: Does $K_n$ have a subgraph $H$ such that the degree sequence of $H$ is $\pi$? The question becomes more general if $K_n$ is replaced by some (simple) graph $G$ on $n$ vertices. If the answer is yes, we say that $\pi$ *can be embedded into* $G$, or equivalently, $\pi$ packs with $\overline{G}$.

Let us mention two classical results in extremal graph theory.

**Theorem 1** (Dirac, [6]). *Every graph $G$ with $n \geq 3$ vertices and minimum degree $\delta(G) \geq \frac{n}{2}$ has a Hamilton cycle.*

[1] In the discrete tomography problem we are given two degree sequences of length $n$, $\pi_1$ and $\pi_2$, and the questions is whether there exists a graph $G$ on $n$ vertices with a red-blue edge coloration so that the following holds: for every vertex $v$ the red degree of $v$ is $\pi_1(v)$ and the blue degree of $v$ is $\pi_2(v)$.

**Theorem 2** (Corrádi-Hajnal, [3]). *Let $k \geq 1$, $n \geq 3k$, and let $H$ be an $n$-vertex graph with $\delta(H) \geq 2k$. Then $H$ contains $k$ vertex-disjoint cycles.*

Observe, that Dirac's theorem implies that given a constant 2 degree sequence $\pi$ of length $n$ and any graph $G$ on $n$ vertices having minimum degree $\delta(G) \geq n/2$, $\pi$ can be embedded into $G$. One can interprete the Corrádi-Hajnal theorem similarly, but here one may require more on the structure of the graph that realizes $\pi$ and in exchange a larger minimum degree of $G$ is needed.

One of our main results is the following.

**Theorem 3.** *For every $\eta > 0$ and $D \in \mathbb{N}$ there exists an $n_0 = n_0(\eta, D)$ such that for all $n > n_0$ if $G$ is a graph on $n$ vertices with $\delta(G) \geq \left(\frac{1}{2} + \eta\right) n$ and $\pi$ is a degree sequence of length $n$ with $\Delta(\pi) \leq D$, then $\pi$ is embeddable into $G$.*

It is easy to see that Theorem 3 is sharp up to the $\eta n$ additive term. For that let $n$ be an even number, and suppose that every element of $\pi$ is 1. Then the only graph that realizes $\pi$ is the union of $n/2$ vertex disjoint edges. Let $G = K_{n/2-1,n/2+1}$ be the complete bipartite graph with vertex class sizes $n/2 - 1$ and $n/2 + 1$. Clearly $G$ does not have $n/2$ vertex disjoint edges.

In order to state the other main result of the paper we introduce a new notion.

**Definition 4.** *Let $q \geq 1$ be an integer. A bipartite graph $H$ with vertex classes $S$ and $T$ is $q$-unbalanced, if $q|S| \leq |T|$. The degree sequence $\pi$ is $q$-unbalanced, if it can be realized by a $q$-unbalanced bipartite graph.*

**Theorem 5.** *Let $q \geq 1$ be an integer. For every $\eta > 0$ and $D \in \mathbb{N}$ there exist an $n_0 = n_0(\eta, q)$ and an $M = M(\eta, D, q)$ such that if $n \geq n_0$, $\pi$ is a $q$-unbalanced degree sequence of length $n - M$ with $\Delta(\pi) \leq D$, $G$ is a graph on $n$ vertices with $\delta(G) \geq (\frac{1}{q+1} + \eta)n$, then $\pi$ can be embedded into $G$.*

Hence, if $\pi$ is unbalanced, the minimum degree requirement of Theorem 3 can be substantially decreased, what we pay for this is that the length of $\pi$ has to be slightly smaller than the number of vertices in the host graph.

## 2 Proof of Theorem 3

*Proof.* First, we find a suitable realization $H$ of $\pi$, our $H$ will consists of components of bounded size. Second, we embed $H$ into $G$ using a theorem by Chvátal and Szemerédi and a result on embedding so called well-separable graphs. The details are as follows.

We construct $H$ in several steps. At the beginning, let $H$ be the empty graph and let all degrees in $\pi$ be *active*. While we can find $2i$ active degrees of $\pi$ with value $i$ (for some $1 \leq i \leq \Delta(\pi)$) we realize them with a $K_{i,i}$ (that is, we add this complete bipartite graph to $H$, and the $2i$ degrees are "inactivated"). When we stop we have at most $\sum_{i=1}^{\Delta(\pi)}(2i-1)$ active degrees. This way we obtain several components, each being a balanced complete bipartite graph. These are the *type 1 gadgets*. Observe that if a vertex $v$ belongs to some type 1 gadget, then its degree is exactly $\pi(v)$. Observe further that if there are no active degrees in $\pi$ at this point then the graph $H$ we have just found is a realization of $\pi$.

Assume that there are active degrees left in $\pi$. Let $R = R_{odd} \cup R_{even}$ be the vertex set that is identified with the active vertices ($v \in R_{odd}$ if and only if the assigned active degree is odd). Since $\sum\limits_{v \in R} d(v)$ must be an even number we have that $|R_{odd}|$ is even. Add a perfect matching on $R_{odd}$ to $H$. With this we achieved that every vertex of $R$ misses an even number of edges.

Next we construct the *type 2 gadgets* using the following algorithm. In the beginning every type 1 gadget is *unmarked*. Suppose that $v \in R$ is an active vertex. Take a type 1 gadget $K$, *mark* it, and let $M_K$ denote an arbitrarily chosen perfect matching in $K$ ($M_K$ exists since $K$ is a balanced complete bipartite graph). Let $xy$ be an arbitrary edge in $M_K$. Delete the $xy$ edge and add the new edges $vx$ and $vy$. While $v$ is missing edges repeat the above procedure with edges of $M_K$, until $M_K$ becomes empty. If $M_K$ becomes empty, take a new unmarked type 1 gadget $L$, and repeat the method with $L$. It is easy to see that in $\pi(v)/2$ steps $v$ reaches its desired degree and gets inactivated. Clearly, the degrees of vertices in the marked type 1 gadgets have not changed.

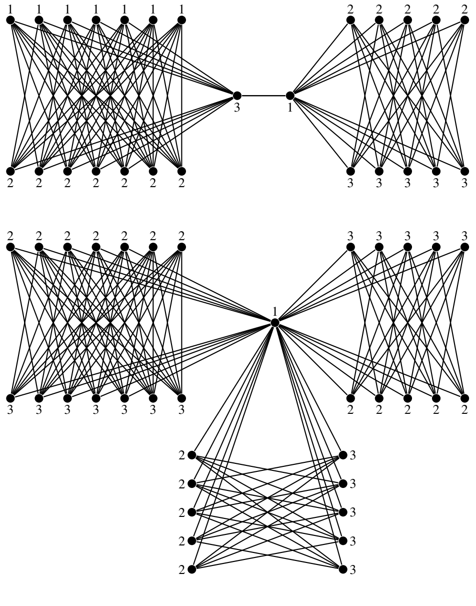


Figure 1: Type 2 gadgets of $H$ with a 3-coloring

Figure 1 shows examples of type 2 gadgets. In the upper one two vertices of $R_{odd}$ were first connected by an edge and then two type 1 gadgets were used so that they could reach their desired degree, while in the lower one we used three type 1 gadgets for a vertex of $R$. The numbers at the vertices indicate the colors in the 3-coloring of $H$.

Let $A \subset V(H)$ denote the set of vertices containing the union of all type 2 gadgets. Observe that type 2 gadgets are 3-colorable and all have less than $5\Delta^2(\pi)$ vertices. Let us summarize our knowledge about $H$ for later reference.

**Claim 6.** *(1)* $|A| \leq 5\Delta^3(\pi)$,

*(2) the components of* $H[V-A]$ *are balanced complete bipartite graphs, each having size at most* $2\Delta(\pi)$,

*(3)* $\chi(H[A]) \leq 3$, *and*

*(4)* $e(H[A, V-A]) = 0$.

We are going to show that $H \subset G$. For that we first embed the possibly 3-chromatic part $H[A]$ using the following strengthening of the Erdős–Stone theorem proved by Chvátal and Szemerédi [2].

**Theorem 7.** *Let* $\varphi > 0$ *and assume that* $G$ *is a graph on* $n$ *vertices where* $n$ *is sufficiently large. Let* $r \in \mathbb{N}$, $r \geq 2$. *If*

$$e(G) \geq \left(\frac{r-2}{2(r-1)} + \varphi\right) n^2,$$

*then* $G$ *contains a* $K_r(t)$, *i.e. a complete* $r$*-partite graph with* $t$ *vertices in each class, such that*

$$t > \frac{\log n}{500 \log \frac{1}{\varphi}}. \tag{1}$$

Since $\delta(G) \geq (1/2+\eta)n$, the conditions of Theorem 7 are satisfied with $r = 3$ and $\varphi = \eta/2$, hence, $G$ contains a balanced complete tripartite subgraph $T$ on $\Omega(\log n)$ vertices. Using Claim 6 and the 3-colorability of $H[A]$ this implies that $H[A] \subset T$.

Observe that after embedding $H[A]$ into $G$ every uncovered vertex of $G$ still has at least $\delta(G) - v(F) > (1/2 + \eta/2)n$ uncovered neighbors. Denoting the subgraph of the uncovered vertices of $G$ by $G'$ we obtain that $\delta(G') > (1/2 + \eta/2)n$.

In order to prove that $H[V-A] \subset G'$ we first need a definition.

**Definition 8.** *A graph* $F$ *on* $n$ *vertices is* well-separable *if it has a subset* $S \subset V(F)$ *of size* $o(n)$ *such that all components of* $F - S$ *are of size* $o(n)$.

We need the following theorem.

**Theorem 9** ([4]). *For every* $\gamma > 0$ *and positive integer* $D$ *there exists an* $n_0$ *such that for all* $n > n_0$ *if* $F$ *is a bipartite well-separable graph on* $n$ *vertices,* $\Delta(F) \leq D$ *and* $\delta(G) \geq \left(\frac{1}{2} + \gamma\right) n$ *for a graph* $G$ *of order* $n$, *then* $F \subset G$.

Since $H[V-A]$ has bounded size components by Claim 6, we can apply Theorem 9 for $H[V-A]$ and $G'$, with parameter $\gamma = \eta/2$. With this we finished proving what was desired.

□

# 3 Further tools for Theorem 5

When proving Theorem 3, we used the Regularity Lemma of Szemerédi, but implicitly, via the result on embedding well-separable graphs. When proving Theorem 5 we will apply this very powerful result explicitly, hence, below we give a very brief introduction to the area. The interested reader may consult with the original paper by Szemerédi [15] or e.g. with the survey paper [10].

## 3.1 Regularity lemma

The *density* between disjoint sets $X$ and $Y$ is defined as:

$$d(X,Y) = \frac{e(X,Y)}{|X||Y|}.$$

We will need the following definition to state the Regularity Lemma.

**Definition 10** (Regularity condition)**.** *Let* $\varepsilon > 0$. *A pair* $(A, B)$ *of disjoint vertex-sets in* $G$ *is* $\varepsilon$*-regular if for every* $X \subset A$ *and* $Y \subset B$, *satisfying*

$$|X| > \varepsilon|A|, \ |Y| > \varepsilon|B|$$

*we have*

$$|d(X,Y) - d(A,B)| < \varepsilon.$$

This definition implies that regular pairs are highly uniform bipartite graphs; namely, the density of any reasonably large subgraph is almost the same as the density of the regular pair.

We will use the following form of the Regularity Lemma:

**Lemma 11** (Degree Form)**.** *For every* $\varepsilon > 0$ *there is an* $M = M(\varepsilon)$ *such that if* $G = (W, E)$ *is any graph and* $d \in [0,1]$ *is any real number, then there is a partition of the vertex set* $V$ *into* $\ell + 1$ *clusters* $W_0, W_1, \ldots, W_\ell$, *and there is a subgraph* $G'$ *of* $G$ *with the following properties:*

- $\ell \leq M$,
- $|W_0| \leq \varepsilon|W|$,
- *all clusters* $W_i$, $i \geq 1$, *are of the same size* $m \left(\leq \left\lfloor \frac{|W|}{\ell} \right\rfloor < \varepsilon|W|\right)$,
- $deg_{G'}(v) > deg_G(v) - (d+\varepsilon)|W|$ *for all* $v \in W$,
- $G'|_{W_i} = \emptyset$ *(*$W_i$ *is an independent set in* $G'$*) for all* $i \geq 1$,
- *all pairs* $(W_i, W_j)$, $1 \leq i < j \leq \ell$, *are* $\varepsilon$*-regular, each with density either 0 or greater than* $d$ *in* $G'$.

We call $W_0$ the *exceptional cluster*, $W_1, \ldots, W_\ell$ are the *non-exceptional clusters*. In the rest of the paper we will assume that $0 < \varepsilon \ll d \ll 1$. Here $a \ll b$ means that $a$ *is sufficiently smaller than b.*

**Definition 12** (Reduced graph)**.** *Apply Lemma 11 to the graph $G = (W, E)$ with parameters $\varepsilon$ and d, and denote the clusters of the resulting partition by $W_0, W_1, \ldots, W_\ell$ ($W_0$ being the exceptional cluster). We construct a new graph $G_r$, the reduced graph of $G'$ in the following way: The non-exceptional clusters of $G'$ are the vertices of the reduced graph $G_r$ (hence $v(G_r) = \ell$). We connect two vertices of $G_r$ by an edge if the corresponding two clusters form an $\varepsilon$-regular pair with density at least d.*

The following corollary is immediate:

**Corollary 13.** *Apply Lemma 11 with parameters $\varepsilon$ and d to the graph $G = (W, E)$ satisfying $\delta(G) \geq \gamma n$ $(v(G) = n)$ for some $\gamma > 0$. Denote $G_r$ the reduced graph of $G'$. Then $\delta(G_r) \geq (\gamma - \theta)\ell$, where $\theta = 2\varepsilon + d$.*

The (fairly easy) proof of the lemma below can be found in [10].

**Lemma 14.** *Let $(A, B)$ be an $\varepsilon$-regular–pair with density d for some $\epsilon > 0$. Let $c > 0$ be a constant such that $\varepsilon \ll c$. We arbitrarily divide A and B into two parts, obtaining the non-empty subsets $A', A''$ and $B', B''$, respectively. Assume that $|A'|, |A''| \geq c|A|$ and $|B'|, |B''| \geq c|B|$. Then the pairs $(A', B')$, $(A', B'')$, $(A'', B')$ and $(A'', B'')$ are all $\varepsilon/c$–regular pairs with density at least $d - \varepsilon/c$.*

## 3.2 Blow-up lemma

Let $H$ and $G$ be two graphs on $n$ vertices. Assume that we want to find an isomorphic copy of $H$ in $G$. In order to achieve this one can apply a very powerful tool, the Blow-up Lemma of Komlós, Sárközy and Szemerédi [8, 9]. For stating it we need a new notion, a stronger one-sided property of regular pairs.

**Definition 15** (Super-Regularity condition)**.** *Given a graph G and two disjoint subsets of its vertices A and B, the pair $(A, B)$ is $(\varepsilon, \delta)$-*super-regular*, if it is $\varepsilon$-regular and furthermore,*

$$deg(a) > \delta|B|, \text{ for all } a \in A,$$

*and*

$$deg(b) > \delta|A|, \text{ for all } b \in B.$$

**Theorem 16** (Blow-up Lemma)**.** *Given a graph R of order r and positive integers $\delta, \Delta$, there exists a positive $\varepsilon = \varepsilon(\delta, \Delta, r)$ such that the following holds: Let $n_1, n_2, \ldots, n_r$ be arbitrary positive parameters and let us replace the vertices $v_1, v_2, \ldots, v_r$ of R with pairwise disjoint sets $W_1, W_2, \ldots, W_r$ of sizes $n_1, n_2, \ldots, n_r$ (blowing up R). We construct two graphs on the same vertex set $V = \cup_i W_i$. The first graph F is obtained by replacing each edge $v_i v_j \in E(R)$ with the complete bipartite graph between $W_i$ and $W_j$. A sparser graph G is constructed by replacing each edge $v_i v_j$ arbitrarily with an $(\varepsilon, \delta)$-super-regular pair between $W_i$ and $W_j$. If a graph H with $\Delta(H) \leq \Delta$ is embeddable into F then it is already embeddable into G.*

# 4 Proof of Theorem 5

Let us give a brief sketch first. Recall, that $\pi$ is a $q$-unbalanced and bounded degree sequence with $\Delta(\pi) \leq D$. In the proof we first show that there exists a $q$-unbalanced bipartite graph $H$ that realizes $\pi$ such that $H$ is the vertex disjoint union of the graphs $H_1, \ldots, H_k$, where each $H_i$ graph is a bipartite $q$-unbalanced graph having bounded size. We will apply the Regularity lemma to $G$, and find a special substructure (a decomposition into vertex-disjoint stars) in the reduced graph of $G$. This substructure can then be used to embed the union of the $H_i$ graphs, for the majority of them we use the Blow-up lemma.

## 4.1 Finding $H$

The goal of this subsection is to prove the lemma below.

**Lemma 17.** *Let $\pi$ be a q-unbalanced degree sequence of positive integers with $\Delta(\pi) \leq D$. Then $\pi$ can be realized by a q-unbalanced bipartite graph H which is the vertex disjoint union of the graphs $H_1, \ldots, H_k$, such that for every i we have that $H_i$ is q-unbalanced, moreover, $v(H_i) \leq 4D^2$.*

Before starting the proof of Lemma 17, we list a few necessary notions and results.

We call a finite sequence of integers a *zero-sum sequence* if the sum of its elements is zero. The following result of Sahs, Sissokho and Torf plays an important role in the proof of Lemma 17.

**Proposition 18.** *[14] Assume that K is a positive integer. Then any zero-sum sequence on $\{-K, \ldots, K\}$ having length at least $2K$ contains a proper nonempty zero-sum subsequence.*

The following result, formulated by Gale [7] and Ryser [13], will also be useful. We present it in the form as discussed in Lovász [11].

**Lemma 19.** *[11] Let $G = (A, B; E(G))$ be a bipartite graph and f be a nonnegative integer function on $A \cup B$ with $f(A) = f(B)$. Then G has a subgraph $F = (A, B; E(F))$ such that $d_F(x) = f(x)$ for all $x \in A \cup B$ if and only if*

$$f(X) \leq e(X, Y) + f(\overline{Y}) \tag{2}$$

*for any $X \subseteq A$ and $Y \subseteq B$, where $\overline{Y} = B - Y$.*

We remark that such a subgraph $F$ is also called an $f$-factor of $G$.

**Lemma 20.** *If $f = (a_1, \dots, a_s; b_1, \dots, b_t)$ is a sequence of positive integers with $s, t \geq 2\Delta^2$, where $\Delta$ is the maximum of $f$, and $f(A) = f(B)$ with $A = \{a_1, \dots, a_s\}$ and $B = \{b_1, \dots, b_t\}$ then $f$ is bigraphic.*

*Proof.* All we have to check is whether the conditions of Lemma 19 are met if $G = K_{s,t}$.

Suppose indirectly that there is an $(X, Y)$ pair for which (2) does not hold. Choose such a pair with minimal $|X| + |Y|$. Then $X = \emptyset$ or $Y = \emptyset$ are impossible, as in those cases (2) trivially holds. Hence, $|X|, |Y| \geq 1$. Assuming that (2) does not hold, we have that

$$f(X) \geq e(X,Y) + f(\overline{Y}) + 1, \tag{3}$$

which is equivalent to

$$f(X) \geq |X||Y| + f(\overline{Y}) + 1, \tag{4}$$

as $G$ is a complete bipartite graph. Furthermore, using the minimality of $|X| + |Y|$, we know that

$$f(X - a) \leq |X - a||Y| + f(\overline{Y}) \tag{5}$$

for any $a \in X$. (5) is equivalent to

$$f(X) - f(a) \leq |X||Y| - |Y| + f(\overline{Y}). \tag{6}$$

From (4) and (6) we have

$$f(a) - 1 \geq |Y| \tag{7}$$

for any $a \in X$, which implies

$$\Delta > |Y|. \tag{8}$$

The same reasoning also implies that $\Delta > |X|$ whenever $(X, Y)$ is a counterexample. Therefore we only have to verify that (2) holds in case $|X| < \Delta$ and $|Y| < \Delta$. Recall that $f(B) \geq t$, as all elements of $f$ are positive. Hence, $f(X) \leq \Delta|X| \leq \Delta^2$, and $f(\overline{Y}) = f(B) - f(Y) \geq t - \Delta^2$, and we get that

$$f(X) \leq \Delta^2 \leq t - \Delta^2 \leq f(\overline{Y}) \leq f(\overline{Y}) + e_G(X,Y) \tag{9}$$

holds, since $t \geq 2\Delta^2$. □

*Proof.* (Lemma 17) Assume that $J = \big(S, T; E(J)\big)$ is a $q$-unbalanced bipartite graph realizing $\pi$. Hence, $q|S| \leq |T|$. Moreover, $|T| \leq D|S|$, since $\Delta(\pi) \leq D$. We form vertex disjoint tuples of the form $(s; t_1, \dots, t_h)$, such that $s \in S$, $t_i \in T$, $q \leq h \leq D$, and the collection of these tuples contains every vertex of $S \cup T$ exactly once. We define the *bias* of the tuple as

$$\zeta = \pi(t_1) + \dots + \pi(t_h) - \pi(s).$$

Obviously, $-D \leq \zeta \leq D^2$. The conditions of Proposition 18 are clearly met with $K = D^2$. Hence, we can form groups of size at most $2D^2$ in which the sums of biases are zero. This way we obtain a partition of $(S, T)$ into $q$-unbalanced set pairs which have zero bias. While these sets may be small, we can combine them so that each combined set is of size at least $2D^2$ and has zero bias. By Lemma 20 these are bigraphic sequences. The realizations of these small sequences give the graphs $H_1, \dots, H_k$. It is easy to see that $v(H_i) \leq 4D^2$ for every $1 \leq i \leq k$. Finally, we let $H = \cup_i H_i$. □

## 4.2 Decomposing $G_r$

Let us apply the Regularity lemma with parameters $0 < \varepsilon \ll d \ll \eta$. By Corollary 13 we have that $\delta(G_r) \geq \ell/(q+1) + \eta\ell/2$.

Let $h \geq 1$ be an integer. An $h$-star is a $K_{1,h}$. The *center* of an $h$-star is the vertex of degree $h$, the other vertices are the *leaves*. In case $h = 1$ we pick one of the vertices of the 1-star arbitrarily to be the center.

**Lemma 21.** *The reduced graph $G_r$ has a decomposition $\mathcal{S}$ into vertex disjoint stars such that each star has at most $q$ leaves.*

*Proof.* Take a partial star-decomposition of $G_r$ that is as large as possible. Assume that there are uncovered vertices in $G_r$. Let $U$ denote those vertices that are covered (so we assume that $U$ has maximal cardinality), and let $v$ be an uncovered vertex. Observe that $v$ has neighbours only in $U$, otherwise, if $uv \in E(G_r)$ with $u \notin U$, then we can simply add $uv$ to the star-decomposition, contradicting to the maximality of $U$. See Figure 2 for the possible neighbors of $v$.

a) If $v$ is connected to a 1-star, then we can replace it with a 2-star.

b) If $v$ is connected to the center $u$ of an $h$-star, where $h < q$, then we can replace this star with an $h + 1$-star by adding the edge $uv$ to the $h$-star.

c) If $v$ is connected to a leaf $u$ of an $h$-star, where $2 \leq h \leq q$, then replace the star with the edge $uv$ and an $(h - 1)$-star (i.e., delete $u$ from it).

We have not yet considered one more case: when $v$ is connected to the center of a $q$-star. However, simple calculation shows that for every vertex $v$ at least one of the above three cases must hold, using the minimum degree condition of $G_r$. Hence we can increase the number of covered vertices. We arrived at a contradiction, $G_r$ has the desired star-decomposition. □

## 4.3 Preparing $G$ for the embedding

Consider the $q$-star-decomposition $\mathcal{S}$ of $G_R$ as in Lemma 21. Let $\ell_i$ denote the number of $(i - 1)$-stars in

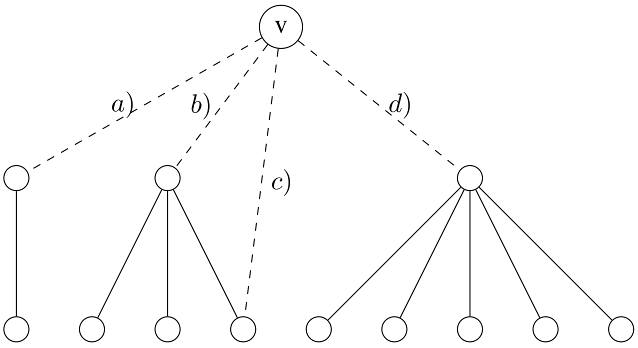


Figure 2: An illustration for Lemma 21

the decomposition for every $2 \le i \le q+1$. It is easy to see that

$$\sum_{i=2}^{q+1} i\ell_i = \ell.$$

First we will make every $\varepsilon$-regular pair in $\mathcal{S}$ super-regular by discarding a few vertices from the non-exceptional clusters. Let for example $\mathcal{C}$ be a star in the decomposition of $G_r$ with center cluster $A$ and leaves $B_1, \dots, B_k$, where $1 \le k \le q$. Recall that the $(A, B_i)$ pairs has density at least $d$. We repeat the following for every $1 \le i \le k$ : if $v \in A$ such that $v$ has at most $2dm/3$ neighbors in $B_i$ then discard $v$ from $A$, put it into $W_0$. Similarly, if $w \in B_i$ has at most $2dm/3$ neighbors in $A$, then discard $w$ from $B_i$, put it into $W_0$. Repeat this process for every star in $\mathcal{S}$. We have the following:

**Claim 22.** *We do not discard more than $q\varepsilon m$ vertices from any non-exceptional cluster.*

*Proof.* Given a star $\mathcal{C}$ in the decomposition $\mathcal{S}$ assume that its center cluster is $A$ and let $B$ be one of its leaves. Since the pair $(A, B)$ is $\varepsilon$-regular with density at least $d$, neither $A$, nor $B$ can have more than $\varepsilon m$ vertices that have at most $2dm/3$ neighbors in the opposite cluster. Hence, during the above process we may discard up to $q\varepsilon m$ vertices from $A$. Next, we may discard vertices from the leaves, but since no leaf $B$ had more than $\varepsilon m$ vertices with less than $(d-\varepsilon)m$ neighbors in $A$, even after discarding at most $q\varepsilon m$ vertices of $A$, there can be at most $\varepsilon m$ vertices in $B$ that have less than $(d-(q+1)\varepsilon)m$ neighbors in $A$. Using that $\varepsilon \ll d$, we have that $(d-(q+1)\varepsilon) > 2d/3$. We obtained what was desired. □

By the above claim we can make every $\varepsilon$-regular pair in $\mathcal{S}$ a $(2\varepsilon, 2d/3)$-super-regular pair so that we discard only relatively few vertices. Notice that we only have an upper bound for the number of discarded vertices, there can be clusters from which we have not put any points into $W_0$. We repeat the following for every non-exceptional cluster: if $s$ vertices were discarded from it with $s < q\varepsilon m$ then we take $q\varepsilon m - s$ arbitrary vertices of it, and place them into $W_0$. This way every non-exceptional cluster will have the same number of points, precisely $m - q\varepsilon m$. For simpler notation, we will use the letter $m$ for this new cluster size. Observe that $W_0$ has increased by $q\varepsilon m\ell$ vertices, but we still have $|W_0| \le 3dn$ since $\varepsilon \ll d$ and $\ell m \le n$. Since $q\varepsilon m \ll d$, in the resulting pairs the minimum degree will be at least $dm/2$.

Summarizing, we obtained the following:

**Lemma 23.** *By discarding a total of at most $q\varepsilon n$ vertices from the non-exceptional clusters we get that every edge in $\mathcal{S}$ represents a $(2\varepsilon, d/2)$-super-regular pair, and all non-exceptional clusters have the same cardinality, which is denoted by $m$. Moreover, $|W_0| \le 3dn$.*

Since $v(G) - v(H)$ is bounded above by a constant, when embedding $H$ we need almost every vertex of $G$, in particular those in the exceptional cluster $W_0$. For this reason we will assign the vertices of $W_0$ to the stars in $\mathcal{S}$. This is not done in an arbitrary way.

**Definition 24.** *Let $v \in W_0$ be a vertex and $(Q, T)$ be an $\varepsilon$-regular pair. We say that $v \in T$* has large degree *to $Q$ if $v$ has at least $\eta|Q|/4$ neighbors in $Q$. Let $S = (A, B_1, \dots, B_k)$ be a star in $\mathcal{S}$ where $A$ is the center of $S$ and $B_1, \dots, B_k$ are the leaves, here $1 \le k \le q$. If $v$ has large degree to any of $B_1, \dots, B_k$, then $v$* can be assigned *to $A$. If $k < q$ and $v$ has large degree to $A$, then $v$* can be assigned *to any of the $B_i$ leaves.*

**Observation 25.** *If we assign new vertices to a $q$-star, then we necessarily assign them to the center. Since before assigning, the number of vertices in the leaf-clusters is exactly $q$ times the number of vertices in the center cluster, after assigning new vertices to the star, $q$ times the cardinality of the center will be larger than the total number of vertices in the leaf-clusters. If $S \in \mathcal{S}$ is a $k$-star with $1 \le k < q$, and we assign up to $cm$ vertices to any of its clusters, where $0 < c \ll 1$, then even after assigning new vertices we will have that $q$ times the cardinality of the center is larger than the total number of vertices in the leaf-clusters.*

The following lemma plays a crucial role in the embedding algorithm.

**Lemma 26.** *Every vertex of $W_0$ can be assigned to at least $\eta\ell/4$ non-exceptional clusters.*

*Proof.* Suppose that there exists a vertex $w \in W_0$ that can be assigned to less than $\eta\ell/4$ clusters. If $w$ cannot be assigned to any cluster of some $k$-star $S_k$ with $k < q$, then the total degree of $w$ into the clusters of $S_k$ is at most $k\eta m/4$. If $w$ cannot be assigned to any cluster of some $q$-star $S_q$, then the total degree of $w$ into the clusters of $S_q$ is at most $m + q\eta m/4$, since every vertex of the center cluster could be adjacent to $w$. Considering that $w$ can be assigned to at most $\eta\ell/4 - 1$ clusters and that $d(w, W - W_0) \ge n/(q+1) + \eta n/2$, we obtain the following inequality:

$$\frac{n}{q+1}+\frac{\eta n}{2}\le d(v,W-W_0)\le \eta\frac{\ell m}{4}+ \sum_{k=1}^{q-1}(k+1)\eta\frac{\ell_{k+1}m}{4}+q\eta\frac{\ell_{q+1}m}{4}+\ell_{q+1}m.$$

Using $m\ell\le n$ and $\sum_{k=1}^{q}(k+1)\ell_{k+1}=\ell$, we get

$$\frac{m\ell}{q+1}+\frac{\eta m\ell}{2}\le \eta\frac{\ell m}{4}+(\ell-\ell_{q+1})\frac{\eta m}{4}+ q\eta\frac{\ell_{q+1}m}{4}+\ell_{q+1}m.$$

Dividing both sides by $m$ and cancellations give

$$\frac{\ell}{q+1}\le q\frac{\eta\ell_{q+1}}{4}+(1-\frac{\eta}{4})\ell_{q+1}.$$

Noting that $(q+1)\ell_{q+1}\le\ell$, one can easily see that we arrived at a contradiction. Hence every vertex of $W_0$ can be assigned to several non-exceptional clusters. □

Lemma 26 implies the following:

**Lemma 27.** *One can assign the vertices of $W_0$ so that at most $\sqrt{d}m$ vertices are assigned to non-exceptional clusters.*

*Proof.* Since we have at least $\eta\ell/4$ choices for every vertex, the bound follows from the inequality $\frac{4|W_0|}{\eta\ell}\le\sqrt{d}m$, where we used $d\ll\eta$ and $|W_0|\le 3dn$. □

**Observation 28.** *A key fact is that the number of newly assigned vertices to a cluster is much smaller than their degree into the opposite cluster of the regular pair since $\sqrt{d}m\ll\eta m/4$.*

## 4.4 The embedding algorithm

The embedding is done in two phases. In the first phase we cover every vertex that belonged to $W_0$, together with some other vertices of the non-exceptional clusters. In the second phase we are left with super-regular pairs into which we embed what is left from $H$ using the Blow-up lemma.

### 4.4.1 The first phase

Let $(A,B)$ be an $\varepsilon$-regular cluster-edge in the $h$-star $\mathcal{C}\in\mathcal{S}$. We begin with partitioning $A$ and $B$ randomly, obtaining $A=A'\cup A''$ and $B=B'\cup B''$ with $A'\cap A''=B'\cap B''=\emptyset$. For every $w\in A$ (except those that came from $W_0$) flip a coin. If it is heads, we put $w$ into $A'$, otherwise we put it into $A''$. Similarly, we flip a coin for every $w\in B$ (except those that came from $W_0$) and depending on the outcome, we either put the vertex into $B'$ or into $B''$. The proof of the following lemma is standard, uses Chernoff's bound (see in [1]), we omit it.

**Lemma 29.** *With high probability, that is, with probability at least $1-1/n$, we have the following:*

- $\big||A'|-|A''|\big|=o(n)$ *and* $\big||B'|-|B''|\big|=o(n)$
- *$deg(w,A'),deg(w,A'')>deg(w,A)/3$ for every $w\in B$*
- *$deg(w,B'),deg(w,B'')>deg(w,B)/3$ for every $w\in A$*
- *the density $d(A',B')\ge d/2$*

It is easy to see that Lemma 29 implies that $(A',B')$ is a $(5\varepsilon,d/6)$-super-regular pair having density at least $d/2$ with high probability.

Assume that $v$ was an element of $W_0$ before we assigned it to the cluster $A$, and assume further that $deg(v,B)\ge\eta m/4$. Since $(A,B)$ is an edge of the star-decomposition, either $A$ or $B$ must be the center of $\mathcal{C}$.

Let $H_i$ be one of the $q$-unbalanced bipartite subgraphs of $H$ that has not been embedded yet. We will use $H_i$ to cover $v$. Denote $S_i$ and $T_i$ the vertex classes of $H_i$, where $|S_i|\ge q|T_i|$. Let $S_i=\{x_1,\dots,x_s\}$ and $T_i=\{y_1,\dots,y_t\}$.

If $A$ is the center of $\mathcal{C}$ then the vertices of $T_i$ will cover vertices of $A'$, and the vertices of $S_i$ will cover vertices of $B'$. If $B$ is the center, $S_i$ and $T_i$ will switch roles. The embedding of $H_i$ is essentially identical in both cases, so we will only discuss the case when $A$ is the center.[2]

In order to cover $v$ we will essentially use a well-known method called Key lemma in [10]. We will heavily use the fact that

$$0<\varepsilon\ll d\ll\eta.$$

The details are as follows. We construct an edge-preserving injective mapping $\varphi: S_i\cup T_i\longrightarrow A'\cup B'$. In particular, we will have $\varphi(S_i)\subset B'$ and $\varphi(T_i)-v\subset A'$. First we let $\varphi(y_1)=v$. Set $N_1=N(v)\cap B'$. Using Lemma 29 we have that $|N_1|\ge\eta m/12\gg\varepsilon m$.

Next we find $\varphi(y_2)$. Since $|N_1|\gg\varepsilon m$, by $5\varepsilon$-regularity the majority of the vacant vertices of $A'$ will have at least $d|N_1|/3$ neighbors in $N_1$. Pick any of these, denote it by $v_2$ and let $\varphi(y_2)=v_2$. Also, set $N_2=N_1\cap N(v_2)$.

In general, assume that we have already found the vertices $v_2,v_3,\dots,v_i$, their common neighborhood in $B'$ is $N_i$, and

$$|N_i|\ge\frac{\eta d^{i-1}}{3^{i-2}\cdot 36}m\gg\varepsilon m.$$

By $5\varepsilon$-regularity, this implies that the majority of the vacant vertices of $A'$ has large degree into $N_i$, at least $d\cdot|N_i|/3$, and this, as above, can be used to find $v_{i+1}$. Then we set $\varphi(y_{i+1})=v_{i+1}$. Since $\eta$ and $d$ is large compared to $\varepsilon$, even into the last set $N_{t-1}$ many vacant vertices will have large degrees.

[2] Recall that if $h<q$ then we may assigned $v$ to a leaf, so in such a case $B$ could be the center.

As soon as we have $\varphi(y_1), \dots, \varphi(y_t)$, it is easy to find the images for $x_1, \dots, x_t$. Since $|N_t| \gg \varepsilon m \gg s = |S_i|$, we can arbitrarily choose $s$ vacant points from $N_t$ for the $\varphi(x_j)$ images.

Note that we use less than $v(H_i) \leq 4D^2$ vertices from $A'$ and $B'$ during this process. We can repeat it for every vertex that were assigned to $A$, and still at most $\sqrt{d}2D^2m$ vertices will be covered from $A'$ and from $B'$.

Another observation is that every $h$-star in the decomposition before this embedding phase was $h$-unbalanced, now, since we were careful, these have become $h'$-balanced with $h' \leq h$.

Of course, the above method will be repeated for every $(A, B)$ edge of the decomposition for which we have assigned vertices of $W_0$.

### 4.4.2 The second phase

In the second phase we first unite all the randomly partitioned clusters. For example, assume that after covering the vertices that were coming from $W_0$ the set of vacant vertices of $A'$ is denoted by $A'_v$. Then we let $A_v = A'_v \cup A''$, and using analogous notation, let $B_v = B'_v \cup B''$.

**Claim 30.** *All the $(A_v, B_v)$ pairs are $(3\varepsilon, d/6)$-super-regular with density at least $d/2$.*

*Proof.* The $3\varepsilon$-regularity of these pairs is easy to see, like the lower bound for the density, since we have only covered relatively few vertices of the clusters. For the large minimum degrees note that by Lemma 29 every vertex of $A$ had at least $dm/6$ neighbors in $B''$, hence, in $B_v$ as well, and analogous bound holds for vertices of $B$. □

At this point we want to apply the Blow-up lemma for every star of $\mathcal{S}$ individually. For that we first have to assign those subgraphs of $H$ to stars that were not embedded yet. We need a lemma.

**Lemma 31.** *Let $K_{a,b}$ be a complete bipartite graph with vertex classes $A$ and $B$, where $|A| = a$ and $|B| = b$. Assume that $a \leq b = ha$, where $1 \leq h \leq q$. Let $H'$ be the vertex disjoint union of $q$-unbalanced bipartite graphs:*

$$H' = \bigcup_{j=1}^{t} H_j,$$

*such that $v(H_j) \leq 2D^2$ for every $j$. If $v(H') \leq a + b - 4(2q+1)D^2$, then $H' \subset K_{a,b}$.*

Observe that if we have Lemma 31, we can distribute the $H_i$ subgraphs among the stars of $\mathcal{S}$, and then apply the Blow-up lemma. Hence, we are done with proving Theorem 5 if we prove Lemma 31 above.

*Proof.* The proof is an assigning algorithm and its analysis. We assign the vertex classes of the $H_j$ subgraphs to $A$ and $B$, one-by-one. Before assigning the $j$th subgraph $H_j$ the number of vacant vertices of $A$ is denoted by $a_j$ and the number of vacant vertices of $B$ is denoted by $b_j$.

Assume that we want to assign $H_k$. If $ha_k - b_k > 0$, then the larger vertex class of $H_k$ is assigned to $A$, the smaller is assigned to $B$. Otherwise, if $ha_k - b_k \leq 0$, then we assign the larger vertex class to $B$ and the smaller one to $A$. Then we update the number of vacant vertices of $A$ and $B$. Observe that using this assigning method we always have $a_k \leq b_k$.

The question is whether we have enough room for $H_k$. If $ha \geq 4hD^2$, then we must have enough room, since $b_k \geq a_k$ and every $H_j$ has at most $2D^2$ vertices. Hence, if the algorithm stops, we must have $a_k < 4D^2$. Since $b_k - ha_k \leq 2D^2$ must hold, we have $b_k < (2h+1)2D^2 < (2q+1)2D^2$. From this the lemma follows. □

## 5 Remarks

One can prove a very similar result to Theorem 5, in fact the result below follows easily from it. For stating it we need the notion of graph edit distance which is detailed e.g. in [12]: the edit distance between two graphs on the same labeled vertex set is defined to be the size of the symmetric difference of the edge sets

**Theorem 32.** *Let $q \geq 1$ be an integer. For every $\eta > 0$ and $D \in \mathbb{N}$ there exists an $n_0 = n_0(\eta, q)$ and a $K = K(\eta, D, q)$ such that if $n \geq n_0$, $\pi$ is a $q$-unbalanced degree sequence of length $n$ with $\Delta(\pi) \leq D$, $G$ is a graph on $n$ vertices with $\delta(G) \geq (\frac{1}{q+1} + \eta)n$, then there exists a graph $G'$ on $n$ vertices such that the edit distance of $G$ and $G'$ is at most $K$, and $\pi$ can be embedded into $G'$.*

Here is an example showing that Theorem 5 and 32 are essentially best possible.

**Example 33.** *Assume that $\pi$ has only odd numbers and $G$ has at least one odd sized component. Then the embedding is impossible. Indeed, any realization of $\pi$ has only even sized components, hence, $G$ cannot contain it as a spanning subgraph.*

Note that this example does not work in case $G$ is connected. In Theorem 3 the minimum degree $\delta(G) \geq (1/2 + \eta)n$, hence, $G$ is connected, and in this case we can embed $\pi$ into $G$.

## Acknowledgement

Partially supported by Ministry of Human Capacities, Hungary, grant 20391-3/2018/FEKUSTRAT, by ERC-AdG. 321400 and by the National Research, Development and Innovation Office - NKFIH Fund No. SNN-117879 and KH 129597.

Supported by TÁMOP-4.2.2.B-15/1/KONV-2015-0006.

# References

[1] N. Alon, J. Spencer. *The probabilistic method.* Third edition, John Wiley & Sons, Inc., 2008. https://doi.org/10.1002/9780470277331

[2] V. Chvátal and E. Szemerédi. On the Erdős–Stone Theorem. *Journal of the London Mathematical Society*, s2-23 (2):207–214, 1981.

[3] K. Corrádi and A. Hajnal. On the maximal number of independent circuits in a graph. *Acta Mathematica Academiae Scientiarum Hungaricae*, 14:423–439, 1963. https://doi.org/10.1007/BF01895727

[4] B. Csaba. On embedding well-separable graphs. *Discrete Mathematics*, 308(19):4322–4331, 2008. https://doi.org/10.1016/j.disc.2007.08.015

[5] J. Diemunsch, M. Ferrara, S. Jahanbekam, and J. Shook. Extremal theorems for degree sequence packing and the 2-color discrete tomography problem. *SIAM Journal of Discrete Mathematics*, 29(4):2088–2099, 2015. https://doi.org/10.1137/140987912

[6] G. Dirac. Some theorems on abstract graphs. *Proceedings of the London Mathematical Society*, s3-2(1):69–81, 1952. https://doi.org/10.1112/plms/s3-2.1.69

[7] D. Gale. A theorem on flows in networks. *Pacific Journal of Mathematics*, 7(2):1073–1082, 1957. https://doi.org/10.2140/pjm.1957.7.1073

[8] J. Komlós, G. Sárközy, and E. Szemerédi. Blow-up lemma. *Combinatorica*, 17:109–123, 1997. https://doi.org/10.1007/BF01196135

[9] J. Komlós, G. Sárközy, and E. Szemerédi. An algorithmic version of the blow-up lemma. *Random Structures and Algorithms*, 12:297–312, 1998. https://doi.org/10.1002/(SICI)1098-2418(199805)12:3<297::AID-RSA5>3.0.CO;2-Q

[10] J. Komlós and M. Simonovits. Szemerédi's regularity lemma and its applications in graph theory. *Combinatorics: Paul Erdős is eighty*, 2:295–352, 1993.

[11] L. Lovász. *Combinatorial problems and exercises.* Corrected reprint of the 1993 second edition, AMS Chelsea Publishing, Providence, RI, 2007.

[12] R. Martin. The edit distance in graphs: Methods, results, and generalizations. In A. Beveridge, J. Griggs, L. Hogben, G. Musiker, and P. Tetali, editors, *Recent Trends in Combinatorics*, pages 31–62. Springer International Publishing, Cham, 2016.

[13] H. Ryser. Combinatorial properties of matrices of zeros and ones. *Canadian Journal of Mathematics*, 9:371–377, 1957. https://doi.org/10.4153/CJM-1957-044-3

[14] M. Sahs, P. Sissokho, and J. Torf. A zero-sum theorem over $\mathbb{Z}$. *Integers*, 13:#A70, 2013.

[15] E. Szemerédi. Regular partitions of graphs. *Colloques Internationaux C.N.R.S* ***260*** *- Problèmes Combinatoires et Théorie des Graphes*, pages 399–401, 1976.

[16] D. West. *Introduction to graph theory*. Prentice Hall, Second edition, 2001.

# A Self-Bounding Branch & Bound Procedure for Truck Routing and Scheduling

Csongor Gy. Csehi
Budapest University of Technology and Economics, Műegyetem rkp. 3-9, Budapest, Hungary
E-mail: cscsgy@cs.bme.hu

Ádám Tóth, Márk Farkas
Nexogen Ltd. Alkotás u. 53, Budapest, Hungary
E-mail: toth.adam@nexogen.hu, mark.farkas@nexogen.hu and nexogen.com

**Keywords:** logistics, route optimization, branch and bound

**Received:** October 30, 2018

*In this paper we study a part of a core algorithm of a complex software solution for truck itinerary construction for one of the largest public road transportation companies in the EU. The problem is to construct a cost optimal itinerary, given an initial location with an asset state, the location and other properties of tasks to be performed. Such an itinerary specifies the location and activity of the truck and the driver until the completition of the last routing task. The calculation of possible itineraries is a branch and bound algorithm. The nodes of the search tree have the following arguments: position, time, driver-state and truck-state. For each node we calculate the cumulative cost for the road reaching that state, and a heuristic lower bound for the cost of the remaining road. In each step the procedure expands the next unexpanded node with the best sum for cumulative and heuristical costs.*
*It is hard to give a sharp lower bound if the model contains time windows. To make a sharp heuristic we run the same branch and bound algorithm (from each node) but with simplified data (hypothetical positions and simplified activities: no refuelling, no road costs, etc.). We have reached significant gains in performance and quality compared to the previous approach.*

*Povzetek: Članek predstavlja aplikacijo za razporejanje tovornjakov v enem največjih transportnih podjetij v EU. Za reševanje problema je uporabljena metoda razveji in omeji. Da bi izboljšali hevristiko, je bila uporabljena ista razveji in omeji metoda iz vsakega vozlišča na poenostavljenih podatkih. Končni rezultat je pomembno izboljšan v primerjavi s predhodnimi pristopi.*

## 1 Introduction

We will study a part of a core algorithm of a complex software solution for truck itinerary construction for one of the largest public road transportation companies in the EU. A minor improvement on the operational cost of each tour can result huge advantage for the freight services company.

The problem is to construct a cost optimal itinerary, given an initial location with an asset state, the location and other properties of tasks (we will call them routing tasks) to be performed. Such an itinerary specifies the location and activity of the truck and the driver until the completition of the last routing task. This means that this itinerary gives every instruction to the driver, including every turn in the road and every stops with exact durations, etc. The working stops can be done only in the places of the tasks, the refueling and resting stops can be done only in previously fixed places (roughly 4000 fixed parking places and 100 fixed filling stations across Europe). To achieve such an itinerary we use mapping software to construct the routes and calculate the distance, duration and cost between any two locations. Clearly the problem is much harder than a path finding in the graph, because we can do many different actions in each place (different amount of fuel taken, different duration of rest, etc).

The software (which also performs the vehicle assignment) is already finished and applied with positive results (from 2015), large cost saving is achieved by the company. For more formal definitions of the problem, and more information of the software one must read [4]. The ongoing researches aim to extend the functionality of the software. One goal is to improve optimality by planning the itinerary for longer timespan. That means more routing tasks have to be calculated each round.

## 2 History

This area of operations research is widely studied [1]. However, to build a complex solver for truck routing and scheduling is a novel concept. This is the reason why we have collected some approaches and solutions that are similar in methodology and solution concept not in the problem in this section. For detailed information on the literature of the problem one can read [4].

There is no such concept in the literature which contains the idea of using the same algorithm on simplified data to get the lower bounds for a branch and bound method. Hence, we have named our method Self-Bounding Branch and Bound (SBBB).

There are many approaches using so called Recursive Branch and Bound (RBB) methods [7]. However, they obtain the bounds by a recursive strategies using the information from preceding calculations [8]. In SBBB we do not use any previously calculated information during the lower bound calculation. In our particular problem the driver state makes this approach impossible because the state is different in all the nodes of the searching tree. In some cases (such as in rectangular guillotine strip packing problem [5]) the recursivity can be formalized in functions which help the calculation of the bounds.

Searching for SBBB alike concepts one can find the Fractal Branch and Bound (FBB) [9]. However, FBB is a very special method where the searching field is self-similar. This means that FBB is more likely an RBB because it uses simplification on the often calculated values. That way, it is different from SBBB and it is not applicable to our problem because of the different driver states of the nodes.

The Double Branch and Bound (DBB) methods are algorithms where the lower bound for a branch and bound method is calculated by another branch and bound method [11], [10]. This means that SBBB is a special type of DBB. However, the DBB concept can be applied for small instances in reasonable timeframe (see some computational results in [10]). It is usual to use RBB like methods for $m$-machine permutation flowshop problems [2]. Moreover there are some approaches where they apply DBB methods [3] instead of the recursive functions.

As we mentioned before no such concept can be found in the literature where the same algorithm on simplified data gives the lower bound of a branch and bound method. However, our results (see Section 4) shows that for some problems with proper reconciliation SBBB can solve large problem instances (larger than DBB).

## 3 The algorithm

The calculation of possible itineraries is a branch and bound algorithm. For detailed information on the widely used algorithms of operations research the reader should see [1] The nodes of the search tree have the following arguments: position, time, driver-state and truck-state (we will call these data the state). For each node we calculate the cumulated cost for the itinerary reaching that state, and a heuristical lower bound for the cost of the remaining road. Each node has a pointer to its parent (this will make it possible to calculate the roue from a proper node). In each step the procedure expands the next unexpanded node with the best sum for cumulated and heuristical cost.

The following oversimplified example of [4] with Figure 1 illustrates the tree of the algorithm. Suppose that we are in position 'Start' in the begining. From 'Start' we can go to different places for example two parking places 'P1' and 'P2' (the state will be different in the two locations if the duration and distance of the drivings are not equal). Supposing that we can rest 9 or 11 hours we get two new nodes from each parking place reaching node. If we can reach place 'P3' from both 'P1' and'P2' then this way we get four different nodes in the same place 'P3'. In general none of the four nodes can be bounded in the algorithm, because the states are different and hence, we can not predict which will give the best solution in the end.

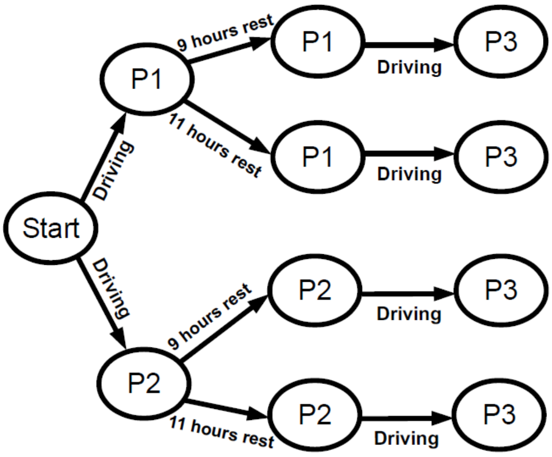


Figure 1: An example subtree of the algorithm

The algorithm has many additional logics, but here we focus on the heuristics only. A more detailed description of the algorithm can be found in [4].

The better the lower bounds are, the less nodes need to be expanded. However, it is always more time-consuming to make better estimations. The difficulty with the heuristics is the presence of the state of the driver (driver state) and the opening times of the routing tasks (time window). Both would be easier to handle separately but together it gives an NP-hard problem. The current solution optimizes the routes for each task in a tour separately, because the current heuristics are too weak and that way the algorithm would be too slow (would expand too many nodes).

The main steps of the algorithm are the following:

**Algorithm 1.**

```
1. Create the starting node of the tree from the initial state and position of the driver. Put it in an empty list L. Set the upper bound $U = \infty$.
2. While L has any element:
    (a) Pick X from L with the best TotalCost value.
    (b) If the itinerary given by X is a complete tour (finishing all the routing tasks), then RETURN X.
```

*(c) Select the best possible activities (set $A$) to do from $X$.*

*(d) BRANCHING: For each element of $A$ create the node (set $N$) which represents the state and position after that activity.*

*(e) For each element of $N$ calculate the cummulated cost (we can get it by adding the cost of the activity to the cummulated cost of $X$).*

*(f) BOUNDING 1: For each element of $N$ calculate the lower bound for the remaining cost (this will be examined in detail in the next sections). Discard the nodes from $N$ which has a total cost (it is the sum of the cummulated and the remaining costs) higher than the upper bound $U$.*

*(g) BOUNDING 2: For each element of $N$ compare the lower bound for the reaching time with the limitations (we get the lower bound during the calculation of the heuristics). If the node can not reach the target in time than delete it from $N$.*

*(h) BOUNDING 3: For each element $Y$ of $N$, where the place of $Y$ is $P$, get the list $L_P$ of the previously examined nodes in place $P$. Compare $Y$ with every element of $L_P$, and if there exists such $Z$ that every state related variable and the cost are not worse in $Z$ than in $Y$, then delete $Y$ from $N$.*

*(i) For each element of $N$ put it into $L$ and into the proper $L_{P_i}$ list according to the place of the nodes.*

*(j) For each element $C$ of $N$ which is a complete tour, let the upper bound $U = min(U, cost(C))$.*

*3. RETURN: Unreachable target. The target can not be reached in the given time limit.*

The first bounding is the usual bounding of BB methods. The second bounding discards the infeasible solutions. The third bounding is a dominance criteria which uses dominance relation between partial solutions.

In this paper a solution is given for the efficient lower bound calculation (in Step 2/f). This will be done by a similar branch and bound structure but with simplified data (relaxed field of POIs).

## 3.1 The concept of the main bounding method

As we mentioned in step 2/f of the algorithm we need a good heuristic to bound the remaining cost from every node. For this we need to calculate a minimal itinerary and we have to bound the needed duration.

First we estimate the remaining distance and driving duration. To optimize the running time we do not want to make the mapping software calculate all these estimations, but store as much of the possibly needed information as we can. We construct two graphs where the nodes are the possible POIs of the tours (parking places, filling stations, ferries, tunnels, etc.) and the length of the edges are the minimal distances and driving durations (we get those values using PTV Group softwares). From these graphs we generate the minimal distances and durations between each pair of nodes with the Floyd – Warshall algorithm (Floyd [6] 1962; Warshall [12] 1962). This happens in a precalculation phase before the itinerary generator algorithm. It is a separate topic how we handle the truck positions and places of the routing tasks (since they are not permanent, hence, they are not contained in the graphs).

These graphs can be used during the algorithm instead of PTV, thus accelerating the computation that way.

After we have a lower bound for the remaining driving time we estimate the total time needed by constructing a hypothetical itinerary. These are the steps of the estimation:

**Algorithm 2.** *1. Let $X$ be the remaining driving time needed to reach the target. Let $D$ be the state of the driver, and $T =$ the actual time.*

*2. $Y =$ the amount the driver with driver-state $D$ can drive continuously.*

*3. Let $Z = min\{X, Y\}$.*

*4. $D =$ the driver-state after a $Z$ long drive.*

*5. $X = X - Z$; $T = T + Z$.*

*6. If $X = 0$ then RETURN $T$.*

*7. $R =$ the amount what the driver with driver-state D should rest.*

We suppose that the driver can drive the maximal available driving time, each time, and then reaches a parking place. In each parking place the driver rests the minimal amount what is needed and then proceeds further. When the driver reaches a routing task sometimes he has to wait for the time-window. However, supposing that there are no time-windows the heuristics can be calculated in linear time (we will call it linear heuristic).

On the other hand, if we think about how to include the time-windows in the linear heuristic we face a problem. Namely, sometimes it would be better to rest more, not just the minimal needed amount before making the task. The following example highlights that behavior.

Suppose that the driver arrives at 6 a clock, after 9 hours of driving, but the routing task opens at 10 a clock. To finish the routing task, the driver has to work 1 hour there and we have one more routing task which is 2 hours far from this. When will we finish the last routing task?

1. If we wait for the first opening and work 1 hour, then we cannot drive further because of the daily driving time limit (9 hours). That way we have to rest at least 9 hours. After the rest we can drive to the next routing task and finish it until **23 a clock**.

2. If we rest 9 hours instead of the 4 hours waiting then

we can start the work with a fresh state, drive to the next routing task and finish it until **19 a clock**.

The above example shows that we can not make good lower bounds with Algorithm 2 (linear heuristic) if we try to optimize with the driver-state and the time-windows at the same time. However, to obtain better estimations for the branch and bound procedure we must include the time-windows in the heuristics. For the best fit (between the heuristics and the algorithm) we apply the same logics to calculate a lower bound for the duration as we use in the branch and bound algorithm itself (see Section 3.2), but relax the field of POIs.

With the linear heuristic we can not efficiently optimize the tours with more than one tasks together. That way the current solution runs Algorithm 1 for each task in a tour separately (after each other, because they use the finishing driver state of the previous one).

## 3.2 The self-bounding branch and bound algorithm

As we mentioned before the main branch and bound algorithm works on nodes with position, time, driver-state and truck-state. The positions are real locations on the map. We present the algorithm B&B heuristic below, to make a sharp heuristic in step 2/f of the original algorithm. Mainly we run the same branch and bound algorithm (from each node) but with hypothetical positions (and simplified activities: no refuelling, no road costs, etc.).

**Algorithm 3.** *1. Create the starting node $S'$ of the inner searching tree. $S'$ has the same state and position of the driver as the node in the outer branch and bound (to which this algorithm results the lower bound).*

*2. Put $S'$ in an empty list $L'$.*

*3. While $L'$ has any element:*

- *(a) Pick $X'$ from $L'$ with the best TotalCost value.*
- *(b) If the itinerary given by $X'$ is a complete tour (finishing all the routing tasks), then RETURN $X'$.*
- *(c) Select the best possible hypothetical activities (set $A'$) to do from $X'$.*
- *(d) For each element of $A'$ create the node (set $N'$) which represents the state and position after that activity.*
- *(e) For each element of $N'$ calculate the cummulative duration (we can get it by adding the duration of the activity to the cummulative duration of $X'$).*
- *(f) For each element of $N'$ calculate the heuristical duration (with the linear heuristics).*
- *(g) For each element of $N'$ compare the lower bound for the reaching time with the limitations. If the node can not reach the target in time than delete it from $N'$.*
- *(h) For each element $Y'$ of $N'$, where the place of $Y'$ is $P'$, get the list $L'_P$ of the previously examined nodes in place $P'$. Compare $Y'$ with every element of $L'_P$, and if there exists such $Z'$ that every state related variable and the cost are not worse in $Z'$ than in $Y'$, then delete $Y'$ from $N'$.*
- *(i) For each element of $N'$ put it into $L'$ and into the proper $L'_{P_i}$ list according to the place of the nodes.*

*4. RETURN: Unreachable target. The target can not be reached in the given time limit.*

This algorithm is similar to the linear heuristic from the aspect, that it generates those positions which was used by the linear heuristic. However, this algorithm lets the different cases compete in total duration. The best solution will give the B&B heuristic, that will be the lower bound for the remaining cost of the node in Algorithm 1 (the main branch and bound algorithm).

Observe that the B&B heuristic needs a lower bound too. For this we can use Algorithm 2 (the linear heuristic).

The real difference with Algorithm 1 is in Step 3/c. Here we generate many hypothetical parking places on the fastest road between $S'$ and the goal. Each time Algorithm 3 reaches Step 3/c new parking places can be generated.

It is easy to see that this extended procedure can give much better lower bounds for the main branch and bound algorithm, but it is questionable that if it is worth the extra time required to construct the nodes (calculating their heuristic values). Observe that it is more likely to get better heuristics this way if we have more routing tasks (with time-windows).

### 3.2.1 Driver state penalty

The objective function includes not just the real costs of the itineraries but the cost of the driver's work, the amortization of the car and many other things. The hardest part is the evaluation of the driver's state in the end of the tour. Of course it is better if the driver can drive more in the day after finishing the last working task because that way the location of the next task can be approached earlier. In fact every variable of the driver-state can be important in the end of the tour. We mainly apply a highly tested linear combination of these quantities.

The original lower bound (Algorithm 2) could not include this type of cost. Algorithm 2 gives a hypothetical fastest solution, this way it can be the case that the driver's state is much worse than in a slower solution, hence, it would not be a lower bound if we add the penalty.

However, Algorithm 3 can also handle the driver state penalty, because if those costs are included to the objective function of the inner branch and bound method, then it will lead to the best hypothetical solution which will give a lower bound on the best possible solution's objective function. It is not trivial why this approach gives a lower bound.

It mainly depends on the fact that the finishing driver state's penalty cannot be higher if we put a parking later. This paper does not aim to prove this fact.

This opportunity is very important because the driver state penalty can give about one third of the total cost of a tour. This means that compared to our previous lower bounds this approach can be much more efficient. Hence, if we include the driver state penalty to the heuristics the searching tree of the main algorithm will be much thinner.

#### 3.2.2 Longer tours

Applying Algorithm 3 we could extend the optimization to tours (with $2-3$ tasks) not just separated tasks. This means that we run Algorithm 1 with more than one targets at once, and in step 2/f we run Algorithm 3. Observe that this means that the lower bound calculation (step 2/f) can get more than one tasks at a time also.

The algorithm was a success in practice. Not just that it could plan a longer time period for a truck, but also gives more profitable plans. Building a route in a chain like concept (such as our original solution) optimizes for mid-route objective functions also and fix the earlier parts of the route in each chain. That way it was anticipated that using the new algorithm we could get better objective function values in the end of the tours. Indeed the new algorithm have given better results than the original.

A detailed description of the performance of the new algorithm can be found in Section 4.

## 4 Results

We evaluated the differences using a sample pack of $4500$ long tours. In average, these tours contains $2-3$ routing tasks with time windows. We have applied Algorithm 1 with the linear heuristic (Algorithm 2) for all tasks of those tours separately. Then we have applied Algorithm 1 with the new heuristic (Algorithm 3) for the same tours, but without decomposing them into tasks.

### 4.1 Sizes and speed

We use about 100 parallel machines. Hence, the software runs in about 20 minutes to construct the $4400$ itineraries. However, here we give the performance data in total (summarized for the parallel machines).

The original branch and bound procedure created $3.42 \cdot 10^4$ nodes, inserted $2.01 \cdot 10^4$ nodes (these are the not bounded ones) and expanded $1.33 \cdot 10^4$ nodes during a tour construction in average. The total time of the algorithm was about $2.6 \cdot 10^3$ minutes. The heuristics was calculated in about $26.3$ minutes in total.

The new branch and bound procedure (with the B&B heuristic) expands about $1.85 \cdot 10^4$ nodes, inserted $1.18 \cdot 10^4$ nodes and expanded $7.22 \cdot 10^4$ nodes during an tour construction in average. This means that the new heuristic reduces the number of node expandations by $55\%$. However, each node creation needs more time. The heuristics was calculated in $927$ minutes in total ($35$ times more than the original). Fortunately the total time of the algorithm was about $3.67 \cdot 10^3$ minutes, which is just $40\%$ more than the original.

### 4.2 Costs

The following numbers are not exactly the profits of the tours, because some costs are calculated in other parts of the software (such as the task assignment). Some costs are also modified for the algorithm (only without changing the optimality of the plans), for example the fuel costs and the wages of the drivers.

The original algorithm results $1.03 \cdot 10^9$ objective function value in total, and 713 Euros of costs in average. The new algorithm results $9.69 \cdot 10^8$ objective function value in total, and 686 Euros of costs in average. This means that we reach about $4\%$ better results with the new algorithm.

These results are promising. Moreover the new solution will be even better for longer tours. We are not capable to make statistics for more than 10 routing tasks in one plan yet. However, with the new algorithm it could be profitable to add more computational capacity, and that way plan more routing tasks in one run.

## Acknowledgement

This work is connected to the scientific program of the "Development of quality-oriented and harmonized R+D+I strategy and functional model at BME" project, supported by the New Hungary Development Plan (Project ID: TÁMOP-4.2.1/B-09/1/KMR-2010-0002) Research has also been partially supported by grant # 124171 of the Hungarian National Research, Development and Innovation Office.

## References

[1] M. W. Carter, C. C. Price (2000) *Operations research: a practical introduction*, Crc Press. https://doi.org/10.1201/9781315274188

[2] C. S. Chung, J. Flynn, Ö. Kirca, (2006) A branch and bound algorithm to minimize the total tardiness for m-machine permutation flowshop problems. *European Journal of Operational Research*, 174(1), 1–10. https://doi.org/10.1016/j.ejor.2004.12.023

[3] R. Companys, M. Mateo (2007) Different behaviour of a double branch-and-bound algorithm on Fm| prmu| Cmax and Fm| block| Cmax problems. *Computers & Operations Research*, 34(4), 938–953. https://doi.org/10.1016/j.cor.2005.05.018

[4] Cs. Gy. Csehi, M. Farkas (2016) Truck routing and scheduling, Central *European Journal of Operations Research*, 1–17.
https://doi.org/10.1007/s10100-016-0453-8

[5] Y. Cui, Y. Yang, X. Cheng, P. Song (2008) A recursive branch-and-bound algorithm for the rectangular guillotine strip packing problem. *Computers & Operations Research*, 35(4), 1281–1291.
https://doi.org/10.1016/j.cor.2006.08.011

[6] R. W. Floyd (1962) Algorithm 97: Shortest path, *Communications of the ACM* 5(6):345
https://doi.org/10.1145/367766.368168

[7] A. Hartwig (1985) Recursive branch and bound. *Optimization*, 16(2), 219–228.
https://doi.org/10.1080/02331938508843011

[8] A. Hartwig, F. Daske, S. Kobe (1984) A recursive branch-and-bound algorithm for the exact ground state of Ising spin-glass models. *Computer Physics Communications*, 32(2), 133–138.
https://doi.org/10.1016/0010-4655(84)90066-3

[9] R. Matsuzaki, A. Todoroki (2007) Stacking-sequence optimization using fractal branch-and-bound method for unsymmetrical laminates. *Composite Structures*, 78(4), 537–550.
https://doi.org/10.1016/j.compstruct.2005.11.015

[10] M. Vanhoucke (2002) Optimal due date assignment in project scheduling. *Working Papers of Faculty of Economics and Business Administration, Ghent University, Belgium*
02/159, Ghent University, Faculty of Economics and Business Administration. https://ideas.repec.org/p/rug/rugwps/02-159.html

[11] M. Vanhoucke (2006) Scheduling an R&D project with quality-dependent time slots. *International Conference on Computational Science and Its Applications* (pp. 621–630). Springer Berlin Heidelberg.
https://doi.org/10.1007/11751595_66

[12] S. Warshall (1962) A theorem on Boolean matrices, *Journal of the ACM* 9(1):11–12
https://doi.org/10.1145/321105.321107

# Improving Flow Lines by Unbalancing

Zsolt Mihály
Optasoft Kft., 1051 Budapest, Hungary
E-mail: zsolt.mihaly@optasoft.hu, www.optasoft.hu

Zoltán Lelkes
Department of Information Technology, Pallasz Athéné University, 6000 Kecskemét, Hungary
E-mail: lelkes.zoltan@gamf.kefo.hu, www.uni-pae.hu

**Keywords:** flow line, throughput, cycle time, work-in-process

**Received:** October 30, 2018

*The paper's aim is to show how the performance of a balanced manufacturing flow line can be improved on the critical WIP level by making it unbalanced to some extent. The explanation and a quantitative example are presented. There is a trade-off between system balance and stability, and it can be handled as an optimization problem. Data are gathered using a physical simulation system. The analysis is carried out with a discrete time simulation program which applies next-event time advance mechanism. The model has been implemented in AIMMS modelling language.*

*Povzetek: V prispevku je prikazana uporaba rahle neuravnoteženosti pri kritičnih opravilih na sicer uravnoteženih opravilih tekočega traku. Model je bil implementiran v jeziku AIMMS in rezultat ovrednoten s pomočjo diskretne simulacije.*

## 1 Introduction

The world makes great efforts to diminish the ecological footprint of humanity. For this reason, companies have to meet more and more regulations; corporations are forced to be more efficient. Thus, researches on environmentally benign business practices receive more and more attention [12]. One of the main interests is to improve production systems, like production of cars, pharmaceutical ingredients or electrical goods. It not only diminishes the ecological footprint, but also increases the profitability of the company by higher productivity, lower response time or lower inventory level.

Discrete manufacturing systems can be classified by several disciplines. Following Govil and Fu [4], the manufacturing systems can be

- job shops,
- flow lines,
- flexible manufacturing systems,
- assembly systems.

Manufacturing systems consist of stations. They are atomic processes, that is to say, they cannot be divided further. The main characteristic of a station is its process time. This is the time needed to process an entity. It is a scalar in a deterministic model, and a random variable in a stochastic model. In the latter case, process time has a mean, a variance, and a type of probability distribution function.

The research of manufacturing systems uses diverse modelling techniques, e.g., simulation models [13], queueing theory and Petri nets [9]. In this paper, flow lines are investigated using physical experiments and a discrete time simulation model. In flow lines, stations are connected in a linear way (see Figure 1). Some examples from the literature contain investigations into flow line with common buffer [16], complex optimization problems where the flow line is only one element in the model [8] or more complicated systems. Huang and Li examined a two-stage hybrid flow shop with multiple product families [7]. Simulation modelling has a wide range of applications in engineering-aided manufacturing regarding system performance. Modelling apparel assembly cells [1], a Mercedes-Benz production facility [10], or analysing the performance of a Korean motor factory [2] are only some of the examples.

Hopp and Spearman [6] investigated flow lines in which there is only one machine per station, one job class, no capacity constraint, and the queueing principle is first in, first out (FIFO). Three main modelling measures are proposed by them:

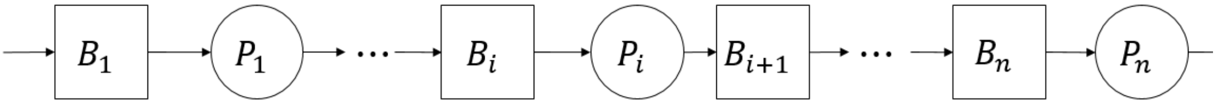


Figure 1: The general structure of a flow line (B: buffer, P: process).

- Throughput (TH): the number of entities (cars, apples, people, etc...) coming out from the system during a given time
- Cycle time (CT): the time an entity spends in the system
- Work-in-process (WIP): the number of entities residing in the system at the same time

The system performs better if it has a higher TH or a lower CT. These parameters are not independent from each other. Little's law makes a connection among them:

$$WIP = TH \times CT$$

According to the previous equation, the optimal value of WIP in a deterministic system is

$$W_0 = T_0 \times r_b$$

Where

- Bottleneck rate $(r_b)$: the rate of the station that has the highest utilization
- Raw process time $(T_0)$: the sum of the average process times in the flow line
- $W_0$ is called the critical WIP level ([6]).

The variability of procedures is measured with the coefficient of variation (CV):

$$CV = \frac{standard\ deviation}{mean}$$

Hopp and Spearman use two so called characteristic functions to analyse the performance. The dependent variables are the TH and the CT, while the independent variable is the WIP level both times. The flow line is modelled as a closed network. It means that the level of WIP is a model parameter [14]. Regarding performance analysis, three important concepts were introduced [6]:

Best case performance: the best possible performance for a line. It is balanced, and there is no batching.

Worst case performance: the worst possible performance for a line. All the entities move in one batch.

Practical worst case (PWC): as the worst case performance is so bad that it is far from practical instances, PWC was introduced to define a realistic worst case.

The paper's aim is to show how the performance of a balanced manufacturing flow line can be improved on the critical WIP level by making it unbalanced to some extent. The explanation and a quantitative example are presented. There is a trade-off between balance and stability, and it can be handled as an optimization problem.

# 2 Method of examination

In this research, the same characteristics are used to evaluate the performance as in [5]. Both physical and simulation model experiments are performed to gather data. Both models are flow lines with FIFO queueing discipline containing single machine stations, one job class and using constant work-in-process (CONWIP) control. It means that a new entity arrives into the system only when another one leaves it.

In the physical model experiment, a toy car factory has been realized with the assumption of infinite raw material stock. The entire process to build a small car takes 4 minutes. In an arbitrary way, the operations could be distributed among 4 production processes where one-one person works with different abilities. Beside the four production processes, there is a transport process as well. Altogether there are 5 stations.

Building a simulation model consists of three levels. There is a superstructure which can represent all the possible flow lines. Beside this, there is a mathematical model and an algorithm. Based on the algorithm, a discrete time simulation program with next-event time advance mechanism is worked out. Comparing with fixed-increment time advance method, it is more complicated, but more efficient regarding computational need [15].

The simulation program is implemented in AIMMS modelling language [11]. It has already been used in other studies with success. E.g., [3] used it on supply chain optimization with homogenous product transport constraints. The simulation program can be easily extended in this environment. AIMMS is linked to the most modern solvers, which are easily integratable. Furthermore, it has an advanced graphical user interface, which can be used for creating simply usable and ergonomic softwares.

# 3 Results

## 3.1 The explanation of the trade-off

Regarding a given raw process time $(T_0)$, a deterministic balanced line performs better than a deterministic unbalanced line. On the other hand, the balanced line has a worse stability regarding variability [6] .

Let us investigate a balanced system whose raw process time is $T_0$, and has n stations. It is true in this case that

$$\forall i \quad T_i = \frac{T_0}{n}$$

$(T_i$ denotes the average process time of the i-th station) That is to say, all the process times are the same. So the bottleneck rate:

$$r_b = \frac{n}{T_0}$$

If the system with the raw process time $T_0$ is unbalanced then

$$\exists i \quad T_i > \frac{T_0}{n}$$

From this, it can be concluded that for the unbalanced system

$$r_b < \frac{n}{T_0}$$

The TH of a deterministic system is calculated in the following way:

$$TH = \min\left\{\frac{WIP}{T_0}, r_b\right\}$$

The two systems have the same performance until

$$WIP \leq r_b T_0$$

After that the balanced system works better because it has a higher $r_b$. This proves that a balanced line has a better performance in this case

It is harder to prove mathematically that a balanced line is less stable. Nonetheless, this effect could be seen when investigating the physical model. In this section, an example is shown. Two flow lines are compared. For both of them, $T_0 = 8h$, and each of them contains four stations. In the balanced line

$$\forall i \quad T_i = 2h$$

In the unbalanced line, the first station is the bottleneck, its process time is 5h, and 1h regarding the rest of them. In the stochastic case for both systems

$$\forall i \quad CV_i = 1$$

The characteristics of the flow lines are summed up in Table 1.

The results of the experiment can be seen in Figure 2. Relative changes are displayed on the ordinate, which shows the deteriorating effect of variability from a different aspect as usual characteristic functions. The reason for applying it is that it is easier to see the difference in the drop of performance regarding WIP. These characteristics are calculated in the following way.

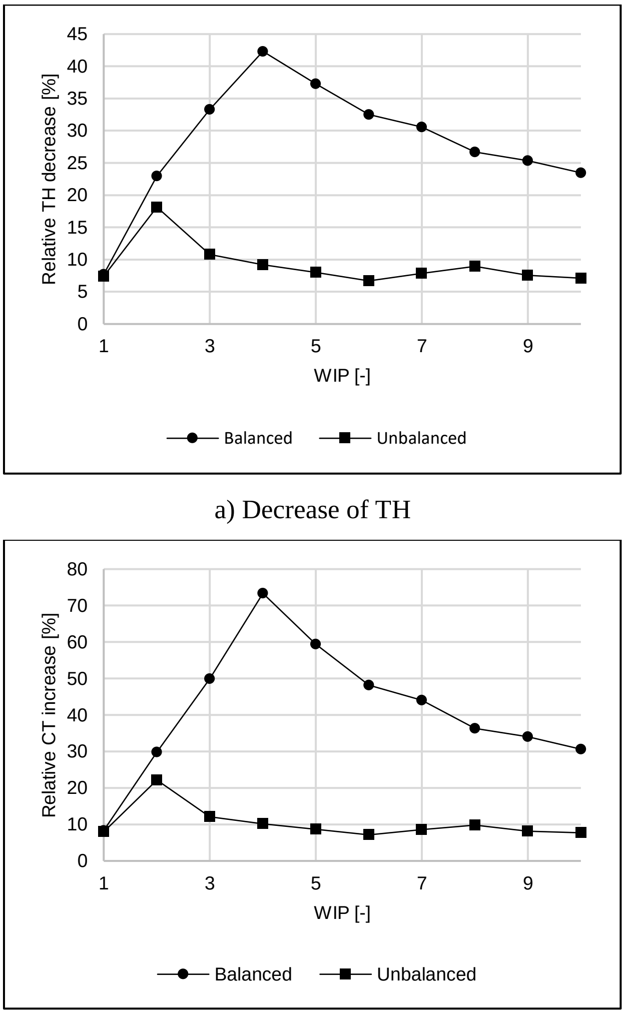
$$TH_{rel} = \frac{abs(TH_{stoch} - TH_{rel})}{TH_{stoch}}$$

$$CT_{rel} = \frac{abs(CT_{stoch} - CT_{rel})}{CT_{stoch}}$$

| | Balanced | Unbalanced |
|---|---|---|
| $T_1$ | 2h | 5h |
| $T_2$ | 2h | 1h |
| $T_3$ | 2h | 1h |
| $T_4$ | 2h | 1h |
| $CV_1$ | 1 | 1 |
| $CV_2$ | 1 | 1 |
| $CV_3$ | 1 | 1 |
| $CV_4$ | 1 | 1 |

Table 1: The characteristics of the investigated flow lines

Both of the diagrams show that the extent of deterioration is bigger when the line is balanced. In this case, the maximal TH decrease is 42% in the balanced system, and 18% in the unbalanced one. The maximal CT increase is 73% when the flow line is balanced; 22% when it is unbalanced. It means that the maximal deterioration of TH is twice as high in balanced lines as in unbalanced lines, and the maximum of CT deterioration is three times as high. That is, evidence is shown that balanced line has less stability regarding variability. The same phenomena could be observed in each physical experiments. The loss of TH and the growth of CT increase until the critical WIP value is reached. After the peak, both functions begin to decrease. At high WIP levels, they will converge into 0. Table 2 sums up the results regarding the peaks.



a) Decrease of TH

b) Increase of CT

Figure 2: The stability of flow lines regarding variability

| | Balanced | Unbalanced |
|---|---|---|
| TH | 42% | 18% |
| CT | 173% | 122% |

Table 2: Comparison of the maximal deteriorations

## 3.2 A quantitative example for system unbalancing

In this section, a quantitative example is shown in which the unbalanced system has a better performance on the critical WIP level. Three systems are compared: a balanced and two unbalanced. The balanced system has uniform process times of 1 hour. The CV of the first three stations are equal to 0.1, and the last station's CV is 1. In the unbalanced systems, the process times of three operations are 1.15 hour, and there is one with 0.55 hour. That is, they differ in the position of the non-bottleneck process. The station with the process time of 0.55 hour has CV = 1. The CV of the bottlenecks are 0.1. The characteristics of the investigated systems can be seen on Table 3.

| | Balanced | Unbalanced (first three are bottlenecks) | Unbalanced (last three are bottlenecks) |
|---|---|---|---|
| $T_1$ | 1h | 1.15h | 0.55h |
| $T_2$ | 1h | 1.15h | 1.15h |
| $T_3$ | 1h | 1.15h | 1.15h |
| $T_4$ | 1h | 0.55h | 1.15h |
| $CV_1$ | 0.1 | 0.1 | 1 |
| $CV_2$ | 0.1 | 0.1 | 0.1 |
| $CV_3$ | 0.1 | 0.1 | 0.1 |
| $CV_4$ | 1 | 1 | 0.1 |

Table 3: The characteristics of the investigated flow lines.

The critical WIP level of the flow line can be calculated in the following way:

$$W_0 = r_b \times T_0 = 1\frac{1}{h} \times 4h = 4$$

The positions of the bottleneck procedures have no effect in the unbalanced systems (Figure 3/b, 3/c). In the deterministic case, the balanced system performs better, as expected. On the other hand, when there is variability, the unbalanced flow line has a better output on the critical WIP level (see Table 4, 5). According to the experiments, the TH of the unbalanced system can be 9-11% higher compared with the balanced line, the CT is 8-9% lower. The results confirm the assumption that there is a trade-off between balance and stability, and it can be handled as an optimization problem.

| | Balanced | Unbalanced |
|---|---|---|
| TH [1/h] | 1 | 0.87 |
| CT [h] | 4 | 4.6 |

Table 4: the performance of the investigated system on the critical WIP level in the deterministic case

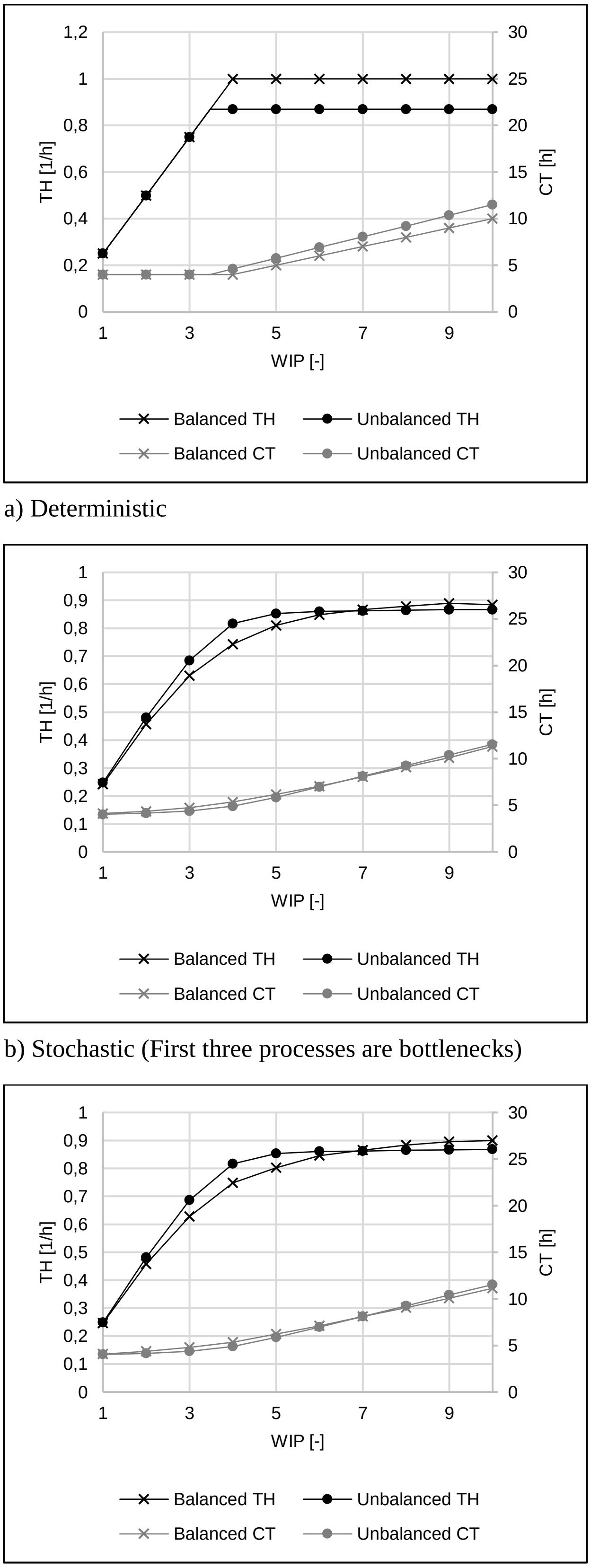
| | Balanced | Unbalanced | Improvement |
|---|---|---|---|
| TH | 0.74 1/h | 0.82 1/h | 11% |
| CT | 5.38 h | 4.90 h | 9% |

Table 5: the performance of the investigated system on the critical WIP level in the stochastic case

# 4 Conclusion

Endeavours are generally made to balance flow lines. This is an intuitive idea, and earlier researches showed examples where unbalanced systems had worse performance. In this paper, it has been shown that unbalancing the flow line in a small extent achieves better performance on the critical WIP level, that is to say, higher TH and lower CT. In the examined case, the TH was 9-11% higher and the CT 8-9% lower on the critical WIP level.

This result is important in flow lines as well where the manufacturing is precise, but there are several products. Each of them might have a low variability, but the variability of the stations' process times can be high if the average process times of products are different.



a) Deterministic

b) Stochastic (First three processes are bottlenecks)

c) Stochastic (Last three processes are bottlenecks)

Figure 3: Comparison of the performance of a balanced and two unbalanced systems.

## List of symbols

| | |
|---|---|
| CT | Cycle time |
| TH | Throughput |
| WIP | Work-in-process |
| $T_0$ | Raw process time |
| $r_b$ | Bottleneck rate |
| $W_0$ | Critical WIP level |
| $T_i$ | Average process time at station i |
| $CV_i$ | Coefficient of variability at station i |
| n | Number of stations |

## References

[1] J. T. Black and B. J. Schroer. Simulation of an apparel assembly cell with walking workers and decouplers. Journal of Manufacturing Systems, 12(2):170-180, 1993. https://doi.org/10.1016/0278-6125(93)90016-M

[2] K. Cho, I. Moon, and W. Yun. System analysis of a multi-product, small-lot-sized production by simulation: A Korean motor factory case. Computers & Industrial Engineering, 30(3):347-356, July 1996. https://doi.org/10.1016/0360-8352(96)00003-4

[3] T. Farkas, Z. Valentinyi, E. Rév, and Z. Lelkes. Supply chain optimization with homogenous product transport constraints. In Computer Aided Chemical Engineering, volume 25, pages 205-210, 2008. https://doi.org/10.1016/S1570-7946(08)80039-9

[4] M. Govil and M.C.Fu. Queueing theory in manufacturing: A survey. Journal of Manufacturing Systems, 18(3):214-240, 1999. https://doi.org/10.1016/S0278-6125(99)80033-8

[5] W. J. Hopp. Supply Chain Science. Waveland Pr Inc, Long Grove, Illinois, 2011.

[6] W. J. Hopp and M. L. Spearman. Factory Physics. McGraw-Hill Education, New York, New York, 2000. https://doi.org/10.1036/0256247951

[7] W. Huang and S. Li. A two-stage hybrid flowshop with uniform machines and setup times. Mathematical and Computer Modelling, 27(2):27-45, January 1998. https://doi.org/10.1016/S0895-7177(97)00258-6

[8] J. Olhager and B. Rapp. Balancing capacity and lot sizes. European Journal of Operational Research, 19(3):337-344, March 1985. https://doi.org/10.1016/0377-2217(85)90130-4

[9] H. T. Papadopoulos and C. Heavey. Queueing theory in manufacturing systems analysis and design: A classification of models for production and transfer lines. European Journal of Operational Research, 92(1):1-27, July 1996. https://doi.org/10.1016/0377-2217(95)00378-9

[10] Y. H. Park, J. E. Matson, and D. M. Miller. Simulation and analysis of the Mercedes-Benz all activity vehicle (aav) production facility. In Proceedings of the 1998 Winter Simulation Conference, pages 921-926, December 1998. https://doi.org/10.1109/WSC.1998.745790

[11] M. Roelofs and J. Bisschop. AIMMS The user's guide. AIMMS B.V., AP Haarlem, The Netherlands, 2016.

[12] J. Sarkis. A strategic decision framework for green supply chain management. Journal of Cleaner Production, 11(4):397-409, June 2003. https://doi.org/10.1016/S0959-6526(02)00062-8

[13] J. S. Smith. Survey on the use of simulation for manufacturing system design and operation. Journal of Manufacturing Systems, 22(2):157-171, 2003. https://doi.org/10.1016/S0278-6125(03)90013-6

[14] W. Whitt. Open and closed models for networks of queues. AT&T Bell Laboratories Technical Journal, 63(9):1911-1979, November 1984. https://doi.org/10.1002/j.1538-7305.1984.tb00084.x

[15] W. L. Winston. Operations Research: Applications and Algorithms. Cengage Learning, Boston, Massachusetts, 2003.

[16] H. Yamashita and S. Suzuki. An approximation method for line production rate of a serial production line with a common buffer. Computers & Operations Research, 15(5):395-402, 1988. https://doi.org/10.1016/0305-0548(88)90056-1

# Incremental 2-D Nearest-Point Search with Evenly Populated Strips

David Podgorelec and Denis Špelič
University of Maribor, Faculty of Electrical Engineering and Computer Science
Koroška cesta 46, SI-2000 Maribor, Slovenia
E-mail: david.podgorelec@um.si, https://gemma.feri.um.si/

**Keywords:** incremental nearest-point search, dynamic partition, deterministic skip list

**Received:** October 30, 2018

*The incremental nearest-point search successively inserts query points into the space partition data structure, and the nearest point for each of them is simultaneously found among the previously inserted ones. The paper introduces a new approach which solves this task in 2-D space in a nearly optimal manner. The proposed dynamic partition into parallel strips, each containing a limited number of points structured in the deterministic skip list, successfully prevents situations with over-populated strips, while its further advanced version with two perpendicular partitions and four categories of deterministic skip lists efficiently decreases the number of strips to be examined in a great majority of practical cases.*

*Povzetek: V članku je predstavljen algoritem za reševanje inkrementalnega problema najbližje točke, ki z dinamično delitvijo ravnine v vzporedne trakove preprečuje prenaseljenost le-teh, z dodatno delitvijo, pravokotno na prvo, pa se večinoma izogne tudi preiskovanju prevelikega števila trakov.*

## 1 Introduction

Let us assume a set $S$ of points in a space $M$ and a query point $\mathrm{p} \in M$. The *nearest-point search* (NPS) aims to find the point in $S$ being the closest to p. In its most common form, $M$ is a $d$-dimensional vector space (2-D in this paper), points correspond to their position vectors, and closeness is expressed with Euclidean distance. Applications can be found in a variety of domains, such as computational geometry [2, 20], geographic information systems – GIS [18], bioinformatics [21], image and video compression [3, pattern recognition [16], computer vision [12], robot motion planning [14] ], telecommunications [9], and computer graphics [6]. There are a number of versatile generalizations where the distance metric between spatial points is extended to any quantitative measure of similarity between two generic objects, as one may also measure closeness of pairs of polygons, text strings, images, audio sequences etc.

If the distance is computable in $\Theta(1)$ time, the brute-force NPS is trivially handled in $\Theta(n)$ time, but the problem becomes more demanding in a *recurring NPS* where a larger set of query points has to be considered. A straightforward repetition of the brute-force approach results in $\Theta(n^2)$ time when applied to $\Theta(n)$ query points. In spatial domains, space partitioning can be used to bound the number of possible nearest-point candidates in each iteration [19]. The partition is accomplished by constructing a hierarchical or a grid data structure, typically a tree [7], the Voronoi diagram [10], a regular grid [4], or a multi-level organization of these structures [19]. Such a data structure is aimed to accelerate solving the point-location problem i.e. determination of the region where a query point lies. The methodology is based on the assumption that the points in the same region or those in adjacent regions are closer to each other than the points in more distinct regions. A static partition does not depend on point distribution or, eventually, it only utilizes the bounding box and/or the number of points in $S$. On the contrary, a dynamic partition maintains the numbers of points in all regions within the previously determined limits. Particularly in higher dimensions, where either the query time or storage space must be sacrificed, a user may also be satisfied by approximate solutions provided by the reasonably fast locality sensitive hashing technique, for example [17].

The *all nearest-points pairs problem* represents a special form of the recurring NPS, where the set of target points $S$ and the set of query points coincide. For each point $\mathrm{p}_i \in S$, we have to find $\mathrm{p}_j \in S$, such that $i \neq j$ and the distance $|\mathrm{p}_i\, \mathrm{p}_j|$ is minimal. Two different approaches can be distinguished.

(a) The two-phase *preliminary points arrangement* approach firstly arranges all the points into the adequate regions of the space partition data structure. Then, in the second phase, this preliminary points arrangement is utilized to determine the nearest point for every query point.

(b) The *incremental nearest-point search* successively inserts points $\mathrm{p}_1, ..., \mathrm{p}_n$ into the space partition data structure and simultaneously searches for their nearest points. The nearest point for $\mathrm{p}_i$, $1 < i \leq n$, is determined among previously inserted points from $S_i = \{\mathrm{p}_1, ..., \mathrm{p}_{i-1}\}$ only. Note that $S_1 = \emptyset$.

In this paper, we introduce an original dynamic plane partition into parallel strips and utilize it to handle the incremental NPS in 2-D space. The incremental search adequately models interactive processing of database queries where the results of previous queries are usually irrelevant for processing the current one. In an expressive everyday life example, a new house is connected to the electrical grid, water supply, road and telecommunication systems in a currently optimal way, although this solution could prove far from optimal after ten more houses are built in the neighbourhood. In computational geometry, a remarkably fast incremental Delaunay triangulation algorithm is based on the incremental NPS [20].

Details on the new algorithm and the problem itself are described in Section 2. Section 3 analyses the time complexity, compares the method with an older algorithm based on static strips, and introduces some additional improvements. Finally, the presented work is briefly summarized and some challenges for further research are discussed in Section 4.

# 2 The DP-DSL approach to the incremental nearest-point search

Subquadratic-time methods for the preliminary points arrangement approach are well-known. Utilization of the Voronoi diagram, together with efficient solutions of the point-location problem, for example, leads to an obvious $O(n \log n)$ time solution [1, 11], where $n$ is the number of points in $S$. On the other hand, utilization of the Voronoi diagram in the incremental NPS requires some of the incremental Voronoi diagram construction algorithms which all, although fast on average, require quadratic time in the worst case [8]. For this reason and because of a relatively complex maintenance of the Voronoi diagrams, we preferably study other space partitioning techniques. First of all, we wish to keep practical advantages of the HT-DSL approach [19] and, simultaneously, to improve its theoretical behaviour. The pioneering HT-DSL approach represents even nowadays the only work where the incremental NPS is explicitly considered. It is based on a uniform plane subdivision into parallel strips. These static strips are directly accessible in $O(1)$ time through a *hash table* (*HT*). On the other hand, our method named the DP-DSL approach uses a dynamic partition (DP) into evenly populated strips. In both methods, the points in a particular strip are stored in ($a$, $b$)-*deterministic skip list* (*DSL*) [13], providing a point insertion in $O(\log n)$ time and, on the average, efficient NPS inside the strip. The DP-DSL approach must additionally provide the functionality of DSL splitting as an over-populated strip has to be split into two (or three) strips.

## 2.1 Deterministic skip lists

Our implementation of ($a$, $b$)-DSL, inherited from [19], consists of a doubly linked list of points sorted in non-descending order on the $x$-coordinate. If more points share the same $x$-coordinate, then the $y$-coordinate is decisive. Double connectivity assures that the move from an arbitrary point to its direct predecessor or successor takes $O(1)$ time. This list represents the basic level (level 1) of the DSL. Its nodes (*leaves*) are accessible from simply linked lists of the *internal nodes* at higher levels. Each parent node P (see Fig. 1) at level $h$, $h > 1$, points to a single child node C followed by the remaining children nodes of P, forming a *gap*. The first node C′ after the gap represents the leading (the one with the lowest $x$-coordinate) child node of the successive parent node (P′) of P. The gap size must be in range [$a$, $b$], except the gap behind the last child node. Access to a particular leaf requires $O(b \log n)$ worst time. By keeping $b$ small, the logarithmic access time is provided. Typical pairs ($a$, $b$) in practice are (1, 2), (1, 3), (2, 5), and (3, 7). Fig. 1 shows an example of (1, 3)-DSL. Values stored in a gap are lower or equal to the value in the parent node. Consequently, $M$ at the root level must be set to some "safely" high value.

The actual search for the nearest point to the query point p was also inherited from [19]. Once it determines a candidate of the nearest-point to p and its distance $d$ to p, it limits the search for better candidates to the interior of the circle with the centre p and radius $d$. The search consists of the *local search* in the strip where p was inserted, and the *inter-strip search* which progresses up and/or down through the adjacent strips within the distance of current $d$. The *local search time* is the time needed to perform local search for a single query point, while the *total local search time* is the time spent for local search operations for all query points. In an analogous manner, the *inter-strip search time* and the *total inter-strip search time* can be introduced, while the *total search time* refers to the sum of both, the total local search time and the total inter-strip search time. Finally, the *total time* is the sum of the total search time and the time spent for the dynamic partition construction.

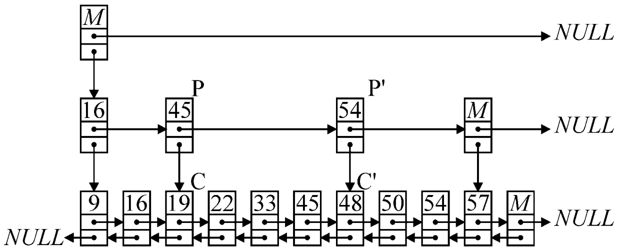


Figure 1: An example of (1, 3)-deterministic skip list.

## 2.2 Dynamic partition

The HT-DSL is remarkably fast for nearly uniform point distributions. However, examples with much slower performance and also strongly affected by the points ordering can effortlessly be constructed and, not rarely, also met in practice. Example in Fig. 2 consists of a cloud of points with rather favourable Gaussian distribution which alone does not result in highly over-populated strips. However, an isolated point concentrates all other points on the opposite end of the region of interest, resulting in a considerable number of empty strips and in the increased population density of those few strips containing the entire point cloud (Fig. 2a). The DP-DSL approach is directly designed to prevent from such situations. The idea is straightforward: when a particular strip contains too many points (the number of points in a strip is labelled $q$ in continuation), the algorithm splits it into a pair of strips, each containing half of the points of the original strip. Under certain conditions, splitting may also result in three strips. The point cloud in Fig. 2b is cut by many narrow strips, while a wide undivided strip is left around the isolated point. Another, more realistic demonstration of the advantageous behaviour of the DP-DSL approach is given in Fig. 3.

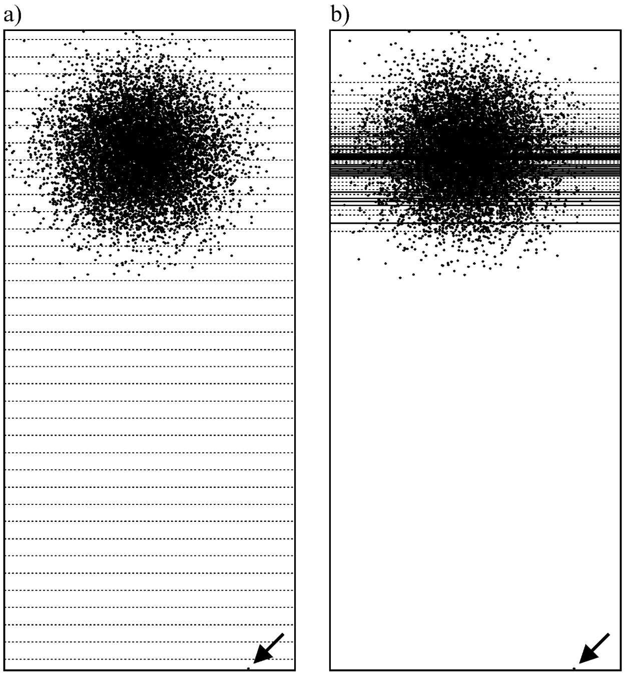


Figure 2: Gaussian distribution with additional point in: a) HT-DSL static partition with uniform strip width, and b) DP-DSL approach with evenly populated strips of variable widths.

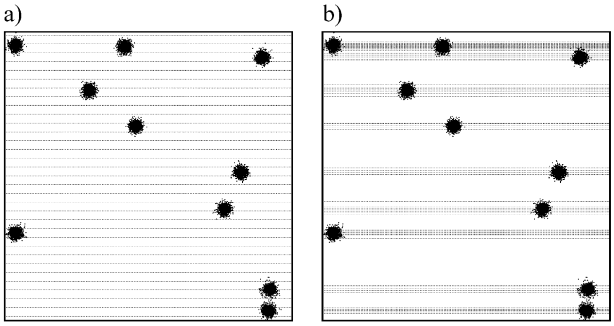


Figure 3: a) HT-DSL and b) DP-DSL approach employed on clusters of points.

The DP-DSL approach requires additional data structure to store the strips' borders i.e. minimum $y$-coordinates for each strip. We use an additional DSL named *Borders* for this purpose. It plays the same role as the hash table in the HT-DSL approach, but requires longer search time (logarithmic instead of constant) and dynamic construction.

Two types of strips are stored in *Borders*. A *line strip* $SoP_i$ is the horizontal line $y = Borders_i$, and an *interval strip* $SoP_j$ is a region between two horizontal lines. The lower borderline is also a part of the interval strip i.e. $SoP_j = \{\mathrm{p}(x, y); Borders_j \leq y < Borders_{j+1}\}$. Exceptionally, the unbounded first and last strip are also the interval strips. Another exception is met when the interval strip $SoP_j$ lies directly above a line strip $SoP_{j–1}$. Then the lower borderline cannot be a part of $SoP_j$ since it represents an independent line strip. A line strip is introduced when the $y$-coordinates of two or more points correspond to the splitting threshold.

Points in each DSL are sorted according to $x$-coordinates, but an over-populated strip should be split with regard to $y$. Splitting is only sensible for the interval strips. A line strip may therefore have O($n$) points, but the local search has to examine only the predecessor and successor of the query point while, on the other hand, the entire interval strip should be examined in the worst case. The role of the line strips is, therefore, to keep the sizes of the interval strips limited.

The splitting algorithm must firstly determine the splitting threshold. We utilize the well-known SELECT algorithm [5] which finds the $i$-th largest element in the set with $q$ points in $\Theta(q)$ time. Simultaneously, the types of the output strips and the numbers of points in each of them are determined. Three diverse output situations can be met: (1) two interval strips, (2) two interval strips and the separating line strip, and (3) an interval strip above a line strip where the latter coincides with the bottom line of the input strip.

The strip splitting is completed by physically splitting the *DSL* into two or three separate skip lists. An intuitive solution rests on the $O(\log n)$-time skip list splitting algorithm [15] which cuts the input DSL at the determined splitting threshold into two separate DSLs. Of course three DSLs may be obtained, when necessary, by performing two cuts. The structure of the input DSL is mostly preserved in the separated DSLs, except that the gaps on the right side along the cut usually require some minor $O(b)$-time corrections. The values of $q$ nodes are maintained in $\Theta(q)$ time afterwards. The method has several desired properties, including the aforementioned inherited structure of higher levels and the ability to reuse allocated nodes of the input DSL. However, the inherited gaps in the output DSLs, varying in size from $a$ to $b$, are often too short for optimal further exploitation. Furthermore, additional short gaps on both sides along a cut are typically produced. In continuation, we propose an original approach, which gives full control over the gap size to the user.

*The bricklaying approach* firstly constructs level 1 for each of the two or three separate DSLs. This is achieved by moving the leaves of the input DSL, one after another, to the end of the corresponding separate list. Upper levels are then built from the elements of the simply linked *global list of recyclable nodes*. At each level, the nodes are grouped into gaps of size $b – gsc$, where $gsc$ is a user-selected gap size correction parameter. A compromise must be found since shorter gaps accelerate later insertions (less gap rearrangements needed), while longer gaps reduce the numbers of nodes at higher levels and consequently decrease the search times.

The global list of recyclable nodes may contain nodes from three different sources, as shown in Fig. 4a. Firstly, eventual unused nodes from previous splitting operations are included. Although the output lists usually contain more internal nodes in total than the input DSL did, the opposite is also possible because of typically longer gaps in split DSLs. The second source consists of the input DSL's internal nodes. They are firstly organized into the linked list, such that the last element of level $k$ is connected to the first one at level $k$ - 1, and this is then appended to the end of the global list. Finally, the third part contains eventual additional nodes allocated just before the actual splitting operation starts. Fig. 4b shows how the nodes of the global list are distributed across three output DSLs. The situation with two output strips is handled in nearly the same manner.

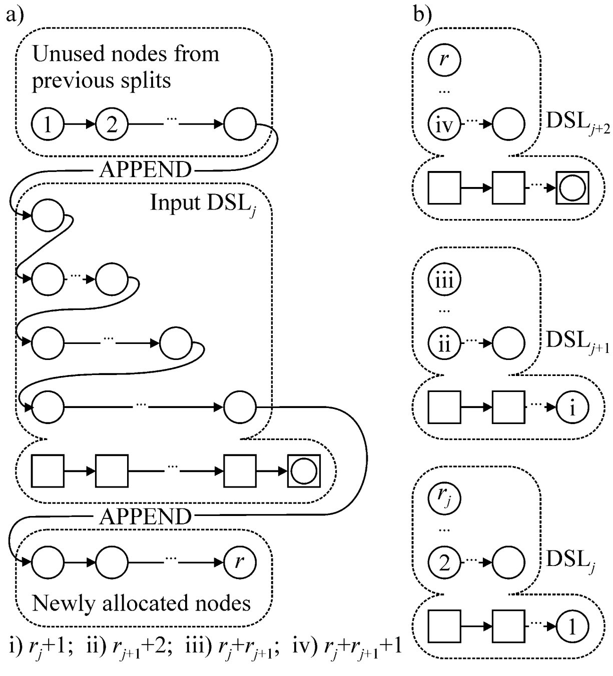

a)
Unused nodes from previous splits
1
2
APPEND
Input DSL$_j$
APPEND
$r$
Newly allocated nodes
i) $r_j$+1; ii) $r_{j+1}$+2; iii) $r_j$+$r_{j+1}$; iv) $r_j$+$r_{j+1}$+1
b)
$r$
iv
DSL$_{j+2}$
iii
ii
DSL$_{j+1}$
i
$r_j$
2
DSL$_j$
1

Figure 4: Bricklaying DSL splitting approach: a) global list of recyclable nodes, and b) three output DSLs.

## 3 Results and analysis

The number of strips in the HT-DSL approach was experimentally set in the range $m = \Theta(\sqrt{n})$. Consequently, the number of points in a strip is $q = O(\sqrt{n})$ in an optimal uniform point distribution. We have retained this result in the DP-DSL approach, and experimentally determined the best performance by splitting the strips reaching $q = \lceil 3\sqrt{n} \rceil$ points. We use (1, 3)-DSLs in the HT-DSL approach, and (2, 5)-DSLs in the DP-DSL approach. The best long-term performance was achieved by using the gap size correction parameter $gsc = 1$.

In Table 1, expected time complexities for handling the considered representative cases by the HT-DSL and DP-DSL approach are given. We consider Gaussian distribution of points with an additional isolated point (Fig. 2), few non-uniformly distributed clusters (Fig. 3), uniform distribution (Fig. 5a), grid (Fig. 5b), two GIS datasets (Figs. 5c and 5d), and the so-called *ladder* with an additional isolated point (Fig. 6b). The time ratios in the second column were obtained for configurations of 5.000.000 points. Much lower cardinalities were used in figures (except Fig. 5) to maintain clarity. The realistic examples in Figs. 5c-d consist of 70.334 and 193.360 points, respectively.

By choosing the number of strips in the range $m = \Theta(\sqrt{n})$, the average horizontal distance between two successive points in the DSL of an interval strip may be considered similar to the average strip width. Consequently, the local search mostly examines only a few nearest-point candidates, while the inter-strip search also traverses only a few strips. Both numbers may be considered O(1) and thus, the expected total search time for $n$ query points does not exceed $O(n)$. The expected total times $O(n \log n)$ for the first six examples in Table 1 are therefore determined by the construction phase (see Table 2). Exceptionally, the HT-DSL approach in the first two examples (Figs. 2 and 3) collects $n$ - 1 points in a single strip, and the local search time can be hardly considered $O(1)$. Significantly slower performance can be noticed in comparison to the DP-DSL approach, although the theoretical worst-case time complexity $\Theta(n^2)$ is not reached in this two cases.

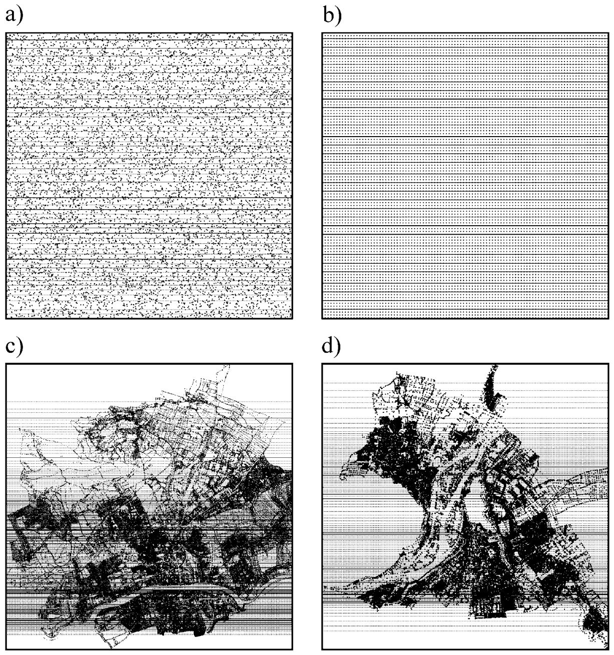


Figure 5: Dynamic partition into strips for: a) uniform points distribution, b) grid, and c-d) two GIS datasets.

While the order of inserting the points was considered random in the above six examples, the ladder illustrated in Fig. 6 and analysed in the last two rows of Table 1 was synthetically generated and represents the worst-case for the local search, which requires $\Theta(r)$ time if there are $2r$ points in an interval strip (see Fig. 6a). The condition $w < h$ assures that the nearest point of any $p_{r+i} \in l_2$ is exactly the other end of the same ladder rung i.e. $p_i \in l_1$. Another requirement $w > x_r - x_1$ provides the arrangement $x_j < x_{j+1}$ for every $j < 2r$. Consequently, exactly $r$ points $p_{r+i-1}$, $p_{r+i-2}$, …, $p_i$ have to be examined for every query point $p_{r+i}$ on $l_2$, $0 < i \leq r$. Thus, the HT-DSL approach requires $\Theta(n^2)$ total local search time to handle the case in Fig. 6b with $2r = n - 1$ points. The DP-DSL approach handles the same case in $\Theta(n\sqrt{n})$ time in a similar manner as the example from Fig. 6c is handled.

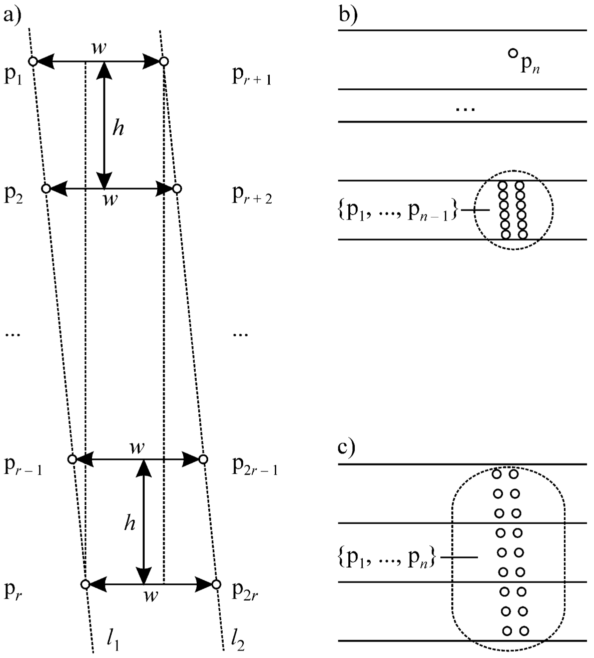


Figure 6: A ladder: a) point organization, b) HT-DSL case with $\Theta(n^2)$ total local search time, and c) case requiring $O(n\sqrt{n})$ time in both approaches.

| Fig. | HT-DSL/DP-DSL | HT-DSL time | DP-DSL time |
|---|---|---|---|
| 2 | 9.91 | $\Theta(n \log n)$ to $O(n^2)$ | $\Theta(n \log n)$ |
| 3 | 7.21 | $\Theta(n \log n)$ to $O(n^2)$ | $\Theta(n \log n)$ |
| 5a | 0.79 | $\Theta(n \log n)$ | $\Theta(n \log n)$ |
| 5b | 0.84 | $\Theta(n \log n)$ | $\Theta(n \log n)$ |
| 5c | 0.50 | $\Theta(n \log n)$ | $\Theta(n \log n)$ |
| 5d | 0.47 | $\Theta(n \log n)$ | $\Theta(n \log n)$ |
| 6b | 203.73 | $\Theta(n^2)$ | $\Theta(n\sqrt{n})$ |
| 6c | 0.59 | $\Theta(n\sqrt{n})$ | $\Theta(n\sqrt{n})$ |

Table 1: Expected time complexities of HT-DSL and DP-DSL approaches in considered representative examples.

## 3.1 Theoretical time complexity analysis

Table 2 lists theoretical worst-case time complexities of all phases of both approaches. The construction of strips and maintenance of DSLs are optimal in both cases, while the total local and total inter-strip search time do not provide the desired $O(n \log n)$ total time and require further consideration.

Besides other interesting cases, we have managed to construct one which requires $\Theta(n^2)$ inter-strip search time in the DP-DSL approach. In Fig. 7, points $p_1$, …, $p_r$ are placed on slightly descending vectors in strips $SoP_2$ to $SoP_m$, and $p_{r+1}$, …, $p_n$ are arranged from left to right in $SoP_1$. These latter ($p_i$, $i > r$) further fulfil the following conditions.

1. In each $SoP_j$, $j > 1$, distances to $p_i$ descend from left to right along the bold line segment (the nearest point to $p_i$ is the rightmost and the bottommost one).
2. $p_u \in SoP_{j+1}$, $p_v \in SoP_j$, $j > 1 \Rightarrow |p_i p_u| < |p_i p_v|$ i.e. upper bold segments are closer to $p_i$ than lower ones.
3. Point $p_{i+1}$ is closer to any point in $S_{r+1}$ than to $p_i$.

We use auxiliary pairs of circular arcs to graphically emphasize the above conditions 1 and 2. The left and right arc in each pair are centred in $p_n$ and $p_{r+1}$, respectively. The third condition is fulfilled by halving the horizontal distance between the query point and the farthest target point candidate $p_{FAR}$ from $S_{r+1}$ in each iteration. For each point from $SoP_1$, all $r$ points from $S_{r+1}$ have to be examined until the nearest point $p_{NEAR}$ is found, thus the total time is raised for $\Theta((n - r)r)$. Selection $r \approx n/2$ obviously leads to $\Theta(n^2)$ time. Note that $SoP_1$ represents a line strip in the DP-DSL approach and thus it is allowed to contain $O(n)$ points. Anyway, $x$-coordinates of $p_{r+1}$, …, $p_{n-1}$ in the considered example represent a geometric progression with ratio 2. Even with relatively low $n$ and really small $d$, the exponential growth quickly produces $x$-coordinates out of the range of the IEEE 754 floating-point specification. If we choose $d = 1$, $n = 1000$, $r = 500$ and $x_{501} = 0$, for example, then we get $x_{1000} = 2^{499} \approx 1.6*10^{150}$, which is usually far beyond the expected range in industrial, GIS and other practical applications.

| Phase | HT-DSL | DP-DSL |
|---|---|---|
| *Construction* | | |
| Strip identification | $O(n)$ | $O(n \log n)$ |
| Point insertion | $\Theta(n \log n)$ | $\Theta(n \log n)$ |
| DSL splitting | 0 | $O(n)$ |
| Maintenance of *Borders* | 0 | $O(\sqrt{n} \log n)$ |
| *Quering* | | |
| Local search | $\Theta(n^2)$ | $\Theta(n\sqrt{n})$ |
| Inter-strip search | $\Theta(n^2)$ | $\Theta(n^2)$ |

Table 2: Worst-case time complexities of particular phases in the HT-DSL and DP-DSL approach.

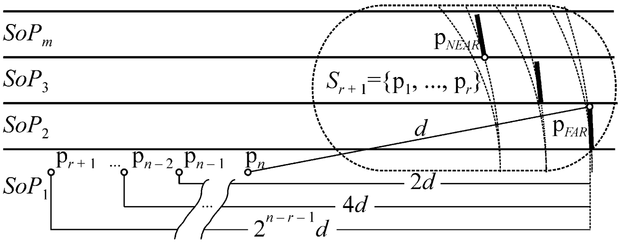


Figure 7: Construction of the $\Theta(n^2)$ time example based on geometric progression.

Note that the total local search time of the DP-DSL approach could be improved from $O(n\sqrt{n})$ to $O(n \log n)$ by splitting the strips of size $q = O(\log n)$ instead of current $q = \lceil 3\sqrt{n} \rceil$. Although this change does not critically increase theoretical worst-case time complexities of other phases, it usually results in slower practical performance due to the increased number of DSL splits and initial positioning operations in much more DSLs during the inter-strip search.

## 3.2 DP-4DSLs approach

We have recently developed an engineering solution which handles the considered problematic examples in the desired (optimal) time bounds. It additionally performs the vertical DP. In each strip, two orthogonal DSLs are constructed, the horizontal one sorted by the *x*-coordinate, and the vertical one sorted by the *y*-coordinate. Each point is therefore placed into four DSLs: XH-DSL (also in the DP-DSL approach) and YH-DSL are assigned to each horizontal strip, and XV-DSL and YV-DSL are constructed in each vertical strip (see Fig. 8). Note that the strip splitting threshold can be found by help of the orthogonal DSL in a quicker way than with the aforementioned SELECT algorithm.

In each iteration of the local search, the method performs one move in each DSL which are all addressing the same radius *d*. The nearest point is found when the first DSL (the winner) examines all the points within the distance *d* around the query point. We have not managed to theoretically prove optimal time complexity but the performance in the considered cases (I to VI in Table 3) appears promising.

- The cases I and II were already considered in Table 1. In the DP-4DSLs approach, the nearest-point of any point p is its direct predecessor in the YH-DSL. The total local search and inter-strip times are both $\Theta(n)$ and thus the total time $\Theta(n \log n)$ is determined by the construction phase.
- In the ladder example rotated 90 degrees (case III), the local search in the HT-DSL and the DP-DSL approach examines at most two points, but the inter-strip search traverses $\Theta(\sqrt{n})$ strips for half of the query points, resulting in $\Theta(n\sqrt{n})$ total time. In the DP-4DSLs approach, XV-DSL has the same role as YH-DSL has in cases I and II, resulting in $\Theta(n \log n)$ total time.
- Case IV was addressed by the HT-DSL and DP-DSL approach in Section 3.1 already. In the DP-4DSLs approach, however, XV-DSL provides $O(1)$ local search and inter-strip search times and thus the total time $\Theta(n \log n)$ is determined by the construction phase.
- In case V where the configuration from case IV is rotated 90 degrees, YH-DSL has the same role as XV-DSL has in case IV, and optimal $\Theta(n \log n)$ is again achieved. For the HT-DSL and the DP-DSL approaches, the same conclusions can be made as in case II.
- "Regular" cases (VI) refer to those configurations, where an optimal $\Theta(n \log n)$ time complexity is expected (see Fig. 5 and Table 1) within both, HT-DSL and DP-DSL approach. Of course, the same optimal time complexity is expected by the DP-4DSLs approach because the winner can either be XH-DSL (also used in the HT-DSL and DP-DSL approaches) or some other DSL outperforming XH-DSL. Note that the HT-DSL and DP-DSL approaches usually outperform the DP-4DSLs approach in "regular" cases as maintenance of two partitions and four DSLs is quite expensive.
- Note that the remaining YV-DSL, which is not met in the considered examples, is also necessary. It is for example the winner if a "regular" case won by XH-DSL is rotated 90 degrees.

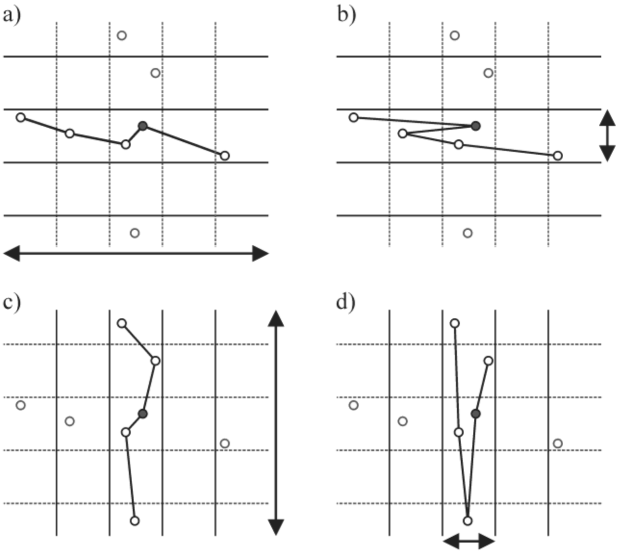


Figure 8: A query point (filled grey) is in four DSLs: a) XH-DSL, b) YH-DSL, c) YV-DSL, and d) XV-DSL.

| Case | HT-DSL | DP-DSL | DP-4DSLs | Winner |
|---|---|---|---|---|
| I | $\Theta(n^2)$ | $\Theta(n\sqrt{n})$ | $\Theta(n \log n)$ | YH |
| II | $\Theta(n\sqrt{n})$ | $\Theta(n\sqrt{n})$ | $\Theta(n \log n)$ | YH |
| III | $\Theta(n\sqrt{n})$ | $\Theta(n\sqrt{n})$ | $\Theta(n \log n)$ | XV |
| IV | $\Theta(n^2)$ | $\Theta(n^2)$ | $\Theta(n \log n)$ | XV |
| V | $\Theta(n\sqrt{n})$ | $\Theta(n\sqrt{n})$ | $\Theta(n \log n)$ | YH |
| VI | $\Theta(n \log n)$ | $\Theta(n \log n)$ | $\Theta(n \log n)$ | various |

Table 3: Comparison of the three approaches in the considered cases: I – ladder with an isolated point, II – ladder, III – ladder rotated 90 degrees, IV – geometric-progression-based case from Fig. 7, V – case from Fig. 7 rotated 90 degrees, and VI – "regular case".

# 4 Conclusion

The paper considers a new (DP-DSL) approach to the incremental nearest-point search in 2-D. It guarantees $\Theta(\sqrt{n})$ interval strips, each containing $\Theta(\sqrt{n})$ points and, therefore, successfully prevents situations with over-populated interval strips and decreases the total local search time from $O(n^2)$ to $O(n\ \sqrt{n})$. In our opinion, this is an important acceleration, although the algorithm still fails to achieve an optimal $O(n \log n)$ time performance characteristic for the preliminary points arrangement approach. In addition, examples can be constructed (although hardly met in practice) which, just as the "traditional" HT-DSL approach still achieve quadratic inter-strip search time. The DP-4DSLs variant seems to solve the considered problematic examples in optimal time, but a formal proof is still missing. Construction of the Voronoi diagram on $\Theta(\sqrt{n})$ points and utilization of two perpendicular DSLs in each Voronoi cell could have a potential, but one should first prove that such dynamic partition is generally possible, and then provide an efficient region splitting algorithm.

## Acknowledgement

This work was partially supported by the Slovenian Research Agency (research programme P2-0041).

## References

[1] F. Aurenhammer F. (1991). Voronoi diagrams – a survey of a fundamental geometric data structure, *ACM Computing Surveys*, ACM, vol. 23, iss. 3, pp. 345-405.

[2] Chan T. M.; Rahmati Z. (2017). Approximating the minimum closest pair distance and nearest neighbor distances of linearly moving points. *Computational Geometry*, Elsevier Science, vol. 60, Jan. 2017, pp. 2-7. https://doi.org/10.1016/j.comgeo.2016.04.001

[3] Chaurasia V.; Chaurasia V. (2016). Statistical feature extraction based technique for fast fractal image compression. *Journal of Visual Communication and Image Representation*, Elsevier Science, vol. 41, Nov. 2016, pp. 87-95. https://doi.org/10.1016/j.jvcir.2016.09.008

[4] Cleary J. G. (1979). Analysis of an Algorithm for Finding Nearest Neighbors in Euclidean Space. *ACM Transactions on Mathematical Software*, ACM, vol. 5, iss. 2, pp. 183-192. https://doi.org/10.1145/355826.355832

[5] Cormen T. H.; Leiserson C. E.; Rivest R. L.; Stein C. (2001). *Introduction to Algorithms, 2nd Edition*, MIT Press and McGraw-Hill.

[6] de Gomensoro Malheiros M.; Walter M. (2016). Spatial sorting: an efficient strategy for approximate nearest neighbor searching. *Computers & Graphics*, Elsevier Science, vol. 57, June 2016, pp. 112-126. https://doi.org/10.1016/j.cag.2016.03.006

[7] Gómez-Ballester E.; Micó L.; Oncina J. (2006). Some approaches to improve tree-based nearest neighbour search algorithms. *Pattern Recognition*, Elsevier Science, vol. 39, iss. 2, pp. 171-179. https://doi.org/10.1016/j.patcog.2005.06.007

[8] Guibas L. J.; Knuth D. E.; Sharir M. (1992). Randomized incremental construction of Delaunay and Voronoi diagrams. *Algorithmica*, Springer, vol. 7, iss. 4, pp. 381-413. https://doi.org/10.1007/BF01758770

[9] Han Y.; Tang J.; Zhou Z.; Xiao M.; Sun L.; Wang Q. (2014). Novel itinerary-based KNN query algorithm leveraging grid division routing in wireless sensor networks of skewness distribution. *Personal and Ubiquitous Computing*, Springer, vol. 18, iss. 8, pp. 1989-2001. https://doi.org/10.1007/s00779-014-0795-y

[10] Kanda T.; Sugihara K. (2002). Comparison of various trees for nearest-point search with/without the Voronoi diagram. *Information Processing Letters*, Elsevier Science, vol. 84, iss. 1, pp. 17-22. https://doi.org/10.1016/S0020-0190(02)00221-1

[11] Kirkpatrick, D. G. (1983). Optimal search in planar subdivisions, *SIAM J. Comput.*, vol 12, iss. 1, pp. 28-35. https://doi.org/10.1137/0212002

[12] Long Y.; Zhu F.; Shao L. (2016). Recognising occluded multi-view actions using local nearest neighbour embedding. *Computer Vision and Image Understanding*, Elsevier Science, vol. 144, March 2016, pp. 36-45. https://doi.org/10.1016/j.cviu.2015.06.003

[13] Munro J. I.; Papadakis T.; Sedgewick R. (1992). Deterministic skip lists. *Proceedings of the Third ACM-SIAM Symposium on Discrete Algorithms*, Society for Industrial and Applied Mathematics, Orlando, USA, pp. 367-375.

[14] Nagasue J.; Konishi Y; Araki N.; Sato T.; Ishigaki H. (2009). Slope-Walking of a Biped Robot with K Nearest Neighbor Method. *Proceedings of the 2009 Fourth International Conference on Innovative Computing, Information and Control*, IEEE Computer Society, Kaohsiung, Taiwan, pp. 173-176. https://doi.org/10.1109/ICICIC.2009.333

[15] Pugh W. (1990). Skip lists: A probabilistic alternative to balanced trees. *Communications of the ACM*, ACM, vol. 33, iss. 6, pp. 668-676. https://doi.org/10.1145/78973.78977

[16] Savchenko A. V. (2017). Maximum-likelihood approximate nearest neighbor method in real-time image recognition. *Pattern Recognition*, Elsevier, vol. 61, Jan. 2017, pp. 459-469. https://doi.org/10.1016/j.patcog.2016.08.015

[17] Wang. H.; Cao J.; Shu L.; Rafiei D. (2013). Locality sensitive hashing revisited: filling the gap between theory and algorithm analysis. *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, ACM, San Francisco, USA, pp. 1969-1978. https://doi.org/10.1145/2505515.2505765

[18] Wei-Kleiner F. (2016). Tree decomposition-based indexing for efficient shortest path and nearest neighbors query answering on graphs. *Journal of Computer and System Sciences*, Elsevier Science, vol. 82, iss. 1, part A, pp. 23–44. https://doi.org/10.1016/j.jcss.2015.06.008

[19] Zadravec M.; Brodnik A.; Mannila M.; Wanne M.; Žalik B. (2008). A practical approach to the 2D incremental nearest-point problem suitable for different point distributions. *Pattern Recognition*, Elsevier Science, vol. 41, iss. 2, pp. 646-653. https://doi.org/10.1016/j.patcog.2007.06.031

[20] Zadravec M.; Žalik B. (2005). An almost distribution-independent incremental Delaunay triangulation algorithm. *Visual Computer*, Springer, vol. 21, iss. 6, pp. 384-396. https://doi.org/10.1007/s00371-005-0293-3

[21] Zheng R. Y. (2010). *Machine Learning Approaches to Bioinformatics*, World Scientific Publishing Co., Inc. https://doi.org/10.1142/7454

# Towards a UML Profile for the Simulation Domain

Maouche Mourad and Bettaz Mohamed
Faculty of Information Technology-Philadelphia University, Jordan
E-mail: mmaouch@philadelphia.edu.jo, mbettaz@philadelphia.edu.jo
http:\\www. philadelphia.edu.jo

**Keywords:** simulation engineering, simulation platforms, model driven approaches, MDA, UML profile**.**

**Received:** May 12, 2017

*Model driven approaches have recently been exploited to implement simulation systems. Most of the reported contributions have adopted the Model Driven Architecture (MDA), a model driven approach widely used in software engineering. Simulation Platform Description Models (SPDM), which are first citizens MDA models intended for the description of simulation platforms supporting the execution of simulation experiments, are not explicitly considered in the previous works. The purpose of this work is to define a UML profile intended for the modelling of both simulation core concepts and simulation platforms. The contribution of this work is threefold: First we review and synthesize recent contributions in modelling and simulation approaches, practices and platforms; second we propose a resource-oriented approach for the modelling of simulation platform elements; third we consider both component- and workflow-based simulation platforms.*

*Povzetek: Predlagana je nova plaltforma za simulacije na osnovi virov.*

## 1 Introduction

Recent research works recommended the practice of the Model Driven Engineering (MDE) in the field of the simulation [2, 3, 4]. The works in [5, 6, 7, 11] adopted approaches based on three steps: A conceptual modelling step where scientists or engineers build models, called Computational Independent Models (CIM) capturing the phenomena under study, a design step where simulation engineers build models called Platform Independent Simulation Model (PISM), and an implementation step where software engineers develop models called Platform Specific Simulation Model (PSSM). A series of model transformations allow to derive PISM models from CIM models, and PSSM models from PISM models. Furthermore the Model Driven Architecture (MDA), a variant of the MDE approach standardized by the Object Management Group (OMG) and targeting the software engineering community, emphasizes also another kind of model called Platform Description Model (PDM) [1]. This kind of model is used for the description of platforms that host the developed software applications.

The MDA approach has been adopted, for instance, in the field of the real-time and embedded systems. UML profiles, like UML-MARTE [8], or Domain Specific Languages like AADL [27] have been defined to support the MDA practice in this field. Both provide mechanisms to describe PDM models, and, seem to be good candidates for the modelling of (software) simulation platforms. In a previous work [22], we proposed a software engineering methodology for the development of multiscale modelling and simulation framework based on the UML-MARTE profile where an attempt to model the multiscale platform MUSCLE using the ingredients of the SRM (Software Resource Model) sub-profile of UML-MARTE have been conducted. The SRM sub-profile is dedicated for the modelling of real time operating systems and middleware. Although it offers a wide range of (software) modelling elements and capabilities, most of them target the specific needs of the real time and embedded systems platforms, and with regard to our previous attempt [22], do not meet specific simulation engineering needs. According to our opinion, it is more natural and comfortable for the simulation engineering community to treat and manipulate their specific native entities and concepts as first class modelling elements.

To the best of our knowledge, none of the current works on the MDE practices in the simulation field, addresses the issue of the Simulation Platform Description Model (SPDM), i.e., the description of simulation platforms that support the execution of simulation experiments. The sole work targeting the modelling of simulation platforms is reported in [9]. Discovering the commonalities and variations among a sample of open source multi-physics simulation platforms has been the main motivation of its authors. Although the work in [9] may serve as a reference architecture for simulation platforms developers, it does not offer, in our opinion, explicit mechanisms to develop models describing simulation platforms in the spirit of the MDA approach.

The objective of this work is to define a UML profile for the simulation field intended to support the MDA practices in this field. The proposed profile particularly provides a set of appropriate modelling mechanisms for the description of simulation platforms.

The contribution of this work is threefold: First we review and synthesize recent contributions in modelling and simulation approaches, practices and platforms; second we adopt a resource-oriented approach for the modelling of simulation platform elements; third we consider both component- and workflow-based simulation platforms. These contributions are illustrated by a set of UML stereotype classes capturing core simulation concepts and platforms elements.

The rest of this paper is organized as follows. Section 2 is devoted to the recent developments in modelling and simulation field. Section 3 presents the simulation field from the workflow perspective. Related works are discussed in Section 4. Our contribution is detailed in Section 5. Section 6 outlines a simple example. Finally conclusions and future works are given in Section 7.

# 2 Recent developments in simulation engineering

Simulation engineering, an emerging discipline that applies the principles of both simulation science and engineering fields, has been widely used to address various complex real-world problems. It mainly involves two complementary activities: 1) a modelling activity where simulation models of physics phenomena or engineering artefacts- are built, 2) a simulation activity where experiments are performed on these simulation models to achieve specific objectives such as understanding of phenomena, predictions, and performance study. The simulation engineering community developed a lot of specific software tools allowing not only to build such models but also to conduct experiments on them. The literature reports various terminology to designate such tools, like simulation frameworks or simulation platforms; simulation platform is the designation that will be used along this paper to designate such simulation tools. A multitude of academic and commercial simulation platforms are available [10]: Some of them are domain dependent while others are generic. MUSCLE [6] and Mapper [12] simulation frameworks proposed generic simulation platforms. Domains where simulation is widely used are numerous: Physics, biology, medicine, and others. Integrated Plasma Simulator (IPS) platform [13], and Virtual Imaging Platform (VPM) [25] are respectively simulation platforms dedicated to the plasma physics and medical imaging domains.

Due to the profusion of concepts, methods, frameworks and tools related to the modelling and simulation field, we present in the following a synthesis addressing advanced issues relevant to this field.

## 2.1 Modelling and simulation core concepts

A model is an abstract representation of reality. One of the practical uses of models is generating the dynamic of systems from their models. Simulation consists in moving a model over time, given some inputs. Models can be either in a mathematical form, i.e., a system of equations for example, or in an algorithmic form: In the first case the simulation takes the form of a kind of software, named simulator, that implements a solver for this system of equations; in this case models, often specified thanks to domain specific modelling languages, and simulators are separated. Solvers may be categorized according to different criteria such as their application domain and their solving methods. They may be either legacy code or newly developed codes. In the second case, models are specified in terms of algorithmic components; models are embedded in the simulation code. In our work we deal with both cases.

Simulation codes accept well defined scripts as inputs. These scripts specify the set-up and the protocol of the targeted experiments. Simulation engines interpret the input scripts and run the simulation of individual models. Simulation scripts are usually written thanks to specific scripting languages like Python, and Ruby, or in the form of standardized data representation languages like XML.

## 2.2 Modelling and simulation approaches

Modern modelling and simulation approaches distinguish between the monolithic approach and the partitioned one. In the first approach a single large scale model capturing the whole phenomena under study is built and then its associated simulation code is executed, while in the second one, a complex model is partitioned into a set of single models and then their associated individual simulation codes are coupled and then executed together.

### 2.2.1 Partitioned methods

A categorization of partitioned methods is given in[26]:

(i) Multiphysics Partitioning
This method is used when the model of the phenomena under study captures multiple physical processes, each of these physical processes belongs to a specific physics such as temperature and viscosity. In this case the model is decomposed into a set of sub-models; each of these sub-models concerns a specific physical process, and all sub-models of the model operate on the same time and space scales.

(ii) Multiscale Partitioning
This method is used when the model of the phenomena under study captures only one physical process; this model, because of its complexity, is decomposed into a set of sub-models that operate on different time and space scales.

(iii) Multiphysics Multiscale Partitioning
Here multi-scale and multi-physics methods are both used. This method is used when the model of the phenomena captures multiple physical processes that don’t operate on the same scales.

Partitioned simulations encompass not only the performance of a set of single simulation experiments but also the interactions between these single simulation experiments. It presupposes the availability of specific mechanisms, called coupling mechanisms, having the mission to drive these interactions. Two issues need to be addressed when coupling single experiments:

(i) The format of the data exchanged between coupled simulation experiments,

(ii) The interaction pattern governing the interaction between coupled simulation patterns.

The same approach, based on a usual programming technique called *wrapping*, is generally used on almost all simulation platforms that deal with the experiments coupling issues. The wrappers are pieces of code that embodies the simulation code of single experiments. For instance the layered architecture of the Integrated Plasma simulation platform described in [13] distinguishes between data wrappers and coupling wrappers:

The data wrapper takes in charge the data conversion from the internal data format used by single experiments into a common exchange data format. The European Fusion research community suggested a generic data structure, named Consistent Physical Objects (CPO), as a common format for the data to be exchanged between single experiments. Data wrappers are not simulation platforms dependent.

The coupling wrapper takes in charge the data motion as well as the pattern of the interaction between coupled single experiments during their data exchanges. Coupling wrappers, contrarily to data wrappers, are simulation platform dependents.

### 2.2.2 Coupling issue

In [14] the authors laid the foundations of multi-scale computing. Their formalization of the multi-scale coupling reveals two complementary features related to this concept:

(i) Coupling template: Specifying the information flow that may occur between any pair of coupled (single) experiments. Unidirectional as well as a bidirectional data flows are admitted.

(ii) Coupling topology: A graph representing the couplings (edges) between pair of single sub-models (nodes) belonging to a partitioned model. The graph edges are labelled by coupling templates. Two kinds of coupling topology are identified:

   a. Acyclic topology: It is characterized by an absence of cycles in the coupling topology. In this case coupled simulation codes can be ordered and executed sequentially; this kind of coupling is also named loose coupling.

   b. Cyclic topology: It is characterized by the presence of cycles in the coupling topology. In this case the order of the execution of individual simulation codes is not predefined; this kind of coupling is also called tight coupling.

Figure 1.a and Figure 1.b [14] depict respectively the loose and the tight coupling of three sub-models belonging to a partitioned model. The arrows show the direction of their interactions.

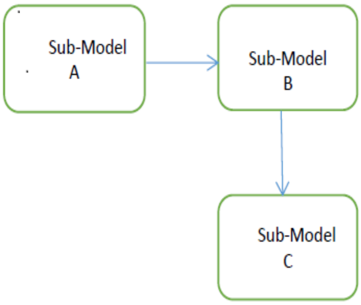


Figure 1.a: Loose Coupling of Sub-Models.

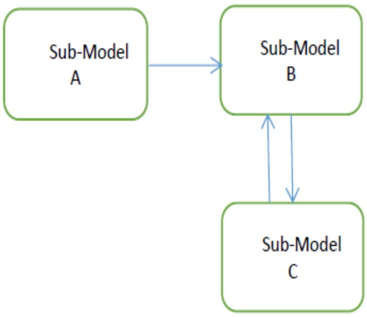


Figure 1.b: Tight Coupling of Sub-Models.

## 2.3 Orchestration of coupled simulations

Three ways to coordinate and orchestrate a set of coupled single experiments are commonly used:

(i) Centralized mode: A dedicated engine orchestrates and coordinates the enacting of single experiments according to predetermined patterns.

(ii) Master/Slave mode: One of the single experiments plays the role of a master. First, the master experiment is enacted and then the master experiment orchestrates the enactment of the other single experiments, called slave experiments, in a sequential way.

(iii) Component based mode: The coordination is distributed over all the participating single experiments.

# 3 Simulation from the scientific workflow perspective

The workflow technology, mainly used by the business community, seems to be one of the promising approaches adopted by the scientific community; the concept of scientific workflows emerged as an alternative to the conventional concept of business workflow. There are similarities as well as differences between the two kinds of workflows. For example, business workflows are control-flow oriented, while scientific workflow are mainly data-flow oriented. The readers interested in more details may refer to [15].

### 3.1 Scientific workflows

A workflow is a pre-defined set of work steps with a partial order on these steps [17]. Work steps represent tasks to be carried out when they are enacted by workflow engines.

Scientific workflows Management Systems have been developed during the last two decades. They are intended to manage, enact and monitor scientific workflows which are a composition of a series of computation and/or data manipulation [13]. Scientific workflows are enacted and orchestrated by specific engines, called workflow engines, forming the core components of scientific workflows Management Systems. Some examples of known scientific workflows management systems are Taverna, Kepler, and Vistrails [16].

Generally, workflows describe control flows and/or data flows. Scientific workflows are usually classified into two categories: Abstract and concrete workflows [19]. Quoting the authors of [18]: "An abstract scientific workflow is a definition of a scientific process with emphasis on the analytical operations or function to be performed rather than the mechanisms for performing these operations". In opposite, concrete scientific workflows bind the work steps to resources that execute the corresponding tasks.

### 3.2 Simulation workflows

Simulations of scientific or engineering models are seen as kinds of scientific workflows. Simulations of models are often described by scientific workflows. These workflows follow specific patterns/motifs and include various kinds of steps: Data processing steps, solving/simulation step, visualization step, and data exchanges step. In [24] the authors elaborated catalogues of common motifs for both scientific workflows and data operations that may be performed when conducting scientific experiments.

The iterative pattern is one of the most used control patterns to describe the workflow of individual experiments. For instance structured loops are a kind of iterative pattern.

In the case of a multi-experiment the workflows of the participating individual experiments are coupled. Their coupling is performed thanks to a set of data exchanges constrained by specific interaction patterns. The authors of [20] suggest the concept of "choreography", borrowed to the business management community, to couple the workflows of single experiments. Every single experiment is realized as an orchestration of scientific services and the whole multi-experiment is described by choreographies without a centralized control.

## 4 Related works

The literature reports two different directions regarding the development of simulation frameworks:

(i) Component based approaches inspired from the software component-based design and programming methods,

(ii) Workflow based approaches inspired from the workflow based business systems. Recent works with respect to each of these two research directions emphasize the MDA practices.

In [7] the authors proposed a simulation framework based on the hierarchical component-based approach. Their framework is supported by well-defined meta-models capturing Conceptual Simulation Models (CSM) as well as Platform Independent Simulation Models (PISM). However they did not define meta-models that capture Platform Specific Simulation Models (PSSM); in fact these are considered as implementations of PISM models. PISM and PSSM terminology used in the simulation field corresponds respectively to the PIM and PSM terminology used in the software engineering field. It is worthwhile to note that the work in [7] does not consider the simulation platform description models as primary models.

The authors in [21] adopted a workflow based approach for the simulation framework they developed. Their approach, based on an MDA approach too, relies on three distinct levels: A conceptual level at which the modellers describe the models that capture the phenomena under study; an abstract level at which PSSM models, independent form the computing infrastructures are conceived; a concrete level at which models are strongly dependent from the computing infrastructure intended to host the simulation experiments; these last models, called Platform Description Models (PDM) refer to the hardware infrastructure rather than to the simulation workflow framework. Conceptual models are first transformed into specific intermediate representations which are themselves converted to abstract workflows to be enacted by a targeted scientific workflow framework.

Both research works does not consider the modelling of simulation platforms. To the best of our knowledge, the sole research work that investigated the issue of simulation platform modelling is described in [9]. Its authors aimed at discovering commonalities and variations among a sample of open source multi-physics simulation platforms, and proposing a feature model capturing the discovered commonalities and variations using the feature-oriented modelling approach. According to the authors, one of the possible uses of their produced feature model is to serve as a reference for simulation platforms developers.

Our research work, contrarily to [9], targets the modelling of simulation platforms in the context of the MDA approach for the simulation domain, i.e., providing a UML profile intended to build Simulation Platform Description Models (SPDM) for simulation experiments; in opposite to [21], PDM models here refer to simulation platform models rather than to computing infrastructure models.

The present work considers scientific workflows for the description of scientific experiment behaviors, and relies on the concept of generic resources as defined in [8] to model elements of simulation platforms.

# 5 The proposed UML profile

In this section we develop our UML profile intended for the simulation field. A set of UML stereotypes intended to capture core concepts of the simulation domain are exposed.

## 5.1 Linking PISM and SPDM models

The proposed profile focuses on the SPDM modelling. Figure 2 depicts the well-known relationship between the PISM, and PSSM models. Elements of PISM models are mapped to elements of SPDM leading to PSSM models.

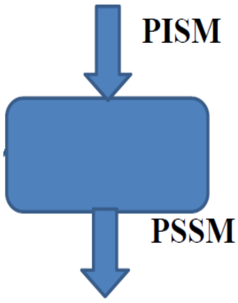


Figure 2: Linking PISM and PSSM.

Our approach relies on two first class UML model elements to describe simulations:

-*Experiment*: intended to describe the simulation of either a monolithic model or the simulation of a single model (member of a partitioned model).

-*Simulation*: intended to describe the architecture of the simulation of a whole model (either monolithic or partitioned model) according to a desired simulation approach (monolithic/partitioned) and design (component-based/ workflow-based).

## 5.2 PISM model elements

In this section we identify and define a set of UML stereotypes that constitutes the main PISM model elements of our profile.

### 5.2.1 Simulation stereotype

The simulation and experiment concepts, as defined above, are modelled as stereotypes. Both extend the UML *BehavioredClassifier* metaclass which is a UML classifier that owns behaviors.

A. The class diagram depicted in Figure 3.a describes the *Simulation* stereotype and the hierarchy of its refined stereotypes covering various kinds of simulation approaches.

Comments:

*(i). Simulation* Stereotype includes at least two properties.

*IdentifierElts* reports a set of required elements that may identify and characterize conducted simulations such their identification number, their date, the target domain, the version number.

*SimulParam* is used to report some parameters related to the simulation itself; for instance the duration of the simulation, the space dimension of the simulated model and others.

*(ii). PartitionedSimulation* and *MonolithicSimulation* are refinements of the *Simulation* Stereotype. *Expnumber* property defined in *PartitionedSimulation* records the number of single experiments participating to a partitioned simulation instance.

*(iii).MultiscaleSimulation is* a refinement of the *PartitionedSimulation stereotype.* Its *scales* property records the kinds of scale dimension (time, space, time and space) characterizes a simulation instance.

*Dimension* is an enumeration type intended to carry various kinds of scales.

Dimension == time | space | time&space |…..

A.The class diagram shown in Figure 3.b presents a hierarchy of various multiscale simulation design approaches according to the way coordination and orchestration of coupled single experiments are done.

Comments:

(i). *CompBasedMsc* stereotype represents multiscale simulations designed according to the component based approach. *Conf* property records the configuration of multiscale simulations i.e., its topology (refer to section 2.2.2). We introduce the stereotype *Coupling* as an extension of the UML Association metaclass to model the simulation configuration. The details of this stereotype are shown in Figure 3.d.

(ii). *WrkFlowBasedMsc* stereotype represents multiscale simulations designed according to the workflow based approach. It is mainly characterized by two properties. *Wbeh* property specifies the abstract workflow associated with the workflow based multiscale simulation. The *WorkFlowBeh* stereotype is defined in the part A of section 5.2.3

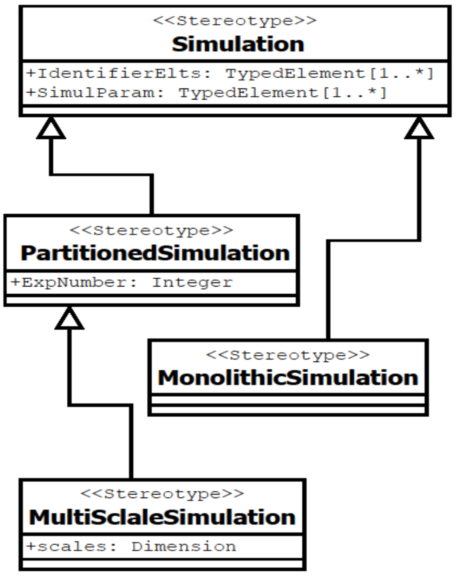


Figure 3.a: Simulation Stereotype and its Refined Stereotypes. Remark: UML TypedElement refers to a pair (named element, its associated type). TypedElement [0..*] means zero or more.

*Map* property specifies the mapping between workflow nodes and their corresponding workflow *call actions*. The *Mapping* class is a datatype that records (workflow node, action to be called) pairs. The concept of *UML call action* is detailed in the part B of section 5.2.3.

(iii). *CentralizedMsc* stereotype represents multiscale simulations designed according to the centralized version of the workflow based approach. It refines *WrkFlowMsc* stereotype. Its *coord* property (instance of the Coordinator class) is intended to represent the central coordinator that orchestrates the whole simulation workflow. The *Coordinator* class is not detailed in this paper.

(iv). *MasterSlaveMsc* stereotype represents multiscale simulations designed according to the master/slave version of the workflow based approach. It refines the *WrfFlowMsc* stereotype. Its *Master* property records the single experiment that plays the role of master in the whole multiscale simulation.

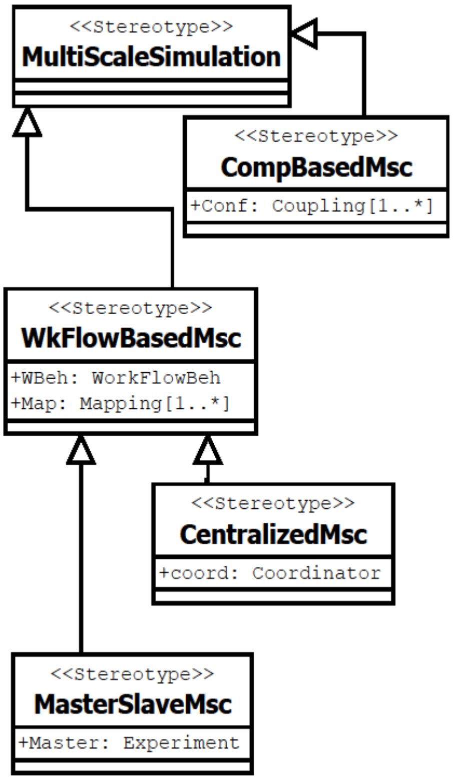


Figure 3.b: Hierarchy of MultiScale Design Approaches.

Figure 3.c shows the relationship between the stereotypes *Monolithic/Partitioned* and *Experiment* stereotypes (more details on the *Experiment* stereotype are given in the section 5.2.3)

Monolithic simulations include only one single experiment whilst partitioned simulations include more than one single experiment.

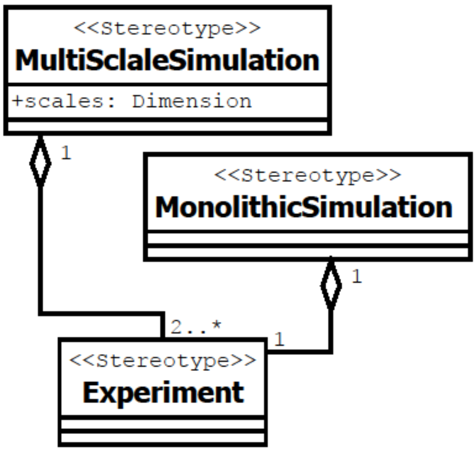


Figure 3.c Linking Simulation with Experiments.

### 5.2.2 *Coupling* stereotype:

Various kinds of couplings are identified:

- Direct coupling between pairs of experiments participating to component based multiscale simulations. This kind of coupling may various forms. For instance the designers of the MUSCLE multiscale platform use the term “coupling template” to refer to these coupling forms.
- Indirect coupling between slave experiments through a master experiment in case of master-slave multiscale simulations.
- Indirect coupling between experiments through a coordinator in case of centralized multiscale simulations.

Figure 3.d shows the specification of the proposed *Coupling* stereotype. This stereotype extends the UML association metaclass and it is characterized by the following properties:

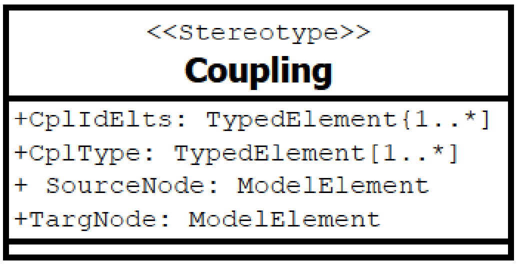

<<Stereotype>>
Coupling
+CplIdElts: TypedElement{1..*]
+CplType: TypedElement[1..*]
+ SourceNode: ModelElement
+TargNode: ModelElement

Figure 3.d: *Coupling* Stereotype.

+ *CplIdElts*: Specifies suitable information susceptible to identify its instances.
+ *CplType:* Set of suitable typed elements allowing to specify the kind of the coupling.
+ *SourceNode*, *TargetNode*: These attributes play the role of the UML association end. They specify the model elements that are coupled.

### 5.2.3 Simulation behavior stereotype

Instances of both *Experiment* and *Simulation* stereotypes own their specific behaviours. The stereotype

*SimBehavior* is intended to capture various simulation and experiment behaviors.

A. Figure 4.a shows a class diagram depicting the usual behaviors met in the simulation world. *The SimBeh* stereotype is intended to model the behavior of experiments and simulations. Two categories of behavior are identified. The opaques ones characterized by their unknown structure, and the regular ones characterized by well-defined, regular and known structures. For instance, workflows and automata-like structures are kinds of regular behavior.

**Comments:**

(i) Opaque behaviors, as defined in the UML infrastructure, are usually characterized by their body (body source plus the language used to express the source); in the context of our work, *Opaque Experiment* stereotype represents experiments driven by simulation engines. Here we adopt the UML Opaque Behavior metaclass as a base class.

(ii) Automata-based behavior which are explicitly described by automata-like formalisms such as Cellular Automata or others. Such kind of behaviors may, for instance, characterizes the behavior of single experiments that participate to multiscale simulations. Here we adopt the UML State Machine metaclass as a base class.

(iii) Workflow-based behaviors which are explicitly described by abstract workflows. Such kind of behavior may for instance characterizes the behaviour of monolithic simulation as well as multiscale simulations. These are often expressed in terms of Petri nets or UML activity diagrams. The authors of [23] defined a profile for scientific workflows by proposing a refinement of the UML Activity metaclass tailored to their own abstract workflow language. In our work we define the *WorkFlowBeh* stereotype to represent abstract simulation workflows by extending the UML Activity metaclass. *SimMotif* is one of the properties associated with the *WorkFlowBeh* stereotype. It is intended to specify the abstract motif/pattern of simulation workflows. Abstract simulation workflows are composed by sets of workflow nodes assembled according to a particular structure. We assume the availability of a library of UML model elements regrouping a catalogue of usual simulation workflow motifs.

B. More on Workflow based Experiments

Workflow-based experiments are usually composed of work steps structured and organized according to a specific workflow motif/pattern. In order to be independent from specific abstract workflow language, we adopt a solution, used by some workflow engines, that uncouples the workflow motif nodes from the task to be performed at the node level. To achieve this objective, we rely on the UML Behavior metaclass infrastructure to define the *SimulationWorkflowStep* stereotype.

This stereotype extends the UML *Call Operation* and *Call Behavior* metaclasses which are themselves two refinements of the UML *Execution Action* metaclass:

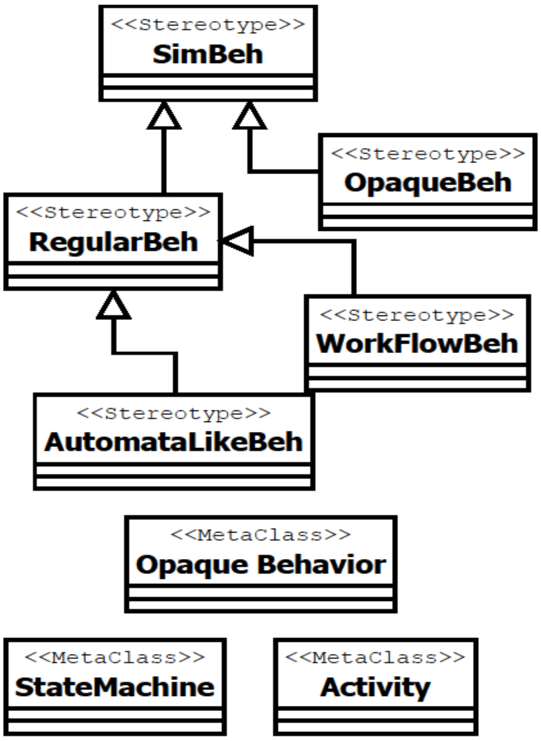


Figure 4.a Typology of Simulation Behaviors

- *Call Operation* is used to trigger atomic operations that correspond to simulation processing steps, like solving, data processing or data interaction steps.
- *Call Behavior* is used to trigger behaviors that correspond to potential sub-workflows contained in simulation workflows (hierarchical workflow motifs). It is useful to handle the master/slave approach (a master experiment enacting a slave experiment) and the centralized approach (a coordinator enacting the workflow of single experiments).

Figure4.b shows two refinements of the *SimulationWorkflowStep* stereotype:

*SimAction* stereotype representing various kinds of atomic simulation actions call (solving, data processing, data interaction operations) that may be associated with nodes of abstract workflow motifs. It extends the UML *Call Operation* metaclass.

*WrkFAction* stereotype representing sub-workflows with call action that may be associated with nodes of workflow motifs. It extends the UML *Call behaviour* metaclass.

### 5.2.4 Experiment and simulation model stereotypes

The *Experiment* stereotype represents PSIM elements. Figure 5 shows the features of this stereotype.

(i) *IdentifierElts* property records any useful information susceptible to identify the experiment (identifier number, experiment date, version, and eventually others).

(ii) *ExpParam* property records experiment parameters (experiment duration, and eventually other parameters).

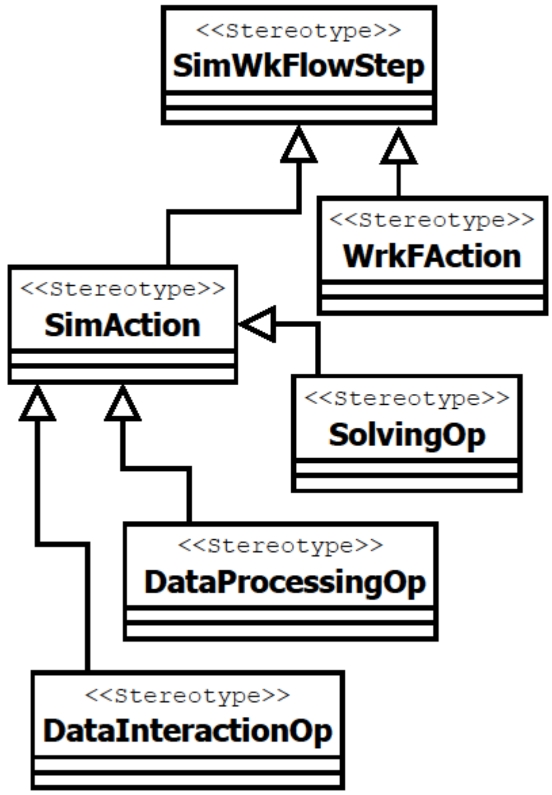


Figure 4.b: Simulation WorkFlow Step Stereotypes Hierarchy**.**

(iii) *ArchElts* property is intended to record any useful information related to the various simulation design approaches. The type of *ArchElts* property type is kept flexible in order to describe various simulation design approaches (monolithic, component based, centralized workflow based, master/slave workflow design approaches). *ModelElement* is a UML-MARTE defined metaclass that refers to any UML classifier.

(iv) *Smod* property specifies the simulation model targeted by the experiment. It may be either a whole simulation model (monolithic simulation) or a single simulation model (partitioned simulation).

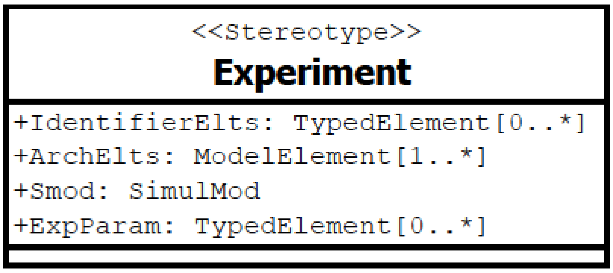

<<Stereotype>>
Experiment
+IdentifierElts: TypedElement[0..*]
+ArchElts: ModelElement[1..*]
+Smod: SimulMod
+ExpParam: TypedElement[0..*]

Figure 5. *Experiment* Stereotype.

Figure 6 shows the features of the *SimulModel* class:

(i) *Field* property specifies the application domain concerned by the simulation (engineering, physics, biology, and others). *Domain* class represents the various domains where simulations may be conducted.

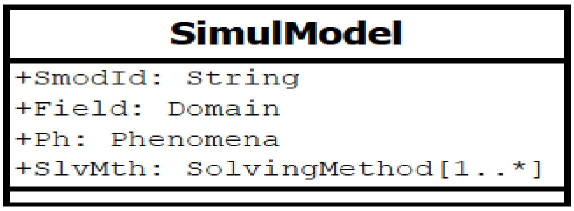


Figure 6. *SimulModel* Class.

(ii) *Ph* property specifies the domain specific phenomena targeted by the simulation.

(iii) *SlvMth* property specifies the set of mathematical methods that may be used to solve the simulation model. We define *SolvingMethod* a stereotype as an extension of the UML *OpaqueExpression* metaclass.

## 5.3 SPDM model elements

Simulation and experiments, as previously mentioned, are hosted and executed by simulation platforms.

UML-MARTE profile provides the concept of *Resource* to model in a uniform way hardware as well as software elements. Resources are abstract entities that provide services and they are themselves composed of other resources. We refine the concept of abstract resource to concrete (software) elements of simulation platforms.

In the present work we focus on only two core stereotypes that may be used to model PDSMs: Engines and Data Processor resources.

### 5.3.1 Engine resources

The concept of “engine’ is often used in the simulation field as well as in the workflow technology. Here engines represent virtual computing resources that interpret and run scripts or workflows written in specific formalisms. *Engine* refines the abstract *Resource* stereotype class defined in UML-MARTE profile.

This abstract resource provides a set of services common to all kinds of resources.

Figure 7 shows two kinds of engines: Simulation and Workflow engines.

*A. SimulationEngine:* An engine that interprets opaque simulation code written in specific formalism/language. It may also be a simulation tool, called simulator, that performs solving methods; simulators accept models and simulation scripts as inputs.

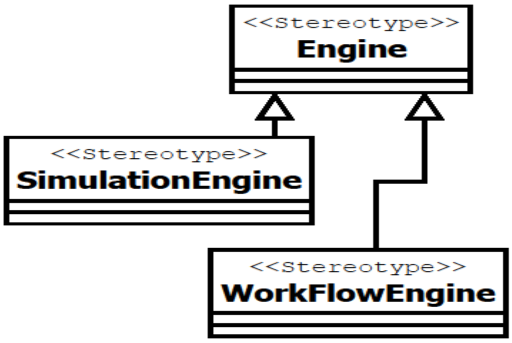


Figure 7. Simulation and Workflow Engines Stereotypes.

(i) I*nterpreter:* Specifies the formalism that is interpreted by the simulation engine,
(ii) K*ind:* Specifies the type of simulation engine. *SimulEngineEnumeration==*
simulator| embedded simulation code |….
(iii) *Slv-method*: Specifies the set of numerical method that are supported by the simulation engine.
(iv) *Computation:* Specifies if the engine performs sequential or parallel computations.

Figure8 represents the main features of the *Simulation Engine* stereotype.

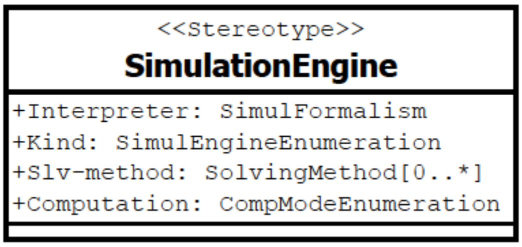

<<Stereotype>>
SimulationEngine
+Interpreter: SimulFormalism
+Kind: SimulEngineEnumeration
+Slv-method: SolvingMethod[0..*]
+Computation: CompModeEnumeration

Figure 8: Simulation Engine Stereotype.

*B. WorkflowEngine***:** An engine that is responsible for the interpretation of executable workflow and the orchestration of workflows. It is a kind of scheduling resource. Workflow steps may be either basic/atomic tasks or sub-workflows. Modellers specify their workflows using either a human readable textual script or a diagram-based workflow language (Front-

End workflow language), while workflow engines interpret platform readable and executable workflow languages (Back-End language).
Figure 9 depicts the main features of the Workflow Engine stereotype.

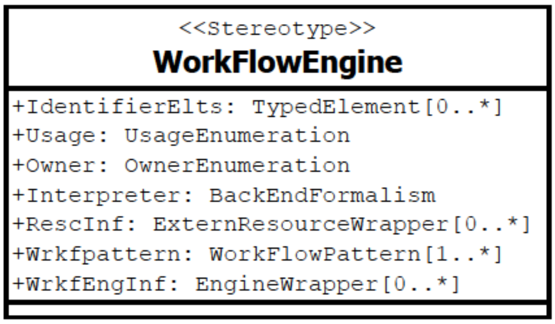

<<Stereotype>>
WorkFlowEngine
+IdentifierElts: TypedElement[0..*]
+Usage: UsageEnumeration
+Owner: OwnerEnumeration
+Interpreter: BackEndFormalism
+RescInf: ExternResourceWrapper[0..*]
+Wrkfpattern: WorkFlowPattern[1..*]
+WrkfEngInf: EngineWrapper[0..*]

Figure 9. *WorkflowEngine* Stereotype.

(i) W*orkFlowPattern is a* sub-class of the *Control Node* meta-class. It includes the usual set of control nodes found in simulation workflows like sequence, loop, and parallel.
(ii) *ExternResourceWrapper*, and *EngineWrapper* are derived from the UML *Adapter* pattern. *External ResourceWrapper* refers to wrappers that encapsulate data processing operators, and *EngineWrapper* refers to wrappers that encapsulate simulation engines in case of cooperation between workflow engines.

### 5.3.2 Data processor elements

In the following, we present a set of stereotypes aiming to model a set of specific computing resources that are able to support the execution of specific operations: data operation, and data interaction. We model these resources as kinds of virtual processor.

Our approach to categorize the data operations is slightly different from the one reported in [20]. We differentiate the data processing operations that may operate inside individual experiments, the intra-experiment case, from the operations on data that are performed along the data motion from one single experiment to another experiment, the inter-experiment case. A categorization of these Data processors is shown in Figures 10a, 10b, and 10c. The following kinds of data processor are identified:

*A. Inter-Experiment Data Processor*

Data are potentially subject to manipulation during their motion between single experiments. Each kind of manipulation is described by a specific (mathematical) function or algorithm. Two kinds of manipulations are identified:

(i) Data transformation: filtering,
(ii) Data combination: usually carried out by operators called Mappers.
(a) Data aggregation: aggregating multiple data sources to one data source,
(b) Data dis-aggregation: separating one data source into multiple data sources.

*B. Intra-Experiment Data Processor*

Usually the input data need to be set into a specific format before to be submitted to simulation engines. The output data (produced by simulation engines) need also to be set in specific formats before to be visualized to the modellers. Commercial and academic libraries provide such data processors.

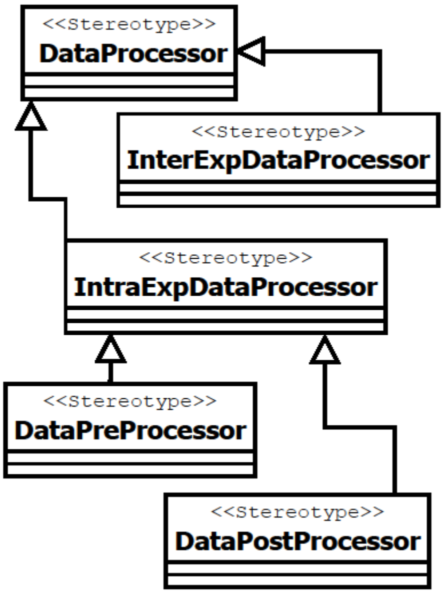


Figure 10 a. Data Processors Classification.

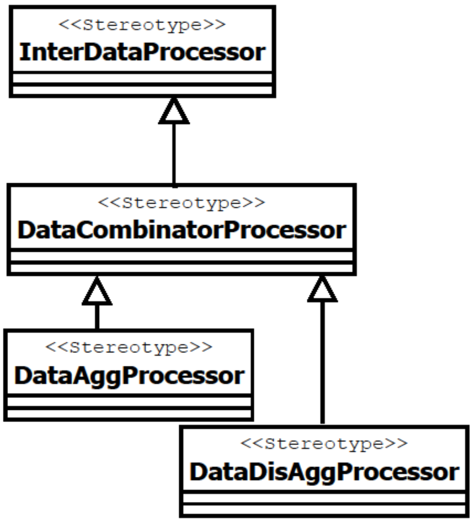


Figure 10 b. Data Combinator Processors Classification.

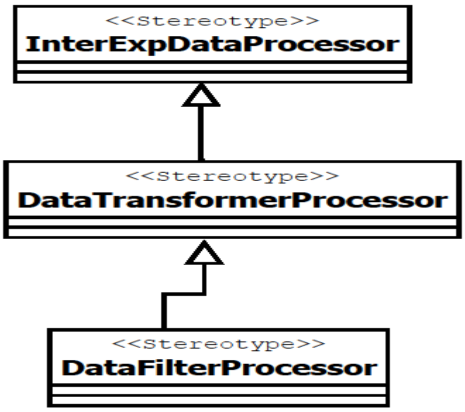


Figure 10.c Data Transformer Processor.

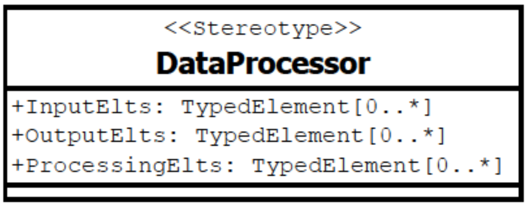

<<Stereotype>>
DataProcessor
+InputElts: TypedElement[0..*]
+OutputElts: TypedElement[0..*]
+ProcessingElts: TypedElement[0..*]

Figure 11: Data Processor Stereotype.

C. The stereotype *Data Processor* inherits from the *Resource* class. Its main features are:

*(i)* *InputElts*: Specifies the number and types of inputs which depend from the kind of data processor,

*(ii)* *OutputElts*: Specifies the number and types of outputs depend from the kind of data processor,

*(iii)* *ProcessingElts*: Specifies an algorithm (body) that implements the analytic (mathematical) operation to be performed as well as a set of appropriate parameters qualifying its performance.

### 5.3.3 Data interaction operator

Single experiments participating to multiscale simulations are coupled according to specific coupling mechanisms. They exchange data either in a direct way, in case of a component based multiscale simulation approach or in an indirect way in case of master/slave and centralized multiscale simulation approaches.

Our profile provides a stereotype class named *DataInteractionOperator* intended to run various kinds of coupling (data motion according to specific templates). It represents an abstraction of the so-called coupling wrappers mentioned in the section 2.2.1. We adopt and refine the UML Adapter pattern to define this stereotype.

## 6 Example

In this section we introduce a simple example to illustrate the (partial) use of our proposed profile. The example exposes only the PISM model elements.

The example presents a component based multiscale simulation which consists of two single scale experiments namely C1 and C2 interacting through two couplings namely Cp12 from C1 to C2 and Cp21 from C2 to C1.

C1 is the experiment on the simulation model Mod1 and C2 is the experiment of the simulation Mod2. Both Mod1 and Mod2 are single scale models of the partitioned simulation model Mod.

a. Instantiation of the stereotype *Simulation* with the following tags:

+ *IdentifierElts* =
SimlId: String
SimDt: Date
SimVersion: String
*SimDm:* Domain /Domain: a data type/

+ *SimulParam* =
SimDuration: Time
SimSpace: Space /Space: a data type/
SimMd: SimulMod / Simulation Model/

b. Instantiation of the stereotype *MultiscaleSimulation* with the following tags:

+ *scales* = time&space /value of Dimension Enumeration type/
+*ExpNumber* = 2 /property of *PartitionedSimulation*

c.Instantiation of the stereotype *CompBasedMsC*
+*Conf* =
*Cp12: Coupling*
*Cp21: Coupling*
Two couplings in our example Cp1 and cp2

d. Instantiation of the stereotype *Coupling*
+ *CplType=*
*CcplK: CouplingKind* */CouplingKind:* enemuration data type/
CpT: *CouplingTemplate* /CouplingTemplate is a *data type/*
+ *CpMeth: OpaqueExpression* /coupling code algorithm

+ SourceNode =
Src: InPort /InPort: a UML model element/
+ TargNode =
Targ: OutPort /OutPort: a UML model element

For Cp12 Instance
CplK = directcoupling /direct coupling between two experiments/
CpT = tempX / templateX is one instance of CouplingTemplate/
CpMeth = MethX /MethX: a coupling algorithm for tempX/
Src =out1
Targ= in2

```
  For Cp21 Instance
  CplK =  directcoupling   /direct coupling between two experiments/
   CpT  =    tempY    / templateY is one instance of CouplingTemplate
  CpMeth = methY      /MethY a coupling algorithm for  tempY /
  Src =out2
  Targ= in1
```

e. Instantiation of the stereotype *Experiment*

From the architectural point of view Experiment instances are seen as components owning an internal behavior and characterized by a set of input and output ports for their interaction (coupling) with other experiments. In this example we use the SEL (SubModel Execution Loop) behavior borrowed from the MUSCLE multiscale framework.

```
+ IdentifierElts =
     ExpId: String
     ExpDt: Date
     ExpVersion: Integer

+ArchElts =
 ExpBeh: SEL      / SEL:  Behaviour of specific cellular automata/
 ExpIn: InPort [1..*]                   / InPort: model element/
 ExpOut:OutPort [1..*]                /  OutPort: model element/

     For C1 Experiment instance:
     ExBeh =sel1   \ an instance of the SEL data type\
      ExpIn = {in1}
      ExpOut = {out1}
     For C2 Experiment instance:
     ExBeh =sel2   \ an instance of the SEL data type\
    ExpIn = {in2}
ExpOut = {out2}

+ExpParam =
 ExpTimeScale: Time               /Time scale of the Experiment/
 ExpSpaceScale: Space            /space scale for the experiment

     For C1 Experiment instance:
      ExpTimeScale = t1
      ExpSpaceScale = sp1
      For C1 Experiment instance:
      ExpTimeScale = t2
      ExpSpaceScale = sp2

+SMod =_Mod1     (Mod2 for the C2 Experiment instance).
```

Realistic and complete case studies are currently under construction.

## 7 Conclusion and future works

In this work we present a synthesis of recent contributions in the modelling and simulation field encompassing up-to-date simulation topics. Model driven approaches for the simulation field are discussed. Multi-scale and multi-physics simulation methods and their related issues are outlined. Modern simulation platforms adopting a component- as well as a workflow-based approach are exposed.

We also propose modelling mechanisms intended for the description of simulation platforms, thus making possible the development of a kind of MDA primary model called SPDM. For this purpose we define a UML profile including a set of useful UML stereotypes that capture core simulation concepts as well as core simulation platforms elements such as simulation engines, workflow engines, and simulation data processors. In this work, a resource-based approach, similar to the one used for the UML-MARTE profile, is adopted for the modelling of simulation platforms elements.

As a first future work we plan also to develop UML meta-models for a set of widely used simulation model specification formalisms, thus enabling PISM-to-PISM transformations.

## Acknowledgement

This research work is fully supported by the Deanship of the Scientific Research - Philadelphia University (Jordan).

## References

[1] A.-R. Da Silva, Model Driven Engineering: A Survey supportedby the Unified Conceptual Model, *Computer languages, Systems and Structures*, Vol. 43, pp. 139-155, Elsevier, 2015.

[2] S. Wagner, D. Pfluger, and M. Mehl, Simulation Software Engineering: Experiences and Challenges, in*Proc.of the International Workshop on Software Engineering for High Performance Computing in Computational Science and Engineering*, pp. 1-4, 2015.

[3] O. Topcu, U. Durak, H. Oguztuzun, and L. Yilmaz, Distributed Simulation: A Model Driven Engineering Approach, *Simulation Foundations, Methods and Applications, Springer*, 2016. https://doi.org/10.1007/978-3-319-03050-0

[4] G. Wagner, Model-Driven Engineering of Second-Life-Style Simulations, in *the Proc. of the Winter simulation conference*, 2010.

[5] A. Yang, and W. Marquard, An Ontological Conceptualization of Multiscale Models, *Computer ChemicalEngineering*, Vol. 33, pp. 822-837, (2009). https://doi.org/10.1016/j.compchemeng.2008.11.015

[6] J. Borgdorff, M. Mamonski, B. Bosak, K. Kurowski, M. Ben Belgacem, B, Chopard, D. Groen, P.-V. Covery, and A.-G. Hoekstra, Distributed Multiscale Computing with MUSCLE2, the Multiscale Coupling Library and Environment, *Journal of Computational Science*, Vol. 5, Issue. 5, pp. 719-731, Elsevier, 2014.

[7] D. Cetinkaya, A. Verbraeck, A., and D.-M Seck, Applying a Model Driven Approach to Component Based Modeling and Simulation, in *Proc. of the Winter Simulation conference*, 2010.

[8] OMG, A UML Profile for MARTE: Modeling and Analysis of Real-Time Embedded Systems, Beta 2, *OMG Document Number: ptc/2008-06-09,* 2008.

[9] O. Babur, V. Smilauer, T. Verhoeff, and M.V-D. Brand, A survey of Open Source Multiphysics Frameworks in Engineering", *Procedisa Computer Science*, Vol. 151, pp. 1088-1097, Elsevier, 2015.

[10] O. Babur, T. Verhoeff, and M.G.-J. Van Den Brand, Multiphisics and Multiscale Sofware Frameworks: An Annotated Bibliography, Computer Science Reports, Technische University Eindhoven, Vol. 1501, 2015.

[11] Y. Zaho, C. Jiang, and A. Yang, Towards Computer-Aided Multiscale Modeling: An Overarching Methodolgy and Support of Conceptual Modeling, *Computer and Chemical Engineering,* No.36, pp. 10-21, Elsevier, 2012.

[12] M. Ben Belgacem, and al, Distributed Multiscale Computations Using the MAPPER Framework”, *Procedia Computer Science*, Vol. 13, pp. 1106-1115, Elsevier, 2013.

[13] O. Hoenen, D. Coster, S. Petruczynik, and M. Plociennick,Coupled Simulations in Plasma Physics with the Integrated Plasma Simulator Platform, *Procedia Computer Science*, Vol. 5, pp. 1138-1147, Elsevier, 2015.

[14] J. Borgdorff, J.-L. Falcone, E. Lorenz, C. Bona-Casas, B. Chopard, and A.-G. Hoekstr, Foundations of Distributed Multiscale Computing: Formalization, Specification and Analysis, *Journal of Distributed Computing*, Elsevier, Vol.73, pp. 465-483. 2013. https://doi.org/10.1016/j.jpdc.2012.12.011

[15] U. Yildez, A. Guabtni, and A.H-H. Ngu, Business versus Scientific workflow: A Comparative Study, Research Report No. 2009-3, Project DAKS, Department of Computer Science, UC Davis University of California, 2009.

[16] T. Buchert, L. Nusbaum, and J. Gustedt, A Workflow-Inspired, Modular and Robust Approach to Experiments in Distributed Systems, Project-Team Algorille, Research Report n0 8404, Research Center Nancy-Grand Est, 2013.

[17] C.-A. Ellis, Workflow Technology, Chapter No. 2 in Computer Supported Cooperative Work, Edited by Beaudouin-Lafon, John Wiley and Sons Ltd, 1999.

[18] B. Dashtban, Scientific Workflow Patterns, Msc Dissertation, School of Advanced Computer Science, Manchester University, 2012.

[19] X.-R. Xiang, andG. Madey, Improving the reuse of scientific workflows and their by-products, in *Proc.IEEE International Conference on Web Services*, pp. 792-799, 2007.

[20] A. WeiB, andD. Karastoyanova, A Life Cycle for Coupled Multi-Scale, Multi-field Experiments Realized through Choreographies, in *Proc. Enterprise Distributed Object Computing Conference (EDOC)*, 2014.

[21] N. Cerezo, J. Montagnat, and M. Baly-Fornarino, Computer-Assisted Scientific Workflow Design, *Journal of Grid Computing*, Vol. 11, No. 3, pp. 585-610, Springer Verlag, 2013.

[22] M. Maouche, M. Bettaz, Towards a Software Engineering Approach to Multi-Scale Modelling and Simulation, *IJSEIA Journal*, Vol. 10, 2016.

[23] J. Qin, T. Fahringer, andS. Pllana, UML Based Grid Workflow Modeling Under ASKALON, Chapter in Distributed and Parallel Systems Book, pp. 191-200, Springer, 2007.

[24] Garijo, D., Alper, P., Belhajjame, K., Corcho, O., Gi, Y., Common Motifs in Scientific Workflows: An Empirical Analysis, Future Generation Computer Systems, Elsevier, (2013).

[25] R. Ferreira Da Silva, R., S. Camarasu-Pop, B. Grenier, V. Hamar, D. Manset, J. Montagnat, J. Revillard, J,-R. Balderrama, A. Tsaregorodtsev, and T. Glatad, Multi-Infrastructure Workflow Execution for Medical Simulation in the Virtual Imaging Platform, in *Proc. HealthGrid Conference*, pp.1-10, 2011.

[26] D. Groen, S.-J. Zasade, and P.-V. Coveney, Survey of Multiscale and Multiphysics Applications and Communities, *Computing in Science & Engineering*, Vol. 16, Issue. 22, pp. 34-43, 2014. https://doi.org/10.1109/MCSE.2013.47

[27] S. Turki, E. Senn, and D. Blouin, Mapping the MARTE UML profile to AADL, in MoDELS 2010 ACES-MB Workshop Proceedings, pp. 11-20, 2010.

# A New Variant of Teaching Learning Based Optimization Algorithm for Global Optimization Problems

Yugal Kumar
Department of Computer Science and Engineering, JUIT, Waknaghat, Solan, Himachal Pardesh, India
E-mail: yugalkumar.14@gmail.com

Neeraj Dahiya and Sanjay Malik
Department of Computer Science and Engineering, SRM University, Delhi-NCR Campus, India
E-mail: neerajdahiya.cse@gmail.com, skmalik9876@gmail.com

Savita Khatri
Department of Computer Science and Engineering, BMIET, Sonepat, India
E-mail: dhynrj@gmail.com

**Keywords:** teaching learning based optimization, crossover, meta-heuristics, mutation

**Received:** May 19, 2017

*This paper presents a new variant of teaching learning based optimization (TLBO) algorithm for solving global optimization problems. The performance of the TLBO algorithm depends on coordination of teacher phase and learner phase. It is noticed that sometimes performance of TLBO algorithm is affected due to lack of diversity in teacher and learner phases. In this work, a new variant of TLBO algorithm is proposed based on genetic crossover and mutation strategies. These strategies are inculcated in TLBO algorithm for improving its search mechanism and convergence rate. Genetic mutation strategy is applied in teacher phase of TLBO algorithm for improving the mean knowledge of leaners. While, Crossover strategy is applied in learner phase of TLBO algorithm to find the good learner. The effectiveness of the proposed algorithm is tested on several bench mark test functions of CEC'14. From simulation results, it is stated that the proposed algorithm provides more optimized results in comparison to same class of algorithms.*

*Povzetek: V prispevku je opisan nov globalni optimizacijski algoritem na osnovi učenja optimizacije (TLBO).*

## 1 Introduction

Optimization is an active area of research and it provides robust and viable solutions for complex real-world problems. A lot of efforts are needed to find optimal solution for these problems due to increase dimensionality, differentiability, multi-modality and rotation characteristics. Hence, a lot of research has been carried out in this direction to design a real-time numerical optimizer. This optimizer can provide more accurate, fast and computationally efficient optimization algorithms. Large numbers of algorithms have been developed by research community for solving many numerical optimization techniques. These algorithms can solve optimization problems efficiently and effectively. But, according to no free lunch theorem, there is no universal algorithm that can solve all optimization problems accurately. Over the past few decades, population based meta-heuristic algorithms have attained more popularity among research community. These algorithms have ability to turn itself according to problem domain and provide successful results for many complex problems. It is noticed that large numbers of optimization problems exist in the fields of engineering and science. These problems can be categorized as unimodal and multimodal optimization problems. Further, it can be characterized as unimodal separable and inseparable, and multimodal separable and inseparable. In literature, it is found that numbers of algorithms have been reported for solving these problems either maximizing or minimizing the objective function. Moreover, these algorithms are also adopted for solving real-life problems such as clustering, classification, scheduling, path planning, resource allocation, and many other problems. These algorithms are divided into two categories i.e. exact and approximation algorithms [1]. Exact algorithms find the optimal solution within bounded time, but having exponential computational time. The approximate algorithms provide better results in terms of time and solution using heuristics. Further, the meta-heuristic algorithms are also applied for solving the wide range of optimization problems. These algorithms are sub branch of approximate algorithms. In past decade, many meta-heuristic algorithms are developed for finding exact solution of optimization problems. Most of these are inspired through natural phenomena's such as swarm behavior, insect's characteristics, physics law and process etc. Some of are Simulated Annealing (SA) algorithm [2], Genetic Algorithm (GA) [3], Particle Swarm Optimization (PSO)

Corresponding Author: Yugal Kumar, Department of Computer Science and Engineering, JUIT, Waknaghat, Solan, Himachal Pardesh, India, yugalkumar.14@gmail.com

[4], Ant Colony Optimization (ACO) [5], Harmony Search (HS) [6], Artificial Bee Colony (ABC) [7, 33], Firefly Algorithm (FA) [8], League Championship Algorithm (LCA) [9], Water Cycle Algorithm (WCA) [10], CSS [11, 12], MCSS[13, 14, 15], TLBO [16, 17, ], CSO[47,48] and Mine Blast Algorithm (MBA) [18].

Recently, Rao et al., have developed teaching learning based optimization (TLBO) algorithm to solve the constrained and unconstrained optimization problems. This algorithm is inspired from class room based teaching methodology [19]. In short span of time, this algorithm become more popular among researchers and has been applied to solve variety of problems. A lot of optimization problems have been solved by using TLBO algorithm and provides better results in comparison to existing algorithms [20-24]. Still, there are several shortcomings that can affect the performance of TLBO algorithm such as quality of solution, stuck in local optima when solving global optimization problems, premature convergence, tradeoff between searching capability and local search ability. Hence, the aim of this work is to design an effective and efficient algorithm for addressing the convergence and quality of solution issues of TLBO algorithm. In order to overcome the aforementioned issues, this work investigates the capability of genetic crossover and mutation operators for improving the performance of TLBO algorithm. It is observed that proposed algorithm provides better results than traditional TLBO and other existing optimization algorithms.The rest of paper is organized as follows: the section 2 describes the related work on improvements in TLBO algorithm and its applicability in diverse fields. Section 3 introduces basic TLBO algorithm. The proposed genetic TLBO algorithm is illustrated in section 4. Section 5 illustrates the simulation results of proposed TLBO algorithm using benchmark functions. The entire work is concluded into section 6 and future work reported in section 7.

## 2 Related works

This section describes the recent work reported on TLBO algorithm. Several studies have been published on the modifications of TLBO algorithm. Some of these are highlighted in this section. To make the effective tradeoff between exploration and exploitation capabilities, Rao et al. have developed an improved TLBO algorithm, called I-TLBO [25]. In this work, authors have introduced the concept of multiple teachers, adaptive teaching factor, self-motivated learning and tutorial training. The self learning and tutorial training methods can be acted as search methods. Further, to explore the local optimum solution in the hope of global optimum solution. The concept of multiple teachers is incorporated in TLBO algorithm to avoid premature convergence. Moreover, adaptive teaching factor is inculcated for fine tuning between exploration and exploitation capabilities. From results, it is seen that I-TLBO effectively overcome the aforementioned problems. Satapathy et al., have presented a new version of TLBO algorithm, called mTLBO for improving the convergence rate [26]. The proposed mTLBO algorithm is applied on global optimization problems for obtaining optimal solution. The proposed algorithm includes the concept of tutorial classes in learner phase to improve the outcomes of learners. The performance of mTLBO is compared with other state of art algorithm like PSO, DE, ABC and GA and it is observed that addition of tutorial class concept improves the results of TLBO algorithm. To enhance local search ability and quality of solutions, Haung et al., have incorporated levy flight based teaching learning process for TLBO algorithm [27]. The proposed algorithm is adopted to solve several engineering optimization problems especially industrial optimization problems. It is seen that the proposed algorithm outperforms than traditional TLBO algorithm. For improving the global performance of traditional TLBO algorithm, Zou et al., have developed an improved variant of TLBO algorithm based on learning experience [28]. Further, a copy operator is also integrated in TLBO algorithm and called it LETLBO. It is noticed that the learning experiences of learners are evaluated using two random possibilities. The performance of algorithm is tested on eighteen standard benchmark functions and compared with state of art algorithms. It is stated that above mentioned improvements significantly improve the global performance of TLBO algorithm. Ouyanget et al. have presented a new variant of TLBO algorithm to address global search and local optima issues, called GC-TLBO [29]. In GC-TLBO, a global crossover operator is introduced for addressing global search issue. Whereas, the local optima issue is controlled using perturbed mechanism. The experimental results reveal that the proposed improvements make the TLBO algorithm more effective and significant one. Ghasemi et al. have developed gaussian bare bones teaching learning optimization algorithm, called GBTLBO for improving the quality of solutions [30]. The results stated that GBTLBO algorithm provides better performance than other algorithms being compared. To avoid the premature convergence and preserve the population diversity, Zou et al. have developed an improved TLBO algorithm based on dynamic group strategy [31]. Moreover, quantum behaved learning scheme is also inculcated into learner phase of TLBO algorithm to maintain population diversity. The feasibility of proposed algorithm is evaluated on eighteen benchmark functions. Simulation results stated that the proposed algorithm is one of effective and efficient algorithm for solving global optimization problems. Lim and Isa have presented a new algorithm by combining PSO and TLBO algorithms for solving global optimization algorithm, called TPLPSO [32]. The performance of TPLPSO is investigated on twenty benchmark functions and it is found that the TPLPSO exhibits better performance than other algorithm being compared. To identify the most relevant gene in the development of the breast cancer, Sahbeig et al. proposed a combination of TLBO and fuzzy adaptive PSO algorithm, called TLBO-PSO [36]. The performance of the proposed algorithm is evaluated using accuracy, sensitivity and specificity parameters. It is revealed that the proposed algorithm obtains higher accuracy rate i.e. 91.88 as compared to other algorithms. Kumar et al. have applied a hybrid TLBO-TS algorithm to deal with problem of simultaneous selection and scheduling of projects [37]. The proposed approach is tested on several datasets and compared with TLBO and TS algorithms. It is seen that combination of TLBO-TS algorithm provides faster convergence than TLBO and TS algorithms. Patel et al. have applied teaching-learning based optimization (TLBO) to design ultra-low reflective coating over a broad wavelength-band using multilayer thin-film structures for optoelectronic devices [38]. The results of TLBO algorithm are compared with GA using Wilcoxon singed ranked test. It is observed that TLBO algorithm gives more effective results than GA. To enhance the performance of original TLBO algorithm and make the balance between local and global searches, Ji et al. developed an improved version of TLBO algorithm, called I-TLBO [39]. In proposed algorithm, a self feedback phase is incorporated to enhance the performance of original TLBO algorithm. The effectiveness of proposed algorithm is tested on several combinatorial optimization problems. It is stated that the proposed improvements have significant impact on the performance of TLBO algorithm. To solve numerical structural analysis problems, Cheng and Prayogo presented fuzzy adaptive teaching learning based optimization algorithm, called FATLBO [40]. In proposed algorithm, three search strategies are included for improving searching capabilities. The performance of proposed algorithm is examined over five well known engineering structural problems. The results show that the proposed algorithm gives excellent and competitive performance. To analyze and predict the time series data, Das and Padhy desgined a hybrid model based on support vector machine (SVM) and TLBO [41]. The proposed model avoids user defined control parameters. The validity and efficacy of proposed model is evaluated on COMDEX commodity futures index. The results stated that proposed model is more effective and performs better than PSO-SVM and SVM models. Kankal and Uzlu adopted neural network with TLBO approach for modeling and forecasting long term electric energy demand in turkey [42]. In proposed approach, TLBO algorithm is used to optimize the parameters of neural network. The simulation results of ANN-TLBO approach is compared with ANN-BP and ANN-ABC models. It is revealed that ANN-TLBO approach provides efficient results than others. Kiziloz et al. applied multi-objective TLBO algorithm for feature subset selection in binary classifications problems [43]. The performance of the proposed algorithm is evaluated on well known benchmark problems and compared with large number of meta-heuristic algorithms such as GA, PSO, NSGA, TS and SS. It is seen that TLBO is one of the competitive algorithms for feature subset selection problem. Prakash et al. presented quasi-oppositional self-learning teacher-learner-based-optimization (QOSL-TLBO) for solving non-convex economic load dispatch (ELD) problem [44]. In this algorithm, self learning mechanism is incorporated in teacher and learner phases. The robustness of the proposed algorithm is evaluated on standard IEEE generator system. Further, the results are compared with well known algorithms. It is seen that the proposed algorithm achieves minimum total cost for all generations. Zheng et al. adopted TLBO to solve multi-skill resource constrained project scheduling problem (MS-RCPSP) with make-span minimization criterion [45]. To make effective balance between exploration and exploitation processes, the concept of reinforcement learning is introduced in TLBO algorithm. From simulation results, it is stated that the computational cost of proposed algorithm is far better than compared algorithms. Birashk et al. designed a cellular TLBO algorithm for dynamic multi objective optimization (DMOO) problems [46]. The performance of cellular TLBO algorithm is evaluated and compared with state of art algorithms using well known DMOO problems. It is observed that cellular TLBO algorithm gives superior results than other algorithms.

# 3 Teaching learning based optimization (TLBO) algorithm

TLBO is a population based meta-heuristic algorithm. Like other population based algorithms, the global solution is represented using population [19]. TLBO algorithm works on the concept of classroom learning paradigm. The teachers are available to teach and enhance the knowledge of learners. The aim of teacher is to improve the learning capability of learners. Further, a learner can also enhance its skills by acquiring the knowledge from other learners. TLBO algorithm consists of two phases: Teacher phase and Learner phase. The detailed discussions on these phases are outlined as given below.

**Teacher Phase**: The aim of this phase is to improve the learning skills of students such that results of class improve significantly and this can lead the mean result of class. In general, teacher can improve the result up to certain level. In practice, several constraints are responsible for results such teaching method, teachers capability, learners grasping ability, interaction of learners to others and knowledge of learners. In teacher phase, M denotes the learner's knowledge mean and T describes any teacher in iteration. The main task of teacher is to enhance present knowledge of learners. To achieve the same, the present mean knowledge of learners i.e. M to move towards the teacher knowledge i.e. T and it can be described using equation 1.

$$\mathrm{X_{i,new}} = \mathrm{X_{i,old}} + \mathrm{r} * (\mathrm{X_{Teacher}} - \mathrm{T_f} \times \mathrm{X_{mean}}) \qquad 1\,)$$

In equation 1, $\mathrm{X_{Teacher}}$ and $\mathrm{X_{mean}}$ represent the teacher and the mean of the knowledge of learner in $i^{th}$ iteration, $\mathrm{T_f}$ denotes the teaching factor, and r is a random number in the range of 0 and 1. The teaching factor is computed using equation 2.

$$\mathrm{T_f} = \mathrm{round}\big(1 + \mathrm{rand}(0,1)\big) \qquad (2)$$

**Learner Phase:** The aim of the learner phase is to enhance the knowledge of learner from others. Hence, to improve learning ability, a learner can interact with other learners randomly. In learner phase of TLBO algorithm, learners learn knowledge from others. This learning capability of learners can be expressed as follows.

If ith learner wants to interact with the kth learner and the fitness of kth learner is higher than ith learner, then the position of ith learner will be updated otherwise kth learner. This can be summarized in equations 3-4.

$$X_{i,new} = X_{i,old} + r_i \times (X_k - X_i) \tag{3}$$

Else

$$X_{i,new} = X_{i,old} + r_i \times (X_i - X_k) \tag{4}$$

If the fitness of ith learner is better than old position, then new position take over the old one otherwise not.

# 4 Proposed TLBO algorithm

This section describes the working of proposed TLBO algorithm for solving global optimization problems. In this work, two amendments are made in TLBO algorithm for improving search mechanism and convergence rate. Hence, to achieve the same, genetic crossover and mutation operators are incorporated into TLBO algorithm. The genetic mutation operator is adopted in teacher phase. Further, genetic crossover operator is used to enhance learning capability of a learner through different learners. These improvements are illustrated in Algorithm 1 and Algorithm 2.

**Algorithm 1: Teacher Phase of TLBO algorithm**

```
For d=1 to N
For j =1 to D
      $DifferenceMean = r(T_{mean} - T_f \times M_i)$
      $T_f = round(1 + rand(0,1))$
IF ($DifferenceMean < rand()$)
   Apply genetic mutation operator on $T_{mean}andM_i$
   Compute Difference Mean and generate the new mean of knowledge ($X_{i,new}$) using equation 3.
Else
      $X_{i,new} = X_{i,old} + DifferenceMean_i$
End
End
    Accept $X_{i,new}$if f($X_{i,new}$) is better than f($X_{i,old}$)
End
```

The aim of these operators is to maintain population diversity during teacher and learner phases, and further, overcome the chance of trapping in local optima. In learner phase, the knowledge of learner is not enhanced gradually, the fitness of leaner is compared with random function. If, it is less than random number, the crossover operator is applied to find other learner to enhance its skills and knowledge.

**Algorithm 2: Learner Phase of TLBO algorithm**

```
For i =1 to N
Randomly pick two learners $X_i$ and $X_k$ such that i≠ j
IF ($F(X_i) < F(X_k)$)
IF ($F(X_i) < rand()$)
Apply genetic crossover operator on $X_iandX_k$ and generate the new position of learner $X_k$
End IF
        $X_{i,new} = X_{i,old} + r_i \times (X_k - X_i)$
Else
        $X_{i,new} = X_{i,old} + r_i \times (X_i - X_k)$
End
    Accept $X_{i,new}$if f($X_{i,new}$) is better than f($X_{i,old}$)
End
```

The detailed description of proposed TLBO algorithm is given in Algorithm 3. The main steps of proposed algorithm are summarized as below and flowchart is illustrated in Fig. 1.

**Algorithm 3: Algorithmic steps of TLBO algorithm**

Step 1: Initialize the number of learners (X), number of dimension (D) and other algorithmic parameters of TLBO algorithm.

Step 2: Evaluate the positions of learners (X) and compute the fitness function F(X).

Step 3: Determine the best learner and it can be acted as Teacher.

Step 4: Compute the mean of all learners (X) and denoted as Mean

Step 5: While(stopping condition is not met)

Step 6: Apply teacher phase of TLBO algorithm (Algorithm 1)

Step 7: Apply learner phase of TLBO algorithm (Algorithm 2)

Step 8: Update the Teacher and the mean

Step 9: End While

Step 10: Obtain final optimal solution

# 5 Results

This section describes the results of proposed TLBO algorithm using benchmark test functions of CEC’14. These functions are combination of uni-modal and multi-modal test functions that are highly trapped in local optima. The proposed algorithm is implemented in Matlab 2010 (a) environment using window based operating system having core i7 processor and 8 GB RAM. The results of proposed algorithm are taken on average of 30 independent runs for each test functions. Mean and standard deviation are taken as performance parameters to compare the performance of proposed algorithm and other algorithms. The mean parameter demonstrates the efficiency of algorithms, whereas standard deviation parameter illustrates the robustness of algorithms. The experiment is performed using real dimension i.e.30 for all benchmark functions i.e. $F_1$-$F_{16}$. The performance of proposed algorithm is compared with other state of art algorithms like PSO, GA, BA, FPA, ABC, FA, BBO, HS and TLBO. Table 1-2 depict

\

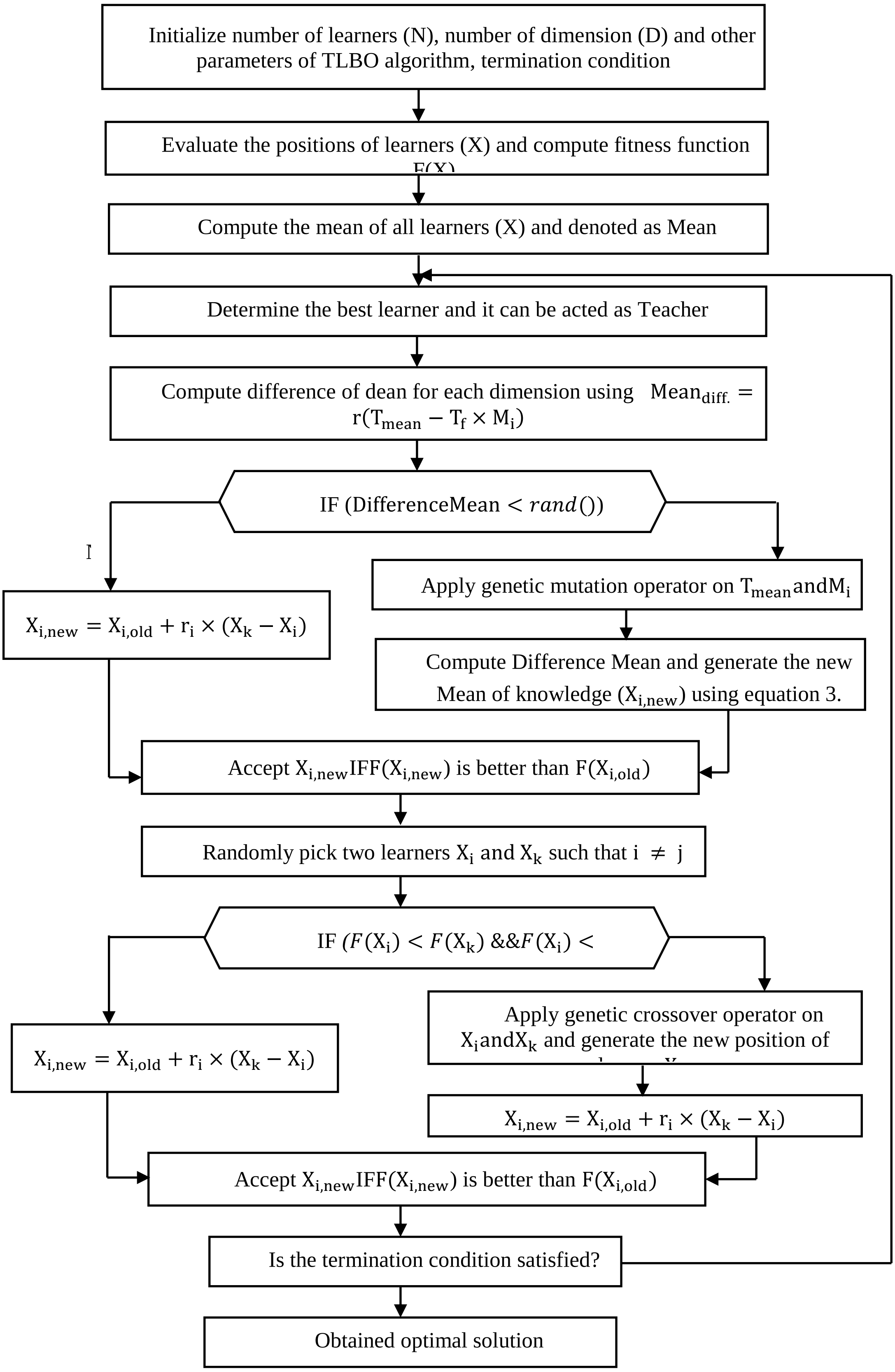

Initialize number of learners (N), number of dimension (D) and other parameters of TLBO algorithm, termination condition
Evaluate the positions of learners (X) and compute fitness function F(X)
Compute the mean of all learners (X) and denoted as Mean
Determine the best learner and it can be acted as Teacher
Compute difference of dean for each dimension using $\mathrm{Mean}_{\mathrm{diff.}} = \mathrm{r}(\mathrm{T}_{\mathrm{mean}} - \mathrm{T}_{\mathrm{f}} \times \mathrm{M}_{\mathrm{i}})$
IF (DifferenceMean $< rand()$)
I
Apply genetic mutation operator on $\mathrm{T}_{\mathrm{mean}}\mathrm{and}\mathrm{M}_{\mathrm{i}}$
$\mathrm{X}_{\mathrm{i,new}} = \mathrm{X}_{\mathrm{i,old}} + \mathrm{r}_{\mathrm{i}} \times (\mathrm{X}_{\mathrm{k}} - \mathrm{X}_{\mathrm{i}})$
Compute Difference Mean and generate the new Mean of knowledge ($\mathrm{X}_{\mathrm{i,new}}$) using equation 3.
Accept $\mathrm{X}_{\mathrm{i,new}}\mathrm{IFF}(\mathrm{X}_{\mathrm{i,new}})$ is better than $\mathrm{F}(\mathrm{X}_{\mathrm{i,old}})$
Randomly pick two learners $\mathrm{X}_{\mathrm{i}}$ and $\mathrm{X}_{\mathrm{k}}$ such that i $\neq$ j
IF $(F(\mathrm{X}_{\mathrm{i}}) < F(\mathrm{X}_{\mathrm{k}})\ \&\& F(\mathrm{X}_{\mathrm{i}}) <$
Apply genetic crossover operator on $\mathrm{X}_{\mathrm{i}}\mathrm{and}\mathrm{X}_{\mathrm{k}}$ and generate the new position of
$\mathrm{X}_{\mathrm{i,new}} = \mathrm{X}_{\mathrm{i,old}} + \mathrm{r}_{\mathrm{i}} \times (\mathrm{X}_{\mathrm{k}} - \mathrm{X}_{\mathrm{i}})$
$\mathrm{X}_{\mathrm{i,new}} = \mathrm{X}_{\mathrm{i,old}} + \mathrm{r}_{\mathrm{i}} \times (\mathrm{X}_{\mathrm{k}} - \mathrm{X}_{\mathrm{i}})$
Accept $\mathrm{X}_{\mathrm{i,new}}\mathrm{IFF}(\mathrm{X}_{\mathrm{i,new}})$ is better than $\mathrm{F}(\mathrm{X}_{\mathrm{i,old}})$
Is the termination condition satisfied?
Obtained optimal solution

Figure 1: Flowchart of the proposed TLBO algorithm.

Table 1: List of test functions used for experimentation.

| No. | Function Name | Definition | Parameter |
|---|---|---|---|
| $F_1$ | Sphere | $F_1(x) = \sum_{i=0}^{D} x_i^2$ | [-100, 100] |
| $F_2$ | Rosenbrock | $F_2(x) = \sum_{i=1}^{D} 100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2$ | [-30,30] |
| $F_3$ | Rastrigin | $F_3(x) = \sum_{i=1}^{D} (x_i^2 - 10\cos(2\pi x_i) + 10)$ | [-5.12, 5.12] |
| $F_4$ | Griewank | $F_4(x) = \frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$ | [-600, 600] |
| $F_5$ | Ackley | $F_5(x) = 20 + e - 20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D} \cos(2\pi x_i)\right)$ | [-32, 32] |
| $F_6$ | Step | $F_6(x) = \sum_{i=0}^{D} (x_i + 0.5)^2$ | [-100, 100] |
| $F_7$ | Schwefel | $F_8(x) = 418.9828D - \sum_{i=1}^{D} \left(x_i \sin\left(\sqrt{\lvert x_i \rvert}\right)\right)$ | [-500, 500] |
| $F_8$ | Schaffer | $F_9(x) = 0.5 + \frac{\sin^2(x_i^2 - x_i^2) - 0.5}{[1 + 0.001(x_i^2 - x_i^2)]^2}$ | [-100, 100] |
| $F_9$ | Powell | $F_{10}(x) = \sum_{i=1}^{D/4} (x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - 10x_{4i})^2 + (x_{4i-2} + 10x_{4i-1})^4 + 10(x_{4i-3} + 10x_{4i})^4$ | [-4, 5] |
| $F_{10}$ | Zakharov's | $F_{10}(x) = \sum_{i=0}^{D} (x_i)^2 + \left(\frac{1}{2}\sum_{i=0}^{D} ix_i\right)^2 + \left(\frac{1}{2}\sum_{i=0}^{D} ix_i\right)^4$ | [-5, 10] |
| $F_{11}$ | Michalewicz | $F_{11}(x) = \sum_{i=1}^{D} \mathrm{Sin}x_1 \left(\sin\left({ix_i}^2/\pi\right)\right)^{20}$ | [0, π] |
| $F_{12}$ | Quartic | $F_{12}(x) = \sum_{i=1}^{D} ix_i^4 + \mathrm{rand}(0,1)$ | [-1.28, 1.28] |

Table 2: Benchmark test functions from CEC'14 suite.

| Function No. | Functions | Search Range | Global Optimum |
|---|---|---|---|
| $F_{13}$ | Rotated High Conditioned Elliptic Function | [-100, 100] | 100 |
| $F_{14}$ | Rotated Bent Cigar Function | [-100, 100] | 200 |
| $F_{15}$ | Rotated Discus Function | [-100, 100] | 300 |
| $F_{16}$ | Shifted and Rotated Rosenbrock's Function | [-100, 100] | 400 |

Table 3: Results of proposed TLBO and other existing algorithm with test functions $F_1$-$F_6$.

| **Algorithm** | **Parameter** | **Function** | | | | | |
|---|---|---|---|---|---|---|---|
| | | **$F_1$** | **$F_2$** | **$F_3$** | **$F_4$** | **$F_5$** | **$F_6$** |
| PSO | average | 3.00E-04 | 2.59E+01 | 3.58E-01 | 1.05E-01 | 5.21E-02 | 4.00E-04 |
| | std. | 1.50E-03 | 1.63E-01 | 6.98E-01 | 4.62E-02 | 4.06E-02 | 2.60E-03 |
| GA | average | 8.38E-01 | 4.53E+01 | 1.00E+00 | 8.61E-01 | 7.93E-01 | 7.87E-01 |
| | std. | 5.14E-01 | 2.16E-01 | 6.92E-01 | 6.48E-02 | 3.22E-01 | 5.64E-01 |
| BA | average | 1.00E+00 | 3.90E+01 | 4.27E-01 | 8.21E-01 | 1.00E+00 | 1.00E+00 |
| | std. | 1.00E+00 | 1.56E+01 | 1.00E+00 | 8.14E-02 | 1.00E+00 | 1.00E+00 |
| FPA | average | 4.28E-01 | 3.69E+01 | 5.92E-01 | 1.00E+00 | 3.17E-01 | 2.76E-01 |
| | std. | 8.26E-02 | 1.76E+01 | 3.54E-01 | 2.09E-02 | 7.36E-02 | 1.97E-01 |
| ABC | average | 6.00E-04 | 2.35E+01 | 3.29E-01 | 1.51E-02 | 9.63E-02 | 1.46E-02 |
| | std. | 1.00E-04 | 1.36E+01 | 4.25E-02 | 1.24E-02 | 7.86E-02 | 3.50E-03 |
| FA | average | 3.79E-02 | 1.72E+02 | 2.29E+01 | 1.05E-01 | 2.05E+00 | 3.11E-01 |
| | std. | 5.35E-02 | 1.30E+02 | 7.14E+00 | 5.45E-02 | 3.57E-01 | 1.18E-01 |
| BBO | average | 2.40E-03 | 5.91E+01 | 7.08E-02 | 1.89E-01 | 1.01E-01 | 1.71E+00 |
| | std. | 4.56E-04 | 1.27E+00 | 5.56E-02 | 3.36E-02 | 5.13E-02 | 3.71E-01 |
| HS | average | 5.22E-04 | 1.90E+02 | 1.69E+01 | 1.60E-01 | 1.09E+00 | 4.37E+00 |
| | std. | 3.29E-05 | 5.16E+01 | 2.66E+00 | 5.34E-02 | 1.36E-01 | 9.83E-01 |
| TLBO | average | 0 | 26.6567 | 1.87E−12 | 0 | 3.55E−15 | 2.74E−09 |
| | std. | 0 | 2.94E−01 | 6.66E−12 | 0 | 8.32E−17 | 5.36E−09 |
| Proposed TLBO | average | 0 | 23.9648 | 1.983 E-12 | 0 | 3.39 E−15 | 2.56E−11 |
| | std. | 0 | 3.26E−01 | 3.68E-12 | 0 | 6.53E−16 | 4.98E−13 |

Table 4: Results of proposed TLBO and other existing algorithm using test functions $F_5$-$F_8$.

| **Algorithm** | **Parameter** | **Functions** | | | | | |
|---|---|---|---|---|---|---|---|
| | | **$F_7$** | **$F_8$** | **$F_9$** | **$F_{10}$** | **$F_{11}$** | **$F_{12}$** |
| PSO | average | 9.82E+03 | 3.86E+02 | 5.13E-01 | 6.14E-10 | -2. 96847 | 4.36E-03 |
| | std. | 6. 412E+02 | 1.13E+02 | 3.46E-03 | 1.35E-11 | 6.52E-01 | 5.71E-04 |
| GA | average | 4.62E+04 | 8.30E+02 | 5.40E-01 | 5.50E-10 | -5.86E+00 | 2.35E-01 |
| | std. | 1.59E+03 | 3.93E+02 | 7.31E-03 | 7.62E-10 | 9.16E-01 | 6.71E-02 |
| BA | average | 2.76E+02 | 1.14E+02 | 8.25E-02 | 9.83E-14 | -4.54E+00 | 4.13E-03 |
| | std. | 1.12E+02 | 8.61E+01 | 1.83E-02 | 4.86E-14 | 7.16E-01 | 7.87E-03 |
| FPA | average | 3.53E+02 | 8.62E+01 | 7.92E-02 | 5.22E-14 | -3.86E+00 | 3.86E-03 |
| | std. | 1.26E+03 | 8.07E+01 | 3.82E-02 | 4.80E-14 | 8.35E-01 | 1.55E-03 |
| ABC | average | 2.79E+04 | 2.35E+01 | 3.29E-01 | 1.51E-02 | 9.63E-02 | 1.46E-02 |
| | std. | 2.10E+01 | 1.36E+01 | 4.25E-02 | 1.24E-02 | 7.86E-02 | 3.50E-03 |
| FA | average | 3.79E+02 | 1.72E+02 | 2.29E+01 | 1.05E-01 | 2.05E+00 | 3.11E-01 |
| | std. | 5.35E+01 | 1.30E+02 | 7.14E+00 | 5.45E-02 | 3.57E-01 | 1.18E-01 |
| BBO | average | 1.38E+02 | 7.50E-01 | 2.46E+02 | 1.30E-03 | -3.92E+00 | 2.00E-05 |
| | std. | 1.14E+02 | 3.86E-02 | 6.47E+01 | 1.85E-03 | 1.31E+00 | 3.14E-06 |
| HS | average | 4.59E+02 | 1.53E-01 | 2.83E+02 | 1.03E-11 | -4.32E+00 | 8.44E-04 |
| | std. | 5.86E+01 | 3.10E-02 | 1.06E+02 | 1.31E-12 | 1.06E+00 | 2.21E-05 |
| TLBO | average | 124.1484 | 0.0066 | 0.0066 | 4.64E-14 | -4.352678 | 3.25E-04 |
| | std. | 2.60E+02 | 4.50E-03 | 4.50E-03 | 2.34E-14 | 1.44E-02 | 1.59E-04 |
| Proposed TLBO | average | 104.526 | 0.0058 | 0.0052 | 8.35E-16 | -3.73415 | 5.14E-03 |
| | std. | 4.13E+01 | 3.66E-03 | 4.48E-03 | 1.42E-16 | 2.43E−2 | 2.86E-03 |

Table 5: Performance comparison of proposed algorithm and other meta-heuristic algorithms with extended benchmark functions of CEC'14.

| Algorithm | Parameter | Functions | | | |
|---|---|---|---|---|---|
| | | $F_{13}$ | $F_{14}$ | $F_{15}$ | $F_{16}$ |
| PSO | Average | 9.78E+04 | 1.69E+03 | 5.73E+02 | 4.04E+02 |
| | Std | 2.82E+03 | 7.32E+02 | 1.92E+02 | 1.46E+01 |
| GA | Average | 9.02E+05 | 7.34E+03 | 1.18E+03 | 2.13E+03 |
| | Std | 3.16E+08 | 3.18E+03 | 2.68E+02 | 1.67E+03 |
| BA | Average | 3.46E+03 | 9.84E+02 | 3.74E+02 | 4.27E+02 |
| | Std | 1.73E+02 | 3.11E+02 | 1.38E+02 | 2.43E+01 |
| FPA | Average | 4.41E+03 | 7.63E+02 | 3.48E+02 | 4.19E+02 |
| | Std | 2.26E+02 | 2.73E+02 | 100% | 1.78E+01 |
| ABC | average | 1.02E+03 | 5.35E+02 | 2.14E+02 | 3.52E+02 |
| | std. | 7.94E-05 | 1.47E+02 | 5.64E+01 | 1.24E+02 |
| FA | average | 3.79E+03 | 4.87E+02 | 2.30E+02 | 3.75E+02 |
| | std. | 9.91E+01 | 1.49E+02 | 7.14E+01 | 6.85E+01 |
| BBO | average | 1.73E+03 | 4.38E+02 | 4.09E+02 | 3.90E+02 |
| | std. | 4.36E+01 | 9.13E+01 | 5.88E+01 | 4.73E+01 |
| HS | average | 9.23E+02 | 3.19E+02 | 2.68E+02 | 2.16E+02 |
| | std. | 1.33E+02 | 2.66E+01 | 6.69E+01 | 5.35E+01 |
| SFLP | average | 1.09E+03 | 2.98E+02 | 3.28E+02 | 2.57E+02 |
| | std. | 1.08E+02 | 3.15E+01 | 5.32E+01 | 3.79E+01 |
| TLBO | Average | 3.82E+03 | 7.42E+02 | 3.26E+02 | 4.08E+02 |
| | Std | 2.58E+02 | 2.42E+02 | 1.42E+02 | 1.96E+01 |
| Proposed TLBO | Average | 3.27E+03 | 5.24E+02 | 3.04E+02 | 3.41E+02 |
| | Std | 1.92E+02 | 2.16E+02 | 1.98E+02 | 7.05E+01 |

the various unimodal and multi-modal test functions taken from CEC, 14. Table 1 shows the normal benchmark function of the CEC'14, whereas Table 2 contains the some extended benchmark function of CEC'14.

Table 3 demonstrates the results of proposed algorithm and other algorithm like PSO, GA, BA, FPA, ABC, FA, BBO, HS and TLBO using test functions $F_1$-$F_4$.These functions are widely adopted to investigate the performance of newly developed algorithms. It is observed from the results that the proposed TLBO algorithm achieves global optimum value for rosenbrock and rastrigin functions within specified number of iterations. It is seen that the performance of proposed algorithm and TLBO algorithm is same for sphere and griewank functions. Further on the analysis of standard deviation parameter, it is observed thatproposed TLBO algorithm obtains minimum standard deviation value in comparison to other algorithms being compared. Its reveals that the proposed algorithm provides more stable result for solving benchmark test functions.

Table 4 illustrates the experimental results of proposed algorithm and other algorithm for standard benchmark functions $F_7$-$F_{12}$. It is seen that proposed algorithm obtains global optimum values i.e. minimum values among all other compared algorithm using most of functions. It is observed that significant difference occurs between the performance of the proposed algorithm and rest of algorithms being compared. On the analysis of standard deviation parameter, it is stated that again the proposed algorithm gets minimum value for standard deviation parameter among all other algorithms. This indicates that the proposed algorithm provides more stable results for solving these functions. From tables 3-4, it is stated that the proposed algorithm is an effective and efficient algorithm for solving bench mark test function and this algorithms also provides more stable results in comparison to other algorithms being compared. Table 5 demonstrates the comparison of the proposed TLBO algorithm and other meta-heuristic algorithms with the extended benchmark functions ($F_{13}$-$F_{16}$) of the CEC'14. To show the effectiveness of the proposed algorithm four well known benchmark functions are taken from the extended benchmarks functions set of CEC'14. It is observed that the proposed algorithm obtains better optimum value as compared to other algorithms being compared. Hence, it can be concluded that the proposed algorithm is one of the efficient algorithm for solving global optimization problems.

Figs. 2-3 show the convergence pattern i.e. cost function of proposed TLBO and original TLBO algorithm using rastrigin and rosenbrock functions. From these, it is also stated that the convergence of the TLBO algorithm is significantly improved.From Fig. 2, it is seen that the proposed algorithm requires less number of iterations to converge than TLBO algorithm. Further, it is also observed that proposed algorithm obtains minimum cost than original TLBO algorithm. On analysis of Fig. 3, it is noted that the proposed algorithm converges fast than the original TLBO algorithm. It is also noticed that there is the significant difference between the initial solution obtained through proposed algorithm and original TLBO algorithm. Finally, it is concluded that the proposed modifications not only improve the performance of TLBO algorithm, but also enhance the convergence rate of algorithm.

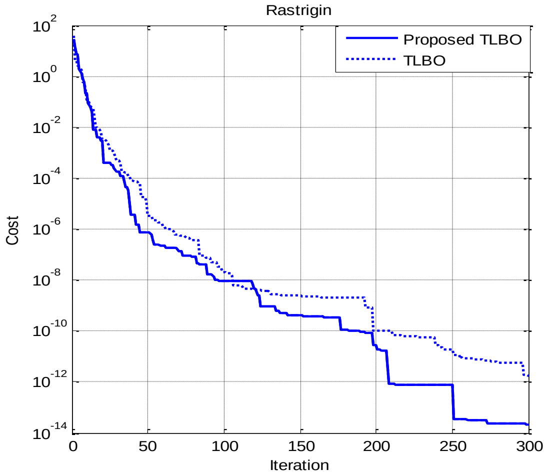


Figure 2 shows the convergence of TLBO and Proposed TLBO algorithm for rastrigin function

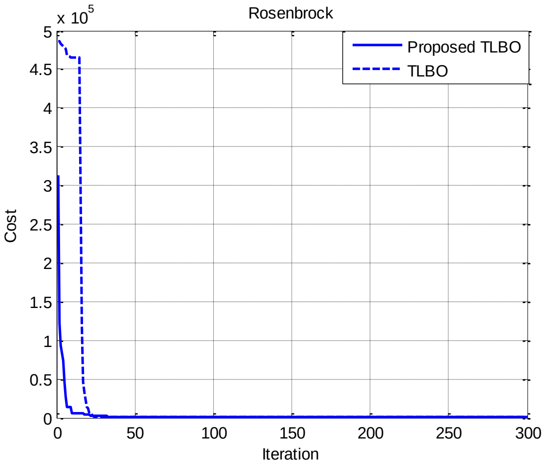


Figure 3: shows the convergence of TLBO and Proposed TLBO algorithm for rosenbrock function

## 6 Conclusion

In this work, a new variant of TLBO algorithm is presented for solving the global optimization problems. For improving the performance of TLBO algorithm, two modifications are incorporated into teacher and learner phases of TLBO algorithm. These modifications are genetic crossover and mutation operators and the aim of these operators to generate diverse population and to improve searching ability and convergence rate of TLBO algorithm. The genetic crossover operator is applied in learner phase for determining the good learner from the set of learners. The aim of genetic mutation operator is to minimize the knowledge gap between teacher and learners. So, this operator is applied in teacher phase of TLBO algorithm to generate diverse population. The performance of the proposed algorithm is evaluated using a set of benchmark test functions using mean and SD parameters and the results are compared with some of state of art algorithm available in literature. From experimental study, it is seen that the performance of the proposed algorithm is better than other algorithms being compared. It is also observed that proposed algorithm provides state of art results with most of benchmark test functions.

## 7 Future work

In future research work, the proposed TLBO algorithm is adopted for solving single objective and multi-objective constrained optimization problems. Further, neighborhood search mechanism is introduced to explore good candidate solution and also for improving convergence rate. In teaching learning process, selection of teacher also impact on the performance of learners. In future work, the effect of number of teachers will be evaluated on the fitness value of objective function. Apart from above, the capability of TLBO algorithm will explore in different research problems such as classification, feature selection, document clustering, parameter optimization of ANN and SVM techniques etc.

## References

[1] Stützle, T. (1998). Local search algorithms for combinatorial problems. Darmstadt University of Technology PhD Thesis, 20.

[2] Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. science, 220(4598), 671-680.

[3] Holland, J. H. (1992). Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press.

[4] Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. Proceedings of the Sixth IEE International Symposium on Micro Machine and Human Science, 39-43.

[5] Dorigo, M., Birattari, M., & Stützle, T. (2006). Ant colony optimization artificial ants as a computational intelligence technique. IEEE Comput. Intell. Mag. 1, 28–39.

[6] Lee, K. S., & Geem, Z. W. (2004). A new structural optimization method based on the harmony search algorithm. Computers & structures, 82(9-10), 781-798.

[7] Karaboga, D., & Basturk, B. (2007). Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems. In International fuzzy systems association world congress, 789-798.

[8] Yang, X. S. (2009). Firefly algorithms for multimodal optimization. In International symposium on stochastic algorithms, 169-178.

[9] Kashan, A. H. (2011). An efficient algorithm for constrained global optimization and application to mechanical engineering design: League championship algorithm (LCA). Computer-Aided Design, 43(12), 1769-1792.

[10] Eskandar, H., Sadollah, A., Bahreininejad, A., & Hamdi, M. (2012). Water cycle algorithm–A novel metaheuristic optimization method for solving constrained engineering optimization problems. Computers & Structures, 110, 151-166.

[11] Kaveh, A., & Talatahari, S. (2010). A novel heuristic optimization method: charged system search. Acta Mechanica, 213(3-4), 267-289.

[12] Kumar, Y., & Sahoo, G. (2014). A charged system search approach for data clustering. Progress in Artificial Intelligence, 2(2-3), 153-166.

[13] Kaveh, A., Share, M. A. M., & Moslehi, M. (2013). Magnetic charged system search: a new meta-heuristic algorithm for optimization. Acta Mechanica, 224(1), 85-107.

[14] Kumar, Y., & Sahoo, G. (2015). Hybridization of magnetic charge system search and particle swarm optimization for efficient data clustering using neighborhood search strategy. Soft Computing, 19(12), 3621-3645.

[15] Kumar, Y., Gupta, S., Kumar, D., & Sahoo, G. (2016). A clustering approach based on charged particles. In Optimization Algorithms-Methods and Applications. InTech.

[16] Rao, R. V., Savsani, V. J., & Vakharia, D. P. (2011). Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. Computer-Aided Design, 43(3), 303-315.

[17] Sahoo, A. J., & Kumar, Y. (2014). Modified teacher learning based optimization method for data clustering. In Advances in Signal Processing and Intelligent Recognition Systems, 429-437.

[18] Sadollah, A., Bahreininejad, A., Eskandar, H., & Hamdi, M. (2013). Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems. Applied Soft Computing, 13(5), 2592-2612.

[19] Rao, R. V., Savsani, V. J., & Balic, J. (2012). Teaching–learning-based optimization algorithm for unconstrained and constrained real-parameter optimization problems. Engineering Optimization, 44(12), 1447-1462.

[20] Rao, R. V., Savsani, V. J., & Vakharia, D. P. (2012). Teaching–learning-based optimization: an optimization method for continuous non-linear large scale problems. Information sciences, 183(1), 1-15.

[21] Zhile, Y. A. N. G., Kang, L. I., Qun, N. I. U., Yusheng, X. U. E., & Foley, A. (2014). A self-learning TLBO based dynamic economic/environmental dispatch considering multiple plug-in electric vehicle loads. Journal of Modern Power Systems and Clean Energy, 2(4), 298-307.

[22] Chen, C. H. (2013, December). Group Leader Dominated Teaching-Learning Based Optimization. In International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT), 304-308.

[23] Yang, Z., Li, K., Foley, A., & Zhang, C. (2014, July). A new self-learning TLBO algorithm for RBF neural modelling of batteries in electric vehicles. In IEEE Congress on Evolutionary Computation (CEC), 2685-2691.

[24] Sahoo, A. J., & Kumar, Y. (2018). A chaotic teaching learning based optimization algorithm for clustering problems. Applied Intelligence (Accepted) DOI: 10.1007/s10489-018-1301-4

[25] Rao, R. V., & Patel, V. (2013). An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems. Scientia Iranica, 20(3), 710-720.

[26] Satapathy, S. C., & Naik, A. (2014). Modified Teaching–Learning-Based Optimization algorithm for global numerical optimization—A comparative study. Swarm and Evolutionary Computation, 16, 28-37.

[27] Huang, J., Gao, L., & Li, X. (2015). An effective teaching-learning-based cuckoo search algorithm for parameter optimization problems in structure designing and machining processes. Applied Soft Computing, 36, 349-356.

[28] Zou, F., Wang, L., Hei, X., & Chen, D. (2015). Teaching–learning-based optimization with learning experience of other learners and its application. Applied Soft Computing, 37, 725-736.

[29] Ouyang, H. B., Gao, L. Q., Kong, X. Y., Zou, D. X., & Li, S. (2015). Teaching-learning based optimization with global crossover for global optimization problems. Applied Mathematics and Computation, 265, 533-556.

[30] Ghasemi, M., Taghizadeh, M., Ghavidel, S., Aghaei, J., & Abbasian, A. (2015). Solving optimal reactive power dispatch problem using a novel teaching–learning-based optimization algorithm. Engineering Applications of Artificial Intelligence, 39, 100-108.

[31] Zou, F., Wang, L., Hei, X., Chen, D., & Yang, D. (2014). Teaching–learning-based optimization with

dynamic group strategy for global optimization. Information Sciences, 273, 112-131.

[32] Lim, W. H., & Isa, N. A. M. (2014). An adaptive two-layer particle swarm optimization with elitist learning strategy. Information Sciences, 273, 49-72.

[33] Kumar, Y. & Sahoo, G. (2017). A two-step artificial bee colony algorithm for clustering. Neural Computing and Applications, 28(3), 537-551.

[34] Ghasemi, M., Ghavidel, S., Rahmani, S., Roosta, A., & Falah, H. (2014). A novel hybrid algorithm of imperialist competitive algorithm and teaching learning algorithm for optimal power flow problem with non-smooth cost functions. Engineering Applications of Artificial Intelligence, 29, 54-69.

[35] Ghasemi, M., Ghanbarian, M. M., Ghavidel, S., Rahmani, S., & Moghaddam, E. M. (2014). Modified teaching learning algorithm and double differential evolution algorithm for optimal reactive power dispatch problem: a comparative study. Information Sciences, 278, 231-249.

[36] Shahbeig, S., Helfroush, M. S., & Rahideh, A. (2017). A fuzzy multi-objective hybrid TLBO–PSO approach to select the associated genes with breast cancer. Signal Processing, 131, 58-65.

[37] Kumar, M., Mittal, M. L., Soni, G., & Joshi, D. (2018). A hybrid TLBO-TS algorithm for integrated selection and scheduling of projects. Computers & Industrial Engineering, 119, 121-130.

[38] Patel, S. J., Toshniwal, A., & Kheraj, V. (2017). A novel teaching-learning based optimization approach for design of broad-band anti-reflection coatings. Swarm and evolutionary computation, 34, 68-74.

[39] Ji, X., Ye, H., Zhou, J., Yin, Y., & Shen, X. (2017). An improved teaching-learning-based optimization algorithm and its application to a combinatorial optimization problem in foundry industry. Applied Soft Computing, 57, 504-516.

[40] Cheng, M. Y., & Prayogo, D. (2017). A novel fuzzy adaptive teaching–learning-based optimization (FATLBO) for solving structural optimization problems. Engineering with Computers, 33(1), 55-69.

[41] Das, S. P., & Padhy, S. (2018). A novel hybrid model using teaching–learning-based optimization and a support vector machine for commodity futures index forecasting. International Journal of Machine Learning and Cybernetics, 9(1), 97-111.

[42] Kankal, M., & Uzlu, E. (2017). Neural network approach with teaching–learning-based optimization for modeling and forecasting long-term electric energy demand in Turkey. Neural Computing and Applications, 28(1), 737-747.

[43] Kiziloz, H. E., Deniz, A., Dokeroglu, T., & Cosar, A. (2018). Novel multiobjective TLBO algorithms for the feature subset selection problem. Neurocomputing, 306, 94-107.

[44] Prakash, T., Singh, V. P., Singh, S. P., & Mohanty, S. R. (2018). Economic load dispatch problem: quasi-oppositional self-learning TLBO algorithm. Energy Systems, 9(2), 415-438.

[45] Zheng, H. Y., Wang, L., & Zheng, X. L. (2017). Teaching–learning-based optimization algorithm for multi-skill resource constrained project scheduling problem. Soft Computing, 21(6), 1537-1548.

[46] Birashk, A., Kordestani, J. K., & Meybodi, M. R. (2018). Cellular teaching-learning-based optimization approach for dynamic multi-objective problems. Knowledge-Based Systems, 141, 148-177.

[47] Kumar, Y., & Singh, P. K. (2017). Improved cat swarm optimization algorithm for solving global optimization problems and its application to clustering. Applied Intelligence, 48(9), 2681–2697.

[48] Kumar, Y., & Sahoo, G. (2017). An improved cat swarm optimization algorithm based on opposition-based learning and Cauchy operator for clustering. JIPS (J Inf Process Syst), 13(4), 1000-1013.

# An Empirical Study for Detecting Fake Facebook Profiles Using Supervised Mining Techniques

Mohammed Basil Albayati and Ahmad Mousa Altamimi
Applied Science Private University, Amman, Jordan
E-mail: mohammed.sabri@asu.edu.jo, a_altamimi@asu.edu.jo

**Keywords:** data mining, online social networks, facebook, fake profiles, data science

**Received:** April 29, 2018

*Our social life and the way of people communicate are greatly affected by the social media technologies. The variety of stand-alone and built-in social media services such as Facebook, Twitter, LinkedIn, and alike facilitate users to create highly interactive platforms. However, these overwhelming technologies made us sank in an enormous amount of information. Recently, Facebook exposed data on 50 million Facebook unaware users for analytical purposes. Fake profiles are also used by Scammers to infiltrate networks of friends to wreak all sorts of havoc as stealing valuable information, financial fraud, or entering other user's social graph. In this paper, we turn our focus to Facebook fake profiles, and proposed a smart system (FBChecker) that enables users to check if any Facebook profile is fake. To achieve that, FBChecker utilizes the data mining approach to analyze and classify a set of behavioral and informational attributes provided in the personal profiles. Specifically, we empirically examine these attributes using four supervised data mining algorithms (e.g., k-NN, decision tree, SVM, and naïve Bayes) to determine how successfully we can recognize the fake profiles. To demonstrate the validity of our conceptual work, the selected classifiers have been implemented using RapidMiner data science platform with a dataset of 200 profiles collected from the authors' profile and a honeypot page. Two experiments are developed; in the first one, the k-NN schema is applied as an estimator model for imputation the missing data with substituted values, whereas in the second experiment a filtering operator is applied to exclude the profiles with missing values. Results showed high accuracy rate with the all classifiers, however, the SVM outperforms other classifiers with an accuracy rate of 98.0% followed by Naïve Bayes.*

*Povzetek: Opisana je metoda iskanja lažnih profilov na Facebooku s pomočjo strojnega učenja.*

## 1 Introduction

In recent years, social media technologies (e.g., Facebook, Twitter, LinkedIn, etc.) have become a vital part of our life [1]. They are designed and maintained by social media organizations presenting a portal for facilitating communication, interaction, sharing information, and entertainment via virtual communities and networks. Users typically utilize these services by creating their own profiles and then connecting them with others' profiles through various technologies that offer social media functionality [2]. By using such services, users can create digital contents, such as text posts, comments, digital photos, videos, or data generated through all online interactions [3].

Social media sites have presented a various service included with high levels of quality, consistency, and The Introduction should provide a clear background, a clear statement of the problem, the relevant literature on the subject, the proposed approach or solution, and the new value of research which it is innovation and availability. This results in huge registered users [4]. Some of the most popular social media websites are Facebook (and its associated Facebook Messenger), Gab, Google+, MySpace, Instagram, LinkedIn, and others. Statistics and surveys for example the one that conducted by the American Academy of Pediatrics exhibit that about 84% of adolescents in America registered on Facebook social online site [5], also showed that the average users spend more than two hours on social network and even more on social online sites such as Facebook, Twitter and else more than any other sites or platform [6]. The benefit of engaging and participating in social online sites have gone beyond simply social online activities, sharing information, or communication but to building careers, making business opportunity, financial income, etc. [7].

Historically, according to Mark Zuckerberg, a co-founder of Facebook which is the largest social network site, Facebook have more than 175 million active users registered in 2009 after just five years' time frame. Nowadays, Facebook has 1.94 billion users on the last official announcement on March 31st, 2017, according to Facebook newsroom administration [8], which exceeds the population of some big countries. With this rapid revolution in this technology, number of negative consequences and risks are raised such as security risks, privacy violation, cloning, hacking, spamming, and others [3]. For example, Spam on social media repeatedly posts the same thing over and over or causes a sudden spike in messaging activity [9]. Fake profiles on the other hand, allow scammers to infiltrate networks of friends to wreak all sorts of havoc such as: stealing valuable information, financial fraud, or entering other user's social graph [10].

It is important to mention here that according to the Facebook's Statement of Rights and Responsibilities; users should provide their real and legit information once they created their profiles. Facebook urges its users to be committed to these policies and terms in order to have an experience in an environment of safety, security, and privacy [11]. In this work, we focused on the problem of detecting fake profiles in Facebook and presenting a smart detection system (FBChecker) to handle this problem based on the prediction and classification techniques of data mining.

Our work is motivated by works presented in [12-15], where researchers employed data mining approach for extracting hidden knowledge within social media. For example, authors of [12] utilized data mining tools for accurately capturing the behavior of intrusions and normal activates in an anomaly detections approach. One can consider also the Web mining that applies data mining tools onto web resources to further developments in World Wide Web mining [15].

In our model, supervised mining techniques are applied to classify Facebook's profiles into fake and real profiles based on a set of behavioral and informational attributes. These attributes are provided in their personal profiles and used to identify the reality of user's identity such as: person's legal name, location, workplace, age, education, and others. The required data set for the training and testing purposes in our work has been collected from the authors' personal profiles considered as a source of real profiles, and from a created honeypot page, which is fake Facebook's pages used for the purposes of data harvesting [10] to attract and collect these profiles. As a collecting tool we wrote our own script to develop a special CRAWLER for gathering the required data set. To underscore the practical viability of our approach, the selected classifiers (e.g., SVM, Naïve Bayes, k-NN, and Decision Tree) have been implemented using RapidMiner data science platform for the mining tasks. These classifiers were evaluated using 10-folds cross validation method and conducted on the collected data set. It is important to mention here that 33 records have some missing values of their attributes.

To solve this problem, two empirical studies were developed, in the first one, the k-NN schema was used as an estimator model for imputation the missing data with substituted values. Results showed that the classifiers (SVM, Naïve Bayes, k-NN, and Decision Tree) achieved (0.9850, 0.9700, 0.8400, 0.9650), respectively. In the second experiment, a filtering operator is applied to exclude the profiles with missing values. Here, the classifiers showed (0.9880, 0.9641, 0.8443, 0.9461), which are relatively equal to the results of the first experiment. The numbers and the ROC graph (Receiver Operating Characteristics) which is a graphical plot utilized to assess the classifiers performance ability showed that, in the both experiments SVM classifier achieved the highest accuracy rates while, the k-NN performance showed the lowest accuracy detection rate among the classifiers. These experiments are discussed in more details in Section 5.

The remaining of this paper is structured as following: Section 2 reviews the related works to the proposed approach and to the fake profiles in online social networks, specifically the Facebook. Section 3 describes the background material of the research work along with brief description of the employed supervised algorithms. Section 4 explains the research methodology while the proposed system along with its main components presents in section 5. Section 6 discusses the implementation of the FBChecker system, the evaluation and experimental results are given in Section 7. Finally, section 8 offers the conclusion and the possible future work. introduction.

## 2 Related work

Many studies and works have been conducted focusing on the phenomena of fake profiles on online social networks, each researcher tried to came up with new way to detect and handle this problem. Studies in this field differ according to how they look at the problem from their own perspectives. Each of which is raised for solving a certain problem and faces certain challenges and difficulties. In this regard, many approaches presented in the literature for handling fake profiles.

One can consider for example, the work in [16]. Here, the authors present a machine learning pipeline framework consists of three components for detecting clusters of duplicate accounts (cluster level detection) rather than making a prediction for an individual account. Here, the pipeline uses simple information that is provided at the registration time, so the profile is detected before it is activated. Moreover, the classifier determines whether the clusters of accounts were created by the same actor, showing a strong evaluation on sample grouping based on the simple text information like name, email, company, etc. and the IP address. Practically. The system captures more than 250,000 fake accounts in practical use. In contrast, [17] proposed a behavioral approach for detecting fake accounts on Facebook. It is designed using information regarding user profile's activities and interactions with other users. Authors characterized these activities through an extensive set of 17 features like (likes, comments, shares, tag, and apps usage on Facebook). To ground their idea, these features are applied on a total of 12 supervised machine learning techniques. The system's performance showed an accuracy of 79%, which may not be impressive results, but the author considered it as a first step or baseline work for further improvements.

Detecting Spam profiles, which is one of the fake profiles types [10], has also considered in the literature. Authors of [18] proposed a statistical analyzing model with 14 generic features from Facebook and Twitter data set regarding 4 basic kinds of social interactions including (profile interaction features, posts/ tweets, URLs and tags & mentions). The model identifies spam profiles on Facebook and Twitter based on information collected manually through scanning these networks for both normal and spam profiles using three different supervised classification algorithms (naive Bayes, Jrip, and J48). Then two different experiments were performed: firstly, examining the role of the whole feature set and calculate the accuracy of the proposed system. And secondly, removing each one of the features and analyzing the results of the system to discover the impact of each features and find out which one can play the key role in the classification model.

Detecting spam profiles is also presented in the literature as in [19], Presented Social Privacy Protector software (SSP) for detecting fake profiles on Facebook, the SSP consist of three protection layers: The software first identifies a user's friends who might pose a threat and then restricts this "friend's" exposure to the user's personal information (The Friends Analyzer Facebook Application). The second layer is an expansion of Facebook's basic privacy settings based on different types of social network usage profiles (The Social Privacy Protector Firefox Add-on). The third layer alerts users about the number of installed applications on their Facebook profile, which have access to their private information (The HTTP Server). The software present convenient method for restrict the users that may be suspected as fake profiles without removing it from the user's friends list.

The Friends Analyzer Application on the Facebook scans the user's friends list and returns a credibility score. Each friend analyzed by machine learning algorithms which takes into account the strength of the connection between the user and his friends. The strength of each connection is based on a set of fifteen connection features depends on three types of the collected dataset, such as the number of common friends between the user and his friend and the number of pictures and videos the user and his friend were tagged in together. Applying eight supervised algorithms such as (Naive-Bayes, Bagging, Random-Forest, J48, and others). The Social Privacy Protector add-on in the Firefox browser help improve the user privacy with simple steps. Finally, The HTTP server responsible for connecting the SPP Firefox Add-on to the SPP Facebook application. Authors of [20] proposed a framework for detecting spammers/ fake profiles on online social network using Facebook as test case in a machine learning approach by exploiting a behavioral and community-based features (attributes) that include the structure of the nodes and some topological features (attributes) in the network.

The framework implemented using WEKA tool as mining environment, using ten discriminative topological attributes (Total out-degree, Total in/out ratio, Total reciprocity, Core node, Community memberships, Foreign out-degree, Foreign in/out ratio, Foreign out-link probability, Foreign reciprocity, and Foreign out-link grouping) regarding the social interactive of the profiles like number of posts, number of sent/ received messages…etc. Four experiments are conducted using two datasets: Facebook dataset and Enron network (Email messages dataset). Four supervised classifiers are employed in this work (Naïve Bayes, J48, k-NN, and Decision Tree).

Ultimately, authors of [21] proposed a machine learning approach for detecting spam bots in Twitter online social network through exploiting two main spam features, which are: The graph-based features including the number of friends, number of followers and the follower's ratio (the ratio of the number of peoples following you to the number of peoples you follow). And the content-based approach which is the number of duplicated tweets, number of HTTP links, and the number of replays/mentions. Regarding the detection process, the approach applied different classification methods such as decision tree, neural network, support vector machines, and k-nearest neighbors to identify spam bots on Twitter. The evaluating results showed that the Bayesian classifier has a better overall performance.

# 3 Background material

Data Mining basically is the process of extracting Knowledge from a huge amount of data, by looking for a pattern, identified, validated, make a prediction and summarize it into useful information. Data mining process goes through a sequence of procedures, applying set techniques, combining several of discipline and fields like statistics, machine learning, database, algorithms visualization methods, pattern recognition and other disciplines [22].

## 3.1 Machine learning techniques in data mining

Machine learning is a branch of computer science, which deals with algorithms that have the ability to learn and adapt to make a decision [22]. One of the most common tasks that data mining offers is Classification & Predication in which they fall into the machine learning techniques.

In machine learning, there are two main techniques known as Supervised Learning, where the training dataset has a class label, and Unsupervised Learning, where the data are grouped together based on observable behavior or features. In other words, in supervised, a labeled set of training data is used to estimate or map the input data to the desired output. In contrast, under the unsupervised methods, no labeled examples are provided and there is no notion of the output during the process, instead the data with similar attributes or similar behavior are grouped together (clustered) [22, 23].

In this work only the supervised techniques have been employed as mentioned, particularly four supervised techniques that are: SVM, Decision Tree, k-NN, and Naïve Bayes. A brief description about these classifiers will be presented in the next subsection.

### 3.1.1 Supervised learning

Supervised machine learning is a heuristic process of mapping inputs to specific output, estimating unknowns based on labeling samples. The objective of supervised learning technique is to build a model with distinguished features and predefining labels with a known class, then using this model to classify or predict a new data with unknown class.

The process of classification and prediction in supervised machine learning involves two major steps:

- The learning step: the model constructed, analyzed and trained with known label dataset called "training set", then the classification and prediction rules are generated.
- The classification and prediction step: the model (classifier) used for classifying or predicting a given data based on the gained experience from the training set. The model's results are evaluated through testing and evaluating process to estimate its accuracy.

The test metrics use to assess how good or how accurate the classifier was. If the it reaches a level of accuracy that is acceptable based on specific standards, then the model can be deployed on new unknown labeled data, otherwise it will be modified [24].

In our work, we choose to employ the most common supervised machine learning algorithms that are:

1- Decision Tree is a predictive model takes a tree structure that generates the classification rule by breaking down the dataset into smaller and smaller subset until the decision node (class label) is met. Each node in the tree represents an attribute of the training set, however, leaf nodes hold the class label (final outcome), while the root node represents the attribute with highest information gain that determines the tree branches in which each branch represents one of the outcomes of the model.

2- k-NN is one of the simplest algorithms perform similarity functions, which store all cases with a known label and classifies new data based on the similarity measures or distance function. k-NN classify new data by using k value to find the nearest case in the data set, for example if (k =1) then simply assign the new case to the class of its first nearest neighbor, if the (k = 3) then k-NN calculate the distance of the nearest three cases and apply majority vote on the class of these cases to decide the class of the new data. The distance measures for finding the nearest neighbor for the numerical data is calculated by the Euclidian distance function and for the categorical data hamming distance measure.

3- Support vector machine algorithm is a classification technique designed to define a hyperplane that classify the training data vectors into classes, the goal or the best choice is to find a hyperplane with widest margin to separate the data classes. The support vector are the data points which are closest to the hyperplane.

4- Finally, Naïve Bayes or simple Bayesian classifier is considered also in the mining process as a supervised classification technique as it is simple and prove its effectiveness, Naïve Bayes is probabilistic algorithm depends on applying Bayesian theorem with naïve assumption that the occurrence of one of the attributes\ predictors are independent of the occurrence of other attribute and regardless of any correlation between these attributes in the classification process. Bayes rules adopted in this algorithm stated a conditional probability of certain event based on previous knowledge about that event [22, 23].

## 4 Research methodology

In this work, we developed a smart system (FBChecker) that enables users to detect the fake Facebook profiles by utilizing the supervised data mining techniques. To do so, the system firstly, collects the data of a set of behavioral and informational attributes derived from the user's friends' profiles (listed in table 1). To achieve this, a special purpose module (called CRAWLER) is developed to collect the required attributes from the user's friends list. CRAWLER is running at the user level for collecting this data. Secondly, the collected data is validated to increase the accuracy of the detection process. Specifically, the problem of missing values has been solved using two methods, the k-NN scheme and a special operator to exclude them. Ultimately, a set of supervised mining algorithms are implemented using the RapidMiner data science platform to detect the fake profiles. The main objective of using the supervised machine learning techniques is to build a model with distinguished features and predefining labels with a known class, then using this model to classify or predict a new data with unknown labels. This process involves two major steps. Firstly, the learning step that includes constructing, analyzing and training with known label data set (training set), then the classification and prediction rules are generated. Secondly, the classification and prediction step that the learner model (classifier) gives data based on the gained experience from the training set.

## 5 The FBChecker smart system

Figure 1 illustrates the main components of proposed FBChecker System. In this section, we discuss the steps that followed carefully to build up the system along with its main components.

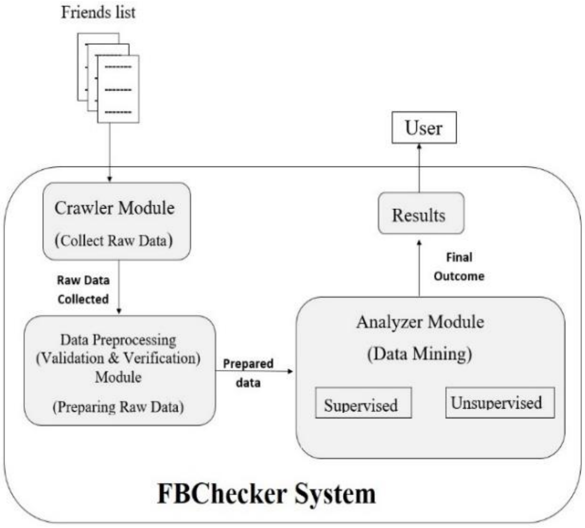


Fig. 1: FBChecker System Components.

1- Collecting the required data: first thing needed to be considered in building a machine learning system is collecting the required data for the training and testing purposes. In this regard, a special purpose module (CRAWLER) was developed and written in

Java Script for collecting the required attributes form the user's friends list. The considered attributes are listed in table 1 along with their description and their using justifications.

**2-** Preparing the data: the raw data need to be prepared and validated to increase the data quality and to be eligible for applying the mining techniques. Here, the preparation process is done as following:

- *Missing Values*: we note that some profiles have missing values due to privacy issues or the users did not fill these attributes with required information. To solve this problem, two methods have been applied, the k-NN schema is applied as an estimator model for imputation the missing data with substituted values, and a filtering operator is applied to exclude the profiles with missing values.
- *Profile Picture*: it is recognized by the user himself as a real picture or not.
- *Education*: it is validated according to a multi-lingual database of size ~10,000 records of colleges and universities existed around the world.
- *About "Bio." Section*: making a textual condition, if the number of words in this section greater or equal 5 return true/real otherwise false/fake value.
- *Other attributes*: such as *Relationship Status, Life Events, Living Place*, and *Check Ins* do not need to be validated as Facebook evaluates the attributes' values. So, the CRAWLER module retrieves them as is.

**3-** Training and Appling the Supervised data mining algorithms: after the data is prepared and ready for mining, a supervised data mining technique is applied (Analyzer module). The classifiers are trained with known class data that are (Fake, Real) profiles. At this step, the system gaines the experience and the ability to classify and detect the fake profiles. In addition, the classification rules are generated and prepared through applying the supervised algorithms. Finally, the selected supervised data mining algorithms are applied using the prepared collected data for detecting fake profiles.

| Attribute | Description | Justification |
|---|---|---|
| Profile Picture | Visual identification of the user | Real users use their real pictures more often than fake users |
| Work place | Workplace or job title's information | Real users more often use their real workplace information than fake users |
| Education | Attended (school, college, university…etc.) information | Real users mentioned their education information in their Facebook profiles more often than fake users |
| Living Place | Living place address<br>(city, town, state…etc.) information | Real users more often use their real living place information than fake users |
| Relationship Status | Social relation status (married, single, engaged, etc.) information | Real users share their real social relation status than fake users |
| Check In | Information for announcing user location | Real users check into places in their Facebook's profiles more often than fake users |
| Life Events | Information for the users to tell their stories | Real users share their life events more often than fake users. |
| Introduction "Bio." | Introduction information about Facebook's users | Real users are more often write something about themselves than fake users |
| No. of Mutual Friends | Number of the people who are Facebook friends with both users and the target profiles | Real users have more mutual friends with target profile than fake users, hence gives profile more incredibility |
| No. of Pages Liked | Number of pages liked | Real users usually liked more pages than fake users |
| No. of Groups Joined | Number of groups joined by the target profile. | Real users usually join groups more than fake users. |

Table 1: Attributes used by FBChecker.

# 6 The FBChecker implementation

## 6.1 Data set description

We note that there is no available standard data set with the required information. Thus, we choose to prepare our own one. The CRAWLER is employed on the author's profile for gathering real profiles and returns 151 profiles friends out of 151. However, 18 profiles were excluded as they were faked, underaged, or duplicated. This ends up with 133 real profiles. Regarding the fake profiles, a honeypot page is created and utilized as a source for collecting fake profiles. The inspecting of the fake profiles was finalized with selecting of 83 fake profiles as some of the collected profiles were not stable with their liking activity in which they drop their likes from our page after few days. As a result, 200 profiles were collected, 117 real and 83 fakes, as summarized in Figure 2.

## 6.2 Building the FBChecker system

After collecting the 200 profiles data set, we are ready to generate the classification and prediction rules. In this regard, RapidMiner 8.0.1 platform was utilized as a mining tool, which offer the use of various machine learning algorithms easily and provides a flexible environment designed specifically for data science and

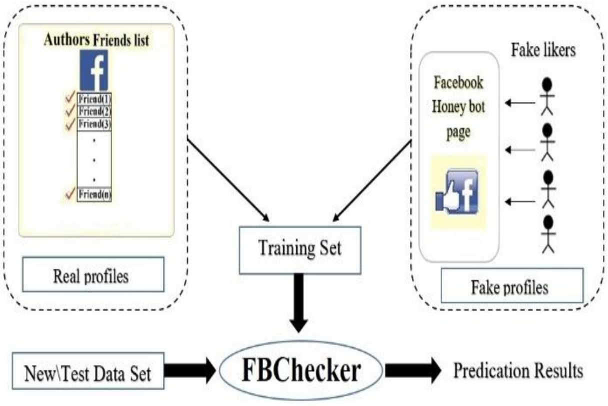


Figure 2: Collecting Training Data Set.

data mining purposes. For the training and testing processes, the (K fold) cross-validation with 10 folds was applied to evaluate the results accuracy as it is considered as one of the most effective methods for evaluating the predictive models with relatively small data set.

# 7 Evaluation process

To evaluate the FBChecker performance, the selected classifiers were tested with two experiments. In the first experiment, the k-NN schema was utilized to substitute the missing values, while in the second experiment, profiles with missing value were excluded. These experiments are discussed in detail in the following subsections. Finally, metrics for the validation process were calculated and proper justifications were provided.

## 7.1 Performance metrics

A group of common metrics are applied in the validation process, in this work the following metrics are used: Recall, Precision, Accuracy, F-measure, and specificity [25]. Next, we give a brief description for each one:

1) *Recall* true positive rate (total numbers of true positive divided by the total number of actual positives)
2) *Precision*: Measure the probability that the positive predications is correct (total numbers true positives divided of total number of predicted positives)
3) *Accuracy* Measure the performance of the classification model (total numbers of correct examples divided by total number of the example set)
4) *Specificity* true negative rates (total numbers of true negatives divided by the total number of actual negatives)
5) *F-measure* is an overall measure of a model's accuracy that combines precision and recall.

## 7.2 The experimental results

Four supervised algorithms were applied on the collected data set based on the following cases:

### 7.2.1 Estimating the missing values using k-NN schema

In this case, the k-NN schema is utilized for handling the missing values. After that the four supervised algorithms (e.g., Decision Tree, k-NN, SVM, and Naïve Bayes) are tested. In addition, the Cross-validation technique with 10 folds is used for performance assessments of these classifiers. The results showed that while the Decision Tree and Naïve Bayes exhibit close results with accuracy of 0.9650 and 0.9700 respectively, the SVM classification registered higher performance accuracy with 0.9850. On the other hand, k-NN algorithm with k=1 showed accuracy of 0.8400. Table 2 shows the complete results along with the validation metrics of these algorithms. Also, Figure 3 shows the accuracy of the classifiers and Figure 4 shows the ROC graph comparison of these classifiers Moreover, Figures 5, 6, 7, and 8 illustrate the ROC of each and every classifier's performance in this experiment. ROC graph is graphical plot that diagnosis the classifier performance by analysis the its work based on the rates of true positive predication against the true negatives predication [26].

| Validation metrics | Decision Tree | k-NN | SVM | Naïve Bayes |
|---|---|---|---|---|
| Accuracy | 0.9650 | 0.8400 | 0.9850 | 0.9750 |
| Recall | 0.9658 | 0.8291 | 1.0000 | 1.0000 |
| Precision | 0.9741 | 0.8899 | 0.9750 | 0.9590 |
| F-measure | 0.9700 | 0.8584 | 0.9873 | 0.9791 |
| Specificity | 0.9639 | 0.8554 | 0.9639 | 0.9398 |

Table 2. Supervised performance with k-NN estimator.

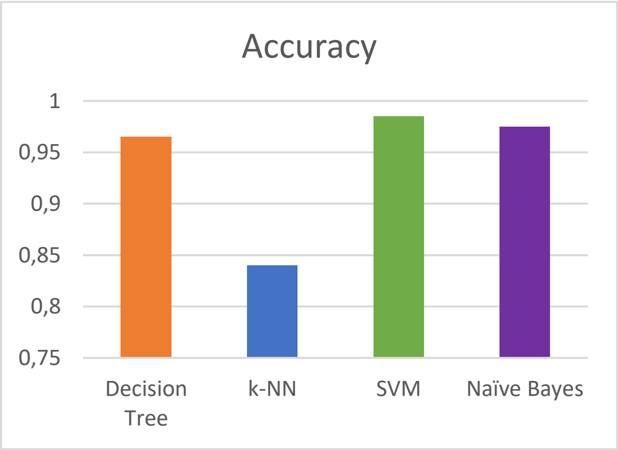


Figure 3: Supervised accuracy with k-NN estimator.

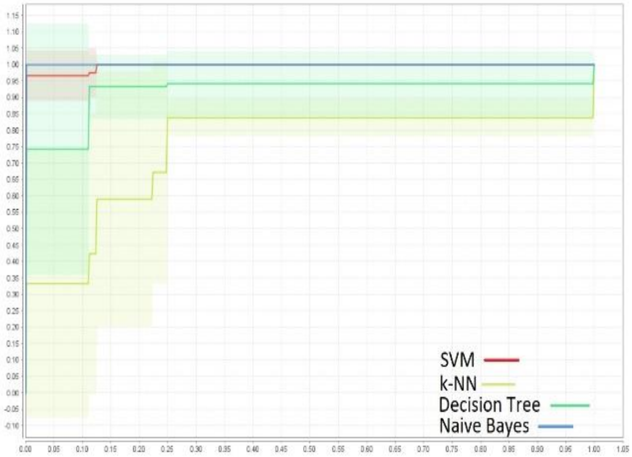


Figure 4: ROC graph comparison of the all classifiers with k-NN estimator.

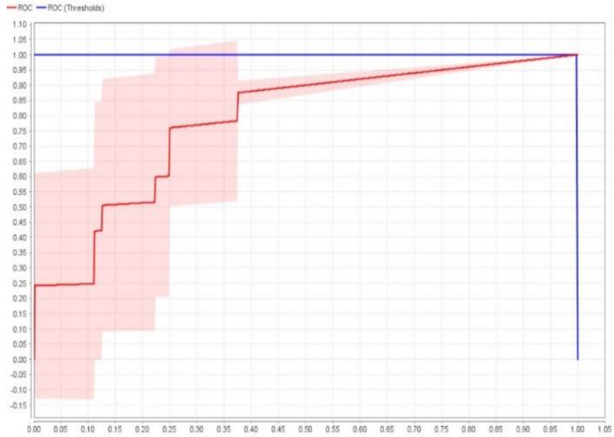


Figure 5: ROC graph of the k-NN performance with k-NN estimator.

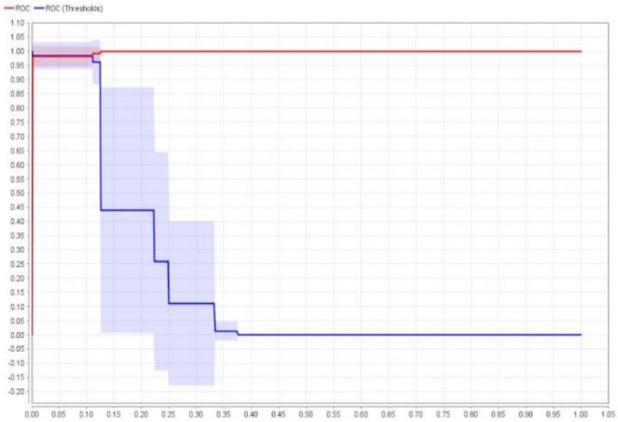


Figure 6: ROC graph of the Naïve Bayes performance with k-NN estimator.

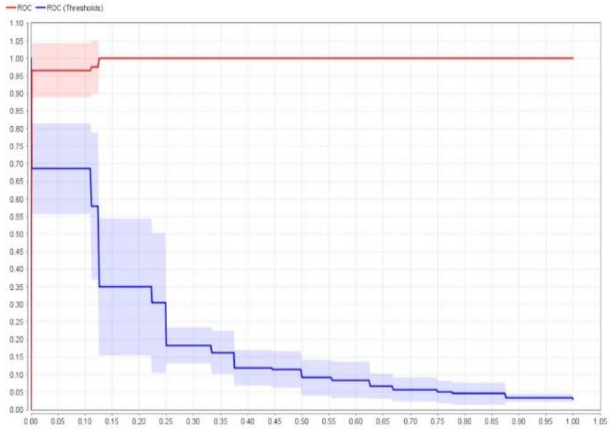


Figure 7: ROC graph of the SVM performance with k-NN estimator.

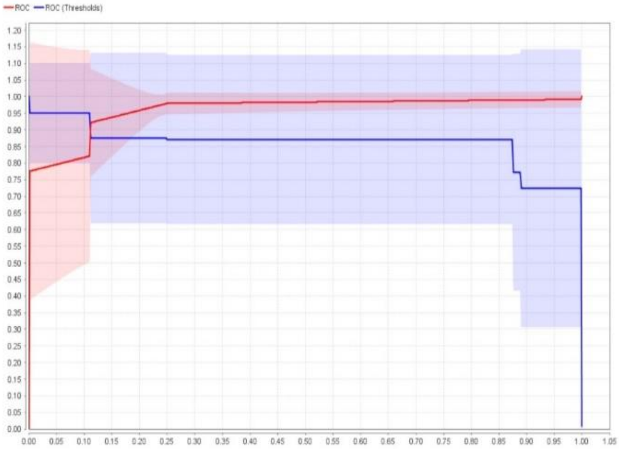


Figure 8: ROC graph of the Decision Tree performance with k-NN estimator.

Although all the classifiers achieve high accuracy rate, however, the SVM outperforms other classifiers as it employs “Nominal to Numerical” operator to map the different types of data to numerical type, so SVM can calculate the distance of these attributes to the hyperplane that separates the concept classes. Specifically, SVM proved its efficiency for application of two concepts classes due to find the optimal decision boundary (Hyperplane) that separate the two class in which are (Fake and real) and calculate the distance of each case (profile) to its nearest class label for the classification process.

### 7.2.2 Excluding the missing values using filtering operator

In the second case, the profiles with missing attributes are excluded by employing a special filtering operator provided by the RapidMiner, which filter the profiles based on specific conditions to keep/remove the profiles that met these conditions. Practically, the conditions of the Filter are set to remove any profile with missing values in anyone of their attributes. By applying this operator, a total of 33 profiles were removed from the collected data set leaving 167 profiles to be considered in this experiment. The main purpose behind this experiment is to eliminate any factor that could affect the model's classification process or the accuracy because we estimated the missing values in the first experiment.

| Validation metrics | Decision Tree | k-NN | SVM | Naïve Bayes |
|---|---|---|---|---|
| Accuracy | 0.9461 | 0.8443 | 0.9880 | 0.9641 |
| Recall | 0.9406 | 0.8317 | 1.0000 | 1.0000 |
| Precision | 0.9694 | 0.9032 | 0.9806 | 0.9439 |
| F-measure | 0.9548 | 0.8660 | 0.9902 | 0.9712 |
| Specificity | 0.9545 | 0.8636 | 0.9697 | 0.9091 |

Table 3. Supervised performance with filtering operator.

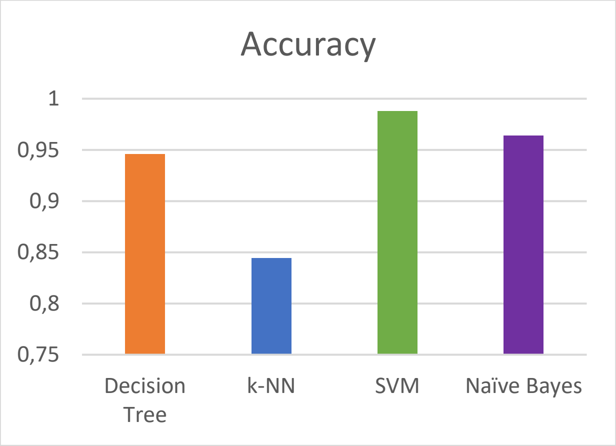


Figure 9. Supervised accuracy with filtering operator.

After that, the supervised algorithms are applied on the data, results showed the following accuracy rate (0.9461, 0.8443, 0.9880, and 0.9641) for Decision Tree, k-NN, SVM, and Naïve Bayes, respectively. Again, SVM exhibits the highest detection performance with accuracy of 0.9880, while K-NN the lowest with accuracy of 0.8433. Other performance indicators for these supervised algorithms are showed in table 3. And following the same vein of the previous experiment Figure 9 illustrates the accuracy results the employed classifiers and Figure 10 the ROC graph comparison of all classifiers employed in this experiment, Figures 11, 12, 13, and 14 shows the ROC graph for each one.

However, although our results are stable and good, one limitation that affects the validity of our study is that the used dataset is relatively small. Therefore, further validations over large datasets is required.

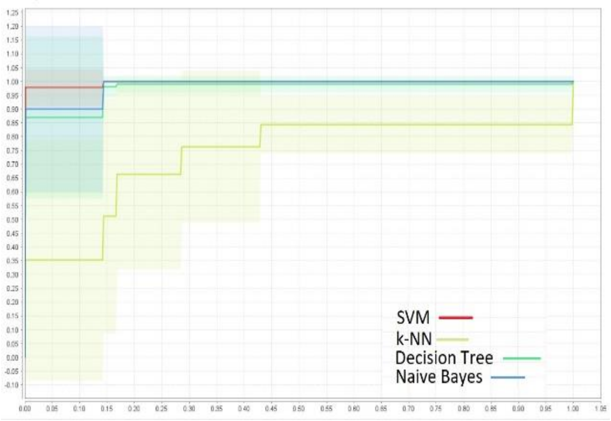


Figure 10. ROC curve of the supervised algorithms with filtering operator.

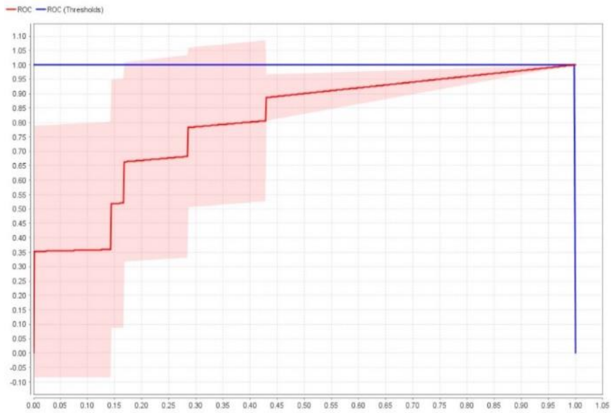


Figure 11: ROC graph of the k-NN performance with filtering operator.

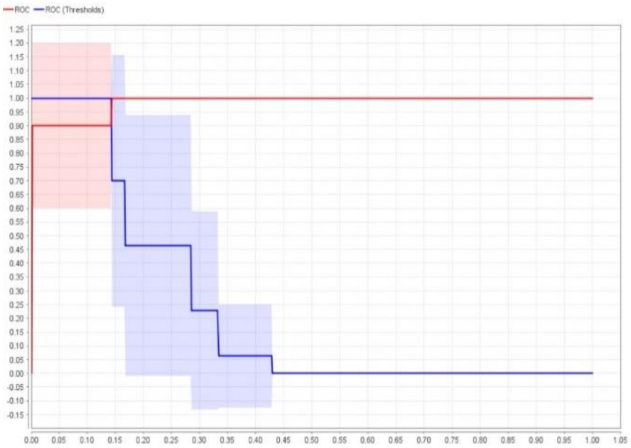


Figure 12: ROC graph of the Naïve Bayes performance with filtering operator.

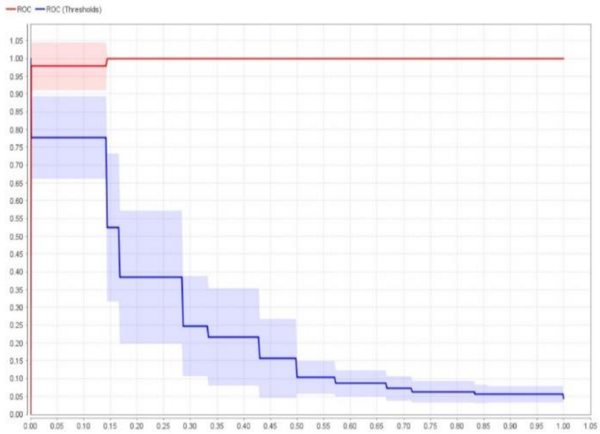


Figure 13: ROC graph of the SVM performance with filtering operator.

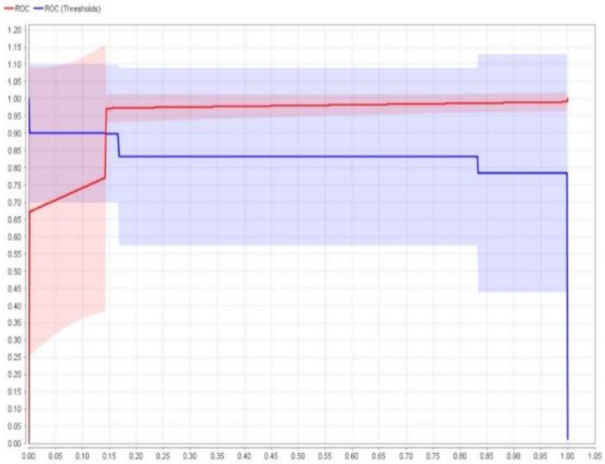


Figure 14: ROC graph of the Decision Tree performance with filtering operator.

## 8 Conclusion and future work

In this work, a smart system FBChecker is presented that have been designed specifically for detecting Facebook fake profiles. FBChecker consists of several components that collecting, preparing, validating, and mining the users' profiles using four supervised data mining techniques. These supervised techniques were implemented using the open source RapidMiner data science platform. The proposed system shows high efficiency performance for detecting fake profiles with accuracy rates reached %98, which represents a successful and promising result.

As a future work, we are aiming to use a large data set size and include more attributes that may employed in the detection model as discriminative features, and also apply more data mining techniques (unsupervised/Clustering algorithms) then evaluate which technique among them perform best.

**ACKNOWLEDGMENT**

The authors are grateful to the Applied Science Private University, Amman-Jordan, for the full financial support granted to cover the publication fee of this research article.

## References

[1] Romero, Daniel M., Wojciech Galuba, Sitaram Asur, and Bernardo A. Huberman. "*Influence and passivity in social media*." In Proceedings of the 20th international conference companion on World Wide Web, 2011, ACM, pp. 113-114. https://doi.org/10.1145/1963192.1963250

[2] Ngai, E. W., Tao, S. S., & Moon, K. K. (2015). Social media research: Theories, constructs, and conceptual frameworks. International Journal of Information Management, 35(1), 33-44. https://doi.org/10.1016/j.ijinfomgt.2014.09.004

[3] Kaplan, Andreas M., and Michael Haenlein. "*Users of the world, unite! The challenges and opportunities of Social Media*." Business Horizons 53, no. 1: 59-68,.2010. https://doi.org/10.1016/j.bushor.2009.09.003

[4] Agichtein, Eugene, Carlos Castillo, Debora Donato, Aristides Gionis, and Gilad Mishne. *"Finding high-quality content in social media."* In Proceedings of the 2008 international conference on web search and data mining, 2008, ACM, pp. 183-194. https://doi.org/10.1145/1341531.1341557

[5] O'Keeffe, Gwenn Schurgin, and Kathleen Clarke-Pearson. *"The impact of social media on children, adolescents, and families."* Pediatrics 127, no. 4: 800-804, 2011. https://doi.org/10.1542/peds.2011-0054

[6] Hajirnis, Aditi. "*Social media networking: Parent guidance required*." The Brown University Child and Adolescent Behavior Letter 31, no. 12: 1-7, 2015. https://doi.org/10.1002/cbl.30086

[7] Tang, Qian, Bin Gu, and Andrew B. Whinston. "*Content contribution for revenue sharing and reputation in social media: A dynamic structural model.*" Journal of Management Information Systems 29, no. 2: 41-76, 2012. https://doi.org/10.2753/MIS0742-1222290203

[8] Facebook Newsroom. https://newsroom.fb.com/ company-info/ (24th July 2017)

[9] Aswani, R., Kar, A. K., & Ilavarasan, P. V. (2018). Detection of spammers in twitter marketing: a hybrid approach using social media analytics and bio inspired computing. *Information Systems Frontiers*, 1-16. https://doi.org/10.1007/s10796-017-9805-8

[10] Singh, N., Sharma, T., Thakral, A., & Choudhury, T. (2018, June). Detection of Fake Profile in Online Social Networks Using Machine Learning. In *2018 International Conference on Advances in Computing and Communication Engineering (ICACCE)* (pp. 231-234), IEEE. https://doi.org/10.1109/ICACCE.2018.8441713

[11] Facebook Terms of Service. https://www.facebook.com/legal/terms (18th august 2017).

[12] Moustafa, N., & Slay, J. (2016). The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the

comparison with the KDD99 data set. *Information Security Journal: A Global Perspective*, *25*(1-3),18-31. https://doi.org/10.1080/19393555.2015.1125974

[13] Fürnkranz, Johannes. "*Separate-and-conquer rule learning."* Artificial Intelligence Review 13, no. 1: 3-54,1999. https://doi.org/10.1023/A:1006524209794

[14] Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials, 18(2), 1153-1176. https://doi.org/10.1109/COMST.2015.2494502

[15] Büchner, Alex G., and Maurice D. Mulvenna. "*Discovering internet marketing intelligence through online analytical web usage mining*." ACM Sigmod Record 27, no. 4: 54-61, 1998. https://doi.org/10.1145/306101.306124

[16] Xiao, Cao, David Mandell Freeman, and Theodore Hwa. "Detecting clusters of fake accounts in online social networks." In *Proceedings of the 8th ACM Workshop on Artificial Intelligence and Security*, pp. 91-101.ACM,2015. https://doi.org/10.1145/2808769.2808779

[17] Gupta, Aditi, and Rishabh Kaushal. "*Towards detecting fake user accounts in Facebook*." In Asia Security and Privacy (ISEASP), 2017 ISEA, pp. 1-6. IEEE,2017. https://doi.org/10.1109/ISEASP.2017.7976996

[18] Ahmed, Faraz, and Muhammad Abulaish. "*A generic statistical approach for spam detection in Online Social Networks*." Computer Communications 36, no.10:1120-1129,2013. https://doi.org/10.1016/j.comcom.2013.04.004

[19] Fire, Michael, Dima Kagan, Aviad Elyashar, and Yuval Elovici. "Friend or foe? Fake profile identification in online social networks." *Social Network Analysis and Mining* 4, no. 1 (2014): 194. https://doi.org/10.1007/s13278-014-0194-4

[20] Bhat, Sajid Yousuf, and Muhammad Abulaish. "Community-based features for identifying spammers in online social networks." In *Advances in Social Networks Analysis and Mining (ASONAM), 2013 IEEE/ACM International Conference on*, pp. 100-107. IEEE, 2013. https://doi.org/10.1145/2492517.2492567

[21] Wang, Alex Hai. "Detecting Spam Bots in Online Social Networking Sites: A Machine Learning Approach." DBSec 10: 335-342, 2010. https://doi.org/10.1007/978-3-642-13739-6_25

[22] Han, Jiawei, Jian Pei, and Micheline Kamber. Data mining: concepts and techniques. Elsevier, 2011.

[23] Cook, Diane J., and Lawrence B. Holder, eds. Mining graph data. John Wiley & Sons, 2006. https://doi.org/10.1002/0470073047

[24] Kotsiantis, Sotiris B., I. Zaharakis, and P. Pintelas. "Supervised machine learning: A review of classification techniques." Emerging artificial intelligence applications in computer engineering 160: 3-24, 2007.

[25] Powers, David Martin. "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation." 2011.

[26] Hanley, James A., and Barbara J. McNeil. "The meaning and use of the area under a receiver operating characteristic (ROC) curve." Radiology 143, no. 1: 29-36: 198.

# An Energy Efficient Architecture of IoT Based on Service Oriented Architecture (SOA)

Sudhanshu Maurya
Department of Computer Science & IT, Jharkhand Rai University, Ranchi, Jharkhand, India
E-mail: sam.sudhanshu@gmail.com, [1]WebAddress: www.jru.edu.in

Dr. Kuntal Mukherjee
Department of Computer Science & Engineering, BIT Mesra, Ranchi, Jharkhand, India
E-mail: kmukherjee@bitmesra.ac.in, [2]WebAddress: www.bitmesra.ac.in

**Keywords:** cloud, IoT, energy Efficient, architecture, SOA

**Received:** August 25, 2017

*Abstract: In the present scenario, IoT is a platform where everyday gadgets are becoming smarter, each day processing will become shrewd, and every day verbal exchange is becoming informative. Even as the IoT is still searching for its basic form, its result has already started in making exquisite strides as a universal solution media for the associated situation. The study which focused on architecture always paves the conformation of associated discipline. The shortage of overall architectural abilities is at this time resisting the researchers to get a way of the scope of procedures based on energy efficient IoT. In this endeavor Service Oriented Architecture (SOA) based IoT architecture and algorithm have been proposed to address energy efficiency in IoT environment. Experimental result has been shown to lay bare the effectiveness of the proposed approach.*

*Povzetek: V tem rokopisu je bila predlagana energetsko učinkovita arhitektura interneta stvari na osnovi storitveno usmerjene arhitekture (SOA) z algoritmom.*

## 1 Introduction

Internet of Things (IoT) is currently turning into a novel upheaval after the internet in information and communication industry. It is the mixture of innovation of internet and hardware equipment, which is able to receive and send the signals. The fundamental thought of this idea is an unavoidable availability of things near us. A spread of factors and articles like Sensors [1], Smartphones, Radio-Frequency Identification (RFID) tags [2], actuators and so on through one of a kind addressing schemes, are capable of engaging with one another and coordinate with their friends to reach commonplace desires [3]. By end of the year 2020, total number of different inter-connected devices will reach on IoT network into several billion. Devices may be actuators, mobile phones, laptops, sensors, appliances used in the office or home and other nodes that can be actuated or connected [4]. There is numerous bridge credibility among IoT and distinctive technologies which incorporate senor, communication, SCM, storage, cloud computing, visualization and mining of data, and its miles generally used in warehousing and logistics, medicine and healthcare, acumen and amusing ambiance [5]. Owing to the fast improvement of IoT, numerous associations and manufacturers resolve to search for out procedures that make the IoT more conventional. For an ample duration of time, IoT will remain at the theoretical level. As it may, get to be distinctly indicated in the IoT planned research roadmap of Europe, devices like cellular gadgets and Machine to Machine (M2M) or Things to Things (T2T), can be primary leaders for the advance development of IoT [6]. The cachet and gesture of an article can be detected with the help of sensor (like RFID) in special condition [7]. For example, take a container, with the help of RFID gadget, the sensor can identify it as a teacup, cup or glass [8]. We likewise know that it had been lifted gradually with the help of a sensor on the container we are able to receive the message like "fluid of glass is excessively warm, making it difficult to drink". The pertinence of syntactic research with regards to the IoT was addressed by authors in [6]. Where substances ontologies may want to offer advantages in institutionalization (managing associations among heterogeneous gadgets and information suppliers), thing disclosure and pursuit (where metadata and semantic comments are profoundly important), or interoperability (particularly tended to by advances in IoT-particular semantic innovations) [9] and additionally "semantically determined code era of gadget interfaces". The purpose of this manuscript is to deliberate the basic concept of IoT and propose a cloud-centric view of IoT framework and new energy efficient architecture of IoT based on SOA (Service Oriented Architecture).

SOA is basically a pool of services, which are able to communicate mutually. Here communication includes the passing of data or managing the activities. Sometimes, connecting between services is desirable.

Section 1 discusses the introductory part of IoT, then in section 2 literature review has been done which includes the architecture of IoT and IP based connectivity architecture of IoT. In Section 3, we have proposed an energy efficient IoT architecture based on SOA with the algorithm. In Section 4, the experimental setup and result have been shown and section 5 has given conclusion of the manuscript.

## 2 Literature review

IoT is a brand new archetype through which commonly recognizable things are virtually characterized, which will allow in establishing the communication with them. Originally IoT was invented in the year of 1999, its miles nevertheless in its beginning in development unexpectedly to 26 billion gadgets in 2020 [10]. The perception and prescient of the ERC placed the IoT in relation to an arrangement of innovation empowering agents where we can recognize knowledge, segment or morphological advancements [6]. Some portion of the upcoming technologies of the internet, IoT is thrice - A worldview which is Anytime, Anything and Anywhere, that is progressively turning into the best components of the the Internet [5]. It permitting "humans and things to be related preferably the use of any route, any network and any carrier" in a six - A connectivity paradigm as shown in Figure 1 [11].

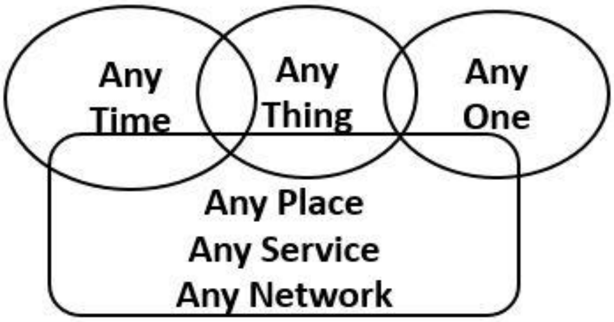


Figure 1: The Six - A Connectivity of IoT

The majority of the assets required in the IoT system are trading data utilizing an extensive variety of communicable hardware and in some of the scenarios, the sources' existence cycle is certain to the battery lifestyles [12]. Along with these, a component for designing the system hardware progressively could enhance the operational time of the gadgets and set up imperative connections by the approach of Green ICT. To overcome this issue an automated many-operator framework is discussed in the research work of [13], which influencing system demonstrations and enhancing the functioning time of gadgets. To ensure interoperability of entire systems, research work is done by authors in [14], [15], [16] concentrated using the idea of SOA directly on devices. In regards to sensor networks, energy proficient data spread had been researched. Authors in their research work, created different models for components of cloud computing environment like communication, computation and storage device [17]. An area assisted methodology is projected with the help of a protocol that supports the situation data to scale back repetitive transmissions of data [18]. Work was done in [19], [20], authors focus on demand allocation of resources which used prediction algorithm for forecasting of resources which are going to be used. In the research work of [21] authors proposed a system that manages complexity and changes consequently in the IoT environment. To accomplish this objective, authors have proposed a framework incorporates the accompanying developments which include dependable, strong, adaptable, expandable and object-oriented layouts to encourage interfaces and reuses of framework segments. Making a cosmology for IoT is a troublesome deed to accomplish as it included a wide range of pitches [22]. Moreover, ontologies which are associated with statistic figures, data received from sensors and different statistics generating gadgets should be taken into consideration so that it will have the ability if you want to work approximate data. For minimum consumption of energy authors [23] in their research work, using few servers in place of multiple servers and distribute the tasks to various virtual machines inside the server of the cloud environment. Authors [24] in their research work projected an algorithm that backs transmission of spatial information and process inquiries in light of their area. Another research work in [25] tended to energy potency in information scattering for IoT. For the minimization of energy consumption, authors [26] in their research work proposed a scheduling algorithm. To provide energy efficient job allocation and load balancing authors [27] proposed a scheduling algorithm named e-STAB for data centers.

### 2.1 Architecture of IoT

IoT architecture comprises of different layers, shown in Figure 2.

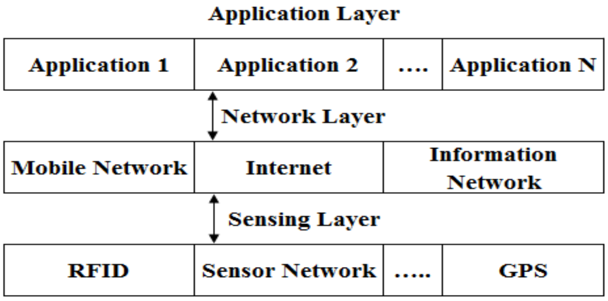


Figure 2: Architecture of IoT [28].

The lowermost layer that is sensing layer, comprises of different sensing devices. The information that is processed by these sensing devices is linked with different applications present at the topmost layer – application layer through the cloud computing [28]. Thus, a cloud-centric hallucination is the utmost need in order to employ IoT extensively.

Obviously, the whole IoT technology pivots on the lowermost layer that is sensing layer. The sensing layer comprises RFID (Radio - Frequency Identification), sensor network and Global Positioning System (GPS) [28]. Thus in the epoch of IoT, all objects and devices are interconnected and interact with each other [29]. Cloud computing platform is built over the different components of the middle layer namely mobile networks, Internet backbone, and the various information networks. A cloud-centric view of IoT framework is depicted in Figure 3.

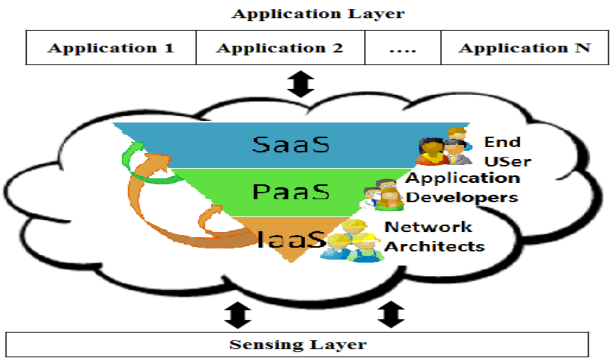


Figure 3: Cloud Centric view of IoT Framework.

Further, for an effective communication among the objects and devices in IoT, unique IP address is the utmost need for all objects. This unique IP address is provided by IPv6 protocol used in IoT. In order to distinguish all computers and pervasive devices, the IPv6 protocol provides 2128 IP addresses [28]. Further, three types of communications exist in IoT, which are H2H, H2T, and T2T, where H stands for Human, T stands for Things.

## 2.2 IP based connectivity architecture of IoT

It has been observed that the building block of IoT network comprises of smart object network and the Internet. The Internet is based on TCP/IP architecture and each machine on the Internet has a unique IP address [30]. Now, it's a big question, what would be the architecture for a smart object network? What would be the addressing scheme for it? Obviously, for the effective design of IoT, the first step will be assigning a unique address to all the things surrounding us including all computers and pervasive devices [28]. Due to interoperability across devices, scalability, and ease of deployment ability features of IP architecture, it is a natural choice to adopt in order to develop smart object network [31].

Nowadays, the wireless sensor network community has widely accepted IP architecture to establish wireless sensor network system – the largest subset of sensing network [32]. Furthermore, the interoperability with existing systems characteristic of IP architecture has lead to evolve IP-based sensor network system [33]. The overall IoT technology integrates a variety of legitimate heterogeneity in the used networking technologies with the smart object networking [5]. The smart object networking comprises of cellular, sensor network, RFID etc., which are different from each other [8]. To bind all such heterogeneities together, IP communications is best suited, shown in Figure 4. IP plays the central role for all types of communication irrespective of their nature. All the network applications depend on IP.

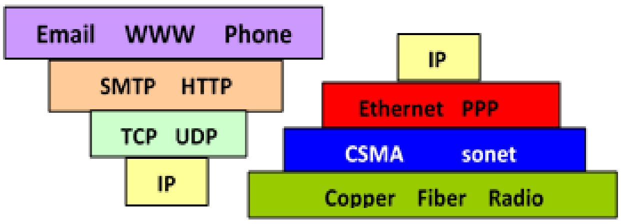


Figure 4: IP-based Communication Model.

Hence, we can say that IP-based connectivity model is best suited in designing IoT network [30]. After assigning unique IP addresses to all devices, the next question is to design an energy efficient network of these devices and hence IoT, which has been discussed in the subsequent sections.

# 3 Proposed energy efficient design of IoT algorithm based on SOA

It is obvious that when the sensor nodes are active then they consume more energy but when they are in the dormant state then they consume less energy. For this endeavor, the random geometric graph concept has been used [34]. A link, hence energy consumption, is established between two sensor nodes when they are very close to each other. For this, different sensor nodes are arranged according to the Poisson point process that is

$$\frac{e^{-\lambda}\lambda^{x}}{x!} \qquad \text{(i)}$$

Where $x$ is random variable and $\lambda$ is parameter of the distribution. Here as the value of $\lambda$ increases more and more nodes exists that are close to each other and start to communicate mutually, in other words, they become active. Hence, it is the utmost need to find out the critical value of $\lambda$, say, $\lambda_c$ for which only the required number of sensor nodes would be made active and rest would be made in the dormant state. Thus the proposed Hypothesis and the proposed algorithm for the same are discussed in next sub sections.

## 3.1 Hypothesis

The hypothesis of proposed algorithm is given below: Hypothesis 1: The sensor nodes are arranged as per the Poisson point process, that is, $\frac{e^{-\lambda}\lambda^{x}}{x!}$

Here, $x$ is the random variable, $\lambda$ is known as a parameter of the distribution, $e$ is exponential. Obviously, $\lambda = np$ where $n$ is number of sensor nodes and $p$ represents the probability of finding them in correct order.

## 3.2 SOA for IoT

The SOA-based architecture of IoT should be discussed in view of security, Quality of Service (QoS), energy management, infrastructure monitoring etc. Out of these parameters, here we only focus on the energy management and QoS aspects.

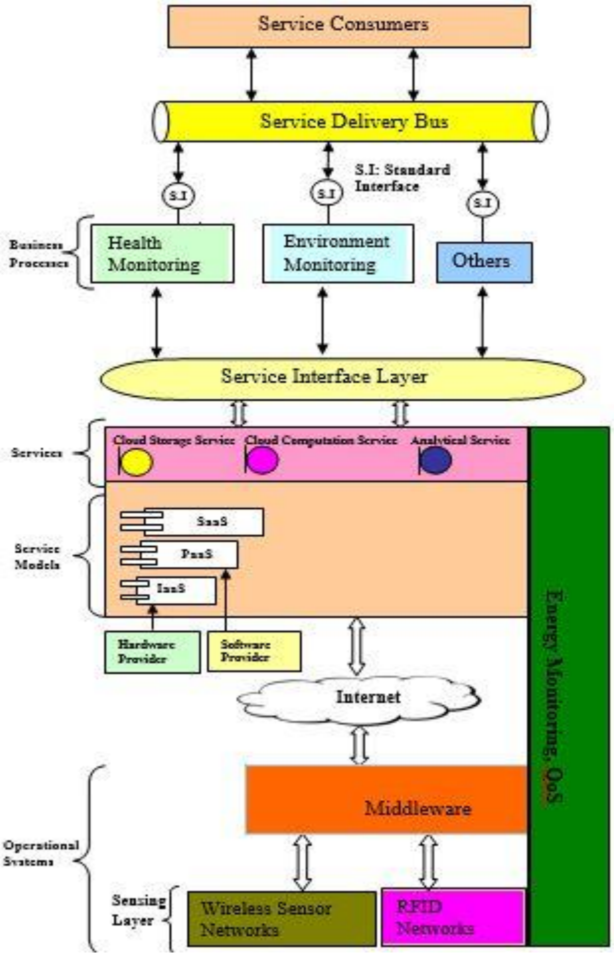


Figure 5: Proposed Energy efficient IoT Architecture based on SOA.

In the proposed architecture, the raw data captured by the sensing layer which comprises of RFID and WSN are integrated by the middleware shared by both the networks, so that data can be seamlessly accessed by the upper layer irrespective of their source. Now, the upper layer, which is cloud computing platform, can provide a different type of services like storage, computation, analytical services and so on, using the different service components offered by cloud computing environment. The services compose of different business processes like health monitoring, environment monitoring, surveillance, smart transportation etc., which are seamlessly accessed by consumers irrespective of time and place.

In figure 5, the raw data captured by the sensing layer are transferred to middleware and from there to cloud computing infrastructure. The different layers of cloud computing, process the raw data. Now this processed data is sent to upper layer to provide different services, which end user can access through Standard Interface. Now, the IaaS of cloud exploits the massive cluster of servers at the back end. The operation of these clusters of servers is associated with consumption of a lot of energy in view of their working along with their cooling process. Again, proper energy management is also an important concern to report at the sensing layer. In Figure 5, the energy monitoring controller manages the energy of hardware at service level as well as operational level. In this perspective, this manuscript has proposed two techniques to monitor the energy management, which is at IaaS level and at sensing layer level. But during the optimization process of energy, one should not compromise with the QoS. Energy optimization along with the maintenance of QoS is today's need. The rest part of this manuscript has presented the graph-theoretic model for energy saving scheme at sensing layer level along with at IaaS level.

## 3.3 Proposed algorithm

```
Step 1. [Initialization]
    Sensor nodes are distributed in a plane or in three-dimensional space as per Hypothesis 1.
    Initially, $\lambda = 0$ [Sensor nodes are in dormant state.]
Step 2. [Increasing the value of $\lambda$]
    When $\lambda = \lambda_1$, consider a specific value of x, say $x_1$. With $x_1$ as a center, construct a circle or sphere of radius 'r', where r is a fixed positive real number. Join all the sensor nodes within the circle (sphere) and made them active and rest are made in the dormant state.
Step 3. [Finding the critical value of '$\lambda$', say, $\lambda_c$]
    IF the sensed value by the sensor meets the need of our requirement and no further sensing of data is needed then stop.
    ELSE increase the value of $\lambda = \lambda_1 < \lambda_2 < \lambda_3 < \lambda_c$ so that we get a larger connected graph to meet our requirement.
Step 4. End.
```

## 3.4 Explanation, justification and threat of proposed algorithm

The proposed algorithm is based on the concept of partition of the plane or three-dimensional space into two parts namely active zone and passive zone. The sensor nodes at active zone are made active whereas the nodes at passive zone are made in the dormant state. The partition is made on the basis of random geometry graph concept. In this way, only a few parts of the plane or three-dimensional space are made active to provide necessary raw data to upper layer with spending less energy as compared to make the whole plane or three-dimensional space active.

The major threat associated with this approach is, the selected zone may have a sensor node that all are out of order and thus may cause the overall failure of capturing the raw data. But in practical sense, the probability of such scenario is very less.

# 4 Experimental setup & result

We evaluate the performance of our proposed approach through simulations. Here MATLAB, Release: R2013a software has been used for the purpose of simulation of sensor nodes in the plane. Let us now assume that the nodes are uniformly scattered in a k×k square space according to Poisson point process where the number of nodes is n. For a given value of r, we intend to find the critical value of $\lambda_c$ so that the random geometric graph G (n, r) has larger connected components. The experimental results are shown in Figure 6 for different values of $\lambda$ (Case (a) for $\lambda$ =200, Case (b) for $\lambda$ =500, Case (c) for $\lambda$ =1000, Case (d) for $\lambda$ =2000, Case (e) for $\lambda$ =2500 and Case (f) for $\lambda$ =5000); where red dots are Active nodes and blue dot are hibernated nodes.

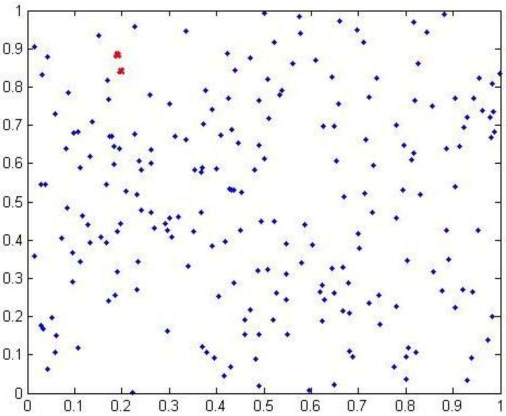


Figure 6: Case (a) for $\lambda$ =200.

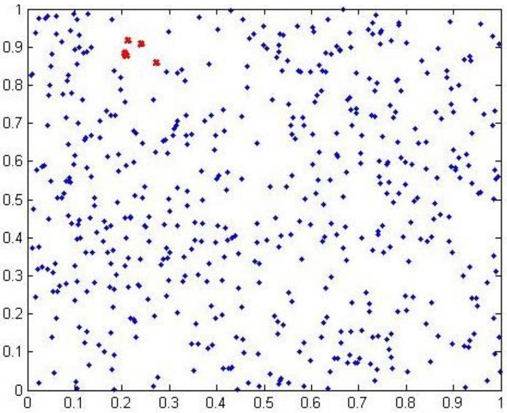


Figure 6: Case (b) for $\lambda$ =500.

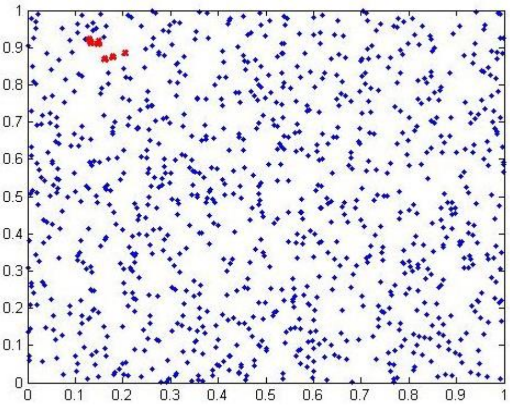


Figure 6: Case (c) for $\lambda$ =1000.

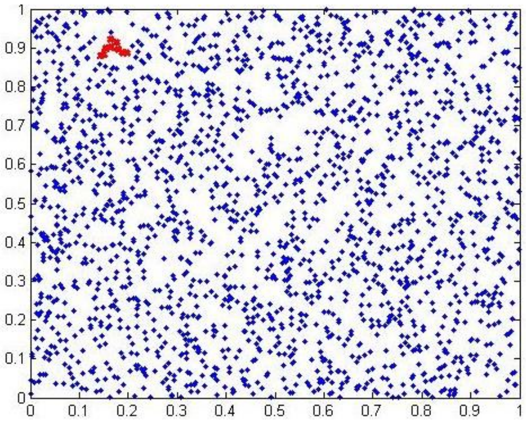


Figure 6: Case (d) for $\lambda$ =2000.

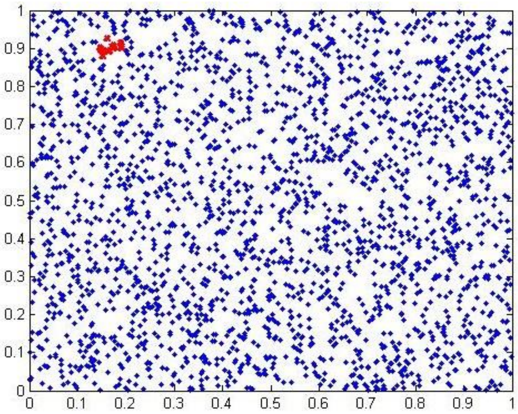


Figure 6: Case (e) for $\lambda$ =2500.

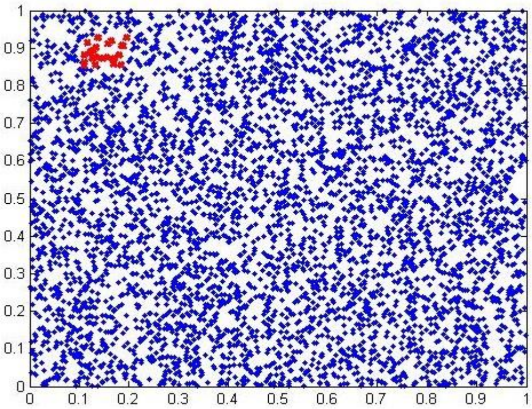


Figure 6: Case (f) for $\lambda$ =5000.

Points arranged as per Poisson point process for different values of $\lambda$ .

In Case (a) of Figure 6, few nodes are in active state (shown using red color), whereas the rest of the nodes are in the dormant state (shown using blue color) for $\lambda$ =100. The experimental result points out that only those nodes are made active which are needed for communication and the rest of the nodes are made in dormant state in order to save energy. Now if a few nodes are active then it is not convenient for establishing efficient communication between the nodes. That is why it is needed to increase the value of λ and to get a critical value of $\lambda_c$ for sufficient enough active nodes.

Based on the above experiments, the critical value of $\lambda_c$ using the interpolation method is shown in Figure 7. We observe that as λ increases, the nodes come very close to each other and hence more nodes are connected with one another. Thus there is a critical value of $\lambda_c$ for which G (n, r) has a larger connected component within, which is shown in Figure 7.

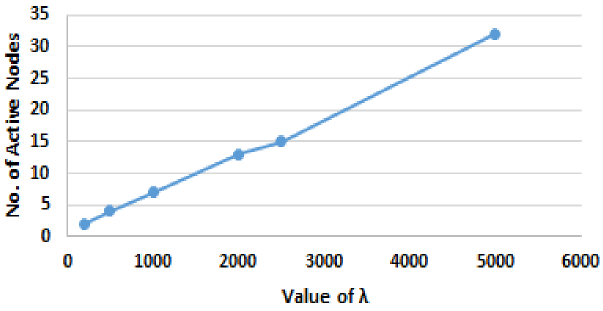


Figure 7: Threshold value of (r, n) for which G (n, r) has larger connected components

## 5 Conclusion

In the current scenario, IoT uses static graph which consumes more energy as it makes active every node all the time. Energy is only used in an active state while the dormant state consumes very less energy. Thus, keeping this in mind, we have proposed an energy efficient architecture based on SOA with their algorithm. In the proposed architecture, only the required number of nodes become active while another stay in dormant state, so that the energy consumption will reach at a minimal level. Energy being a precious resource, always demands optimization. Hence, the issue of drainage of energy in IoT can be well addressed by the implementation of above said energy-efficient architecture.

## References

[1] Bui N., A. Georgiadis, M. Miozzo, M. Rossi, X. Vilajosana (2011). SWAP Project: Beyond the state of the art on harvested energy-power wireless sensors platform design. *Proceedings of 2011 IEEE International Conference on Mobile Ad-Hoc and Sensor Systems*, Valencia, Spain. https://doi.org/10.1109/MASS.2011.97

[2] Sung J., Lopez T.S., and Kim D. (2007). The EPC Sensor Network for RFID and WSN Integration Infrastructure. *Proceedings of the 5th Annual IEEE International Conference on Pervasive Computing and Communications Workshops*, New York, USA. https://doi.org/10.1109/PERCOMW.2007.113

[3] Giusto D., Iera A., Morabito G., Atzori L. (2010). *The Internet of Things, Springer*. ISBN: 978-1-4419-1673-0. https://doi.org/10.1007/978-1-4419-1674-7

[4] Biswas, A. R., & Giaffreda, R. (2014). IoT and cloud convergence: Opportunities and challenges. *IEEE World Forum on Internet of Things (WF-IoT)*, pp. 375-376. https://doi.org/10.1109/WF-IoT.2014.6803194

[5] Atzori L., Lera A., Morabito G. (2010). The Internet of Things: A Survey. *Computer Networks*, Vol. 54, No. 15, pp. 2787-2805. https://doi.org/10.1016/j.comnet.2010.05.010

[6] Vermesan O., Friess P., Guillemin P., Gusmeroli S., Sundmaeker H., Bassi A., Jubert I. S., Mazura M., Harrison M., Eisenhauer M., and Doody P. (2011). Internet of Things - Global Technological and Societal Trends. Capter: Internet of Things Strategic Research Roadmap, *River Publishers Series in Communications*, pp. 9–52. PMid:20184663

[7] Kranz, M., Roalter, L. and Michahelles, F. (2010). Things that Twitter: social networks and the Internet of things. *What can the Internet of Things do for the Citizen (CIoT) Workshop at The Eighth International Conference on Pervasive Computing (Pervasive 2010).*

[8] Liu H., Bolic M., and Nayak A. (2008). Taxonomy and Challenges of the Integration of RFID and Wireless Sensor Networks. *Journal of IEEE Network*, Vol. 22, No. 6, pp. 26-35. https://doi.org/10.1109/MNET.2008.4694171

[9] Vermesan, O. and Friess, P. (2014). Internet of Things – From Research and Innovation to market Deployment. *River Publishers Series in Communications*, Denmark, pp. 07-112

[10] Gartner (2014). Internet of things will transform the data center. *Research Report by IoT - Internet of Things*. Available online at: https://iot.do/gartner-says-internet-things-will-transform-data-center-2014-03

[11] Miorandi D., Sicari S., Pellegrini F. D., and Chlamtac I., (2012). Internet of things: Vision, applications and research challenges. *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497–1516. https://doi.org/10.1016/j.adhoc.2012.02.016

[12] Peng Y., He U. and Choi J. (2012). Energy Efficient Hop Length Optimization for Laterally Connected Wireless Sensor Networks. *Proceedings of IoT workshop*, Berlin Germany. https://doi.org/10.1007/978-3-642-32427-7_8

[13] Pujolle, G. (2006). An autonomic-oriented architecture for the internet of things. *IEEE International Symposium on Modern Computing, JVA'06*, pp. 163–168. https://doi.org/10.1109/JVA.2006.6

[14] Jammes, F. and Smit, H. (2005). Service-Oriented Paradigms in Industrial Automation. *IEEE Trans. Industrial Informatics*, vol. 1, no. 1, pp. 62-70. https://doi.org/10.1109/TII.2005.844419

[15] De Souza L.M.S., Spiess P., Guinard D., Kohler M., Karnouskos S., and Savio D. (2008). SOCRADES: A Web Service Based Shop Floor Integration Infrastructure. *Proceedings of Internet of Things Conference (IoT '08),* Switzerland, pp. 50-67. https://doi.org/10.1007/978-3-540-78731-0_4

[16] Priyantha N. B., Kansal A., Goraczko M., and Zhao F. (2008). Tiny Web Services: Design and Implementation of Interoperable and Evolvable Sensor Networks. *Proceedings of 6th ACM Conf. Embedded Network Sensor Systems*, pp. 253-266. https://doi.org/10.1145/1460412.1460438

[17] Gurun S., Krintz C. and Wolski R. (2004). NWS Lite: A Light- Weight Prediction Utility for Mobile Devices. *Proceedings of the 2nd international conference on Mobile systems, applications, and services, Boston, USA*, pp. 2-11. https://doi.org/10.1145/990064.990068

[18] Sabbineni H., and Chakrabarty K. (2005). Location-aided flooding: An energy-efficient data dissemination protocol for wireless sensor networks. *IEEE Transactions on Computers,* Vol. 54, pp. 36–46. https://doi.org/10.1109/TC.2005.8

[19] Adachi M., Hiraoka T., Komatsu N. (2008). A study on a resource allocation algorithm for on-demand data center services. *Proceeding of the 10th international conference on advanced communication technology – ICACT.* pp. 295-300. https://doi.org/10.1109/ICACT.2008.4493765

[20] Lubin B., Kephart J. O., Das R., Parkes D. C. (2009). Expressive power-based resource allocation for data centers. *Proceedings of 21st international joint conference on artificial intelligence, IJCAI – ACM Digital Library*. pp. 1451-1456.

[21] Wang, C., Bi, Z., & Da Xu, L. (2014). IoT and Cloud Computing in Automation of Assembly Modeling Systems. *IEEE Transactions on Industrial Informatics,* Vol. 10, Issue 2, pp. 1426-1434. https://doi.org/10.1109/TII.2014.2300346

[22] Duroc Y. and Andia, V. G. (2014). Towards Autonomous Wireless Sensors: RFID and Energy Harvesting Solutions. *Internet of Things - Challenges and Opportunities,* Springer, Switzerland. https://doi.org/10.1007/978-3-319-04223-7_1

[23] Yamini B. and Selvi D. V. (2010). Cloud virtualization: A potential way to reduce global warming. *IEEE International Conference on Recent Advances in Space Technology Services and Climate Change (RSTS & CC)* Chennai, India, pp. 55-57. https://doi.org/10.1109/RSTSCC.2010.5712798

[24] Park K. and Valduriez P. (2011). Energy Efficient Data Access in Mobile P2P Networks, *IEEE Transactions on Knowledge and Data Engineering.* Vol. 23, No. 11, pp. 1619–1634. https://doi.org/10.1109/TKDE.2010.194

[25] Liu D., Zhang J., and Yu C. (2012). Energy efficient massive data dissemination protocols for the International Network of Things. *Frontiers in Computer Education*, pp. 627–634. https://doi.org/10.1007/978-3-642-27552-4_84

[26] Yuan W, Nahrstedt K. (2003). Energy-efficient soft reat-time CPU scheduling for mobile multimedia systems. *ACM SIGOPS Operating Systems Review*, Vol. 37, Issue 5. https://doi.org/10.1145/1165389.945460

[27] Dzmitry K., Sisay T. A., Fabrizio G. (2013). e-STAB: Energy-Efficient Scheduling for Cloud Computing Applications with Traffic Load Balancing. *IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing,* Beijing, China, pp. 07-13. https://doi.org/10.1109/GreenCom-iThings-CPSCom.2013.28

[28] Hwang K., Fox G. C. and Dongarra J. J. (2012). Distributed and Cloud Computing From Parallel Processing to the Internet of Things. *Morgan Kaufmann (An imprint of Elsevier),* USA.

[29] Zhang L., Wang X., and Dou W. (2004). A K-Connected Energy-Saving Topology Control Algorithm for Wireless Sensor Networks. *Proceedings of the 6th International Workshop on Distributed Computing,* Kolkata, India. https://doi.org/10.1007/978-3-540-30536-1_57

[30] Jin J., Gubbi J., Lou T., and Palaniswami M. (2012). Network Architecture and QoS Issues in the Internet of Things for a smart city. *Proceedings of International Symposium on Communications and Information Technologies,* Gold Coast, QLD. https://doi.org/10.1109/ISCIT.2012.6381043

[31] Vasseur J. P. and Dunkets A. (2010). Interconnecting Smart Objects with IP: The Next Internet. *Morgan Kaufmann (An imprint of Elsevier), USA.*

[32] Estrin D., Govindan R., Heidemann J., and Kumar S., (1999). Next century challenges: scalable coordination in sensor networks. *Proceedings of the 5th annual ACM/IEEE International Conference on Mobile Computing and Networking, Seattle,* Washington, USA. https://doi.org/10.1145/313451.313556

[33] Hui J. W. and Culler D.E. (2008). IP is dead, long live IP for wireless sensor networks. *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems,* New York, USA. https://doi.org/10.1145/1460412.1460415

[34] Penrose M., (2003). Random Geometric Graphs. *Oxford University Press,* USA. https://doi.org/10.1093/acprof:oso/9780198506263.001.0001

# The Student-Self Oriented Learning Model as an Effective Paradigm for Education in Knowledge Society

Vladimir A. Fomichov
Department of Intelligent Monitoring Systems, Institute No. 3 Control Systems
Informatics and Electric Power Industry, Moscow Aviation Institute (National Research University)
Orshanskaya street 3, 121552 Moscow, Russia
E-mail: vfomichov@gmail.com, https://www.researchgate.net/profile/Vladimir_Fomichov

Olga S. Fomichova
Division "Dialogue of Sciences", State Budget Professional Educational Institution "Sparrow Hills"
Universitetsky prospect 5, 119296 Moscow, Russia

**Keywords:** emotional intelligence, STEM education, STEAM education, creativity development, development of figurative reasoning, art cognitonics, thought producing self, early socialization of children, theory of dynamic conceptual mappings, system of emotional-imaginative teaching, cognitive engagement, serendipity, anthropocentric approach to education

**Received:** June 21, 2018

*Proceeding from broadly accepted role of emotional intelligence (EI) in professional and personal life, the paper suggests a new learning model (LM) called Student-Self Oriented LM (SSOL-model). It is defined as the model being beneficial for self-cognition and self-construction through the prism of the acquired knowledge and life experience. Two successful implementations of the SSOL-model are shortly described: the system of emotional-imaginative teaching (the EIT-system) developed by the authors in the 1990s and expanded in the 2000s and art cognitonics. The EIT-system is underpinned by the authors' Theory of Dynamic Conceptual Mappings (the DCM-theory). The EIT-system includes an original method of developing figurative thinking and creativity at the lessons of second language (English for Russian children), literature and poetry in English and Russian, symbolic language of painting, and communication culture. It is stated that this method may be used as an effective starting framework of STEAM education. It is mentioned that the DCM-theory and the EIT-system became the starting point for developing the foundations of a new scientific discipline called cognitonics. The basic principles of art cognitonics – a well developed branch of cognitonics and the concept of cognitive engagement are described. The significance of art cognitonics for helping the learners to answer the encountered moral questions is explained. The broad possibilities of designing intelligent tutoring systems (ITSs) of new generation are indicated. These are culture and moral experience oriented systems aimed at developing figurative thinking, creativity, early socialization and helping the students to find in the outstanding art works the answers to emerging moral questions. An original, detailed script of an ITS contributing to early socialization of the learners is described, its literary source is the fairy-tale "Sleeping Beauty". Since the sphere of using the SSOL-model is not restricted, it is suggested to interpret this model as an effective paradigm for education in knowledge society and a strong support to anthropocentric approach to education in the digital age.*

*Povzetek: Predstavljeno je samo-modeliranje študentov kot metoda učenja v družbi znanja.*

## 1 Introduction

The progress of science and technology in the end of the 20th – the beginning of the 21st century has posed new demands to education. Likely, the most significant demand is the formation of the preconditions of mastering several professions during the life: for the most part of people, it is impossible in knowledge society (or smart society) to have only one profession during the life. The second most significant demand seems to be the ability to quickly generate new knowledge, the skill of integrating knowledge pieces obtained from numerous dispersed information sources, the ability of creative thinking while processing new knowledge and its connections with available knowledge. A popular answer to the challenge of time in the sphere of education is *smart learning.* Many scholars interpret smart learning as learning with broad use of mobile devices: tablet computers, androids, etc. This definition reflects the focus on technical means of learning. It explicates the essence of technological approach to education in the digital age.

We believe that the following metaphor shows the roots of this approach: it is a consequence of the admiration experienced by the six-year old child just discovered astonishing possibilities of computer.

There are reasons to believe that the dominant part of the scholars don't notice the fundamental problem of (a) perfecting, improving the principles and methods of teaching and learning, (b) combining these principles and methods with the principles and methods of developing personality of the learners, their system of values. The analysis of scientific literature shows that a high proportion of elementary, middle, and high school students encounter considerable difficulties as concerns grasping the main ideas of the pieces of theory to be studied (see Section 2).

Parallelly with the term "smart learning", the notion "*emotional intelligence*" (EI) belongs to the set of concepts most often used in scientific publications in the field of education. According to [1], EI is the other kind of smart. The studies carried out during two decades after the birth of this notion in 1995 [25] have shown that EI is the crucial factor distinguishing star performers of various professional roles among all performers.

EI determines the manner of a person to manage behavior, deal with social complexities, and make decisions leading to positive results. EI is the unity of four core skills forming two primary competences: *personal competence* (*PC*) and *social competence (SC)*. According to [1], PC is composed by two skills: *self-awareness* and *self-management*. The first skill is the ability of a person to accurately perceive his/her emotions and stay away of them as they happen. self-management is the ability of a person to use awareness of his/her emotions for staying flexible and for positively directing his/her behaviour. SC is defined in [1] as the ability of a person to understand other people's moods, behaviour, and motives for improving the quality of his /her relationships. SC is composed by *social awareness* and *relationship management*.

During last two decades, the psychologists have discovered a huge role of well-developed EI in taking successful business decisions. That is why now the companies throughout the world pay a very high attention to the state of EI while hiring, promoting, and increasing qualification of their employees [26].

The next term of high significance for education is *creativity*. The analysis of the literature shows that the scholars from many countries consider the problem of supporting and developing creativity of the learners as a highly acute problem. Until the 2000s, the majority of publications on creativity studied the peculiarities of intelligent activity of outstanding scientists, painters, writers, poets, etc. (see., e.g., [24]). This kind of creativity is often called in modern literature "big C creativity" (BCC), this term was introduced in [34].

However, the realities of information society transforming in many countries into knowledge society (KS) caused the emergence of the term "little C creativity" (LCC). The birth of this term reflects the demand of everyday creative thinking [2]. The two criteria of BCC are the originality and high significance for big groups of people. Creativity demonstrated by children usually is subjective, it is determined by their prior knowledge. An important characteristic of children's creativity is *imagination* [3].

The realities of KS demand to support and develop LCC in order to increase the proportion of the specialists possessing BCC. But it is not obvious how to achieve this global goal. The paper [33] describes the results of a large scale study carried out in USA and focused on the evolution of average level of intelligence and creativity during two decades, since the early 1990s. This study showed a considerable increase of average intelligence level during two decades. To the contrary, according to this study, the level of creative thinking considerably decreased. The study showed the increase of children's ability to produce ideas and the openness to new ideas until the age of nine. After this moment, these characteristics remain rather stable during approximately one year, and then they steadily decline. Children's curiosity follows the same dynamics.

A considerable diminishment of creativity level discovered in USA [33] became a significant standpoint for the development of an education brand called STEAM education (STEAMed). Here STEAM is an abbreviation for Science, Technology, Engineering, Art, Mathematics. During this decade, the ideas of STEAMed have become rather popular in the USA and South Korea.

During over two decades, we have been looking for more effective principles of teaching and learning in comparison with the broadly used ones. The accumulated theoretical and practical experience shows that modern education as a whole underestimates the significance of basing on EI for making easier for the students grasping central ideas of theoretical materials to be learned and for developing creativity.

The principal aim of this paper is to suggest a new paradigm for education in KS, in particular, for STEAMed. It is the Student-Self Oriented Learning Model (SSOL-model). Our model is called Student-Self oriented, because Self is always emotionally coloured. If we take something close to our heart or understand that it is not our cup of tea, we always experience emotions. Strong emotions leave a semantic trace in our inner world's picture [39].

The proposed SSOL-model has at least two successful implementations. The first one is the System of Emotional-Imaginative Teaching (the EIT-system) underpinned by our Theory of Dynamic Conceptual Mappings. The system is composed by (a) several complex methods combing teaching/learning with the development of the student's personality, (b) an original program of extra-scholastic humanitarian education covering 9 years of continuous studies, where the starting age is five – seven years [6-12, 14-22]. Our new theory and the system of methods became the starting point for developing the foundations of a new scientific discipline called cognitonics [13-18, 22]. The second implementation of the SSOL-model is art cognitonics [16 - 18].

The structure of this article is as follows. Next section describes the related approaches. The principal attention is given to (a) the difficulties encountered by the students trying to grasp the pieces of materials to be learned; (b) the approaches to STEAM education. Section 3 introduces the SSOL-model. Section 4 illustrates one of its components: a conceptual learning environment. Section

5 interprets the EIT-system as a successful implementation of the SSOL-model. Section 6 outlines a balanced approach to starting and continuing STEAM education, it develops EI, reasoning skills, and creativity. Section 7 interprets art cognitonics as the second successful implementation of the SSOL-model and explains the concept of cognitive engagement. Finally, the prospects of using the EIT-system for designing intelligent tutoring systems developing EI of the learners are discussed. A big part of this article is based on our papers [17, 23].

# 2 Related approaches

## 2.1 The problem of grasping the materials to be learned

The analysis of scientific literature shows that a high proportion of elementary, middle, and high school students encounter considerable difficulties as concerns grasping the main ideas of the pieces of theory to be studied, Let's consider the main social consequences of this situation.

1. The US Public Health Service prepared in the year 2000 a report on children's mental health [46]. According to this report, approximately one fifth of children and adolescents experienced problems showing their need for mental health services. One of the main reasons for this need is the intellectual and emotional obstacles on the way of successful learning faced by the school students in conditions of too short time for relaxation and refreshing the brain as a consequence of many-hour interaction with the Internet, in particular, with computer games.
2. Rather often, the breaks of discipline at lessons encountered by the students in the process of grasping the materials to be learned cause the growth of aggressiveness towards the teachers and the classmates with higher grades. The breaks of discipline negatively influence the total learning result of the class. Besides, these breaks and the aggressiveness of some students towards the teacher prevent a considerable part of gifted persons with the abilities of good educator from choosing the profession of teacher for himself/herself.
3. The negative consequences of students' aggressive attitude towards their classmates may be very severe (posting in social networks false information about a classmate, false erotic pictures, etc.) and even tragic in cases of cyber bullying.
4. In many cases, the attacks of young hackers against socially important technical systems may be explained as a result of expressing the aggressiveness. The intelligent power of applied computer systems is being permanently increased. That is why the negative consequences of the hackers' attacks promise to be increasing too. Taking this dangerous tendency into account, and, besides, the aggressiveness of some students towards their classmates and the teachers, a socially very significant task is to find the ways of making easier for the students the grasping of the pieces of theory to be learned (there are reasons to believe that, very often, the experience of success in the process of learning eliminates the student's aggressiveness).
5. In KS, many countries encounter the problem of insufficiently developed social lifts. The following scientific fact says about the high significance of solving this problem: in different strata of people living in one country, various gifts are distributed approximately in the same way. That is why it would be important to have the situation when the adolescents from different social-economic strata possessing the gifts being crucial for a certain socially significant profession (a medicine, a lawyer, etc.) would enter a university for mastering this profession.

Unfortunately, in many countries the real situation is quite opposite. For instance, it applies to UK. As it is shown in [30], the Organisation for Economic Co-operation and Development (OECD) describes the United Kingdom's troubling social mobility problems [38]: more than 50% of youngsters will grow up to have the same salary as their father. The Sutton Trust [45] shows that 53% of the UK's most influential people were independently educated, including 24% of university vice-chancellors, 32% of Members of Parliament, 51% of medical consultants, 54% of top journalists, 70% of High Court judges… when only 7% of the UK population are.

## 2.2 From STEM to STEAM education

The U.S. National Academies of Science, Engineering and Medicine prepared in 2005 a report underlining that for the global leadership of the U.S., their students would be better prepared in STEM disciplines (Science, Technology, Engineering, Mathematics). As a reaction, President Obama announced in 2009 the initiative called "Educate to Innovate". This initiative, first of all, proposed to increase federal investment in STEM education and to prepare 100,000 new STEM teachers by 2021. Regretfully, the U.S. Department of Education stated in the year 2011 that only 16% of high school students are interested in a STEM profession and had proven a proficiency in mathematics. Besides, 57% of high school students with initial interest in a STEM-related field lose interest before graduating from high school [28].

A dramatic diminishment of average creativity level discovered in USA [33] demanded a strong compensating response. The idea of response was prompted by the well known fact: the brightest works of art emerged due to the creative acts of consciousness. That is why a way out was seen in including art lessons into STEM curricula. The resolution No. 319 of the U.S. House of Representatives (June 21, 2011) indicated the significance of STEAM education (Science, Technology, Engineering, Art, Mathematics) and asked the Director of NSF to develop a STEM to STEAM committee [35, 41]. It seems that this resolution is underpinned not only by the ideas to increase the creativity of young generation but also by the idea to make the STEM disciplines more attractive for middle and high school students and, as a consequence, to contribute to solving the main task: to increase the effectiveness of STEM education and the proportion of young people choosing a STEM-related profession. During recent years, the ideas of STEAMed have become rather popular in several other countries, first of all, in South Korea[31, 32].

Our analysis of publications on implementations of STEAMed shows the lack of adequate theoretical foundations of creating the effective cognitive preconditions of STEAMed in kindergartens and elementary school. We mean here the ability to perceive beauty in various manifestations (the openness to perceiving beauty) and realization of basic creative mechanisms being equally necessary for future artists, mathematicians, dress designers and airplane designers. The significance of developing creativity in kindergartens and elementary school is underlined in [33].

The scientific literature presents only a rather restricted spectrum of the approaches to developing creativity in pre-school and elementary school. The prevailing approach is the plays of young children with physical objects, including the sorting of objects in accordance with certain properties (colour, size, etc.). During last decades, it has been possible to observe the shift from using for sorting mechanical objects to the usage of plants, sands, and water [27]. However, rather often the parents of young children (in particular, in Singapore) are anxious about the lack of learning outcomes as a result of the lessons when young children just free play [27].

That is why a topical problem is to discover the effective ways of developing the creativity at pre-school and elementary school lessons with considerable learning outcome.

## 3 Student-self oriented learning model

We believe that modern education may find the ways to effectively deal with numerous open problems as a result of accepting a new LM taking into account the significance of EI in professional and personal life.

The broadly accepted student oriented LM determines the activities launched by the goal to discover the world: acquisition of information, information processing, knowledge construction. The resulting activities are constructing a new text and constructing a new sense. Then the achieved cognitive-emotional state is as follows: a student is well-educated but not intellectually and spiritually mature.

Discovering the world is based on a brand-new culture on the basis of digital opportunities and ideology. Its essence is to catch up with new technologies (but not to find one’s way and incorporate it into modern reality as a new vision). It is underpinned by the curiosity and strong aspiration to discover the digital world, on the one hand, and by the desire to emulate the grown-ups and become as smart and powerful as the grown-ups or even much smarter and much more powerful.

The concept of Self is based upon our images of ourselves. The Self develops as it interacts with the most important of environmental influences. Through this social interaction the Self defines itself as a social being, which influences and is influenced by others [37].

*Student-self oriented learning model (SSOL-model) is defined as the model being beneficial for self-cognition and self-construction through the prism of the acquired knowledge and life experience* [17].

Natural language is the tool for constructing social reality [44]. The Self develops through the social interaction and co-creative work, because creative work suggests personal involvement and is underpinned with strong emotions (e.g., inspiration). The process is always emotionally coloured.

Under the framework of standard model, the process of knowledge acquisition often seems to the students to be first rather gloomy (no interest, no personal involvement), then pleasant. In case with the SSOL-model, the process of learning seems to the students to be pleasant and curious from the very beginning. Afterwards it is filled with never ending delight. The new model helps to exclude from the perception of educational process such characteristic as “gloomy”. It arises the interaction with the environmental influences and causes cognitive engagement of the students.

This idea is intuitively clear to very many experienced lecturers. EI suggests Self, because Self is always emotionally coloured. One is never tired when the subject of the conversation (or lecture) touches his/her Self in a positive and curious way. You are never tired if we are speaking about you and want to know your life experience, you are in the centre of attention.

Strength of materials (or mechanics of materials) is known as one of most difficult disciplines for the university students – future engineers. However, 62 years ago one scientist found a thrilling way to introduce basic ideas of his discipline. This scientist is Charles Seim, he wrote the article “A Stress Analysis of a Strapless Evening Gown” in the year 1956. This article was published in the book "A Stress Analysis of a Strapless Evening Gown and Other Essays for a Scientific Age" (Robert A. Baker, 1969, 212 pages). The translation of this book under the title “The physicists are joking” became very popular in Russia in the 1970s.

The core of the proposed model consists of self-cognition, self-construction, and self-regulation of self-conscious emotions. Let’s explicate these notions. Self-cognition is active transfiguration but not passive reflexion. Self is constructed through the interaction with the world (through the discovery of the world). The particular facets of the personality are improved as a result of new experience and as a result of processing the semantic trace left by a strong emotion caused by that new experience (it is the improvement of emotional experience). Self is always coloured by emotions. That is why this semantic trace deepens the emotional experience and, as a consequence, improves the emotional intelligence of the student [18, 39] .

Firstly, the realization of this new model leads to Student – Digital World balanced partnership. It means that the spiritual maturity and cultural level of the student become equal to the outstanding breakthrough in digital technologies. Secondly, the realization of the new model leads to the improvement of serendipity and turns information into serendipitous information (unexpected but desirable). Serendipity is the ability to make pleasant and unexpected discoveries entirely by chance [29, 43]. It leads to much higher level of socialization and to much higher level of responsibility.

To sum up, the suggested model determines the humanistic filling of education in the digital world. The new model suggests:

- a conceptual learning environment instead of a memorization-based one (it means making emotionally coloured the concepts to be learned and, as a consequence, making much easier grasping these concepts);
- the methods of achieving cognitive engagement of the students;
- a system of self-oriented questions in the process of knowledge acquisition;
- the methods of encouraging the students to discover the world aimed at self-cognition and self-construction;
- a method of teaching the students how to process serendipitous information.

# 4 Examples: a conceptual learning environment for studying second language

**Example 1.** The experience shows that it is very difficult for five - six year old Russian students to understand why they should use in simple phrases different words "am", "is", "are" and how one should combine these words with the words "I", "you", "he", "she", "it", "we", "they". Our approach to this problem is as follows. Assume that a teacher knows that her young student Julia has a beautiful dress for theatre, a dress for kindergarten, and a dress for a bathroom. Julia agrees that she never doubts what dress to wear. Then the words "am", "is", "are" may be called the different dresses of the verb "to be" (children at this age have very vivid imagination). Besides, "am" may be called a dress for visiting the house where the word "I' lives, "are" - a dress for visiting the house where the words "you", "we", "they" live, and "is" - a dress for visiting the house where the words "he", "she", "it" live [8].

**Example 2.** In English grammar we have the Present Continuous Tense. We propose a new approach to explaining this Tense, the motive is that our many year experience has shown that this approach provides the possibility to minimize the number of errors. The peculiarity of this piece of grammar for Russian learners is the lack of similar tense in the grammar of Russian language. This tense is very important tense, because it reveals the emotional state of an interlocutor and emotionally colours the speech. It is possible to call it the Emotional Tense – the tense which reveals our emotions. For example, saying "Look, she is reading", we attract somebody's attention to something or somebody, because we are not indifferent to it. When we are talking and drinking in a cafe, we say: "I am reading an interesting book now", it means that we are carried away by the book (at least we are not indifferent towards it if we mention it while talking and drinking). We use this tense speaking about the weather, about changing situation, irritation ("You are always wearing my slippers"), to express admiration ("What a nice hat you are wearing!"), personal arrangements, etc.

This tense shows emotionally coloured attitude towards something, it doesn't just state the fact, doesn't denote something which is true in general. The personal involvement is high, the state of minute is obvious (when somebody scoops a precious, significant, or just a particular minute out of the river of time). Due to the usage of the proposed approach at lessons of English as a second language (SL), the Russian students begin to employ this tense eagerly while speaking. It makes the lessons of English grammar socially coloured and more interesting for the students. Besides, this method reveals the essence of the English character.

# 5 The system of emotional-imaginative teaching as a successful implementation of the SSOL-model

In early 1990s we came to the conclusion that educational potential of young learners (5-6-7 years old) is much higher then it was broadly accepted to believe. The key to more effective realization of this potential should be the ways of establishing a correspondence between a piece of material to be studied and a certain *bright* fragment of the learner's conceptual picture of the world. We called such correspondences *dynamic conceptual mappings* [7, 8, 13]. That is why we started in early 1990s a study aimed at finding more effective ways of teaching and learning due to systemic basing on young learners' emotional experience accumulated, in particular, during the breakfasts and lunches, the walks in gardens and parks and along a river, while visiting school and theatres, playing various games, sport activities, etc.

Step by step, we obtained several scientific and practical results of high social significance, and these results stood apart from the principal trends in education of the 1990s and early 2000s [15]. It was done due to our original *Theory of Dynamic Conceptual Mappings* (*the DCM-theory*) [7-12, 19-21] and our *System of Emotional-Imaginative Teaching* (*the EIT-system*), based on the DCM-theory. The EIT-system is aimed at systematic development of EI, reasoning skills, sound creativity, language skills, and communication culture at the lessons of language - mother tongue and SL, literature and poetry in two languages (on the example of Russian and English), symbolic languages of painting, sculpture, garden-park art, classic dance. We have accumulated the 28-year-long successful experience of using the EIT-system in extra education in Russia. Many aspects of the EIT-system are described in our papers published in the proceedings of the First – Fifth international conferences on cognitonics (see, in particular, [16, 17, 22, 23]), in the papers [6-11, 13-15, 19-21] and in the monograph [18].

Let’s consider now such aspects of the EIT-system that concern basing on and developing EI and Self of the students. Self is always creative, because it is a personal way of viewing the world, based on the world’s conceptual picture of the beholder and his/her estimation of the events.

**Example.** The famous Russian poet Boris Pasternak gives the picture of the early spring, writing:

- Is it only dirt you notice
- Does the thaw not catch your glance?

We ask the students what makes Pasternak think that the thaw is beautiful. The answers of young students (7 years old) are as follows:

- The thaw is like a herd of dapple grey deer basking in the spring sun;
- It is like a surface of the moon dotted with craters;
- A table served for breakfast with blue cups and black ice tea with a piece of Sun.

This approach helps young students to understand and penetrate the very essence of beautiful poetical lines written by the great poet. Besides, it expands their way of viewing the world, makes the surrounding world much more colourful, and their way of viewing it much more creative. This approach helps them also to understand painting (and modern painting, in particular).

An acute educational problem is early socialization of children in KS. Let’s illustrate the approach of the EIT-system to solving this problem.

**Example.** In the fairy-tale “Snow White” the Queen asks: “Looking-glass on the wall who is fairest of us all?”. The students are asked whether it is a question in fact or she is sure that she is beautiful. The young students give the following explanations:

- If she wants to know as a researcher, she wouldn’t be furious.
- She does it every morning simultaneously with having coffee or brushing her hair. It means that she is sure in the answer.
- She is selfish and she doesn’t think about the good for the others, even the King. That is why she can’t be beautiful. May be attractive, like Cinderella’s sisters, but not beautiful.
- When the hunter promises to take Snow White into the woods, he doesn’t promise to kill her. But the Queen is sure that he does. It is a cognitive trap: she doesn’t expect anybody to protest, to disagree, to disobey her. It is one more prove that she is selfish and doesn’t listen to anyone. It will mislead her.

Our educational results obtained in the 1990s due to the EIT-system were interpreted during last decade as a significant contribution to *developmental psychology* and to *positive movement in psychology* [6, 9, 10, 15, 16].

# 6 A balanced approach to developing emotional intelligence, reasoning skills, and creativity as an effective framework for STEAM education

Comparing our results with the approaches to initial, start stages of STEAMed described in the papers from North America and Asia, we have come to the conclusion that *now STEAMed has no appropriate theory of start mechanism*. Showing the diminishment of creativity level in USA during two last decades, Kim [33] indicated the necessity of starting the development of creativity in kindergartens. An important role in achieving this goal is to be played by calm, free, friendly atmosphere at lessons [33, 34].

We believe that now, as a whole, cognitive potential of five-seven year olds is underestimated. The analysis shows that the DCM-theory and the EIT-system may be interpreted as an effective theoretical framework for starting STEAMed. Figure 1 shows the system of cognitive transformations corresponding to creating initial cognitive preconditions of effective STEAMed concerning a concrete young learner.

The principal advantages of our approach to creating the preconditions of effectively starting STEAMed are as follows. Young learners (five-seven years old) get accustomed to the beauty expressed in various ways. It is well known that it is highly important not only for the painters, sculptors, poets, dress designers but also for mathematicians, physicists, designers of ships and airplanes to have a well developed feeling of harmony.

A fundamental significance of our approach for STEAM education is determined also by the formulation of the cognitive precondition of the situation when it is possible to start systematic acquaintance of children with the computer. It is the realization of the Thought-Producing Self of the child [12, 13, 18, 20].

As a consequence of getting a developed figurative reasoning (due to of a kind of intellectual game, intellectual competition), young learners get a developed creativity. Our approach to early creativity development excellently correlates with the opinion of Piaget [40] about the significance of "reflective abstraction'. i.e., about the crucial role of processing and constructing knowledge in the course of mental actions performed on the perceived and imaginary objects and causing generation of new ideas.

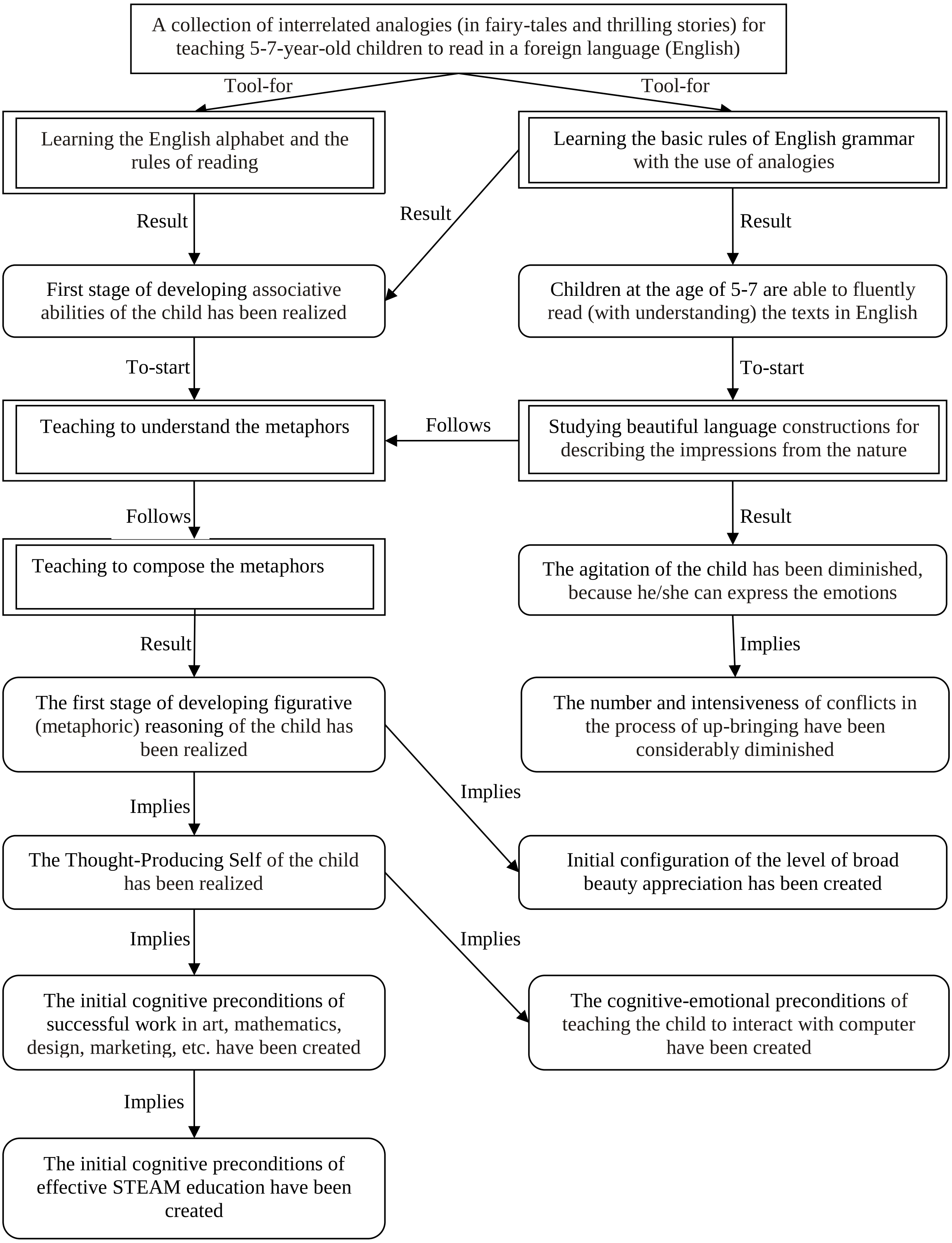

A collection of interrelated analogies (in fairy-tales and thrilling stories) for teaching 5-7-year-old children to read in a foreign language (English)
Tool-for
Tool-for
Learning the English alphabet and the rules of reading
Learning the basic rules of English grammar with the use of analogies
Result
Result
Result
First stage of developing associative abilities of the child has been realized
Children at the age of 5-7 are able to fluently read (with understanding) the texts in English
To-start
To-start
Teaching to understand the metaphors
Follows
Studying beautiful language constructions for describing the impressions from the nature
Follows
Result
Teaching to compose the metaphors
The agitation of the child has been diminished, because he/she can express the emotions
Result
Implies
The first stage of developing figurative (metaphoric) reasoning of the child has been realized
The number and intensiveness of conflicts in the process of up-bringing have been considerably diminished
Implies
Implies
The Thought-Producing Self of the child has been realized
Initial configuration of the level of broad beauty appreciation has been created
Implies
Implies
The initial cognitive preconditions of successful work in art, mathematics, design, marketing, etc. have been created
The cognitive-emotional preconditions of teaching the child to interact with computer have been created
Implies
The initial cognitive preconditions of effective STEAM education have been created

Figure 1: A map of cognitive transformations corresponding to creating initial cognitive preconditions of effective STEAM education.

As for early socialization, the young students became careful, tactful, thoughtful, they acquire the feeling of empathy and start appreciating the harmony in everything, including human relationships. In this case we say that children have reached the level of consciousness development called *the level of broad beauty appreciation* [6, 15]. It is important to do before the age of "teen", when children are ready to discuss and follow the social rules. In this case, beauty becomes the core of their system of values [15, 16], and it helps a lot at the moment they are twelve and are going on thirteen – the transition age. Figure 2 illustrates a new look at combined development of EI, reasoning skills, and creativity.

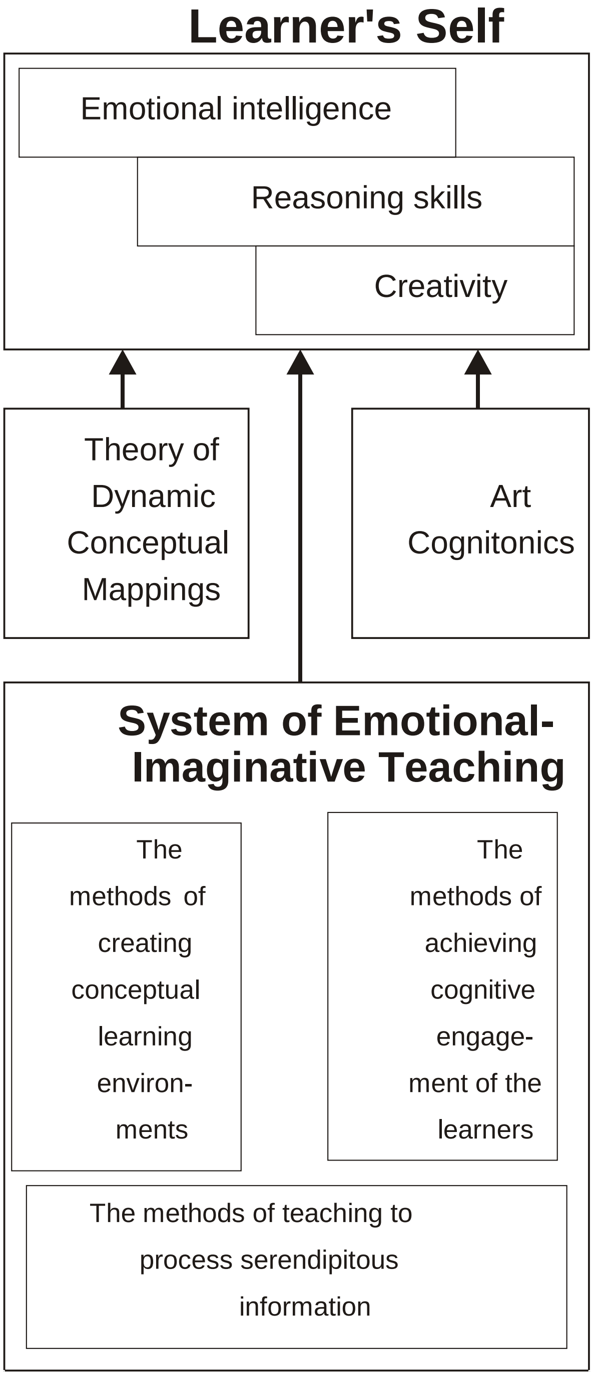

Learner's Self
Emotional intelligence
Reasoning skills
Creativity
Theory of Dynamic Conceptual Mappings
Art Cognitonics
System of Emotional-Imaginative Teaching
The methods of creating conceptual learning environ-ments
The methods of achieving cognitive engage-ment of the learners
The methods of teaching to process serendipitous information

Figure 2: A scheme of a new look at combined development of emotional intelligence, reasoning skills, creativity, and serendipity.

The EIT-system includes the original methods of teaching to process serendipitous information. According to Kim [33], a very large scaled study carried out in USA showed that during last decade of the XXth century and first decade of the XXIst century children became less able to connect seemingly irrelevant things. That is why our methods of teaching how to process serendipitous information are very topical.

Well developed feeling of beauty creates for the student the preconditions of being successful at arts lessons. It is broadly accepted to believe that art education supports and develops creativity of young children and teenagers, develops EI, improves emotional well-being, self-confidence, and life skills of the students [42].

The well developed feeling of beauty also creates for the students the preconditions of being successful at art lessons, hence contributes to the success of STEAMed. Hence the broad usage of our approach would contribute to the success of STEAMed.

## 7 Art cognitonics and cognitive engagement at art lessons

Art cognitonics (AC) [16-18] is one of the principal branches of cognitonics, or the science about the human being in the digital world [13 - 15]. The first objective of cognitonics is to explicate the distortions in the perception of the world caused by the information society and globalization. The second, principal objective of cognitonics is to cope with these distortions in different fields by means of elaborating systemic solutions for compensating the negative implications of the kind for the personality and society, in particular, for creating cognitive-cultural preconditions of the harmonic development of the personality in the information society and smart society and for ensuring the successful development of national cultures and national languages.

From the standpoint of educational practice, cognitonics proposes an answer to the following question: what precious ideas and images accumulated by the mankind, at what age, and in what a way are to be inscribed into the world's conceptual picture of a person in order to harmonize his/her intellectual and spiritually-coloured emotional development and to contribute to the successful development of national cultures and national economics?

Cognitonics formulates a new, large-scale goal for the software industry and Web science: to develop a new generation of culture-oriented computer programs and online courses (in the collaboration with educators, linguists, art historians, psychologists) - the computer programs and online courses intended for supporting and developing positively-oriented creativity, emotional intelligence, the appreciation of the roots of the national cultures, the awareness of the integrity of the cultural space in the information and smart society, and for supporting and developing symbolic information processing and linguistic skills, associative and reasoning abilities of children and university students.

The objectives, ideas, and methods of Cognitonics have been broadly supported by the international scientific community. We successfully organized (as the Co-Chairs) five international scientific conferences on Cognitonics (Cognit-2009 – Cognit-2017) under the framework of the international scientific multi-conferences "Information Society" (IS-2009, IS-2011, IS-2013, IS-2015, IS-2017, Slovenia, Ljubljana, Jozef Stefan Institute, October 2009, 2011, 2013, 2015, 2017). The access to the Proceedings of the conferences Cognit-2009 – Cognit-2017 is open, see https://is.ijs.si. A part of cognitonics-based scientific and practical results is presented in the Second Edition of the International Encyclopedia of Social and Behavioral Studies [6].

The DCM-theory and the EIT-system belong to the constructive core of cognitonics. AC aims at tuning the EI of the young children and adolescents with the help of well-known works of art. The goal is to create a bright semantic trace in the world's conceptual picture of the learner corresponding to an idea explaining or illustrating a moral value, communicative situation, a situation of making a decision, cognitive process itself, the process of self-cognition and consideration, the seething cocktail of emotions, a way of viewing the world around, etc.

AC establishes the links between the objects, situation, processes, views of a person (a beholder) and the work of art that becomes a metaphor or a vivid illustration (vivid mental representation) of something the beholder is considering about. That is why the consciousness of the beholder receives a considerable impulse to developing the ability of establishing diverse analogies and consequently to finding a new look at a situation [16-18].

**Example.** For enriching the colour of their canvases, the impressionists made use of what is known as division of colour and optical blending. E.g., to represent a green meadow, they put little tabs of blue and yellow on the canvas which are supposed to be combined to form green in the eye of the beholder – a far more intense green than one taken straight from the artist's palette. That is why it is impossible to understand the idea of a picture standing close to the canvas. We have to step aside and look at it from a certain distance to enjoy it and to have the desired effect.

The same situation we have in every-day life. "Multiple debs, reflections" prevent us from grasping the sense of what is happening. As in case with impressionists' canvases, we have to have a look at the situation from a distance, and distance in this case is equal to time distance. We need some time to better understand what has happened, and this will help us to cope with the situation (see another examples in [14]).

The paper [14] contains an algorithm of resisting emotional attacks from social networks by means of transforming the negative emotions into the positive ones. This algorithm is based on the idea described immediately above.

*Cognitive engagement* (*CE*) is defined in [15-18] as the process of highly motivated intellectual activity when the interest towards the subject under discussion is so strong that the students loose the track of time and, as a result, they are not tired. The students' interest determines the level of involvement. The emotional response is very close to inspiration, because they are making their own discoveries, and their mental efforts are appreciated. It helps to provide a conceptual learning environment instead of a memorization based one and enhances the motivation. CE is created mainly by the components called in [15, 18] *focused attention, positive effect, aesthetics*, *endurance*, *novelty, motivation*.

AC may be interpreted as second successful implementation of the SSOL-model.

# 8 A script of an intelligent tutoring system contributing to early socialization of the learners

During last twenty years, the intelligent tutoring systems (ITSs) have been broadly used throughout the world for helping children and adolescents to grasp theoretical materials. The big subclasses of ITSs help to study (a) second language (SL), (b) mathematics. However, one has been able to find in the literature only separate examples of the systems oriented at developing the personality of the learners. In particular, the interactive multimedia courseware package CITRA is a tool for moral values education using traditional Malay oral narratives [36]. Two collaborative videogames described in [4, 5] not only develop mathematical and language skills of the eight – ten year old children in Mexico but also support and develop in Mexican children the skill of effective collaboration in a team, hence develop social competence.

The analysis shows that the methods of cognitonics open broad prospects for the development of a new generation of ITSs. Their principal distinguished features should be orientation at culture, at developing EI of the learners. New, culture-oriented scripts under the framework of cognitonics may be divided into three main groups.

**Group 1:** Socialization-oriented scripts.
The scripts of the kind help to improve the process of better understanding of the goal and the nature of a communicative act. As a result, the acquired communication skills diminish the number of conflicts caused by the misunderstandings and the level of aggression caused by these conflicts.

**Group 2.** Improvement of the language (mother tongue and SL) as a tool of thinking in order to oppose the phenomenon of poor language and, as a consequence, poor cognitive process, that is, an underdeveloped tool of constructing social reality.

**Group 3.** The scripts aimed at demonstrating the possibilities of expressing the same idea by means of different languages, e.g., by means of natural language and the language of painting. The goal is the development of the ability to see something extraordinary in an ordinary thing or situation, to find a new look at an object of interest and to make a discovery, to develop the ability of processing serendipitous information.

Let’s consider a script of a culture-oriented ITS based on the idea of social conventions and belonging to the Group 1. The literary source of this script is the fairy-tale “Sleeping Beauty”. The script is associated with two aims. The first aim is to explain how it is possible for the student to escape in the life the meeting with the 13th fairy. It means not to make a person act in a provocative way. The reason is that such kind of behaviour would make harm both to an initiator and to a person. In case of the considered fairy-tale, a fairy turned into a witch, because she could not cope with emotions and gave way to hatred. The second aim is to develop the Ecological Self of the student.

**Instruction 1 for the designers.**
Construct a dwelling (a hut, a castle, a palace, a cottage, etc.) being appropriate for a King and a Queen and for the 13th fairy. Put the dwelling into appropriate surrounding (garden, park, edge of the forest, etc.).

Choose the interior revealing the characters of the story. Choose the time of the day, the season. Dress the characters up and choose some occupations for them.

**Instruction 2.**
According to the logic of Instruction 1, create a big album containing the photos of the characters in different situations. One part of the photos adequately illustrates the life of the personages. Another part falsely illustrates the actions of the personages (in such cases an action or situation contradicts the properties of the character).

Motivate the students to select the photos for the album of each character. The aim of this subsystem of the ITS is to develop the ability of the student to correctly associate the actions of a character with the essence of this character.

**Instruction 3.**
Create a subsystem motivating students to construct a dynamic picture showing the extensive preparations in the Kingdom for the Christening Party.

Step 1: The construction of a picture showing all kinds of the living beings (in particular, the carpenter, the animals, and the birds) in the Royal Park.

Step 2: Ask the student to select the living beings for active preparation for the Christening Party.

Step 3: For each considered living being, select one of four - five actions.

**Example.** It is possible that for the birds a student will select the action “sing the songs”.

**Instruction 4.**
In general terms, the task is to realize the step explicating the essence of social responsibility. The details of this step are as follows.

The King should be sure that every guest has received the invitation and has accepted the invitation. In order to be sure, the King is to receive a confirmation from every guest that the guest has received and has accepted the invitation. The violation of the rule leads to misunderstanding. In our case, the 13th fairy didn’t receive the invitation, though the King had sent an invitation, and regarded the lack of invitation as a mark of disrespect on his part.

**Instruction 5.** Explain to children how the violation of etiquette will mislead them. Preventing a violation of etiquette means not to make a person act in a provocative way. The reason is that such kind of behaviour would make harm both to an initiator and to a person. In case of the considered fairy-tale, the fairy turned into a witch, because she could not cope with emotions and gave way to hatred. Consider possible examples.

**Example 1.** One meets a classmate but doesn’t greet him/her. It may lead to offense.

**Example 2.** One may take a pencil of a classmate without the permission. The classmate may become cross with him/her.

**Example 3.** One may eat a cake without expressing his/her gratitude to a classmate. The classmate may think that he/she is not polite.

**Example 4.** When he/she does something wrong and doesn’t apologize, then the classmate may think that he/she is rude with him/her.

**Example 5.** When a classmate brings a mouse, though he/she knows that the girl is afraid of mice, it means that he/she is selfish, because he/she doesn’t take into account the peculiarities of the girl.

**Instruction 6.** The essence of this step is to attract the attention to the episodes revealing the behaviour of the character of the book who is thinking and acting in terms of public good.

**Example 1.** The 12th fairy was attentive and ready to help, she made up the situation and tried to make not only the princess but the whole kingdom fall asleep. The motive of the fairy was not to make the princess lonely when she woke.

**Example 2.** The people of the kingdom were ready to help, and they brought their spindles to the square to make a fire. They were ready to sacrifice the necessary things and not to have new clothes, because they would not have the spindles to spin.

**Instruction 7.**

**Preliminary stage.** Ask five-six-seven-year-old children being acquainted with the fairy-tale “Sleeping Beauty” to describe the preparations in the Kingdom to the birth of a princess. Construct a collection containing all proposed creatures and their actions of the kind.

**Main stage.** Ask children to select the creatures and their preparations to the birth of a princess.

# 9 Children's speech as a source of vocabulary and images for the designers of tutoring systems

The inner world’s picture of young children is very different from the picture of adults. Young children have a vivid imagination, and they easily go from the reality into the world of fantasy. That is why it is very important for the designers of ITSs to use in the computer systems the vocabulary and images extracted from children's examples collected at the preliminary stage of developing a system. We have collected, in particular, the following examples given by children:

**Preparation of gifts**
(1) The gardener prepares fountains and flower beds; (2) the carpenter makes the cradle shaped like (a) a swan, (b) dolphin which always rescues, (c) see-shell in which the princess will be like a pearl, (d) a flower which opens its petals at dawn; (3) the beasts prepare (a) milk taken from forest plants, (b) pick up glow-worms; the birds sing songs; the kittens are purring a lullaby; the baby-squirrels have picked up nuts; the mother-dogs are knitting mittens; the mother-squirrels are sewing the dresses for the dolls of the princess.

**Preparation in the palace**
(1) The birds are bringing in the beaks the field flowers; (2) the chipmunks are bringing the baskets with drops of dew in order to water field flowers; (3) in the evening the star peeps through the curtain to light the room; (4) the little angel descends in order to fill the nursery with kind dreams and to kiss the princess good night.

## 10 Possible directions of future studies

The considered script allows us to get an initial impression about the possibilities of using the methods of emotional-imaginative teaching as the basis for developing ITSs of a new generation. This script may be compared with a single piece of a big, complicated mosaic picture to be created. The EIT-system provides original effective methods for designing ITSs solving the following tasks:

- developing imagination, creativity by means of teaching to decode metaphors and invent metaphors;
- contributing to early socialization of the learners on the example of etiquette as a social agreement (etiquette makes the behaviour of the humans predictable, it is very important for understanding each other and in order not to heart the feelings of people);
- making thrilling the mastering of SL grammar (on the example of English);
- teaching the learners to integrate information dispersed in various sources and to establish time-causal relationships between the extracted facts;
- revealing cross-culture differences for avoiding misunderstanding during communication.

The EIT-system has been mainly realized at lessons of English as a SL for Russian-speaking children and at the lessons of poetry and literature in English, at lessons devoted to explaining the symbolic language of painting, the culture of communication, and the symbolic language of classical dance. These kinds of lessons are considered in numerous countries as highly appropriate for young children and teenagers. The carefully selected collection of texts used at lessons is provided by a number of classical, world-known fairy-tales and novels, in particular, “Snow White”, “Cinderella”, “Sleeping Beauty”, “Pinocchio”, “Pollyanna”, “The Life and Adventures of Santa Claus“ by L. Frank Baum, “Alice in Wonder Land” by Lewis Carroll, “The Wind in the Willows” by Kenneth Grahame, “The Hundred and One Dalmatians” by Dodie Smith, etc. That is why the EIT-system may be used (after a certain adaptation requiring a small time) in English-speaking countries and in numerous countries where the English language is learned as a SL.

## 11 Conclusion

We believe that the proposed SSOL-model possesses the properties enabling its usage as a paradigm for education in KS. The focus on the student's Self at the lessons means that the lessons are emotionally coloured, and this very much contributes to the success of the learning process.

Now there are at least two successful implementations of the SSOL-model: the EIT-system, tested during 28 years in Russia, and art cognitonics. The principal distinguishing feature of the EIT-system is an effective, many-staged method of sustaining and developing creativity in young children and adolescents, supporting and developing EI, basing on EI for making much easier the grasping of the materials to be studied.

Our numerous publications in English describe many aspects of the EIT-system. The scholars from various countries do have the possibility to develop their original implementations of the SSOL-model with respect to their mother tongue and national culture. The stated ideas and the EIT-system provide a strong support to STEAM education and to anthropocentric approach to education in the digital age as a whole.

## Acknowledgement

We are grateful to the anonymous reviewers of our paper for precious remarks.

## References

[1] Bradberry, T. (2014). *Emotional Intelligence - EQ*; http://www.forbes.com/sites/travisbradberry/2014/01/09/emotional-intelligence/#216e16f61ac0; retrieved 21.07.17.

[2] Craft, A. (2001). Little c creativity. In A. Craft, B. Jeffrey and M. Leibling (Eds.), *Creativity in Education.* Continuum, London, 45-61.

[3] Craft, A. (2005). *Creativity in Schools: Tensions and Dilemmas*. Routledge, Abingdon. https://doi.org/10.4324/9780203357965

[4] Craig, P., Roa-Seïler, N., Martínez Díaz, M., and Lara Rosano, F. (2013). Assessing the Potential of Colaborative Video Games to Improve Education in La Mixteca Region of Mexico. In: Gams, M., Piltaver, R., Mladenic, D. et al., Eds. (2013). *Proceedings of the 16th International*

*Multiconference Information Society – IS 2013, Slovenia, Ljubljana, 7 – 11 October 2013. The Conference Kognitonika/Cognitonics*. Jozef Stefan Institute; pp. 413-417; https://is.ijs.si/archive/proceedings/2013/ .

[5] Craig, P., Roa-Seiler, N., Martínez Díaz, M., and Lara Rosano, F. (2014). A Cognitonics Approach to Computer Supported Learning in the Mexican State of Oaxaca. *Informatica. An Intern. Journal of Computing and Informatics*, 38, 3, pp. 241-248.

[6] Fomichov, V. A. (2015). Conscious Control during Childhood, Development of. In *International Encyclopedia of the Social and Behavioral Sciences, Second Edition* .Vol. 4. Elsevier, Oxford, pp. 666-672. https://doi.org/10.1016/B978-0-08-097086-8.23082-1.

[7] Fomichov, V. A., Fomichova, O. S. (1994). The Theory of Dynamic Conceptual Mappings and its Significance for Education, Cognitive Science, and Artificial Intelligence. *Informatica. An International Journal of Computing and Informatics,* Vol. 8, No. 2, pp. 31-148.

[8] Fomichov, V. A., Fomichova, O. S. (1995). The artificial intelligence theory and highly effective methods of teaching young children foreign languages. *Cybernetica.* XXXVIII, pp. 321-344.

[9] Fomichov, V.A., Fomichova, O.S. (1997). An Informational Conception of Developing the Consciousness of the Child. *Informatica. An Intern. Journal of Computing and Informatics (Slovenia),* Vol. 21, pp. 371-390.

[10] Fomichov, V., Fomichova, O. (1998) A new theoretical and practical approach to early positive developing child's consciousness. In *R. Trappl (Editor), Cybernetics and Systems'98. Proceedings of the 14th European Meeting on Cybernetics and Systems Research.* Vol. 1, Austrian Society for Cybernetic Studies, Vienna, pp. 276-281.

[11] Fomichov, V.A., Fomichova, O.S. (1998). Early development of natural-language-processing abilities as a key to up-bringing of children without conflicts. In: *G. Ritschard, A. Berchtold, F. Duc, D.A. Zighed (Eds). Apprentissage: des principes naturels aux methodes artificielles.* Editions HERMES, Paris, pp. 67-81.

[12] Fomichov, V. A., Fomichova, O. S. (2000). The social responsibility of computer science specialists for the creative potential of the young generation. *Intern. Journal of Artificial Intelligence in Education.* Vol. 11, pp. 208-219.

[13] Fomichov, V.A., Fomichova, O. S. (2006). Cognitonics as a New Science and Its Significance for Informatics and Information Society; *Informatica. An Intern. Journal of Computing and Informatics (Slovenia),* Vol. 30, pp. 387-398.

[14] Fomichov, V. A., Fomichova, O. S. (2012). A Contribution of Cognitonics to Secure Living in Information Society. *Informatica. An Intern. Journal of Computing and Informatics*, Vol. 36, pp. 121-130.

[15] Fomichov, V.A., Fomichova, O.S. (2014). An Imperative of a Poorly Recognized Existential Risk: Early Socialization of Smart Young Generation in Information Society. *Informatica. An International Journal of Computing and Informatics*, Vol. 38, No. 1, pp. 59-70.

[16] Fomichov V. A., Fomichova, O. S. (2015). Early Development of the Human Being Ideal Constituent and Art Cognitonics as the Answer to the Challenge of Knowledge Society. In: *Informacijska druzba - IS 2015. Proc. of the 18th International Multiconference - IS 2015, Edited by V. A. Fomichov, O. S. Fomichova*. *Vol. F: Kognitonika/Cognitonics. Oct. 12th-13th, 2015, Ljubljana*. Jozef Stefan Institute, pp. 27-32.

[17] Fomichov, V. A., Fomichova, O. S. (2017). The Student-Self Oriented Learning Model as a Paradigm for Supporting and Developing Emotional Intelligence and Creativity.In *Informacijska druzba - IS 2017. Proceedings of the 20th International Multiconference - IS 2017, Edited by V. A. Fomichov, O. S. Fomichova*. *Vol. Kognitonika/Cognitonics. October 9th-10th, 2017, Ljubljana, Slovenia* . Jozef Stefan Institute, Ljubljana, pp. 11-16.

[18] Fomichov, V. A., Fomichova, O. S. (2018). *Cognitonics and Its Significance for Education in the Digital Age*. Moscow, Publishing House MAKS Press. 344 p. (in Russian).

[19] Fomichova, O.S., Fomichov, V.A. (1996). Theoretical Foundations of a New Method of Teaching Children Effective Information Processing. *Informatica. An Intern. Journal of Computing and Informatics,* Vol. 20, pp. 381-399.

[20] Fomichova, O. S., Fomichov, V. A. (2000). Computers and the Thought-Producing Self of the Young Child; *The British Journal of Educational Technology,* Vol. 31, No. 3, pp. 213-220. https://doi.org/10.1111/1467-8535.00152.

*[21]* Fomichova, O.S., Fomichov, V.A. (2003). A new paradigm for constructing children-oriented web-sites of art museums. *Intern. Journal of Educational Technology and Society,* Vol. 6, No. 3, pp. 24-29.

[22] Fomichova, O. S. and Fomichov, V. A. 2009. Cognitonics as an Answer to the Challenge of Time. In *Proceedings of the 12$^{th}$ International Multiconference Information Society - IS 2009, Slovenia, Ljubljana, 12 – 16 October 2009.* The Conference Kognitonika/Cognitonics. Jozef Stefan Institute, Ljubljana, 2009, pp. 431-434..

[23] Fomichova,, O. S., Fomichov, V. A. (2017). The Methods of Cognitonics as the Basis for Designing Intelligent Tutoring Systems Developing Emotional Intelligence of the Learners. In *Informacijska druzba - IS 2017. Proc. of the 20th Intern. Multiconference - IS 2017, Edited by V. A. Fomichov, O. S. Fomichova*. *Vol. Kognitonika/Cognitonics. October 9th-10th, 2017, Ljubljana, Slovenia*. Jozef Stefan Institute, Ljubljana, pp. 17-21.

[24] Gardner, H. (1993). *Creating Minds: an Anatomy of Creativity Seen Through the Lives of Freud,*

*Einstein, Picasso, Stravinsky, Eliot, Graham, and Gandhi*. Basic Books, New York.

[25] Goleman, D. (1995). *Emotional Intelligence.* Bantam Books, New York.

[26] Goleman, D. (2016). *Emotional Intelligence*; http://www.danielgoleman.info/topics/emotional-intelligence; retrieved 12.09.2016.

[27] Goy, P. (2015). Pre-school kids play and learn. *The Straits Times*, Nov. 23, 2015; http://www.straitstimes.com/singapore/pre-school-kids-play-and-learn (retrieved 14.12.2017).

[28] Hom, E.J. (2014). What is STEM Education. *LiveScience,* Feb. 11, 2014; https://www.livescience.com/43296-what-is-stem-education.html (retrieved 10.12.2017).

[29] Hornby, A. S. (1995). *Oxford Advanced Learner's Dictionary of Current English. Fifth edition*. Oxford University Press.

[30] Kane, T. B. (2014) Using Cognitive Tunnels in a New Approach to Building Social Elevators in the Information Society. *Informatica. An International Journal of Computing and Informatics (Slovenia),* Vol. 38, pp. 263-271.

[31] Kim, B.-H., Lim, Y.-D., Jung, M.-Y. et al. (2017). The effects of STEAM class using science-art-IT convergence art work for middle school education under a free semester system in Korea. International Workshop on Intelligent Information Technology (2WINTECH), Cheonan, South Korea, Oct. 20-22, 2016. *Advanced science letters*, Vol. 23, No. 3, pp. 1700-1704. https://doi.org/10.1166/asl.2017.8634.

[32] Kim, D., Bolger, M. (2017). Analysis of Korean elementary pre-service teachers' changing attitudes about integrated STEAM pedagogy through developing lesson plans. *Intern. journal of science and mathematics education.* Vol. 15, No, 4, pp. 587-605. https://doi.org/10.1007/s10763-015-9709-3.

[33] Kim, K. H. (2011). The creativity crisis: the decrease in creative thinking scores on the Torrance Tests of Creative Thinking. *Creativity Research Journal.* Vol. 23, No. 4, pp. 285-295. https://doi.org/10.1080/10400419.2011.627805

[34] Kudryavtsev, V. T. (2011). The phenomenon of child creativity. *International Journal of Early Years Education.* Vol. 19, No. 1, pp. 45-53. https://doi.org/10.1080/09669760.2011.570999.

[35] Maeda, J. (2012). STEM to STEAM; Art in K-12 is key to building a strong economy. EDUTOPIA. George Lucas Educational Foundation, Oct. 2, 2012; https://www.edutopia.org/blog/stem-to-steam-strengthens-economy-john-maeda (retrieved 14.02.2018).

[36] Mukti, N.A., Hwa, S.P. (2004). Malaysian perspective: designing interactive multimedia learning environment for moral values education. *Educational Technology and Society.* Vol. 7, No. 4, pp. 143-152.

[37] Neisser, U. (1997). The roots of self-knowledge: perceiving Self, It, and Thou. In: *Snodgrass, J.G. and Tompson, R.L. (eds.). The Self Across Psychology. Self-Recognition, Self-Awareness, and the Self Concept. Annals,* Vol. 818. The New York Academy of Sciences, pp.19-33. https://doi.org/10.1111/j.1749-6632.1997.tb48243.x.

[38] OECD (2010). A Family Affair: Intergenerational Social Mobility across OECD Countries, in OECD, *Economic Policy Reforms 2010: Going for Growth*, OECD Publishing. Doi: 10.1787/growth-2010-38-en. https://doi.org/10.1787/growth-2010-38-en.

[39] Padovan, C., Versace, R. (1998). The represen-tation of emotion in long term memory. *Apprentissage: des principes naturels aux methodes artificielles, sous la direction de G. Ritschard, A. Berchtold, F. Duc, and D.A. Zighed.* Editions HERMES, Paris, pp. 111-123.

[40] Piaget, J. (1981). Creativity. In *J. M. Gallagher and D. K. Reid (Eds.), The Learning Theory of Piaget and Inhelder*. Brooks-Cole, Monterey, CA, pp. 221-229.

[41] Pomeroy, R.S. (2012). From STEM to STEAM: Science and Art Go Hand-in-Hand. *Scientific American*, Aug. 22, 2012; https://blogs.scientificamerican.com/guest-blog/from-stem-to-steam-science-and-the-arts-go-hand-in-hand/ (retrieved 14.02.2018).

[42] Roege, G. B., Kim, K. H. (2013). Why we need arts education. *Empirical Studies of the Arts.* Vol. 31, No. 2, pp. 121-130. https://doi.org/10.2190/EM.31.2.EOV.1

[43] Sawaizumi, S., Katai, O., Kawakami, H., Shiose, T. (2007). Using the concept of serendipity in education. *Proc. of the KICSS 2007: The Second International Conference on Knowledge, Information and Creativity Support Systems, Nomi, Ishikawa, Japan. JAIST Press*; https://dspace.jaist.ac.jp/dspace/handle/10119/4087

[44] Searle, J. R. (1995). *The Construction of Social Reality.* The Penguine Press.

[45] The Sutton Trust, Educational Backrounds Reports, http://www.suttontrust.com/our-work/research/professions/

[46] *US Public Health Service. Report on the Surgeon's General's Conference on Children's Mental Health: a national action agenda* (2000). Washington, DC., Department of Health and Human Services.

# Noise-tolerant Modular Neural Network System for Classifying ECG Signal

Alberto Ochoa
Universidad de Colima, 28400 Colima, Mexico
E-mail: aochoa@ucol.mx

Luis J. Mena and Vanessa G. Felix
Universidad Politecnica de Sinaloa, 82199 Mazatlan, Mexico
E-mail: lmena@upsin.edu.mx, vfelix@upsin.edu.mx

Apolinar Gonzalez and Walter Mata
Universidad de Colima, 28400 Colima, Mexico
E-mail: apogon@ucol.mx, wmata@ucol.mx

Gladys E. Maestre
University of Texas Rio Grande Valley School of Medicine, Brownsville, TX 78520, USA
E-mail: gladys.maestre@utrgv.edu

**Keywords:** electrocardiogram signals, machine learning, classification, neural network

**Received:** April 18, 2017

*Millions of electrocardiograms (ECG) are interpreted every year, requiring specialized training for accurate interpretation. Because automated and accurate classification ECG signals will improve early diagnosis of heart condition, several neural network (NN) approaches have been proposed for classifying ECG signals. Current strategies for a critical step, the preprocessing for noise removal, are still unsatisfactory. We propose a modular NN approach based on artificial noise injection, to improve the generalization capability of the resulting model. The NN classifier initially performed a fairly accurate recognition of four types of cardiac anomalies in simulated ECG signals with minor, moderate, severe, and extreme noise, with an average accuracy of 99.2%, 95.1%, 91.4%, and 85.2% respectively. Ultimately we discriminated normal and abnormal heartbeat patterns for single lead of raw ECG signals, obtained 95.7% of overall accuracy and 99.5% of Precision. Therefore, the propose approach is a useful tool for the detection and diagnosis of cardiac abnormalities.*

*Povzetek: V članku je opisana metoda modularnih nevronskim mrež za prepoznavanje šumnih ECG signalov.*

# 1 Introduction

The electrocardiogram (ECG) is a non-invasive clinical test that measures and records electrical changes that take place in the heart when it beats [1]. ECG is vastly used for screening, diagnosis, and monitoring of several heart conditions. Most ECGs are recorded and interpreted by health professionals, few of which have received formal training and proper assessment of competency in recording and interpreting ECGs [2,3], and many self-reported their ECG reading skills as inadequate [4]. Therefore, several automated approaches have been developed to increase efficiency and enhance accuracy in interpreting ECG waveforms [5]. This is the case of classification systems based on artificial neural networks (NN), which have become very popular and most widely employed for successful classification of ECG signals [5] because of their natural ability to deal with incomplete or ambiguous input in pattern recognition tasks [6].

An ECG signal consists mainly of five continuous electromagnetic waves namely, P, Q, R, S, and T (Fig. 1). The amplitude, direction, and duration of the waves, and their morphological aspects are analyzed for specific abnormalities. Other important information includes the peak area, called the QRS complex, the duration of the PR and QT intervals, and the deviation of the PR and ST segments. These characteristics can be contaminated by the physical parameters of electronic and mechanical devices, electrical activity of muscles, degradation of the electrode-skin contact, and other causes [7-9]. Noise corruption can generate similar morphologies to the ECG waveform, reducing the discriminating power of heartbeat patterns, and increasing the rate of false alarms for cardiac monitors [9]. Therefore, a large number of NN approaches for ECG classification have included signal preprocessing for noise reduction, using a wavelet transformer (WT) [6,11-13], nonlinear cubic spline interpolation (CSI) [14], fast Fourier transformation (FFT) [15] or band-pass filters [16-19].

Nevertheless, current strategies for this critical step, the preprocessing for noise removal, are still unsatisfactory because clinical interpretation often requires even higher signal quality to detect cardiac disorders [20, 21]. For that reason, NN systems for ECG classification that are robust and efficient, and have greater noise tolerance, are needed. In this paper, we develop and test a noise-tolerant ECG signal classifier based on an NN approach. The method uses a modular NN architecture to perform initial training and testing on a simulated dataset. Ultimately we discriminated normal and abnormal heartbeat patterns for single lead of real ECG signals.

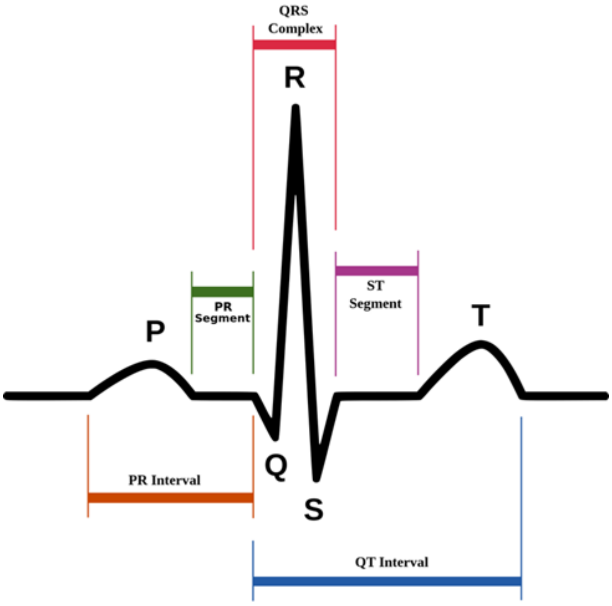


Figure 1: Typical one-cycle ECG signal.

## 2 Literature review

Most NN systems were tested using ECG data from the MIT-BIH arrhythmia database [5], which contains 48 ECG recordings with signals that were band-pass filtered in the frequency range of 0.1 to 100 Hz and sampled at 360 Hz [22]. In this sense, Rohan and Patil also used a low pass filter with a cut-off frequency of 30-100 Hz to pre-process 16 records from the MIT-BIH database [16]. They then employed an NN approach composed by two hidden layers with eight neurons and classified four types of cardiac arrhythmias with an overall accuracy of 99.9%. Asl et al. used a 5-15 Hz band-pass filter to remove contamination from 2009 ECG segments, each with 32 RR intervals [17]. The authors developed a three-layered NN with one hidden layer of 20 neurons that classified RR interval signals into four arrhythmia categories with an average accuracy of 99.3%. Das and Ari employed an NN approach with a pre-processing band-pass filter (3-20Hz) to reduce noise in 44 records of the database [18]. They classified five types of ECG beats with an S-transform NN approach and achieved 97.9% of average classification accuracy. In contrast, Tang and Shu eliminated noise from an ECG waveform using high-pass filter with 0.7Hz and low-pass filter with 100Hz [19]. Their quantum NN model recognized ECG signals with an overall accuracy of 91.7%.

In two studies the authors used WT technique to remove noise from MIT-BIH records. Javadi et al. proposed a modular NN based on a mixture of experts and negatively correlated learning using stationary WT as a tool for noise reduction [6]. The NN ensemble produced a recognition rate of 96% for classifying normal heartbeats, premature ventricular contraction arrhythmias and other cardiac abnormalities. Vijayavanan et al. preprocessed ECG signal to remove different kinds of artifacts using discrete WT [12]. They proposed a probabilistic NN approach to discriminate the difference between a normal ECG signal and an arrhythmia affected signal with an accuracy of 96.5% classification rate. Similarly, Naima and Timemy used discrete WT denoising procedure on ECG data collected from two hospitals in Bagdad [13]. Their discrete WT-NN classifier with six neurons in the hidden layer detected acute MI with 95% accuracy. On the other hand, Güler and Übeyli decomposed ECG signals from the Physiobank database [23] into time-frequency representations also using discrete WT. They classified four types of ECG beats with a total accuracy of 96.9% through a combined NN model composed by 30 hidden neurons [11]

In its place, Setizwan et al. employed the nonlinear CSI method to estimate and eliminate noise from the baseline ECG of MIT-BIH registers. The implemented fuzzy-neuro learning vector quantization algorithm produced 95.5% of the overall accuracy rate to classify normal beat and 11 types of arrhythmias [14]. Meanwhile, Vishwa et al. applied direct FFT to remove low frequencies and restore an ECG signal from the MIT-BIH arrhythmia database with the help of inverse FFT [15]. The NN model composed by three and five neurons in first and second hidden layer respectively obtained a detection accuracy of 96.7%.

In contrast, Garg and Sharma used an NN model with two hidden layers to analyze ECG records from the MIT-BIH database with no additional filter and correctly detected normal vs. arrhythmic ECGs with a general accuracy of 96.6% [24]. Finally, Jadhav et al. used records from the Cardiac Arrhythmia Database of the UCI Machine Learning Repository [25] with no prior filtering for classification of normal and abnormal ECG signals. Their NN approach with two hidden layers resulted in 82.4% correct classifications [26].

## 3 Materials and methods

The NN classification comprises five stages: (i) simulation of ECG signal, (ii) extraction of features that indicate cardiac abnormalities, (iii) computer generation of normal and abnormal heartbeat patterns, (iv) artificial noise injection, and (v) cardiac rhythm classification on simulated and real ECG signals.

## 3.1 ECG signal simulation

We used a standard electrocardiographic 12 lead representation of the heart electrical activity [27] divided in the P, PQ, QRS, ST, T and TP sections (Fig. 1), to simulate an ECG signal with specific parameters (Table 1). The resulting signal was composed of different waveforms and frequencies.

| Section | Minimum Voltage | Maximum voltage | Time |
|---|---|---|---|
| P | 0 | 0.125 mV | 95 ms |
| PQ | 0 | 0 | 40 ms |
| QRS | -0.1 mV | 0.97 mV | 65 ms |
| ST | 0 | 0 | 120 ms |
| T | 0 | 0.16 mV | 180 ms |
| TP | 0 | 0 | 130 ms |

Table 1: Parameters of typical ECG lead.

The P and T sections of the simulated ECG signal were similar to waveforms generated by the movement of a piston, which allowed generation of a mathematical model of their behavior [28]. The piston describes an oscillatory motion that can be approximated by a simple harmonic. The position equation of the piston is a function of the angular velocity: $x(t)=r\cdot\cos(\omega t)$, where $x(t)$ is the piston position versus time, $r$ is the radius of the crank and $\omega t$ is the angular velocity of rotation in radians.

The P wave modeled by the piston motion equation had maximum amplitude of 0.125 mV. The T wave was split into two sections: the first section had maximum amplitude of 0.16 mV, a positive slope, and a period of $T_1$; the second section had the same maximum amplitude, a negative slope, and a period, $T_2$, which was less than $T_1$. The QRS segment used corresponding voltages at the Q, R, and S points and intermediate voltages for the PQ-Q, Q-R, R-S, and S-ST sections. A positive off-set voltage was then added to each value. Each section was further characterized by amplitude and corresponding slope (Table 1). The final output ECG signal, simulated using MATLAB software [29, 30], had a duration of 700 ms for each cycle and was mounted on a signal base of 512 mV amplitude (Fig. 2).

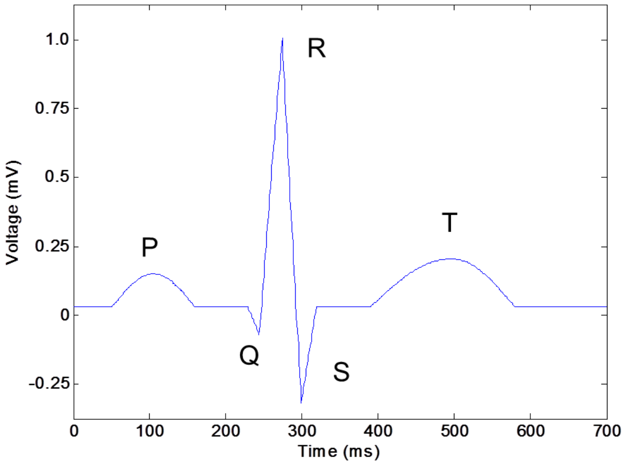


Figure 2: Typical ECG signal simulated in MATLAB.

## 3.2 Cardiac abnormalities

Feature extraction is a key issue in recognition and classification tasks. We used a combination of morphological and timing features to distinguish between a normal heartbeat (NH) and disorders of heart rate and cardiac rhythm. The shape, position, and time duration of P, Q, T waves and the ST segment, were used to identify specific abnormalities (Fig. 3).

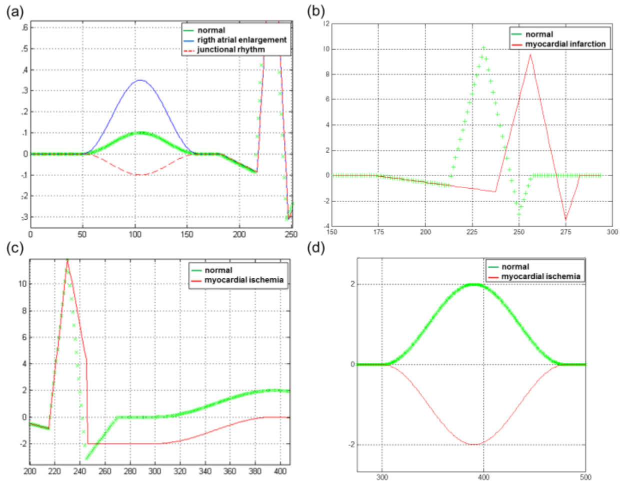


Figure 3: Morphological and timing features of (a) the P wave; (b) the Q wave; (c) the ST segment; and (d) the T wave, used to detect specific cardiac abnormalities

The P wave is the first positive deflection on ECG signal, with a normal duration <120 ms and amplitude rarely exceeding 0.25 mV. Greater amplitude suggests right atrial enlargement (RAE) [31], and an inverted P wave can indicate junctional rhythm (JR) [32]. The Q wave is the first downward deflection on ECG signal. Pathological Q waves, with a duration >40 ms or depth >0.1 mV, can be a sign of current or previous myocardial infarction (MI) [33]. Greater than 0.2 mV depression of the ST segment, which connects the QRS complex and the T wave, is attributable to cardiac ischemia (CI) [34]. Widespread inversion of the T wave, the first deflection following the QRS complex, is also associated with CI [35].

## 3.3 Dataset

After extracting the morphological and timing characteristics of the simulated ECG signal, we generated 10000 heartbeat feature vectors (normal and abnormal) for each ECG segment. The dimensions of the feature vectors for the P, Q, ST and T waves were 102, 115, 120, and 200-dimensional respectively. We then built matrices ($X_p$, $X_q$, $X_{st}$ and $X_t$) from randomly selected pattern vectors, using 900 for each matrix. These were organized as interspersed normal ($x_N^i$) and abnormal patterns ($x_a^i$). In the case of a P wave with two associated pathologies ($x_a^i$ $x_b^i$), both abnormal patterns were inserted after each NH pattern:

$$X_p = [x_N^1\, x_a^1\, x_b^1\, x_N^2\, x_a^2\, x_b^2 \ldots x_N^n\, x_a^n\, x_b^n] \qquad (1)$$

Later, we randomly combined the pattern vectors to generate a total training dataset, composed of 5400 samples from five classes. The first class was a NH; the other four classes were specific cardiac pathologies for each ECG wave. This takes into account that CI could be attributable to depression of the ST segment or widespread inversion of the T wave [34]. The total training data was partitioned into two datasets: training and testing set. The testing set was not seen by NN classifier during the training phase. It is only used for testing the generalization of NN approach after it was trained. We randomly selected the 80% examples for training, and the rest 20% examples as testing data.

To assess the robustness of the learned patterns within noisy conditions, and to improve the generalization capability of the resulting NN system, we created artificial corruption in all ECG segments of testing set (Fig. 4) using a Gaussian white-noise model [40]. We injected 1 to 12% of randomly generated Gaussian white noise [41,42]. We defined quality categories to describe the noise level: minor (1-2%), moderate (4-6%), severe (8-10%), and extreme (12%). The corrupted testing set was used during the training phase to improve the behavior of NN ensembles while they were trained within noise conditions.

Later we build a corrupted dataset for testing of trained NN with a total of 21600 contaminated ECG segments (17100 NH, 1800 CI, 900 RAE, 900 JR, and 900 MI segments). In addition, the NN approach was tested on real ECG records of the Physikalisch-Technische Bundesanstalt (PTB) Diagnostic ECG Database [36]. The PTB database contains digitized ECG signals provided by the National Metrology Institute of Germany. This ECG collection included 15 simultaneously measured signals: the conventional 12 leads together with the 3 Frank lead ECGs. Each signal is digitized at 1000 samples per second, with 16 bit resolution over a range of ± 16 mV and sampling frequency equal to 1 KHz [36].

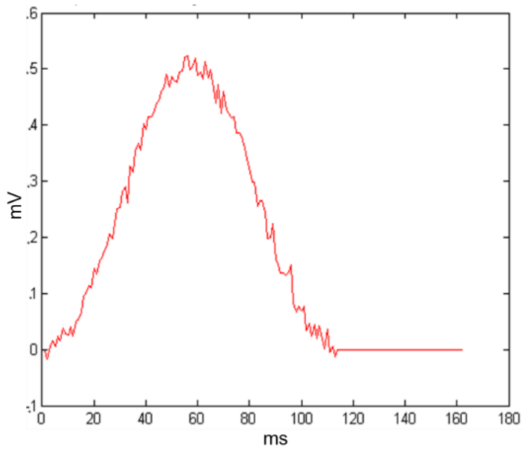


Figure 4: P wave corrupted with 6% Gaussian white noise.

We selected data from 221 subjects with a clinical summary available, which included ECG records classified as NH (n=52) or cardiac abnormalities (148 MI, 14 dysrhythmia and 7 myocardial hypertrophy). For the testing purpose, we considered an unbalanced dataset in favor of arrhythmia data (76.5%) to improve the testing generalization capabilities of the NN classifier to recognizing cardiac abnormalities. Lead V1 was chosen for the whole analysis; because it has the largest ratio of atrial to ventricular signal amplitude and therefore can offer more representative characteristics for identifying the common heart diseases [37,38]. The final test set consisted of 884 ECG traces built from 4 heartbeats per individual.

## 3.3 Neural network classifier

The modular NN classifier consisted of four, three-layered, feedforward micro NNs built through Matlab NN toolbox, one for each ECG interval analyzed. A back-propagation algorithm in batch gradient descent with momentum mode [39] and random weights/bias initialization were used for training. Transfer function group was of the hyperbolic tangent-logarithmic sigmoid-linear type for input-hidden-output layers. The learning rate and momentum coefficient were selected as 0.05 and 0.9 respectively. Performance was tested using the mean squared error parameter, computed for differences between the actual outputs and the outputs obtained in each trained micro NN. The training ended, if the total sum of the squared errors was <0.001, or when 3000 epochs were reached. The target outputs for NH, RAE, JR, MI, and CI were given by (0,0,0,0), (0,0,0,1), (0,0,1,0), (0,1,0,0), and (1,0,0,0), respectively.

# 4 Results

## 4.1 Simulated ECG dataset

When the NN system was trained using pattern vectors of clean and noised simulated ECG signals, the MSE convergence goal (0.00096) was reached in 109 epochs. The best performance was obtained using 10 (P and T) or 5 (Q and ST) neurons in the hidden layer of the micro NNs. In the first scenario of testing with an artificial corrupted dataset, correct classifications over 10 runs averaged 99.2%, 95.1%, 91.4%, and 85.2% for minor, moderate, severe, and extreme noise (Table 2) respectively. Total confusion matrix of each micro NN model for all the levels of noise is shown in Table 3.

| Noise Level | Micro NN Accuracy (%) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | | | Q | | ST | | T | |
| | NH | RAE | JR | NH | MI | NH | CI | NH | CI |
| Minor | 100 | 99 | 100 | 100 | 100 | 100 | 100 | 97 | 97 |
| Moderate | 97 | 88 | 96 | 100 | 100 | 100 | 100 | 85 | 90 |
| Severe | 96 | 85 | 86 | 100 | 100 | 99 | 100 | 78 | 79 |
| Extreme | 94 | 70 | 74 | 95 | 100 | 96 | 100 | 67 | 71 |
| Average | 96.8 | 85.5 | 89 | 98.8 | 100 | 98.8 | 100 | 81.8 | 84.3 |

Table 2: Classification performance for contaminated ECG segments.

| Estimated Output | True output | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | | | Q | | ST | | T | |
| | NH | RAE | JR | NH | MI | NH | CI | NH | CI |
| NH | 66184 | 522 | 396 | 67588 | 0 | 67598 | 0 | 55972 | 564 |
| Cardiac anomaly | 2216 | 3078 | 3204 | 812 | 3600 | 802 | 3600 | 12428 | 3036 |

Table 3: Confusion matrix for classification of artificially corrupted ECG segments.

Overall classification accuracy was 93.9%. The best performance was achieved for the micro NNs, Q and ST, correctly classifying 98.8% of NH and 100% of abnormal segments.

## 4.2 Real ECG dataset

For the last stage of the study the trained NN approach was tested directly on raw ECG traces from PTB database, exclusively for discriminating between NH and cardiac abnormality. Overall classification accuracy was 95.7%. The results are shown by a confusion matrix, where each cell contains the number of ECG traces classified for the corresponding combination of estimated and true outputs (Table 4).

| Estimated output | True output | |
|---|---|---|
| | NH | Cardiac abnormality |
| NH | 205 | 35 |
| Cardiac abnormality | 3 | 641 |

Table 4: Confusion matrix for classification of real ECG signals.

The total test performance of the NN classifier is displayed in Table 5. This was determined by the computation of evaluation metrics such as Specificity (number of correctly classified NH over total number of NH), Sensitivity (number of correctly classified cardiac abnormalities over total number of cardiac abnormalities), and Precision (number of correctly classified cardiac abnormalities over total number of estimated cardiac abnormalities).

| Evaluation metrics | Values (%) |
|---|---|
| Specificity | 98.6 |
| Sensitivity | 94.8 |
| Precision | 99.5 |
| Overall classification accuracy | 95.7 |

Table 5: Total test performance of the NN classifier.

In Table 6 the overall performance of our proposed NN classifier is compared with the recognition rate of previous NN approaches for EGC classification found in the literature.

# 5 Discussion

Overall, results of the previous studies make it clear that suppression of noise corruption embedded in analysed signals improves the accuracy of NN classifiers. However, filtering parameters, particularly cut-off frequencies and phase response characteristics, should be chosen such that clinical information in ECG signals remains undistorted, while as much noise as possible is removed [43]. This is difficult because the signal and noise often share the same frequencies. Furthermore, adaptive filters do not normally have a sharp delineation between the pass-bands and cut-bands, but rather a slow transition in the filter response. If the clinical signals and noise are close, it may not be possible to remove the noise without removing some of the clinical signal [44]. Due to filtration problems, several previous studies found that adaptive filtering affected the estimation of morphological parameters in ECG signals [45-47], resulting in changes that simulated CI [48-53]. A reduction in the number of peaks and valleys was particularly misleading [54].

| NN approach | Filtering technique | Accuracy (%) |
|---|---|---|
| *Rohan et al. [16] | 30-100 Hz | 99.9 |
| *Asl et al. [17] | 5-15 Hz | 99.3 |
| *Das et al. [18] | 3-20 Hz | 97.9 |
| Güler et al. [11] | WT | 96.9 |
| *Vishwa [15] | FFT | 96.7 |
| *Garg et al. [23] | Non | 96.6 |
| *Vijayavanan [12] | WT | 96.5 |
| *Javadi et al. [6] | WT | 96.0 |
| *Setiawan et al. [14] | CSI | 95.5 |
| Naima et al. [13] | WT | 95.0 |
| *Tang et al. [19] | 0.7-100 Hz | 91.7 |
| Jadhav et al. [26] | Non | 82.3 |
| **Method proposed** | **Non** | **95.7** |

* ECG signals that were previously band-pass filtered in the frequency range of 0.1 to 100 Hz and sampled at 360 Hz.

Table 6: Comparison of the overall classification accuracy of the proposed method and previous NN approaches in literature.

Converse to previous NN approaches for classification of ECG signals, the system proposed trains with clean and noisy data [55]. By using inputs corrupted with randomly sampled noises and various signal-to-noise ratios, we were able to build a robust classifier without an adaptive filter, because the injected noise improves the generalization capability of the NN model [56]. The rationale of this approach is that the perturbation introduced in training by the injected noise can be learned by the NN structure and recognized in the test phase. More exactly, noise injection during the training favors an optimal solution at which the objective function is less sensitive to the change of the input [57]. On the other hand, this approach is based on the premise that an NN method, which can provide accurate classification with noise, is preferable to methods that modify the original signal. Furthermore the modular design based in micro NNs provides a more specific classification for each considered kind of cardiac abnormality. In this sense, the analysis and experiments suggest that by injecting a minor to extreme level of noise in training of NN, the noise patterns can be effectively learned, and the generalization capability of the micro NNs can be improved. Both of these advantages result in substantial performance improvement of NN for ECG classification in noise conditions, without the inclusion of adaptive filters.

However, although the average classification accuracy and precision of our NN system is competitive, the system was tested for the detection of only four types of cardiac pathologies. On the other hand, the first results of the trained NN approach were achieved with artificially generated random Gaussian white noise, without any specific assumption on the origin of the noise. Moreover, the additive noise model can differ to some extent from real ECG records that are corrupted by physiological noise and exhibiting spatial correlation across the individual ECG signals [16]. These limitations require additional research for situations where the nature of the contaminating noises are better known, and the additive artificial noise may be selected according to the particular situation. Therefore, our system requires further verification including information about the noise sources using actual ECG data and classifying other specific types of cardiac disorders.

# 6 Conclusion

We developed a robust and fairly accurate, noise-tolerant NN approach for detecting and diagnosing specific cardiac abnormalities. The modular NN system discriminates between simulated normal and abnormal cardiac rhythms with high accuracy for ECG signals with minor to moderate noise and good accuracy for signals with severe to extreme noise. The previous artificial noise injection stage enables the trained NN classifier to handle noise and recognize cardiac abnormalities on raw ECG signals with high Precision. With further verification, this system could facilitate the use of NN approaches to support clinical decisions.

# 7 Acknowledgments

This paper was supported by the Secretaria de Educación Pública, México (DSA/103.5/15/11115). The authors acknowledge the support of the Programa Interinstitucional para el Fortalecimiento de la Investigación y el Posgrado del Pacífico, which promotes the formation and consolidation of research networks integrated by researchers from universities, technological institutes, and state councils of science and technology of Mexico. GE Maestre is supported partially by the National Institutes of Health and the National Institute of Aging (5 R01 AG036469-05, 1R03AG054186). We acknowledge Sarah Kaip for her editing of English language.

# Conflicts of interest

None declared.

# References

[1] Hurxthal L.M. Clinical interpretation of the electrocardiogram. *The New England Journal of Medicine*, 1934, Vol. 211, No. 10, pp. 431-437. https://doi.org/10.1056/NEJM193409062111001

[2] Wolff A.R., Long S., McComb J.M., Richley D., Mercer P. The gap between training and provision: a primary-care based ECG survey in North-East England. *The British Journal of Cardiology*, 2012, Vol. 19, No. 1, pp. 38-40. https://doi.org/10.5837/bjc.2012.008

[3] Richley D. New training and qualifications in electrocardiography. *British Journal of Cardiac Nursing*, 2013, Vol. 8, No. 1, pp. 38-42. https://doi.org/10.12968/bjca.2013.8.1.38

[4] Som S. Nurse practitioners (and other physician extenders) are not an appropriate replacement for expert physician electrocardiogram readers in routine clinical practice. *Journal of the American College of Cardiology*, 2015, Vol. 65, No. 1, pp. 106-107. https://doi.org/10.1016/j.jacc.2014.09.075

[5] Jambukia S.H., Dabhi V.K., Prajapati, H.B. Classification of ECG signals using machine learning techniques: A survey. *Proc. of IEEE International Conference on Advances in Computer Engineering and Applications*, 2015, pp. 714-721. https://doi.org/10.1109/ICACEA.2015.7164783

[6] Javadi M., Arani S.A.A.A, Sajedin A., Ebrahimpour R. Classification of ECG arrhythmia by a modular neural network based on mixture of experts and negatively correlated learning. *Biomedical Signal Processing and Control*, 2013, Vol. 8, No. 3, pp. 289-296. https://doi.org/10.1016/j.bspc.2012.10.005

[7] Zadeh A.E., Khazaee A., Ranaee V. Classification of the electrocardiogram signals using supervised classifiers and efficient features. *Computer Methods and Programs in Biomedicine*, 2010, Vol. 99, No. 2, pp. 179-194. https://doi.org/10.1016/j.cmpb.2010.04.013

[8] Micó P., Mora M., Cuesta-Frau D., Aboy M. Automatic segmentation of long-term ECG signals corrupted with broadband noise based on sample entropy. *Computer Methods and Programs in Biomedicine*, 2010, Vol. 98, No. 2, pp. 118-129. https://doi.org/10.1016/j.cmpb.2009.08.010

[9] Choudhary M., Narwaria R.P. Suppression of noise in ECG signal using low pass IIR filters. *International Journal of Electronics and Computer Science Engineering*, 2012, Vol. 1, No. 4, pp. 2238-2243.

[10] Li Q., Clifford G.D. Signal quality and data fusion for false alarm reduction in the intensive care unit. *Journal of Electrocardiology*, 2012, Vol. 45, No. 6, pp. 596-603. https://doi.org/10.1016/j.jelectrocard.2012.07.015

[11] Güler, İ., Übeyli, E.D. (2005). ECG beat classifier designed by combined neural network model. *Pattern recognition*, 2005, Vol. 38, No. 2, pp. 199-208. https://doi.org/10.1016/j.patcog.2004.06.009

[12] Vijayavanan M., Rathikarani V., Dhanalakshmi P. Automatic classification of ECG signal for heart disease diagnosis using morphological features. *International Journal of Computer Science & Engineering Technology*, 2014, Vol. 5, No. 4, pp. 449-455.

[13] Naima, F.A, Timemy A.A. Neural network based classification of myocardial infarction: a comparative study of Wavelet and Fourier

transforms. In *Pattern Recognition*, 2009, pp. 337-351. INTECH. https://doi.org/10.5772/7533

[14] Setiawan I.M.A., Imah, E.M., Jatmiko, W. Arrhytmia classification using fuzzy-neuro generalized learning vector quantization. *Proc. of IEEE International Conference on Advanced Computer Science and Information System*, 2011, pp. 385-390.

[15] Vishwa A.M.K., Lal S.D., Vardwaj P. Clasification of arrhythmic ECG data using machine learning techniques. *International Journal of Interactive Multimedia and Artificial Intelligence*, 2011, Vol. 1, No. 4 pp. 67-70. https://doi.org/10.9781/ijimai.2011.1411

[16] Rohan M.D., Patil A.J. Layered approach for ECG beat classification utilizing neural network. *Bioinformatics*, 2012, Vol. 2, No. 6, pp. 1495-1500.

[17] Asl B.M., Sharafat A.R., Setarehdan S.K. An adaptive backpropagation neural network for arrhythmia classification using RR interval signal. *Neural Network World*, 2012, Vol. 6, No. 12, pp. 535-54.

[18] Das M.K., Ari S. ECG arrhythmia recognition using artificial neural network with S-transform based effective features. *Proc. of IEEE India Conference Annual*, 2013, pp. 1-6. https://doi.org/10.1109/INDCON.2013.6726153

[19] Tang X., Shu L. Classification of electrocardiogram signals with RS and quantum neural networks. *International Journal of Multimedia and Ubiquitous Engineering*, 2014, Vol 9, No. 2, pp. 363-372.https://doi.org/10.14257/ijmue.2014.9.2.37

[20] Li Q., Rajagopalan C., Clifford G.D. A machine learning approach to multi-level ECG signal quality classification. *Computer Methods and Programs in Biomedicine*, 2014, Vol. 117, No. 3, pp. 435-447. https://doi.org/10.1016/j.cmpb.2014.09.002

[21] Redmond S.J., Xie Y., Chang D., Basilakis J., Lovell N.H. Electrocardiogram signal quality measures for unsupervised telehealth environments. *Physiological Measurement*, 2012, Vol. 33, No. 9, pp. 1517-1533. https://doi.org/10.1088/0967-3334/33/9/1517

[22] Moody G.B., Mark R.G. The impact of the MIT-BIH arrhythmia database. *IEEE Engineering in Medicine and Biology Magazine*, 2001, Vol. 20, No. 3, pp. 45-50. https://doi.org/10.1109/51.932724

[23] Goldberger A.L., Amaral L.A.N., Glass L., Hausdorff J.M., Ivanov P.Ch., Mark R.G., Mietus J.E., Moody G.B., Peng, C.K., Stanley H.E. Physiobank, Physiotoolkit, and Physionet. Components of a new research resource for complex physiologic signals. *Circulation*, 2000, Vol. 101, No. 23, pp. e215-e220. https://doi.org/10.1161/01.CIR.101.23.e215

[24] Garg P., Sharma A. Detection of normal ECG and arrhythmia using artificial neural network system. *International Journal of Engineering Research and Science & Technology*, 2015, Vol. 4, No. 1, pp. 1-13.

[25] Moses D. A survey of data mining algorithms used in cardiovascular disease diagnosis from multi-lead ECG data. *Kuwait Journal of Science*, 2015, Vol. 42, No. 2, pp. 206-235.

[26] Jadhav S., Nalbalwar S., Ghatol A. Performance evaluation of generalized feedforward neural network based ECG arrhythmia classifier. *International Journal of Computer Science Issues*, 2012, Vol. 9, No. 4, pp. 379-384

[27] Ambrose M. *ECG. Interpretación clínica*. Manual Moderno, Buenos Aires, 2008.

[28] Chierici F., Pignagnoli L., Embriaco D. Modeling of the hydroacoustic signal and tsunami wave generated by seafloor motion including a porous seabed. *Journal of Geophysical Research*, 2010. Vol. 115, No. C3, pp. 1-15. https://doi.org/10.1029/2009JC005522

[29] Islam M.K., Haque A.N., Tangim G., Ahammad T., Khondokar M.R.H. Study and analysis of ECG signal using MATLAB & LABVIEW as effective tools. *International Journal of Computer and Electric Engineering*, 2012, Vol. 4, No. 3, pp. 404-408. https://doi.org/10.7763/IJCEE.2012.V4.522

[30] Guda M., Gasser S., El Mahallawy M.S. MATLAB simulation comparison for different adaptive noise cancelation algorithms. *Proc. of the International Conference on Digital Information, Networking, and Wireless Communications*, 2014, pp. 68-73.

[31] Bille K., Figueiras D., Schamasch P., Kappenberger L., Brenner J.I., Meijboom F.J., Meijboom E.J. Sudden cardiac death in athletes: the Lausanne Recommendations. *European Journal of Cardiovascular Prevention & Rehabilitation*, 2006, Vol. 13, No. 6, pp. 859-875. https://doi.org/10.1097/01.hjr.0000238397.50341.4a

[32] Cools E., Missant C. Junctional ectopic tachycardia after congenital heart surgery. *Acta Anaesthesiologica Belgica*, 2014, Vol. 65, No. 1, pp. 1-8.

[33] Das M.K., Khan B., Jacob S., Kumar A., Mahenthiran J. Significance of a fragmented QRS complex versus a Q wave in patients with coronary artery disease. *Circulation*, 2006, Vol. 113, No. 21, pp. 2495-2501. https://doi.org/10.1161/CIRCULATIONAHA.105.595892

[34] Valo M., Moller A., Teupe C. Markers of myocardial ischemia in patients with diabetes mellitus and severe obstructive sleep apnea impact of continuous positive airway pressure therapy. *Journal of Diabetes & Metabolism*, 2015, Vol. 6, No. 492, pp. 2-5.

[35] Bakhoya V.N., Kurl S., Laukkanen J.A. T-wave inversion on electrocardiogram is related to the risk of acute coronary syndrome in the general population. *European Journal of Preventive Cardiology*, 2014, Vol. 21, No. 4, pp., 500-506.

[36] Bousseljot R., Kreiseler D., Schnabel A. Nutzung der EKG-Signaldatenbank CARDIODAT der PTB über das Internet. *Biomedizinische Technik/Biomedical Engineering*, 1995, Vol. 40, No. s1, pp. 317-318.

[37] Mainardi L., Sornmo L., Cerutti S. *Understanding atrial fibrillation: the signal processing contribution*. Morgan & Claypool Publishers, San Rafael, 2008.

[38] Ramli A.B., Ahmad P.A. Correlation analysis for abnormal ECG signal features extraction. *Proc. of 4th National Conference on Telecommunication Technology*, 2003, pp. 232-237. https://doi.org/10.1109/NCTT.2003.1188342

[39] Paulin F., Santhakumaran, A. Classification of breast cancer by comparing back propagation training algorithms. *International Journal on Computer Science and Engineering*, 2011, Vol. 3, No. 1, pp. 327-332.

[40] Szabó B.T., van der Vaart A.W., van Zanten J.H. Empirical Bayes scaling of Gaussian priors in the white noise model. *Electronic Journal of Statistics*, 2013, Vol. 7, pp. 991-1018. https://doi.org/10.1214/13-EJS798

[41] Grandvalet Y., Canu S. Comments on noise injection into inputs in back propagation learning. *IEEE Transactions Systems, Man and Cybernetics*, 1995, Vol. 25, No. 4, pp. 678-681. https://doi.org/10.1109/21.370200

[42] An G. The effects of adding noise during backpropagation training on a generalization performance. *Neural Computation*, 1996, Vol. 8, No. 3, pp. 643-674. https://doi.org/10.1162/neco.1996.8.3.643

[43] Kumar R., Gupta R., Jyoti K., Ranjan A.K. AT89C51 microcontroller based medical two channel ECG module and body temperature measurement with graphics LCD. *International Journal of Advanced Research in Computer Science and Software Engineerin*g, 2015, Vol. 5, No. 5, pp. 1321-1326.

[44] Longini R.L., Giolma J.P., Wall III C., Quick R.F. Filtering without phase shift. *IEEE Transactions on Biomedical Engineering*, 1975, Vol. BME-22, No. 5, pp. 432-433. https://doi.org/10.1109/TBME.1975.324517

[45] Valverde E.R., Quinteiro R.A., Bertran G.C., Arini P.D., Glenny P., Biagetti M.O. Influence of filtering techniques on the time domain analysis of signal-averaged P wave electrocardiogram. *Journal of Cardiovascular Electrophysiology*, 1998, Vol. 9, No. 3, pp. 253-260. https://doi.org/10.1111/j.1540-8167.1998.tb00910.x

[46] Łęski J.M., Henzel N. ECG baseline wander and powerline interference reduction using nonlinear filter bank. *Signal Processing*, 2005, Vol. 85, No. 4, pp. 781-793. https://doi.org/10.1016/j.sigpro.2004.12.001

[47] Gregg R.E., Zhou S.H., Lindauer J.M., Helfenbein E.D., Giuliano K.K. What is inside the electrocardiograph?. *Journal of Electrocardiology*, 2008, Vol. 41, No. 1, pp. 8-14. https://doi.org/10.1016/j.jelectrocard.2007.08.059

[48] Berson A.S., Pipberger H.V. The low-frequency response of electrocardiographs, a frequent source of recording errors. *American Heart Journal*, 1966, Vol. 71, No. 6, pp. 779-789. https://doi.org/10.1016/0002-8703(66)90599-0

[49] Bragg-Remschel D.A., Anderson C.M., Winkle R.A. Frequency response characteristics of ambulatory ECG monitoring systems and their implications for ST segment analysis. *American Heart Journal*, 1982, Vol. 103, No. 1, pp. 20-31. https://doi.org/10.1016/0002-8703(82)90524-5

[50] Tayler D., Vincent R. Signal distortion in the electrocardiogram due to inadequate phase response. *IEEE Transactions on Biomedical Engineering*, 1983, Vol. BME-30, No. 6, pp. 352-356. https://doi.org/10.1109/TBME.1983.325137

[51] Tayler D.I, Vincent R. Artefactual ST segment abnormalities due to electrocardiograph design. *British Heart Journal*, 1985, Vol. 54, No. 2, pp. 121-128. https://doi.org/10.1136/hrt.54.2.121

[52] Burri H., Sunthorn H., Shah D. Simulation of anteroseptal myocardial infarction by electrocardiographic filters. *Journal of Electrocardiology*, 2006, Vol. 39, No. 3, pp. 253-258. https://doi.org/10.1016/j.jelectrocard.2005.11.001

[53] Censi F., Calcagnini G., Triventi M., Mattei E., Bartolini P., Corazza I., Boriani G. Effect of high-pass filtering on ECG signal on the analysis of patients prone to atrial fibrillation. *Annali dell'Istituto Superiore di Sanità*, 2009, Vol. 45, No. 4, pp. 427-431. https://doi.org/10.1590/S0021-25712009000400012

[54] Buendía-Fuentes F., Arnau-Vives M.A., Arnau-Vives A., Jiménez-Jiménez Y., Rueda-Soriano J., Zorio-Grima E., Osa-Sáez A., Martínez-Dolz L.V., Almenar-Bonet L., Palencia-Pérez M.A. High-bandpass filters in electrocardiography: source of error in the interpretation of the ST segment. *International Scholarly Research Notices Cardiology*, 2012, Vol. 2012, pp. 1-10. https://doi.org/10.5402/2012/706217

[55] Li B., Tsao Y., Sim K.C. An investigation of spectral restoration algorithms for deep neural networks based noise robust speech recognition. *Proc. of 14th Annual Conference of the International Speech Communication Association*, 2013, pp. 3002-3006.

[56] Yin S., Liu C., Zhang Z., Lin Y., Wang D., Tejedor J., Fang-Zhen T., Li Y. Noisy training for deep neural networks in speech recognition. *EURASIP Journal on Audio, Speech, and Music Processing*, 2015, Vol. 2015, No. 1, pp. 1-14. https://doi.org/10.1186/s13636-014-0047-0

[57] Grandvalet Y., Canu S Comments on “Noise injection into inputs in back propagation learning”. *IEEE Transactions on Systems, Man, and Cybernetics*, 1995, Vol. 25, No. 4, pp. 678–681. https://doi.org/10.1109/21.370200

# Facial Expression Recognition Based on Local Features and Monogenic Binary Coding

Zhangbao Chen
School of Electronic and Electrical Engineering, Bengbu University, Bengbu, Anhui 233030, China
Corresponding address: No. 1866, Caoshan Road, Bengbu, Anhui 233030, China
E-mail: zbczbao@126.com

**Keywords:** monogenic binary coding, facial expression recognition, local features, Japanese female facial expression

**Received:** March 11, 2019

*Fast developing facial expression reciognition is one of the significant recognition technologies for biological features with high applied value. In this study, a monogenic binary coding algorithm was considered to illustrate a good matching with local features through the analysis of monogenic signal theory and monogenic binary algorithm. Then, the results of the facial expression recognition simulation experiment of monogenic based classical facial expression database, the Japanese Female Facial Expression (JAFFE) Database, and the results of traditional Local Binary Patterns-Sparse Representation-based Classification (LBP-SRC) residual fusion method were compared to illustrate the efficiency of the monogenic binary coding algorithm in the aspect of facial recognition and to provide a basis for the application of the monogenic signal theory in facial expression.*

*Povzetek: Opisana je nova varianta algoritma za hitro prepoznavanje izrazov na obrazu .*

## 1 Introduction

Language is the most common way to convey information in daily life. However, the study shows that facial expressions contain more information than languages do and some cannot be easily recognized by humans themselves [1]. Thus, using computers to analyze facial expressions and to recognize important information has become a critical research topic in computer realm and the relative results have obtained a remarkable progress in the aspects of mental illness treatment, polygraph detection, distance education, human-computer interaction and so on. The rough operation flow of facial expression recognition is face detection and preprocessing, facial feature extraction, and facial expression classification [2], where face detection and preprocessing refer to the separation of the target face from the image; facial feature extraction refers to the extraction of some facial features that can reflect human emotions from the face, and facial expression classification is a classification of the extracted facial features as a specific expression. In the aspects of the extraction of facial features, there exist many mature technologies, including the extraction methods based on geometric features, overall statistical features, characteristic frequency rate, and motion features. Mollahosseini et al. [3] proposed a method to recognize facial expression with deep neural networks with good performance in blurred images. Dapogny et al. [4] proposed a method of facial recognition based on pairwise conditional random forests and its improvement in recognition was remarkable by learning random forests from different derivative characteristics in paired image. Lee et al. [5] proposed a new method of facial expression recognition based on sparse representation which could reduce the variance within class and emphasize to inquire the facial expressions in face images at the same time. The monogenic binary coding algorithm is relatively more widely used among many facial expression recognition technologies, and many improved methods with high recognition rate have been derived over time. Yang et al. [6] proposed a monogenic binary pattern that combined monogenic orientations and monogenic amplitudes to apply on facial recognition realm, which obtained good results. Zhang et al. [7] proposed a monogenic binary algorithm based on local features which was relatively easier in calculation and improved the recognition accuracy further. In this study, facial expression recognition algorithm based on local features and monogenic binary coding was analyzed to build the facial expression recognition model through the deep understanding of the monogenic signal theory and the monogenic binary coding and the accuracy rate was examined by Japanese Female Facial Expression (JAFFE) Database, the classic facial expression database. Some suggestions about the methods of facial expression recognition were provided through the illustration of the wonderful features of monogenic binary coding in facial expression recognition realm, which was shown by the comparison of accuracy rate with the traditional facial recognition method.

## 2 Monogenic signal theory

Hilbert transform is one of the methods to describe signal analysis and complex envelope in the field of mathematics and signal processing and it can be roughly described as the convolution of a signal $s(t)$ with $\frac{1}{\pi t}$ to get the signal $s'(t)$ [8].

The formula of Hilbert transform is:

$$s'(t)=\frac{1}{\pi t}*s(t)=\frac{1}{\pi}\int_{-\infty}^{+\infty}\frac{s(\tau)}{t-\tau}d\tau \quad . \qquad (1)$$

Hilbert transform keeps the amplitude of each frequency component of the target signal and the phase exhibits a $90^{\circ}$ shift.

Riesz transform is a two-dimensional extension of the Hilbert transform [9]. The monogenic signal based on Riesz transform can convert the one-dimensional signal into two-dimensional signal. The transformation form of Riesz transform in the spatial domain is:

$$f_R(z)=\begin{pmatrix} f_x(z) \\ f_y(z) \end{pmatrix}=\begin{pmatrix} \frac{x}{2\pi\|z\|^3}*f(z) \\ \frac{y}{2\pi\|z\|^3}*f(z) \end{pmatrix}, \qquad (2)$$

where the convolution kernel of Risez transform is $\left(\frac{x}{2\pi\|z\|^3},\frac{y}{2\pi\|z\|^3}\right)$, and * is convolution operator. The two-dimensional frequency response of the Riesz filter is $F_x=-\frac{jx}{\|x\|},F_y=-\frac{jy}{\|y\|}$. The monogenic signal of a picture can be defined as the combination of the signal $f(z)$ and Risez transform. The combination mode is:

$$f_S(z)=f(z)+(i,j)f_R(z)=f(z)+if_x(z)+jf_y(z)\,. \qquad (3)$$

The monogenic signal based on its features similarly as real and imaginary components can be orthogonally decomposed into local amplitude $A$, local phase $\varphi$, and local orientation $\theta$. The three mentioned above relatively describe the local energy information, the local structure information and the geometric orientation information of the monogenic signal. The calculation method is:

$$\begin{cases} A=\sqrt{f^2+f_x^2+f_y^2} \\ \varphi=-\operatorname{sgn}(f_x)\arctan(\frac{\sqrt{f_x+f_y}}{f}) \\ \theta=\arctan(\frac{f_y}{f_x}) \end{cases} . \qquad (4)$$

## 3 Monogenic binary coding

It is necessary to analyze from several perspectives with different indexes to better describe the detailed information of an image. According to monogenic signal theory, an image can be extracted three specific pieces of information, including amplitude $A$, phase $\varphi$, and orientation $\theta$, then the local features of the image in details is described [10].

The feature images which have been decomposed can be named as monogenic binary coding amplitude image $JPG-A$, monogenic binary coding phase image $JPG-\varphi$, and monogenic binary coding orientation image $JPG-\theta$. The coding images which have been decomposed are all composed of two parts, intensity coding of monogenic local imaginary part and monogenic local variable coding. The coding formula after the combination is:

$$JPG-N(z_c)=\left[C_x'(z_c),C_y'(z_c),C_N(z_c)\right], N=A,\varphi,\theta \quad , \qquad (5)$$

where $z_c$ refers to the central pixel of a selected area.

### 3.1 Intensity binary coding of monogenic local imaginary part

Since the Riesz transform is unsymmetrical in the monogenic signal, the features of the real part are suppressed, the representations of imaginary part are obvious, the decomposition effects are good, and the robustness is strong to factors such as illumination changes. In the image, the imaginary part after the monogenic binary coding can represent the intensity of local feature information of the central pixel. As for a pixel $z_c$, its intensity coding of monogenic local imaginary part is $\left[C_x'(z_c),C_y'(z_c)\right]$, and the coding rule is:

$$C_n'(z_c)=\begin{cases} 0, & f_n(z_c)>0 \\ 1, & f_n(z_c)\le 0 \end{cases}, n=x,y, \qquad (6)$$

where $f_x$, $f_y$ are the horizontal and vertical output of the imaginary part of Riesz transform represented by monogenic signal. The quadrant was expressed in the coordinate system in Figure 1.

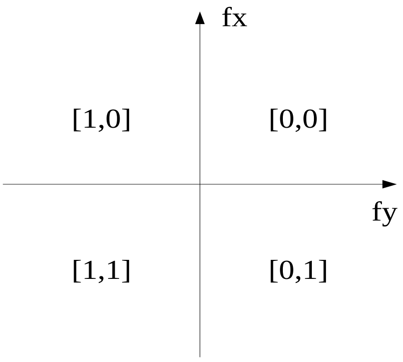


Figure 1: Quadrant coding of features of monogenic imaginary part in each place

### 3.2 Monogenic local variable binary coding

Local binary pattern (LBP) coding can be used in monogenic amplitude, local XOR pattern (LXP) can be used in monogenic orientation and monogenic phase and the robustness of codes can be strengthened through blocking by the angles when generating the specific codes.

Local binary coding of monogenic amplitude usually uses round LBP operators. The working principle is to measure the local energy condition by comparing the amplitudes of the surrounding pixels and the central pixel. $z_c$ is set as the central pixel and there are P proximal points in the surrounding area. The calculation method of amplitude binary coding of $z_c$ is as follows:

$$C_A(z_c) = \sum_{i=0}^{P-1} s(A(i) - A(z_c)) \times 2^i , \tag{7}$$

where $A(i)$ is the amplitude value of the $i$-th proximal point, $A(z_c)$ is the proximal value of central pixel, and $2^i$ is the factor that converts binary system to decimal system. The function $s(x)$ is a sign function and its formula is as follows:

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} . \tag{8}$$

The XOR pattern coding of monogenic orientation and monogenic phase. Local orientation describes the major information of local structure, and local phase describes the texture information of the image. Both of them can be represented with the angles from 0 to 360, however, the special points will be too dispersed and unstable to extract local features, which refers to relatively worse robustness, in the situation where the groups are too much and trivial. Due to the strengthening of robustness of coding, it is necessary to divide those angles into several intervals to generate aggregation effect and highlight the features and steadiness of the centralized region.

In this study, the region, $\lfloor 0,\quad 360)$, was divided into 6 intervals. Different phases and orientation would be distributed to corresponding regions through categories and it was defaulted that the phases and orientation in the same region represented the same local features. The partition function $D(x)$ is shown as follows:

$$D(x) = \begin{cases} 1 & 0 \leq x < 60 \\ 2 & 60 \leq x < 120 \\ 3 & 120 \leq x < 180 \\ 4 & 180 \leq x < 240 \\ 5 & 240 \leq x < 300 \\ 6 & 300 \leq x \leq 360 \end{cases} . \tag{9}$$

The method of XOR pattern coding of central pixel $z_c$ with N proximal points is as follows:

$$C_M(z_c) = \sum_{i=0}^{N-1} C_i^M \times 2^i , M = \varphi, \theta , \tag{10}$$

where $C_i^M$ is defined as follows:

$$C_i^{\varphi}(z_c) = \begin{cases} 0, & D(\varphi(z_c)) = D(\varphi(i)) \\ 1, & D(\varphi(z_c)) \neq D(\varphi(i)) \end{cases} , \tag{11}$$

$$C_i^{\theta}(z_c) = \begin{cases} 0, & D(\theta(z_c)) = D(\theta(i)) \\ 1, & D(\theta(z_c)) \neq D(\theta(i)) \end{cases} , \tag{12}$$

where $C_i^{\varphi}(z_c)$ is the binary coding of local phase, $C_i^{\theta}(z_c)$ is the binary coding of local orientation, $\varphi(z_c)$ is the local phase of the center pixel $z_c$, $\varphi(i)$ is the phase of the $i$-th proximal point, $\theta(z_c)$ is the local orientation of the center pixel $z_c$, and $\theta(i)$ is the orientation of the $i$-th proximal point.

# 4 Facial expression recognition simulation experiment based on local features and monogenic binary coding

## 4.1 Experimental preparation

The experiment was run on a desktop computer with a Windows 10 32-bit system, quad-core 3.30 GHz, and 4 GB memory.

This experiment is a non-specific facial expression experiment, using JAFFE Database which is a classic facial expression database. The database consists of 7 expressions, each with 3 to 4 sample images taken from 10 Japanese women, and there were 213 facial expression images. The seven types of expressions are sadness, joy, anger, nausea, surprise, fear and neutral expression [11].

## 4.2 Experimental procedure

Test set and training set were selected firstly. The experiment was divided into 10 rounds. In every round, 9 out of 10 females and about 190 facial expressions would be selected as the training set, and the remaining one about 20 facial expressions was as the test set. After the end of the 10th round, the average value of the recognition rates was calculated as the result of this experiment. The extracted expressions of a woman in the library are in Figure 2.

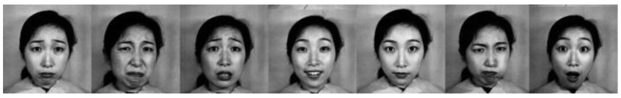


Figure 2: Seven expression examples in JAFFE

In order to ensure the validity of the recognition, it was firstly necessary to perform preprocessing. A uniform cropping template was taken to remove redundant information such as hair and neck, and then the processed

Figure 3: The example of preprocessed results.

images were obtained after being equalized and scaled. The examples are shown in Figure 3.

The feature graphs of monogenic binary coding of each image could be obtained with the method introduced in this study and the feature graphs $JPG-A$, $JPG-\varphi$, and $JPG-\theta$ could be generated correspondingly with amplitude $A$, phase $\varphi$, and orientation $\theta$. Eight proximal points in the proximal region could be obtained with the radius of 2 in the LBP calculation. The coding values of LBP coding and LXP coding were both in the range from 0 to 255; the monogenic local intensity coding was non-successive, from 0 to 1024.

A histogram was established with the three features to further construct the feature histogram of the partial block. The images were divided into 15 non-overlapping blocks provided that the faces were standardized. Then every block were divided again into $3\times3$ sub-blocks. The feature graphs of the sub-blocks in the three dimensions were put in series to obtain the histogram of the sub-blocks. The feature histograms of every partial block could be obtained as a proof after being connected.

Fisher linear discriminant analysis was used to reduce dimension of the training sets block by block [12]. The original image had a large amount of information and a large number of dimensions. If the identification was directly recognized, a "dimension disaster" might occur, thus it was necessary to perform dimension reduction at the beginning. Firstly, the linear discriminant analysis was used to learn the feature histogram matrix of each local block, and the best projection vector was obtained. Then, it was convolved with the histogram to obtain the reduced-dimensional histogram.

The cosine distance method was used to calculate the similarity between the test set and the training set in each local block, and the recognition was accumulated at the final [13]. The facial recognition of monogenic binary coding after fusion was supposed as $JPG-Com$, and the results of weighting fusion is as follows:

$$S_{JPG-Com} = 0.27S_{JPG-A} + 0.27S_{JPG-\varphi} + 0.46S_{JPG-\theta} . \quad (13)$$

The LBP-SRC residual fusion algorithm with high recognition rate in traditional face recognition technology was selected [14], the results are modeled and calculated in the method described in this study, and the calculation results were obtained after being compared with the monogenic binary coding method.

## 4.3 Experimental results and analysis

The comparison results of the recognition rates are showed in Table 1.

| Type of recognition algorithm | LBP-SRC residual fusion | $JPG-A$ | $JPG-\varphi$ | $JPG-\theta$ | $JPG-Com$ |
|---|---|---|---|---|---|
| The success rate of recognition /% | 71.15 | 66.37 | 68.92 | 70.38 | 78.71 |

Table 1: Comparison results of algorithm recognition rate.

It could be easily seen that the traditional LBP-SRC residual fusion method had a relatively good recognition rate, which reached 71.15%; although the three separate monogenic binary coding calculation methods also had relatively higher facial expression recognition rate, their rates were lower than those of the traditional LBP-SRC residual fusion method.

The algorithm based on local features and monogenic binary coding integrated the recognition characteristics of the three features, and reduced the volatility and limitation of recognition. Therefore, the highest facial expression recognition rate obtained was 78.71%.

The corresponding facial expression recognition rate based on different expression categories is shown in the Figure 4.

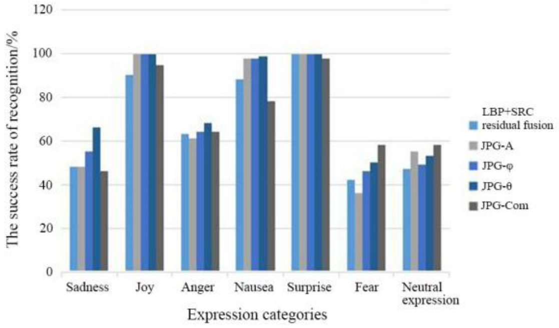


Figure 4: Comparison of five algorithms in the recognition rates of seven categories of expressions.

It could be seen that the recognition rates of the five modes were different in different types of expressions. The total recognition rate seemed to be higher in the expressions of joy, nausea, and surprise. The most easily recognized expression was surprise. Sadness, anger, fear, and neutral expression were more difficult to recognize because of the small degree of facial muscle changes. The most difficult expression to recognize was fear.

After comparing the recognition rates of different algorithms in the same expressions, it was known that the recognition rates of $JPG-Com$ was higher than those of $JPG-A$, $JPG-\varphi$, and $JPG-\theta$ except the neutral expression, and the highest recognition rate of $JPG-Com$ was in the three expressions of joy, nausea, and surprise. After comparing of $JPG-Com$ and LBP-SRC residual fusion, it was known that the recognition rates of $JPG-Com$ in the expressions of sadness, joy, anger, nausea, and surprise were higher than those of LBP-SRC residual fusion, but the recognition rates in the expressions of fear and neutral expression were lower than those of LBP-SRC residual fusion.

Based on the above research and considerations, it could be said that the algorithm based on local features and monogenic binary coding was a more effective facial expression recognition method with higher recognition efficiency.

# 5 Conclusion

Nowadays, facial recognition technology is widely used. In this study, facial expression recognition based on local features and monogenic binary coding algorithm was analyzed to illustrate the principles of monogenic signal theory and monogenic binary coding, and then the corresponding facial expression recognition model of the monogenic binary coding algorithm was constructed according to the theory, which was then applied to the JAFFE Database samples. Compared to the traditional LBP-SRC residual fusion method, the effectiveness of the monogenic binary coding algorithm in the face recognition field was shown, providing some references for related research.

# References

[1] Kim JG, Ko JG, Lee SJ (2016) Demo:MilliCat: Real-Time Autonomous Image Suggestion for Mobile Messaging. *International Conference on Mobile Systems*. https://doi.org/10.1145/2938559.2938585.

[2] Kim BK, Roh J, Dong SY, Lee SY. (2016) Hierarchical committee of deep convolutional neural networks for robust facial expression recognition. *Journal on Multimodal User Interfaces*, 10(2), pp. 173-189. https://doi.org/10.1007/s12193-015-0209-0.

[3] Mollahosseini A, Chan D, Mahoor MH (2016) Going deeper in facial expression recognition using deep neural networks. *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, Lake Placid, NY, USA. https://doi.org/10.1109/WACV.2016.7477450.

[4] Dapogny A, Bailly K, Dubuisson S (2016) Pairwise Conditional Random Forests for Facial Expression Recognition. *IEEE International Conference on Computer Vision*, IEEE, Santiago, Chile. https://doi.org/10.1109/ICCV.2015.431.

[5] Lee SH, Ro YM (2017) Intra-Class Variation Reduction Using Training Expression Images for Sparse Representation Based Facial Expression Recognition. *IEEE Transactions on Affective Computing*, 5(3), pp. 340-351. https://doi.org/10.1109/TAFFC.2014.2346515.

[6] Yang M, Zhang L, Zhang L, Zhang D (2010) Monogenic Binary Pattern (MBP): A Novel Feature Extraction and Representation Model for Face Recognition. *International Conference on Pattern Recognition*, IEEE, Istanbul, Turkey. https://doi.org/10.1109/ICPR.2010.657.

[7] Yang M, Zhang L, Shiu CK, et al. (2012) Monogenic Binary Coding: An Efficient Local Feature Extraction Approach to Face Recognition. *IEEE Transactions on Information Forensics and Security*, 7(6), pp. 1738-1751. https://doi.org/10.1109/TIFS.2012.2217332.

[8] Culiuc A, Kesler R, Lacey MT (2016) Sparse Bounds for the Discrete Cubic Hilbert Transform. https://doi.org/10.2140/apde.2019.12.1259.

[9] Lunyov AA, Malamud MM (2016) On the Riesz basis property of root vectors system for 2 × 2 Dirac type operators. *Journal of Mathematical Analysis & Applications*, 441(1), pp. 57-103. https://doi.org/10.1016/j.jmaa.2016.03.085.

[10] Wang Z, Sun X, Sun L (2016) Face Recognition Using Monogenic Signal Feature and Discriminative Manifold Preserving Embedding. *Journal of Computational and Theoretical Nanoscience*, 13(4):2537-2550. https://doi.org/10.1166/jctn.2016.4615.

[11] Saaidia M, Zermi N, Ramdani M (2016) Multiple Image Characterization Techniques for Enhanced Facial Expression Recognition. *Intelligent Systems Technologies and Applications*. Springer International Publishing. https://doi.org/10.1007/978-3-319-23036-8_43.

[12] Mahdianpari M, Salehi B, Mohammadimanesh F, Brisco B, Mahdavi S, Amani M, Granger JE. (2018) Fisher Linear Discriminant Analysis of coherency matrix for wetland classification using PolSAR imagery. *Remote Sensing of Environment*, 206, pp. 300-317. https://doi.org/10.1016/j.rse.2017.11.005.

[13] Wang X, Yang S, Zhao Y, Wang Y (2018) Lithology identification using an optimized KNN clustering method based on entropy-weighed cosine distance in Mesozoic strata of Gaoqing field, Jiyang depression. *Journal of Petroleum Science and Engineering*, 166, pp. 157-174. https://doi.org/10.1016/j.petrol.2018.03.034.

[14] Ameur B, Masmoudi S, Derbel AG, Hamida AB (2016) Fusing Gabor and LBP feature sets for KNN and SRC-based face recognition. *International Conference on Advanced Technologies for Signal & Image Processing*, Monastir, Tunisia. https://doi.org/10.1109/ATSIP.2016.7523134.

# Application of the Support Vector Machine Algorithm based Gesture Recognition in Human-computer Interaction

Wangcheng Cao
School of Computer and Information Technology, Mudanjiang Normal University
Mudanjiang, Heilongjiang, 157011, China
Corresponding address: Mudanjiang Normal University
No.191, Cultural Street, Aimin District, Mudanjiang, Heilongjiang 157011, China
E-mail: wangchengc80@126.com

**Keywords:** support, human-computer interaction, gesture recognition, image segmentation

**Received:** November 29, 2018

*A gesture recognition technology is an important part of the human-computer interaction. This study focuses on the application of the support vector machine (SVM) in gesture recognition. The gesture image is segmented by YCgCr color space based skin color segmentation method. Then four Hu invariant moments and the ratio of area to circumference of gesture are taken as eigenvalues to extract gesture features. Finally, SVM is used for recognition. It was discovered that the proposed method has a good performance in the gesture recognition and can segment the collected images accurately. The recognition rate of Hu invariant moments based SVM algorithm reaches 99.2% in the recognition of the six gestures designed in this study, which is 9.2% higher than that of the HMM algorithm. The proposed method is reliable and feasible and can contribute to the simple man-machine interaction.*

*Povzetek: Opisana je aplikacija algoritma SVM za prepoznavanje gest pri komunikaciji z računalnikom.*

## 1 Introduction

With the development of science and technology, human-computer interaction has gradually become a part of people's lives [1]. The human-computer interaction technology has been constantly developed [2]. Gesture recognition is an important part of human-computer interaction [4] and plays an indispensable role in people's daily life, and it has a wide application in computer games, virtual reality, medical care and other areas [5,6]. Kuang et al. [7] used the zed stereo camera to get the gesture depth image, segmented the gesture image by the depth information and color information detection, carried out the fingertip detection, and recognized the five kinds of digital gestures by support vector machine (SVM). The average recognition rate was 94.9%, which indicated the high validity of the method. Huang et al. [8] proposed a Gabor filter and SVM based hand gesture recognition method which eliminated the limitation of illumination conditions and obtained a recognition rate of 96.1% in the experiment. Moreover it was found that the use of Gabor filter improved the recognition accuracy from 72.8% to 93.7%, which suggested the high feasibility of the method. Li et al. [9] designed a gesture recognition system, used previous facial knowledge based adaptive skin region segmentation algorithm to segment gesture, and then used SVM to recognize gesture. The experimental results showed that the gesture recognition method had a recognition rate of 95.88%, indicating that the method had excellent performance in gesture recognition and could be applied in real life. Nagarajan et al. [10] proposed a gesture recognition system based on edge histogram features and multi-class SVM to recognize American Sign Language (ASL) and found that the recognition rate of this method was 93.75%. In the present study, the YCgCr color space based skin color segmentation method was used to segment gesture image. Then the gesture image was recognized by Hu invariant moments based SVM algorithm to study the recognition effect of the method.

## 2 Gesture recognition technology

Man-human interaction technology realizes the rapid communication between human and machine, which brings huge convenience to people's lives. Gesture is one of the daily communication means. Recognizing gesture can help computer understand the behavior act of humans and bring an intuitive experience to users, which is a natural human-computer communication means [11]. Gesture recognition is based on a data glove or vision. In data glove based gesture recognition, relevant information of hands are obtained through data glove, and then the collected data are recognized using computer. It is high-efficient, but is high-cost and complex; hence it is difficult to be promoted. Vision-based gesture recognition is to collect hand images by camera and then recognize them by image analysis. It is practical and has been widely studied. It has been widely used in many fields, such as sign language recognition, somatosensory games and smart home.

In the gesture recognition, the acquired image is segmented firstly to obtain gesture image, and the features of the gesture image are extracted. Then, the gesture recognition algorithm was used for recognizing the gesture image. A complete gesture recognition system is shown in Figure 1.

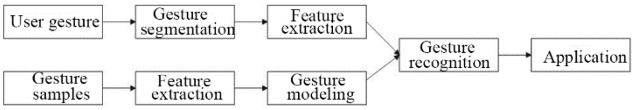


Figure 1: Gesture recognition system.

## 3 YCgCr color space based gesture segmentation method

To accurately recognize gesture, the gesture needs to be separated from the gesture video. The commonly used gesture segmentation methods included skin color segmentation method, background differencing method and pattern matching method. In this study, YCgCr color space based skin color segmentation method was used to segment gesture image.

YCgCr color space has many advantages in gesture segmentation. It is seldom affected by illumination. Y channel can represent the brightness information of the image. Gray image can be extracted directly on Y channel. Cg and Cr components can effectively identify skin color and non-skin color regions.

Fixed threshold value was used to detect skin color. When the pixel value of an image satisfied the ondition $Y \in [35,230], Cg \in [80,127], Cr \in [133,173]$, it was recognized as a skin color pixel and reset to 255; otherwise it was recognized as a non-skin color pixel and reset to 0. Thus a binary image containing noise was obtained. Then the segmentation process was carried out using this image.

(1) The skin color regions with the area smaller than 400 pixels were eliminated.

(2) The skin color regions with the width and height smaller than 20 pixels were removed.

(3) The center of gravity $(x_z, y_z)$ of the remaining skin color region was calculated.

$$x_z = \frac{m_{10}}{m_{00}}, y_z = \frac{m_{01}}{m_{00}} , \quad (1)$$

where

$$m_{10} = \sum_{x=1}^{w}\sum_{y=1}^{h} xf(x,y), m_{01} = \sum_{x=1}^{w}\sum_{y=1}^{h} yf(x,y), m_{00} = \sum_{x=1}^{w}\sum_{y=1}^{h} f(x,y) . \quad (2)$$

(4) The ratio of height (H) to width (W) of the skin color region was calculated and defined as $\sigma$,

$$0.7 \le \sigma = \frac{H}{W} \le 3.0 .$$

## 4 Hu invariant moment based gesture feature extraction

To improve recognition effect, feature extraction was performed on the segmented gesture binary image, i.e., selecting the features which could represent gesture as the feature vector. Features extracted included feature of normalized moment of inertia (NMI), the Fourier descriptor and geometrical characteristic [12]. Hu invariant moment was selected to extract feature of the gesture image.

Hu moments are invariant to target translation and rotation. Hu invariant moment theory include seven t moments defined as:

$$\phi_1 = \eta_{20} + \eta_{02} , \quad (3)$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 - 4\eta_{11}^2 , \quad (4)$$

$$\phi_3 = (\eta_{30} - 3\eta_{21})^2 + (3\eta_{21} + \eta_{03})^2 , \quad (5)$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 , \quad (6)$$

$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} - \eta_{12}) + \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + , \quad (7)$$
$$(3\eta_{21} - \eta_{03}) \times (\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]$$

$$\phi_6 = (\eta_{20} - \eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] + 4\eta_{11}(\eta_{21} + \eta_{03}) , \quad (8)$$

$$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12}) + \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + . \quad (9)$$
$$(3\eta_{12} - \eta_{30}) \times (\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]$$

Due to calculation complexity of high-order moments, $\phi_1 - \phi_4$ were selected as features. Moreover the ratio of area to circumference of the gesture image was calculated and also taken as the feature parameter.

Circumference (L) refers to the sum of pixels on the border line:

$$L = \sum\sum h(x,y) ,$$

where

$$h(x,y) = \begin{cases} 1, when\ point(x,y) is\ the contour\ point of\ gesture \\ 0, when\ point(x,y) is\ the non-contour\ point of\ gesture \end{cases} .(10$$

)

Area (S) refers to the sum of pixels in the hand region in the image:

$S = \sum\sum f(x,y)$, where

$$f(x,y) = \begin{cases} 1, when\ point(x,y) is\ the\ gesturearea \\ 0, when\ point(x,y) is\ the non-gesturearea \end{cases} . \quad (11)$$

The ratio of the area to the circumference of the eighth characteristic parameter is $A = \frac{S}{L}$.

## 5 Gesture recognition algorithm

The commonly used gesture recognition algorithms include dynamic warping algorithm [13], Hidden Markov Model (HMM) and neural network. In recent years, SVM has been frequently used in gesture recognition [14]. In this study, SVM was selected as the gesture recognition algorithm.

## 5.1 SVM algorithm

Suppose there was a sample set $\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$, where $x_i \in X, y_i \in Y=\{1,-1\}$. If there was a hyperplane $(w\cdot x)+b_o=0$, then linear discriminant function was $g(x)=w\cdot x+b$, where w stands for weight vector and b is a constant. Then class interval was:

$$d(w,b)=\min_{\{x_i|y_i=1\}}\frac{w\cdot x_i+b}{|w|}-\max_{\{x_i|y_i=-1\}}\frac{w\cdot x_i+b}{|w|}=\frac{1}{|w|}-\frac{-1}{|w|}=\frac{2}{|w|}. \quad (12)$$

If the condition $y_i[(w\cdot x_i)+b]\geq 0, i=1,2,\cdots,N$ was satisfied and the class interval was the largest, then this was the optimal hyperplane.

Linearly separable SVM could be rewritten as an optimization problem.

$$\begin{cases} \min \quad \dfrac{\|w\|}{2} \\ st. \quad (y_i(w\cdot x_i)+b)-1\geq 0, i=1,2,\cdots,N \end{cases}, \quad (13)$$

where $w$ and $b\in R, X_n \in R^M$ are eigenvectors and $y_n \in(-1,1)$ represents the affiliated category value. Lagrangian multiplier was used for solution. Then the problem could be written as:

$$w(\alpha)=\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j(x_i\cdot x_j)-\sum_{i=1}^{n}\alpha_i \quad (14)$$

where $\alpha_i$ stands for Lagrangian multiplier .

The final classification function was:

$$f(x)=\text{sgn}(\sum_{i=1}^{n}\alpha_i y_i(x_i\cdot x)+b) \quad (15)$$

If it was linearly inseparable, slack variable $\zeta$ needed to be introduced. The objective function was:

$$\begin{cases} \min \quad \dfrac{\|w\|^2}{2}+C\sum_{i=1}^{N}\zeta_i \\ y_i[(w\cdot x_i)+b]\geq 1-\zeta \quad i=1,2,\cdots,N \\ \zeta\geq 0 \end{cases} \quad (16)$$

where $C$ stands for the penalty factor.

The following equation could be obtained after the solution based on Lagrangian multiplier:

$$w(\alpha)=\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j K(x_i\cdot x_j)-\sum_{i=1}^{n}\alpha_i, \quad (17)$$

where $K(x_i,x_j)$ stands for the kernel function .

The final classification function was:

$$f(x)=\text{sgn}(\sum_{i=1}^{n}\alpha_i y_i K(x_i\cdot x)+b) \quad (18)$$

## 5.2 Common kernel functions

(1) Linear kernel function is:

$$K(x,x_i)=x\cdot x_i. \quad (19)$$

(2) Polynomial kernel function is:

$$K(x,x_i)=[(x\cdot x_i)+1]^p, \quad (20)$$

where $p$ stands for the polynomial order.

(3) Radial basis kernel function is:

$$K(x,x_i)=\exp(-\frac{\|x-x_i\|^2}{\sigma^2}. \quad (21)$$

Different kernel functions will affect the classification performance of SVM. It is found that radial basis kernel function has better performance. Therefore radial basis kernel function was used in this study.

# 6 Verification of gesture recognition system

## 6.1 Establishment of sample library

Gestures were collected with the video camera in the laboratory environment. Six gestures were designed, as shown in Figure 2. An experimenter repeated every gesture in front of the camera for 100 times. The first 60 samples of every gesture were taken as the training samples, and the remaining 40 samples were taken as the experimental samples. There were 360 training samples and 240 experimental samples, totally 600 samples.

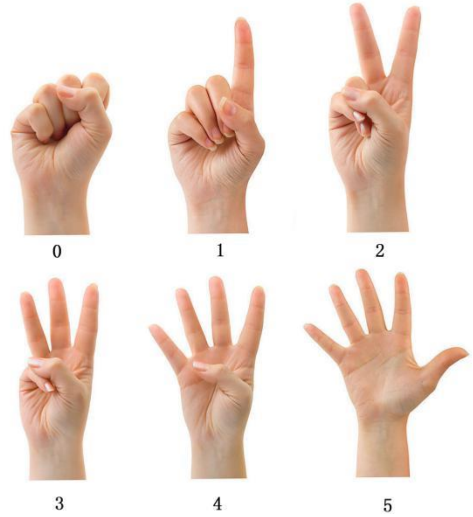


Figure 2: Experimental gestures

## 6.2 Gesture recognition results

Firstly YCgCr color space based skin color segmentation method was used to segment the samples. Images of the gestures obtained after segmentation are shown in Figure 3.

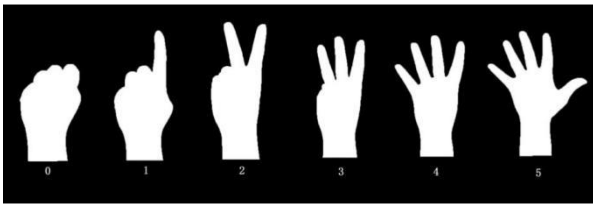


Figure 3: The segmentation results of gestures

It was found that the used gesture segmentation method accurately segmented gestures. According to the obtained segmentation results, features of the training samples were extracted, and five eigenvalues were obtained, including 4 invariant moments and the ratio of area to circumference A, as shown in Table 1.

| | $\phi_1$ $(10^{-3})$ | $\phi_2$ $(10^{-7})$ | $\phi_3$ $(10^{-8})$ | $\phi_4$ $(10^{-9})$ | A |
|---|---|---|---|---|---|
| 0 | 8.72645 | 3.32452 | 1.75426 | 2.71256 | -1.42658 |
| 1 | 1.02154 | 3.59875 | 1.02568 | 2.31456 | -2.25454 |
| 2 | 8.62135 | 2.61245 | 2.26589 | 4.501247 | -3.26589 |
| 3 | 6.62354 | 9.28452 | 4.12523 | 4.12589 | -4.23654 |
| 4 | 5.62147 | 5.16521 | 3.28564 | 3.25489 | -5.07541 |
| 5 | 5.01245 | 1.06241 | 1.24658 | 1.20158 | -5.92587 |

Table 1 The eigenvalues of gestures.

SVM was trained using the selected samples. Then it was used to recognize the experimental samples. To verify the recognition effect of the SVM, HMM and SVM recognition performance were compared. The results are shown in Table 2.

Table 2 shows that the SVM has a very favourable performance. Only 2 out of 240 samples were misjudged. The recognition rate of 4 gestures out of 6 gestures reached 100%, and the overall recognition rate reached 99.2%. The number of errors of HMM algorithm was relatively large, totally 24, and the average recognition rate was 90%. This showed that the effect of gesture recognition based on SVM was better than the one based on HMM. The recognition based on SVM could accurately recognize the gesture samples after gesture segmentation and feature extraction, with few number of wrongly recognized samples and high recognition rate; therefore the method was reliable.

| Recognition results of SVM | | | | | | |
|---|---|---|---|---|---|---|
| Gesture | 0 | 1 | 2 | 3 | 4 | 5 |
| Number of correctly recognized samples | 40 | 40 | 39 | 40 | 39 | 40 |
| Number of wrongly recognized samples | 0 | 0 | 1 | 0 | 1 | 0 |
| Recognition rate | 100% | 100% | 97.5 % | 100% | 97.5 % | 100% |
| Average recognition rate | 99.2% | | | | | |
| Recognition results of HMM | | | | | | |
| Gesture | 0 | 1 | 2 | 3 | 4 | 5 |
| Number of correctly recognized samples | 38 | 36 | 36 | 35 | 34 | 37 |
| Number of wrongly recognized samples | 2 | 4 | 4 | 5 | 6 | 3 |
| Recognition rate | 95% | 90% | 90% | 87.5 % | 85% | 92.5 % |
| Average recognition rate | 90% | | | | | |

Table 2 Comparison of the recognition results of SVM and HMM.

# 7 Discussion

In the aspect of the gesture segmentation, the skin color segmentation method based on YCgCr color space was used in this study [15]. It is less limited by light, and the calculation involved is simpler. It can segment the gesture image more accurately to facilitate feature extraction and recognition.

The segmented gesture image contains a large amount of data. In order to recognize the image effectively, the feature vectors are needed. The extraction of feature vectors has a great influence to the recognition accuracy. In this study, Hu invariant moments were selected to extract the features of the gesture image. In order to reduce the computation, only the first four invariant moments were used together with the ratio of the area to the circumference of the gesture. In total, five feature parameters were used for gesture recognition.

The SVM algorithm was selected due to its unique advantages in classification of small sample sets and non-linearity. It was also successfully used in several other pattern recognition applications, data mining and other aspects. In this paper, the radial basis function is used as the kernel function. In the experiment, it was shown that the support vector machine algorithm has a good recognition performance. In the recognition of six gestures, the recognition rate is as high as 99.2%, while the recognition rate was 90% when HMM was used. The SVM clearly outperformed the HMM.

The gesture recognition approach described in this paper has a good performance and can be used in the actual human-computer interaction. For example, in the application of the intelligent remote control, gesture 0 can be used for indicating shutdown, gesture 1~5 can be used to indicate channel 1~5. It can also be used in intelligent audio, gesture 0~5 corresponding to shutdown, last song, next song, volume plus and volume down. With further training and recognizing additional gestures this could be applied even more extensively.

# 8 Conclusion

Gesture recognition can demonstrate a simple human-computer interaction and can have a great applied value in people's daily life. In this study, firstly, the YCgCr color space based skin color segmentation method was used for segmenting the collected hand gesture images. Then, features were extracted using the first four features of the Hu invariant moment theory together with the ratio of area to circumference of the hand gesture images. Next SVM was used for gesture recognition in images. In experimental setup the recognition rate of SVM was 99.2%, which was 9.2% higher than the HMM approach, indicating the high reliability of the SVM algorithm. This work provides a theoretical support for the application of the SVM in human-computer interaction.

## Acknowledgement

This study was supported by Research on the Application of SVM in Human-Computer Interaction under grant number 1351MSYYB002 and Research on Some Key Technologies of the Internet of Things Architecture and Intelligent Information Processing Theory under grant number YB2018005.

# References

[1] Hasan H. S., Kareem S. A. (2015). Human Computer Interaction for Vision Based Hand Gesture Recognition: A Survey. *Artificial Intelligence Review*, pp. 1-54. https://doi.org/10.1007/s10462-012-9356-9

[2] Grudin J., Carroll J. M. (2017). From Tool to Partner:The Evolution of Human-Computer Interaction. *Extended Abstracts of the Chi Conference. Morgan & Claypool*, Morgan & Claypool, pp. 183. https://doi.org/10.2200/S00745ED1V01Y201612HCI035

[3] Biswas K. K., Basu S. K. (2012). Gesture recognition using Microsoft Kinect®. *International Conference on Automation, Robotics and Applications*. IEEE, Wellington, New Zealand, pp. 100-103. https://doi.org/10.1109/ICARA.2011.6144864

[4] Panwar M. (2012). Hand gesture recognition based on shape parameters. *International Conference on Computing, Communication and Applications.* IEEE, Dindigul, Tamilnadu, India, pp. 1-6. https://doi.org/10.1109/ICCCA.2012.6179213

[5] Ren Z., Meng J., Yuan J. (2011). Depth camera based hand gesture recognition and its applications in Human-Computer-Interaction. *Communications and Signal Processing*. IEEE, Singapore, pp. 1-5. https://doi.org/10.1109/ICICS.2011.6173545

[6] Córdova-Palomera A., Fatjó-Vilas M., Kebir O., et al. (2011). Intelligent Approaches to interact with Machines using Hand Gesture Recognition in Natural way: A Survey. *International Journal of Computer Science & Engineering Survey*, pp. 122-133. https://doi.org/10.5121/ijcses.2011.2109

[7] Kuang D., Yang C., Wang M., Peng G. (2018). *An improved approach for gesture recognition. Chinese Automation Congress*. IEEE, Jinan, China, pp. 4856-4861. https://doi.org/10.1109/CAC.2017.8243638

[8] Huang D. Y., Hu W. C., Chang S. H. (2011). Gabor filter-based hand-pose angle estimation for hand gesture recognition under varying illumination. *Expert Systems with Applications*, pp. 6031-6042. https://doi.org/10.1016/j.eswa.2010.11.016

[9] Li J., Zheng L., Chen Y., et al. (2013). A Real Time Hand Gesture Recognition System Based on the Prior Facial Knowledge and SVM. *Journal of Convergence Information Technology*, pp. 185-193.

[10] Nagarajan S. S., Subashini T. (2013). Static Hand Gesture Recognition for Sign Language Alphabets using Edge Oriented Histogram and Multi Class SVM. *International Journal of Computer Applications*, pp. 28-35. https://doi.org/10.5120/14106-2145

[11] Itkarkar R. R., Nandi A. V. (2017). A survey of 2D and 3D imaging used in hand gesture recognition for human-computer interaction (HCI). *IEEE International Wie Conference on Electrical and Computer Engineering*. IEEE, Pune, India, pp. 188-193. https://doi.org/10.1109/WIECON-ECE.2016.8009115

[12] Zhao S., Zhang Y., Zhou B., Ma D. (2014). Research on gesture recognition of augmented reality maintenance guiding system based on improved SVM. *International Symposium on Advanced Optical Manufacturing and Testing Technologies: Optical Test and Measurement Technology and Equipment. International Society for Optics and Photonics*, pp. 61-81. https://doi.org/10.1117/12.2067852

[13] Hartmann B., Link N. (2010). Gesture recognition with inertial sensors and optimized DTW prototypes. *IEEE International Conference on Systems Man and Cybernetics*, Istanbul, Turkey, pp. 2102-2109. https://doi.org/10.1109/ICSMC.2010.5641703

[14] Aguilar W. G., Cobeña B., Rodriguez G., et al. (2018). SVM and RGB-D Sensor Based Gesture Recognition for UAV Control. *International Conference on Augmented Reality, Virtual Reality and Computer Graphics*. Springer, Cham, pp. 713-719. https://doi.org/10.1007/978-3-319-95282-6_50

[15] AlTairi ZH, Rahmat RW, Saripan MI, Sulaiman PS. Skin segmentation using YUV and RGB color spaces. *Journal of Information Processing Systems*, 2014, 10(2):283-299.

# Collaborative Strategy for Teaching and Learning Object-Oriented Programming Course: A Case Study at Mostafa Stambouli Mascara University, Algeria

Chérifa Boudia
University Ahmed Ben Bella Oran 1
PB 1524, El M'nouer Oran, Algeria
E-mail: cherifa.boudia@univ-mascara.dz

Asmaa Bengueddach
University Ahmed Ben Bella Oran 1
PB 1524, El M'nouer Oran, Algeria
E-mail: asmaa.bengueddach@univ-oran1.dz

Hafid Haffaf
University Ahmed Ben Bella Oran 1
PB 1524, El M'nouer Oran, Algeria
E-mail: haffaf.hafid@univ-oran1.dz

**Student paper**

**Keywords:** university teaching strategies, oriented-object programming, collaborative learning,

**Received:** May 21, 2018

*Several strategies and methods have been designed and tested to allow students to have better teaching and learning of programming concepts and training their programming skills acquired. In our university, old and classic pedagogical methods are still used in the learning of Oriented-Object Programming (OOP). This paper explores the experimentation of a pedagogical approach designed for Mostafa Stambouli Mascara University's computer sciences students to enhance their chances to get an effective teaching using a collaborative learning and early introduction of current technologies. The study was made of 108 second year informatics students' for two semesters (October–January 2016-2017, September-December 2017-2018) who were identified in a controlled experience for collaborative learning OOP in lab session. Students gathered into predetermined groups based on initial test and some setting. The students are supposed to share the same programming assignment by distributing roles according to global pedagogical scenario for problem-resolving activities. This experience allows the teachers to apply this strategy and see students in closer and permit them to express their problems and search together for solutions. The principal objective was to offer a new experience to motivate students and make this design a smart opportunity to have better programming performance. Furthermore, as a result of testing this new strategy, the students' academic progress is clear. A questionnaire was distributed with the purpose of analyzing students' motivation and satisfaction levels alongside the effects of the experiment. Results show that students found that among other benefits this method facilitates the learning process.*

*Povzetek: Poučevanje objektnega programiranja v Alžiriji na osnovi skupinskih pristopov.*

## 1 Introduction

The challenges of the act of teaching are diverse and intricate in Algeria, as in most universities of the world [1], but remain focused on its pedagogical components [2], [3]. Indeed, to make our university credible innovative learning environment, newly recruited teachers are trained in teaching methodology for several months. However, these teaching methods and strategies are not deepened and stay premature [4]. Coming to computer sciences where problems are doubled. In fact, programming is a very hard task for students because it required high metacognitive skills like abstraction, deep comprehension, and long awareness, and many steps in problem-solving from analysis to debugging and testing [5]. In addition, teachers claim further difficulties found during programming lab sessions, due to several reasons such as the large number of students per class [6]. To overcome these challenges, collaborative learning is a good situation to involve students to help each other in groups for the same assignment activity by discussing ideas and giving everyone a simple task [5]. They can switch roles when the activity is completed. Consequently, Students enjoy the learning experience; feel motivated and ready for more improvement, in the other side, the teacher found more time to go further on the course [7]. To elaborate collaborative learning in teaching programming for lab sessions, it is necessary to design a collaborative strategy for each step in problem-solving and hire metacognitive abilities respectively in every simple activity [8]. Based on this idea, the students' final program is the result of a collaborative strategy in teaching programming lab sessions starting from forming adequate groups which students work and learn from sharing, discussing their separated solutions and select the best solution that everyone has participated and understood. The overall purpose of the study is to demonstrate the impact and the effectiveness of a collaborative strategy on student learning in our university. As principal element, we introduce the collaborative learning in programming lab session course. As well, identify the behaviors of students through all designed roles in programming activities at different phases of global learning scenario.

The lack of experimental research of new educational strategies in the field regardless the poor level of students' academic performance in programming. The decision was made to experiment a new strategy and to reveal the results. Our perspectives, as teachers experience's in teaching programming, enable us to discover what challenges are presents, and the strategies claimed to be using successfully. In addition, a variety of pedagogical strategies was recommended from teaching practices. In categorizing approaches taken by teaching to support students, five key themes emerged globally collaborative learning, unplugged type activities, contextualization of tasks, developing computational thinking, and scaffolding programming tasks. Hence, this study aims to investigate the following research questions:

Will collaborative learning with different roles in lab design affect students' academic performance in programming?

Can resolution strategy process or steps guide the students in their achievement?

Is there a relationship between the uses of ICTs to gain computer metacognitive competence?

What is the teacher role in the entire experience?

The rest of the paper is organized as follows: section 2 describes the state of art of the research topic, related works and common definitions; section 4 displays the Algerian universities context about teaching computer-programming course. Section 5 discusses the methodology and its results of the strategy. The last section, presents some conclusions and perspectives associated to the result of our research.

# 2 Related work

## 2.1 Teaching OOP courses

In computer science, the teaching strategies and methods should be completely redesigned especially for programming courses [9]. Relatively to the nature of programming that presents challenges for both teachers and students [10]. It's resulting mainly by the gap between theory and practice [3] in converting the understanding OOP courses' lectures words into specification and coding using keyboards machines. The students seem to be disconnected from the practice of programming in lab sessions, where the smallest difficulties related to misunderstanding of a small semantic detail. Moreover, they are not diligent to work with current technologies in place of known tools [11]. Another difficulty for many programmers' students is the multiplicity of approaches, and tools for program development. An approach or an algorithm that works in one case will sometimes not work in another, even though the problems appear review similar. The identification and selection of alternative approaches is seldom. Most programmers must rely on experience to determine the appropriate way [12].

To facilitate an effective learning experience in programming, and to support students understand the concepts, gain skills and promote their thinking performance, we need a collection of components beside the pedagogical ones. Researchers argues the necessarily of ICT (Information, Communications & Technology) tools in building a solid environment for programming education [13], [15]. Some tools are really serving the whole education process for a good comprehension of concepts and developing programming skills such as game [14], [13] [16], [17], [18]. Education is still, at an important percentage, teacher-centered learning where he is the source of knowledge, key of participation, students' engagement and collaboration [19]. The use of internet and especially Facebook emerged significantly education to student-centred learning [20]. It was proven that providing links and materials is more useful for student to learn and give them extra time outside university doors for communication between teacher and students and among students [21].

Teaching guides the student' progress from beginner [10] to expert by attaining skills, and develop intelligence, and learn to communicate with teachers and his class members [22], [23], [24], explore his own conviction and attitude, and recognize his prospective, complete the task to another high one [25], [26]. As well, learning to be a computer programmer is considered a very hard task especially for beginners [27]. It requires constant improvement in programming skills. If learners are not moving forward, they are going backward.

Learning to program involved many techniques, like verifying arguments using methods provided by programming tools [10], understanding the important each code blocks, experimenting that is performed by executing some test data, learning more by reading the outputs of the program to find errors in the software code, and using the open source codes, in both books and sites that host and release millions of lines of software [5]. Some of those programming techniques can help in a specific pinch, while others focus on the programming tools and environments. Despite the nature of each technique, when used with awareness and diligence, each can help develop students abilities, both beginner, and expert [28]. In the same time, some students show resistance to develop the competencies required to code programs than execute it properly [17]. Consequently, it is imperative to know what barriers make this kind of learning complicated and how students could learn correctly and easily [29]. Furthermore, program execution on a machine is a dynamic process and it is complex to evoke mentally all the treatment seize and track how variables change during program execution [9], [30]. Others difficulties like problems in visualizing all the changes that happen when the program is running. Programming engages problem-solving strategies [31], choosing appropriate resolution methods, coding, debugging and testing a program as a result [32], [15].

Besides, programming courses traditionally emphasize theoretical basis understanding of OO programming concepts, as well as its application in code fragment [33]. The concepts are truly learned during the practical experience to develop very high programming skills level, students needed to do often practice on programming exercises and to master debugging [1]. This course is presently brought to students in different strategies: traditional face-to-face learning [34], online learning [25], or in both situations called: blended learning [35].

## 2.2 Collaborative learning

### Definitions

The collaborative learning is a situation where an interaction between students with exchanging, sharing ideas and information, distributing tasks [5] in order to achieve learning outcomes [37]. The importance of collaborative work is in summoning all the effort for the complementary relationship that conducts to excellent results. In collaborative learning situation, where at least two students work together to solve the same assignment, engage in a discussion in which all elements are working together to resolve a problem, and agreed on the founded solution at the end where various roles can be considered [28]. In psychological sciences and education, a study of Temperman and his colleagues [37] suggested six roles specific to learners according to their learning style (coach, organizer, secretary, moderator, administrator and theorist) in a collaborative remote learning environment. Andriessen [38] talk about discussion through visual representations that can help students learn to think critically and independently, even when students disagree, they still share the common concepts and they are all interested in achieving the same goals, in the meantime, the teacher should supervise the whole session and intervene depending on the problem to solve.

Collaborative learning is used in many education disciplines, rather in engineering science. Computer science courses, mainly in face-to-face lab sessions, when the students are all presents, roles can be distributed at the beginning of each problem-solving and switched at the end. Literature mentions different collaborative learning strategies in programming courses [30], [7]. The most used is pair programming, two students share the same computer and attempt to solve the programming exercise, one of them the ”driver” pilot the coding task and the other the “navigator” detect and correct the eventual errors [39]. When the group is made up of more than two members in small or big groups, different tasks could be proposed according to the collaborative learning activities scenarios or scripts [40]. The distribution of those tasks might be automated in traditional teaching environments[41]. Another technique is TPS (Think-Pair-Share). Students operate in three phases: think and work individually for few minutes on the task, then engage in discussion with neighbors to write the detailed code encouraged by the teacher. At the last phase, the entire class participates in discussion for the proposed and alternative solutions. It is simple and easy to deploy in large class for small groups used in lectures and labs [42]. Learning computer programming accrue many abilities such as thinking critically, analyzing and synthesize information, organizing and planning in groups. Those abilities depend on others skills like social skills, it enhances communication and motivation form group working [43]. A collaborative environment reinforced by ICT tools increase interaction between learners [17]. Learners can discuss the subject problems, share ideas and/or code fragments through online interaction [44], [27].

Collaborative programming on the same programming assignment activity consists on designing several pedagogical scenarios depending on group settings. Many settings for group formation exist: random selection, self-selection, homogenous, and heterogeneous. Random selection generally is used in first year programming courses curriculum or large classes [9], where the students meeting up in groups set from the list of registered students ordered alphabetically. The random segmentation for large classes provides equitable distribution easy to apply offers to students, the same chance to interact socially being in any group [45]. Self-selection is used frequently project work groups. The students are more comfortable in choosing their mate based on friendship, previous programming experience, and previous studies [28].

In the homogeneous group setting, members join a group where similar characteristics and preferences are in common such as learning style, same academic level [36], [23], [46]. Among its advantages, students can promote actively each other by interacting with varied individuals to enhance their abilities [39]. The other sides, students with different characteristics like learning level, make heterogeneous groups. This dissimilarity may be in each student but the same learning objective brings them together. The conditions that determine the choice of the group depend on the learning activity’s objectives [5], [47]. In literature, mixed groups are the most adopted setting [28]. However, the problem lies in heterogeneous elements' determination to configure the group in question [23]. Therefore, frequent researches have experienced the effect of group formation method on group performance [22], [48]. At this point, the importance of identifying witch settings may lead to a fruitful learning experience.

All the reviewed studies provide a big evidence of how academics and teachers are agreed to work hard in order to improve teaching and learning computer programming. Though, they operate at different aspects and use different techniques on teaching methodologies. In such a complex field as education and pedagogy, we focus on two aspects: techniques and methods in teaching computer programming and collaborative learning. Some parameters are needed to determinate a clear vision of each study, such as scope, timeframe, etc. From this state of art, collaborative learning may be a good practice to involve in learning programming. In methodology section, we will provide all settings to define our pedagogical approach.

### Pedagogical scenario

The pedagogical scenario is a central concept called method that defines the learning unit's scenario. This meta-model contains the descriptors necessary to model a learning unit where the role of each intervener is precisely described as well as the objectives and prerequisites of each participating activity to achieve a global goal [49] like in a collaborative learning situation. Therefore, the pedagogical scenario is the product of a design process. Its content consists of objectives, a planning of learning activities, a schedule, a description tasks to be carried out by the students, evaluation methods all defined, arranged and organized during a design process. The Scenario is considered as a structured and coherent set of two parts: the learning scenario and the coaching scenario. The learning scenario' role is to describe the activities designed for use by learners and their assembly to build a learning situation and the productions that are expected. The frame scenario, or scenario as called by [50], specifies how teachers intervene tutors as designed to support the learning scenario which is considered to be a specification guiding the progress of the activity in the IT environment for which it was designed [51].

The passage from the simple textual description describing the activity towards its unfolding through a series of transformations that plays the scenario in the environment selected target computer. In [52], the authors define the pedagogical scenario as the description of a learning situation in terms of roles, activities and environment necessary for its implementation, but also in terms of knowledge manipulated. Also, they make the difference between two types of scenarios: the predictive scenario: defined as established before by a designer for the implementation of a learning situation, and the descriptive scenario defined as a scenario that describes the progress of the situation a posterior learning by including in particular the actors' activity traces (mainly the learners) and their productions.

## 2.3 Algerian universities practice in teaching computer programming

To properly conduct our study, we need to make a diagnosis on our specific problem we are facing and carry out a state of existing to see what strategy of teaching that could success in our environment with our students with the available components. Since 2004, almost in all higher institutions and universities in Algeria, educational system had moved to LMD system (licence, Master, Doctorat). Many malfunctions has accompanied the introduction of this new system, experts and academics argues from an evaluation of this experience in each university, and proposed, in terms of actions and measures that will allow the university to fill in the gaps and overcome those problems [53], this paved different features as a reform at the national level. At Mostafa Stambouli Mascara University, the undergraduate curriculum in computer science has been completely redesigned in the past few years. As part of this redesign, in the first semester, students are taking initiation to algorithmic at the first-year mathematics and informatics. This course introduces basics concepts in computer science as algorithms and basics data structures that include one lecture, one tutorial and one lab session weekly. In second semester, they take, as continuation of initiation to algorithmic, programming and data structure. This course gives a deeper notions and data structures using C programming language. Those two courses are ones of several fundamental teaching units' courses. Among methodological teaching units, a new course (M211) Information and Communications Techniques where students can distinguish new technologies for information and communication. As elective courses, students can choose between (M212) programming tools for mathematics and (M213) Introduction to Object-Oriented programming only one lecture in a week. Because students don't take any elective course in the first-year, they took these courses mandatory, M213 in the second year informatics and M212 in the second year mathematics. They are required to take oriented-object programming course with Java as the programming language with overall hourly volume of 67 hours and 30 minutes. The lectures were optional to attend but tutorials and lab sessions were mandatory. At the beginning of the semester, for the second year informatics class, students are divided into groups by alphabetical order; the average number of students per group is 20 according to the number of students enrolled. The two machines rooms granted to the informatics' students are rented for all courses with lab session within one hour and thirty minutes period weekly. The machine rooms are equipped with computers, and their tables, a whiteboard and a desk for the teacher. This course is generally given by one teacher who is in charge of the lecture, tutorials and the lab sessions or no more than two teachers; one in charge of the lecture and the tutorials, the other in charge of the lab sessions. The official program of this course contains five chapters: introduction to OO Programming, classes, heritage and polymorphism, interface and implementation, and graphic interface and applet with java programming language. The student is expected to acquire during this course the following skills:

1- The fundamental of object programming in java,
2- Reading and understanding programs in java,
3- Writing a program in java as a solution for a given problem,
4- Writing sophisticated applications (use of advanced data structures).

The course's assessment is done in two forms Continued Control (CC) and the final exam. For tutorials and lab sessions respectively, the teacher must do two CCs that could be a tests in machine room or homework and provide the marks before the final exam.

# 3 Method

The objectives of the study are to demonstrate the impact and effectiveness of implementing the collaborative group in learning achievement. Characterizing the nature of student behavior through different roles in programming learning, whether the teaching techniques are useful for students in face-face and distance learning, and to carry out a comparative study at different phases of global learning scenario with teacher and students engagement involved in.

## 3.1 Groups' determination

In this study, 108 students of two different semesters (about 14 weeks per semester) from second year informatics undergraduate OO programming course with Java programming language are involved. The experience is conducted only on lab sessions according to a schedule time. Each semester for lab sessions, students were initially divided into 3 groups by alphabetical order. Separately, each group from the three initial groups, attend the lab session consecutively. As indicated in Table 1, Only 55 students (27 females, 28 males) from 67 enrolled students in semester one and 53 students (22 Females, 31 males) from 55 enrolled students in semester two. At the beginning, the students are randomly divided into 3 groups, and each group is composed of 22 as an average. Starting by setting up the sub-groups according to three steps:

**The initial test (IT)**

As the first step, it consists of making a simple MCQ that includes 16 basis algorithmic questions. It aims to determinate the performance Academic Level AL of each student. Another four questions, which are the age, score in first-year, color, and the last one, is: do you want to be the representative of your group? The color preference may identify similarities for the same group [54]. The answer of the last question can reveal the leadership potential of students [55]. As a result, the MCQ allows us to classify in 3 categories according to scores' intervals: good [16-11[, medium [11-6[, poor [6-0].

| Semesters | Total enrolled students | Total remain students | Initial groups | Sub-groups |
|---|---|---|---|---|
| 1 | 67 | 55 | 3 | 14 |
| 2 | 62 | 53 | 3 | 17 |

Table 1: Information on the number of students and groups formed students.

**Information examination**

At the stage, we have all information like AL according to IT. We make the students together by followings AL and colors preference in a way to formulate in each color group for example: blue group contain one good, two mediums and one poor. Remark: to have more balanced groups, we tried to mix the two types of gender in each group.

**The final step**

In class, we told the students witch group they belong to. If they do not have any other issue about their partners, any preference or suggestions, they join their groups, sit together, and share one computer. According to this, sub-groups are formed of 3 to 5 students each. The teacher recommended all groups to get textbook called Pedagogical Notebook (PN) in order to note every lab sessions activity, they cover it with their favorite color. The teacher checks frequently the PNs notes and gives extra marks for the best pedagogical notebooks at the end of the semester.

## 3.2 Roles distribution

We start to give details how to work in the group with explanation the roles of each member inter and intra group.

### The students' roles

We have determinate a variety of roles that the students must know as programmers and as members in the group. Those roles are quite different from most studies, such as those reported by [37]. The roles are not assigned but taken after discussing. Students' roles are the main piece in our collaborative learning design because when all group members discuss and get a task, the student assumes his work with determination. Roles should rotate at the end of the exercise in order to allow every student can perform and experience all roles so that everyone takes a turn and consequently everyone in the groups should master. Students can complete more than one role at a time, if necessary. Possible roles include the following:

***Coach***, student guides the group in the problem solving by facilitating the communication, motivate group members and if conflict case, he or she decide what strategy to choose or algorithm to apply, shifting the dialogue focus toward program structure. The coach asks the other groups' coaches or/and the teacher for assistance, if necessary.

***Theorist***, student, who can easily explain of the concept discussed, searches quickly for resources, and reminds the group of the steps to follow in problem solving.

***Programmer***, student sits, in front of the computer, typing the code program properly using Intellij and all group members read and correct or share an idea.

***Moderator***, student watches the group and gives feedback. He or she looks for behaviors to praise. The student encourages all group members to participate in the discussion and assist one another. He or she evaluates how well the group has worked together and gives suggestions for improvement.

***Secretary***, during the whole-class wrap-up, the student write on the pedagogical notebook PN all the details from the problem statement to the solution: noting the main ideas the group shared, questions the group has generated, teachers' answers…

***Planner***, Student sets the timer for each part in the lab session and lets the group know when it is time to move on to the next task. He or she makes sure everyone participates and only one person talks at a time (the teacher might do this instead of students).

### The teacher's roles

Besides designing the course, tutorials and Lab sessions and exercises sheets, the teacher also design the timing about exercises to be solve in each lab session according the courses' progress, and also reminds the groups, during the 90-minute lab sessions to swap roles every finished exercise. Teacher can help by actively listening to students' conversations and if necessary clarifying situations, modeling strategy usage, explaining errors after or before debugging, sometimes correcting the code or giving tips about possible solutions, encouraging students to participate, and modeling a helpful attitude. At the end of each exercise, the teacher selects the best solution or final code in the perfect situation or suggests the worst code to be corrected, or to found the mistakes and at last taking notes on the PN. It is expected that students will need assistance learning to work in collaborative groups, implementing the strategies, and mastering academic content.

## 3.3 The semester' content and resolution strategy

### The content

The Java lab sessions is conducted as shown in Table 2. The exercises' sheets are given at the start of each chapter that permits the students to prepare at least one or two exercises solutions before each groups' meeting in the lab session classrooms or other places. The teacher at the beginning of the class checks every student's solution and gives scores. Because lack of time, we select the most important exercises to solve in class and let the students do the rest outside of classroom time.

To foster the collaborative learning, the teacher might give highlights or mention the main idea of the checking skills and knowledge of Java programming. In the same date, we make an announcement of project groups' assignment and collaborative learning conditions to do during 7 weeks.

A project consultation involving collaboration among the students to make a code of simple software with user interface is conducted at the 14th week. The students learn advanced OOP in the latter half of the semester.

| week | Session content |
|---|---|
| 1st | Groups determination, Intellij installation and first uses |
| 2nd | Sitting up groups & Uses java terminology 1 |
| 3rd | Uses java terminology 2 |
| 4th | Java Class 1 |
| 5th | Java Class 2 |
| 6th | Heritage & Polymorphism 1 |
| 7th | Individual CC1 & Project groups Assignment |
| 8th | Heritage & Polymorphism 2 |
| 9th | Abstract class & Interfaces 1 |
| 10th | Abstract class & Interfaces 2 |
| 11th | Graphic Interface 1 |
| 12th | Graphic Interface 2 |
| 13th | Applet |
| 14th | Individual CC2 & Project Consultation |

Table 2: The Java lab sessions content per week.

### The resolution strategy

Figure 1shows an overall scenario for learning activities in case the given exercise is being to be solved in class (solve at minimum two exercises per session: 20 min per exercise), otherwise we jump to phase 3 (the student prepare exercises before class to gain time and we correct more than two exercises in class). That involves:

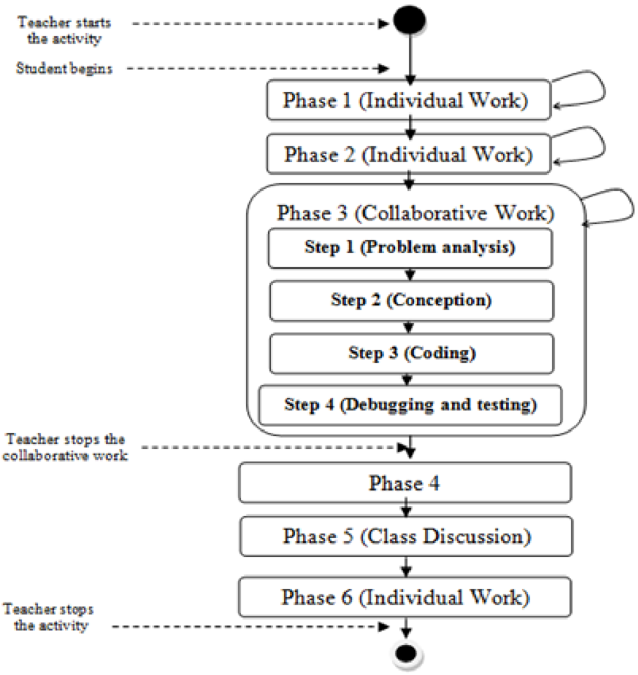

Teacher starts the activity
Student begins
Phase 1 (Individual Work)
Phase 2 (Individual Work)
Phase 3 (Collaborative Work)
Step 1 (Problem analysis)
Step 2 (Conception)
Step 3 (Coding)
Step 4 (Debugging and testing)
Teacher stops the collaborative work
Phase 4
Phase 5 (Class Discussion)
Phase 6 (Individual Work)
Teacher stops the activity

Figure 1: Global scenario for learning activity Problem-solving phases for programming.

*Phase 1 (Individual work):* students try to read and understand the exercise statement after that the teacher can explain the exercise's purpose, give some details and set up the timer for 20 min.

*Phase 2 (Individual work):* students adopt a strategy of resolution and try to solve the problem partially or totally.

*Phase 3 (Collaborative work):* it involves 4 steps:

3.1 Discussion (Problem analysis)

3.2 Choice of resolution strategy & writing algorithms (Conception)

3.3 Implementation (Coding)

3.4 Debugging and testing the final program.

*Phase 4:* the teacher stop the collaborative work, selects a code from any group and ask the group's coach to write it or just fragments on the whiteboard.

*Phase 5:* the teacher starts a discussion with the students about the code, finding bugs, and modifies it together, then ask the students to type it again on machine to display the results.

*Phase 6 (Individual work):* writing notes.

# 4 Results and discussion

So far, we have recorded 31 collaborative sub-groups for the two semesters. The following step is, for each one of them, to trace, in categories their characteristics, communication, benefits, disadvantages and other points of interest such as the technologies used. For instance, Group organizing refers to role division of various types that offer features of organizing a group. Collaboration environments refer to places or environments that allow students to study together in the same physical area that provides the appropriate infrastructure to support the collaboration environment for exchange and interaction used with a common purpose with Github in distance learning. Communication refers to different ways of supporting discussing and interaction between members in a group. Web conference and Webinars are synchronous ways of communication using internet and social network Facebook. Of course, there are other ways (for example asynchronous) that can thoroughly support groups' communication; like telephone messages.

## 4.1 Survey and data analysis

The online survey questionnaire was sent by this link https://app.evalandgo.com/s/?id=JTk1byU5NG4lOTklQUI=&a=JTk1ayU5OG0lOUIlQTk= via the students' Facebook group to about two promotions of 108 2nd year informatics students. The survey was conducted for the first promotion at the second half of January 2017 and at the beginning of February 2018. The survey was done in French language. The students had only one month to respond to the questionnaire, they were not graded. It was optional to complete. The responses were not anonymous at the time that they were submitted but were identified during data analysis. The ongoing collection of student responses was purposefully tied to the content questions so as to get the whole picture of the student experience in the Lab session Collaborative learning.

During this period, in total 97 questionnaires were received with response rate of about 71.59%. After initial analysis of the total responses, 31 of them were judged to be incomplete and were dropped from the analysis, bringing the number of surveys to be analyzed to 66 responses.

Survey consisted of multiple-choice, 4-point Likert scale, and short-answer questions that asked students about various aspects of collaborative learning lab session new design. Briefly, it contained 109 questions divided on 8 parts:

Part 1: Personal Information
Part 2: Use of E-Learning Platforms (Distance Learning)
Part 3: OOP Course
Part 4: Students' perceptions of collaborative work within the group
Part 5: Students' perceptions of group design for Lab session
Part 6: Student Perceptions of Role Division in a Group.
Part 7: Perceptions of students in Collaborative Production
Part 8: Students' Perceptions of the quality and progress of their learning during Learning Scenario Activities (Global Pedagogical Scenario).

***Part 1: Personal Information.*** For the part 1, 44.44% of the responders were females and 55.56% were males. All students had BAC diploma and were between 20 and 25 years old except for one student who was over 25 years old. 53.33% attended the lectures, tutorials and lab sessions where the rest attended tutorials and lab sessions only. We noted that 82.22% of the students prefer to do their important homework outside of the university and the most of 17.78% stay in the university's library. 47.73% study for less than one hour per day, 38.64 % between 2-3 hours and 13.64 % for more than 3 hours. We asked the students about how they contact their teachers as multiple-choice question, 88.64 % answered that they do it directly after the teacher finishes the lectures, or tutorials or lab sessions and/or by social media about 45.45%, 31.82 % by emails and only 6.82% via the class's representative. Another multiple-choice question about the use of the internet: 88.84 % use it in search of exercises and solutions of different course, 29.55 to do tests online, 50% were looking for courses documentations, 20.45% to see their emails, and 25% to do other things.

***Part 2: Use of E-Learning Platforms (Distance Learning).*** Regarding the 2nd part that was 8 multiple-choice, 4-points Likert scale, and one short-answer format as mentioned in Table 3 (see Appendix (A)). We want to know from this section if the students visit e-learning platforms to study and what they do exactly. Apparently, only half of students know the e-learning platforms and 65.91of them believe in its effectiveness for their learning. Generally, they wished for changing content and format of these platforms. For ICT courses taken in first-year, 81.82% took 1-2 courses, 4.55% from 2-4 courses, 4.55% took more than 4 courses and 9.09% haven't took any course.

***Part 3: The OOP course.*** The OOP course design was a 6 of 4-point Likert scale, 1 multiple-choice and 3 short-answer questions format. The students, 65.79%, know that there is no OOP course on the e-learning platform, where the others don't. The division of the pedagogical content on courses, tutorials and lab session was satisfiable for 70.27% of the students, 27.03% disagreed and only 2.7% strongly disagreed. Then 48.65% of the students were strongly agreed to receive the course's content in video format, 21.62% agree, and the rest disagreed. All the students preferred to receive the contents of all sessions (courses, tutorials and lab session) before class, argues that this act will help them to prepare well and have more ideas about the classes 'content, and wish to have it online with tests and exercises' solutions with details, and know what their mates were reading, which courses. While the majority of students accessed to this part to complete all the questions (90–100% of students), a small percentage (5–10%) of students opted to not complete the 2nd part, particularly the short-answer questions only in two weeks. For better response rate, we should divide the survey on short-surveys and ask students to complete in different time of the whole experience. As few students generally wrote a few words for the short-answer questions, it is unlikely that any given short survey takes less than 5 min to complete.

***Part 4: Students' perceptions of collaborative work within the group.*** It consisted of 7 questions of 4-points Likert scale; that allowed students to evaluate their opinions and understanding of the collaboration, coordination and communication in the group with all members.

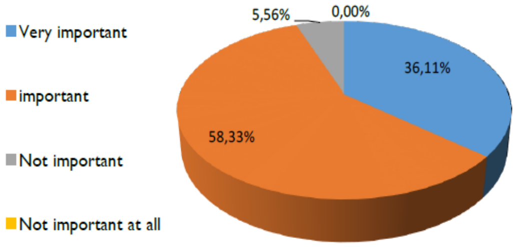


Figure 2: Students' Performances in Collaboration within the group.

Three principles questions were asked within "I would qualify my group during this practical work from a point of view". It was regarding the student experience in the collaborative work were based on the students' interactions with each other's inside the group and emerging themes observed during lab sessions. The data set shown in Table 4 (see Appendix (A)) that all students were happy and agreed of the idea of collaborative group work excepting for 5.56% were disagreed and preferred working solo (see Figure 2).

In coordination theme, tasks' assignment between groups' members were very important at 27.78%, important at 58.33% and not important at 13.89%. Just for information, tasks' division is different from roles' division; roles could be divided into tasks done by one or more than student. This might be more clarifying in part 6. As shown in Figure 3, the crucial element in the whole process was the time management including for the teacher, students in some steps in problem-solving phases overflow time allocated to finish the exercise when they (63.89%) could not decide about a definite point especially discussions and exchanging tasks (55.56% said that is not effective).

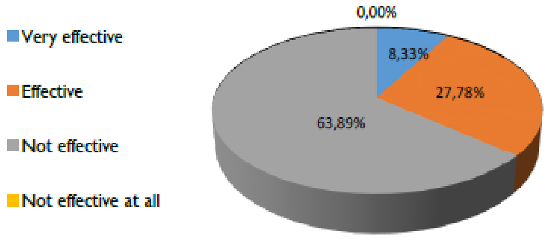


Figure 3: Students' Performances in Coordination (Time management).

They couldn't move on to the next step till it is not resolved. So, the Planner should be more lucid and specific in notifying the group members. This is why the teacher always notices all the groups when they are running out of time. To gain time, we allow groups to separate into subgroups (25.00%, 41.67% confirm its importance), more than one student take over one or more tasks. Some students (33.33%) chose to finish alone their tasks assigned. Communication was the vessel conductor in the group, students enjoy listening among the groups' members to learn and understand the context (Very effective at 33.33%, effective at 58 .33% and not effective at 8.33%), as seen in Figure 4.

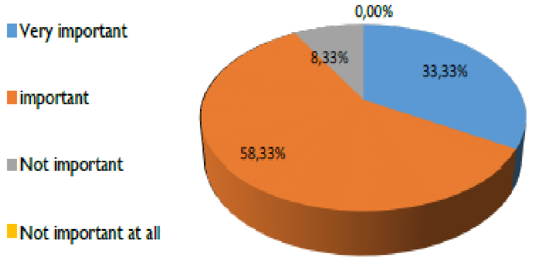


Figure 4: Students' Performances Listening among the groups' members.

When students agreed to divide simple tasks, help and depend on each other, they already know exactly what to do to finish it, as a result fast execution of tasks was very successful (Very effective at 36.11%, effective at 61.11% and not effective at 2.78%).

***Part 5: Students' Perceptions of Group Design for Lab Session.*** This part of the survey is composed of 6 questions of 4-point Likert scale and one short question format (rated from Totally agree, Agree, Disagree, Strongly disagree); it is about groups' design and conditions for group members' adhesion. In Table V (see Appendix (A)), students consider 3 to 5 members in the group was appropriate (47.22% totally agree, 25.00% agree) due to the complexity of problem-solving and short time to realize it. We adopted the flipped-class approach in preparing and solving exercises outdoor class to spend more time in rich discussions at lab sessions. They students know that there is not one way to solve a programming exercise but there is always an optimal one in term of running time, memory allocation, the right algorithm and minimum code lines. 80.55% of students get connected with their partners in the same group, 19.44% were not may be because in few groups some males and females were very shy (7 balanced groups, 2 groups of females and 2 groups of males). Conflicts in those groups were usually more balanced. Thus, the data set confirms that they prefer choosing their group mates and since we don't know the background of the students, it was not possible to let students select their group mates. The greater part of students wished for different academic level groups in order to lower academic level students get help from higher academic level students. 8.33% only which represents almost higher academic level students wants more competitive collaborative groups at the same time.

***Part 6: Students' Perceptions about the roles' distribution within a group.*** The fundamental preoccupation in computer science pedagogy is the improvement of computational thinking skills where all students are involves in this process from every step in solving-problems to implement applications. Students were asked about the impact of the roles proposed for the entire process to perform in collaborative groups for programming assignments; its benefits in facilitating the project's realization. Also, they were asked about further elements that reinforce the group work and what does it take to achieve each role? Table VI (see Appendix (A)) shows that they were agreed (27.78% totally agree, 72.22% agree) that it is facilitating their achievement by helping each other to understand the problems and solutions. Besides developing computational thinking skills (100%), it attributes them to build trust (41.67%), to arise their commitment (44.44%) and motivation (47.22%). It was evident for students to change and switch roles in terms of tasks (25.00% totally agree, 55.56% agree), skills required and also to gain abilities in mastering all roles (16.67% disagree, 2.78% strongly disagree). Students must be engaged in front of all members and fully responsible on his simple goal to realize the group's goal, Fig. 5 (see Appendix (A)) shows that 16,67 % protest doing other roles to accomplish the programming tasks, we should investigate more in order to determinate witch roles could be defined to those defined before. Two roles (programmer, secretary) were done only by 13.89 % and 27.78 % of students respectively. Each one of these roles has its special criteria, required coding skills and abilities like formal pertinent writing and students were less confident and afraid of making errors and bugs. It can be explained that they have poor abilities to do such roles. 47.22 % of students were holding the role of coach imply that they were interacting more in the collaborative learning. Whereas the role of moderator was hold by 44.44 % that characterizes the engagement to encourage participation of all group members, and theorist were 44.44 % that provide confirmation that they understand well the concept being discussed and could easily explain it for others and only 41,67 % for planner role which reflects organizational feature in problem-solving for group learning. All Students might have already some individual skills or knowledge and could foster getting others by practicing, discussing and watching others doing it because they like to ask spontaneously their peers more than the teacher. Consequently, all percentages join the results shown above. As last, two short-questions format for this section, we asked student about benefits and inconveniences. Many students wrote that they were learning fast, getting great ideas form discussion, correction knowledge about wrong concepts' comprehension, working effortlessly without pressing, saving more time for writing algorithms and codes. Little minority seem to dislike this type of group work in distance because of difficulties in unreliability of some groups members, scheduling meetings, and while students divide up the programming task for a project, each do a code fragment, and then combine all pieces into a one finished program.

***Part 7: Students' Perceptions in Collaborative Production.*** That concern what was used in collaborative production as tools and technologies. Table 8 indicates some students' answers performance. It involves different components:

**On the proposed programming environment:** in this section, we demand students about the use of that was chosen in first place for several reasons principally because it is intuitive and intelligent as environment and can help beginners to master programming and lets them to quickly and easily write and change the code. Students shows their resistance and refuse strongly (only 13.89% said that is effective, 44.44% not effective and, 41.67% said it is not effective at all) this new environment because they preferred working with familiar platforms like Eclipse or Netbeans and that Intellij IDEA is not easy to use (16.67% claim that is easy to use and the rest 83.33% assumed the opposite). For the next question they argue about the choice of the programming environment depends on (the tool's mastery 16.67%, the tool's ease of use 38.89%, the community of developers 30.56%, The needs of the work to be solved 13.89%) that explain why they would not exploit Intellij IDEA). Since the student viewed the proposed programming environment as an instrument for programming rather than only the required lab session which they had to pass the computer examination. Besides, they administered various strategies as well as a long period in practicing outside their class for developing the better learning techniques.

**On tools used for communication and publishing tools:** The data collected and evaluated about student use of communication and publishing tools confirmed that the tools were very helpful and appreciated by students for collaborative group in distance learning. In class, the tools were not employed for two reasons: 1) no internet available in the lab session, 2) they do not need it because only one computer was assigned to each group. At first, we must notify that categories, who possess personal laptops, smart phone or other devices with internet, declared that the tools' use was very important. (27.78% were totally agree and 58.33% were agree), they needed to create Facebook groups to communicate via videos or calls and exchanging messages, discussing or at least to schedule meetings sessions and face-to-face appointments (63.89%), Github were used for displaying code fragments (22.23%). The other category were only 13,88% were connected to their colleagues by phone calls and proclaimed that Github was very sophisticated tool, consummate time to master and it deviate them from the principal purpose of programming.

**On the necessary use of Pedagogical Notebook:** The students may take notes, as a complementary learning activity for programming assignment. We decide to make it a good habit by requesting every group. Students responds (13.89% were totally agree, 63.89% were agree, 16.67% were disagree and 5.56% strongly disagree). That can be explicable: taking notes is a complex process from comprehension, selecting pertinent information in limited time. Other explanation, students replaced in most of the time PN as paper support by video, audio and images supports using its smart phones, justifying this action that as electronic support has better writing, and don't contain errors especially programming codes by answering the question:" Is a electronic format of PN better to handle than the paper format?", (36.11% were totally agree, 27.78% were agree, 30.56% were disagree and 5.56% strongly disagree).

**About the lab session room:** To enable students having modern teaching perspectives in lab rooms that allow collaborative learning, they were asked if they prefer working on their laptops instead of machines in the room. The majority of the students preferred their laptops (66.67% were totally agree, 16.67% were agree) to avoid problems like virus transferred with movable devices, and 16.67% were disagree because they do not have personal computers. Other questions regarding Classroom Structure or Computers Disposition form and the ideal arrangement for group work. They were all disagreed about the fixed computers tables and preferred (circle form 33.33%, the V form 52.78%, square form 11.11% and 2.78% wished for other flexible forms). Wi-Fi high-speed Internet access, interactive or intelligent board were the most wanted technologies for lab session. At minimum, students needed suitable spaces with audiovisual components for short demonstrations of tools installation, codes debugging, and lectures delivered during the session. They require also movable tables, computers and networks for higher collaborative organization supplies performance and an impressive room design to satisfying learning goals.

**About the teacher's accompaniment and assistance for lab session:** This might be divided on two: face-to-face and distance support. Students were thoroughly convinced that is the most important and necessary part in learning process. They were satisfied at 94.45% and 5.56% required for more than one teacher in lab sessions, they explain that the teacher should know the progress of each group globally and each student in detail in a special sheet assessment available for consulting every session.

***Part 8: Students' Perceptions of the quality and progress of their learning during Learning Scenario Activities (Global Pedagogical Scenario.*** The students were generally satisfied but it reveals some weakness in different phases. The final set of data related to the global scenario proposed above rejoin that is very large category of students faces enormous difficulties in different steps in problem-solving in engineering sciences, especially in first steps e.g. analysis and designing (writing algorithms). Consequently, it is reasonable to think that students want more interaction with their peers in such phase where they are meeting problems. Moreover, some phases are believed to excel at some tasks more so than others do, e.g., phase 1, phase 2. A minor misunderstanding leads to a greater number of defects. The students were interested in-group discussion by listening attentively.

| | Totally Agree | Agree | Disagree | Strongly disagree |
|---|---|---|---|---|
| Phase 1 | 27.78% | 41.67% | 27.78% | 2.78% |
| Phase 2 | 13.89% | 50% | 33.33% | 2.78% |
| Phase 3.1 | 38.89% | 52.78% | 8.33% | 0 % |
| Phase 3.2 | 41.67% | 38.89% | 19.44% | 0% |
| Phase 3.3 | 30.56% | 50% | 16.67% | 2.78% |
| Phase 3.4 | 47.22% | 50% | 0 % | 2.78% |
| Phase 4 | 17.14% | 31.43% | 48.57 % | 2.86% |

Table 5: Some students' responds performance for part 8 questions.

They collaborated in working with their peers for getting main idea of the exercise. We compared the two first lines with the two next ones, it clearly obvious that the positive and significant role of collaborative work to overcome with discussion the lake of understanding where a few students were disagree especially the best; they preferred not wasting time to explain to their peers, time they need to move on and start coding. It might be explained that they do not esteem the notion of sharing and acquire poor collaborative skills where the lower academic level students appear passive peers by holding the others back. Given the magnitude of the percentage in the last two steps in phase 3 regarding coding and debugging, however, we can reasonably assume that the lack of significance is not related to insufficient data, rather, it is because the behavior of most students in the class is very close to watcher behavior across all sessions. It could be concluded that the collaborative work could alleviate their anxiety in coding step, which would cause their low programming learning achievement. We confirm this by recoding weakness of programming as a gap between theory and practice.

In summary, much effort was expending in collaborative programming but, in other hand, the improved communication skills, enjoying lab sessions and reporting confidence in programming ability were the highest gains.

## 4.2 Assessments of programming improvement

In the context of this study, we use learning to refer to the improvement in programming skill, not knowledge of an abstract concept. Three different assessments in lab session for students are done in the 7th and 14th week: Individual CC1, Individual CC2 and Project Consultation, we calculated individual learning gain as follows:

In individual lab session LCC1 and LCC2: The individual LCC1 and LCC2 assessments were graded using a 10-point rubric. We examined ability for programming by checking source files that each student made in order to measure their programming skill as individual learning gain ILG using formula (1)

$$\mathrm{ILG} = (\mathrm{LCC1} + \mathrm{LCC2})/2 \qquad (1)$$

The Project Consultation PC: This phase endured at the 14th week. The students turned in final projects, presented it as groups. In the lab session room, they found helpful to use PowerPoint slides. After each presentation, the teacher used an oral questioning for each group and it was graded using a 5-point rubric, focusing on collaborative education outcomes. We can determinate three categories of groups: good, medium and lower performing groups depending on project achievement, active contribution of group's members, beating challenges found, etc. The CC score for Lab session LCC is calculated as (2) follow:

$$\mathrm{LCC} = \mathrm{ILG} + \mathrm{PC} + \mathrm{P} \qquad (2)$$

P refers to a score using a 5-point rubric relative to the students' attendance. From the results, initials analysis has shown three different categories: High, Average, and low scores. We compared Initial Test, Lab session LCC, Tutorials TCC and the final exam FE.

## 4.3 Assessments findings

For a good interpretation the results of this study, we remember that every assessment is done differently. The final exam, which constituted 50% of the course grade, consisted of 5 simple questions and 4 exercises including: identifying Java vocabulary, tracing source code outcomes, and implement a code. The TCC grades represent 25% of the course grade equally with LCC grades. The TCC was managed differently from LCC; it consisted of the average of two TCC1 and TCC2 assessment based on implementing a simple code each on paper in 20 minutes period. The final course grade was calculated on the weighted percentage of all assessment activities (FE, TCC and LCC). Therefore, the main reason for comparing those results is to recognize the improvement in computational thinking and programming skills and style in collaborative learning. Fig. 6 displays the key predictors of performance in relation to assessment results in three levels. Starting with IT, at the beginning of the semester, students have low performance level (62.03% of Low performance), only 7.40% have High performance and 30.55% as Average performance. As can be seen, LCC results, correlated with collaborative programming, were positively significant.

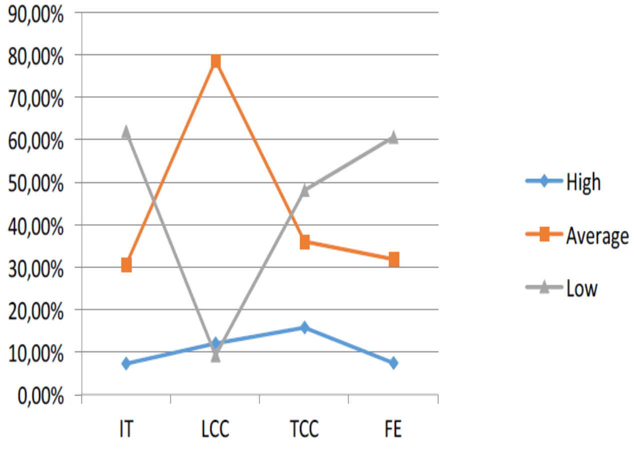


Figure 6: Improvement of students' results across different assessments.

The average level, which represented 78.70 % of the students, dominate the results with the lowest percentage 9.25 % of students in low level that shows a downward change. The percentage of the total number of students in Average performance has jumped highly from 30.55% to 78.70%. In High performance, it moved 7.40% to 12.05%. For TCC, we observe some drop in average and low levels with a small upward change in high level. Next in the transition from learning to the last assessment form, accompanied by teachers' surveillance and exam stress, we again observe poor and negative students performance across different levels.

# 5 Conclusion

This paper has investigated the educational effectiveness of a collaborative learning design in object-oriented programming language lab session at the university Mostafa Stambouli Mascara and students' satisfaction about different related issues. The results obtained allow us to claim that group design, global pedagogical scenario and especially roles division combined with technologies tools are the pedagogical methodology that produce in students. Furthermore, engagement, commitment and motivation of students increased at each phase of programming process.

Besides verifying that the global scores, the academic level increase was verified among the students that had an initial lower academic level. Collaborative learning groups exhibit new social skills in all students. Higher academic level students were the most resistant to collaboration at the beginning of the experiment. It was also observed in these groups that students with low academic level tended to be passive towards high academic level students. But the connection takes place only while the teacher intervenes to weld broken the groups and encouraged the participation of all group members. However, other students were more motivated and collaborative. The teacher assistance was very important, helping students groups to accept critical and constructive feedback reinforce their learning inside and outside lab session room. The experiments also verified that in absence of the teacher, some groups were not effective outdoors lab session, and all members count on high academic level students to do all the work. By contrast, high academic level students impose their solutions for the other members without any discussion. In the same time, it conducts to unfinished discussion, where each member believed that he has the right point. Furthermore, dividing tasks and roles made the learning less difficult and more agreeable; students with high determination continue to work in groups in other courses since they build trust and solid membership. Other courses' teachers mention, in meetings, the good change of students' programming style who still works collaboratively. Selecting an appropriate delivery format becomes even more pronounced when the students are engaged in a collaborative learning environment. Designing a collaborative learning environment should include all element cited before. Obviously, learning becomes more driven by ICT tools. Facebook and Github are emerging technologies that supported students in the successful achievement of collaborative blended learning. Teachers inspected all source codes at the end of the experiment reported by data indicating that collaborative group has an affirmative influence on learning how to program, at least shows improving of students programming style. In sum, our results suggest that ICT tools must be added to the pedagogical methodology to increase the learning progress, but some new suggested tools faced by a large resistance and show opposite expectations. Until this stage, we have not discussed the choice of Java as a programming language because, firstly, it is a powerful language, then students had already studied the C language and it is mandatory in the minister official program. Since our students learned programming in first year and their lower performance level, we believe that we should prepare them as native programmers programming for second-year java collaborative programming for better results.

**Limitations:** The entire design is very exhausting practice from the beginning of the semester to the end, from initial tests to sending the final exam scores. So many difficulties demotivate both teachers and students in absence of minor ICT technologies, time allocated, networks, and management contesting often that conduct to discouragement for teachers and poor emerging skills for students.

**Perspectives:** Finally, working with limited available components infect on learning methodologies, thus it is an open issue for the future. Roles proposed for students in the pedagogical scenario should be studied carefully. The metacognitive abilities behind every task in every role are strongly related; it is essential to inspect and explain witch poor skills behind role were less picked. Our vision is to create a platform for learning Objet-Oriented programming language with all communication and visualization features that allows collaborative teaching and learning programming. We have to unite inspiring all students with clear and visible guidance to have equal opportunities to reach their full potential. Helping our students overcome their obstacles, in order to raise the level of participation in collaborative work, keep it moving forward in order to become more competitive in programming.

# 6 References

[1] J. Carter, P. Dewan, and M. Pichiliani, "Towards Incremental Separation of Surmountable and Insurmountable Programming Difficulties," in *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, New York, NY, USA, 2015, pp. 241–246. https://doi.org/10.1145%2F2676723.2677294

[2] D. Sleeman, "The Challenges of Teaching Computer Programming," *Commun. ACM*, vol. 29, no. 9, pp. 840–841, Sep. 1986. https://doi.org/10.1145%2F6592.214913

[3] P. H. Tan, C. Y. Ting, and S. W. Ling, "Learning Difficulties in Programming Courses: Undergraduates' Perspective and Perception," in *2009 International Conference on Computer Technology and Development*, 2009, vol. 1, pp. 42–46. https://doi.org/10.1109%2FICCTD.2009.188

[4] H. Belhandouz, "Teaching science in Algeria: pedagogical shortfalls and conflicts of meaning," *The Journal of North African Studies*, vol. 16, no. 1, pp. 99–116, Mar. 2011. https://doi.org/10.1080%2F13629387.2010.529655

[5] A. V. P. Arboleda, L. R. R. Mazuera, and G. S. Montemiranda, "Competences that facilitate the achievement of the objectives of an introductory programming course," in *2015 International Conference on Interactive Collaborative Learning (ICL)*, 2015, pp. 1007–1012. https://doi.org/10.1109%2FICL.2015.7318166

[6] A. Robins, J. Rountree, and N. Rountree, "Learning and Teaching Programming: A Review and Discussion," *Computer Science Education*, vol. 13, no. 2, pp. 137–172, Jun. 2003. https://doi.org/10.1076%2Fcsed.13.2.137.14200

[7] C. Baron and L. Baron, "Trois approches d'apprentissage collaboratif dans l'action pour soutenir le développement du leadership," *Humain et organisation*, vol. 1, no. 2, pp. 24–32, Nov. 2015.

[8] G. Alexandron, M. Armoni, M. Gordon, and D. Harel, "Scenario-based Programming: Reducing the Cognitive Load, Fostering Abstract Thinking," in *Companion Proceedings of the 36th International Conference on Software Engineering*, New York, NY, USA, 2014, pp. 311–320. https://doi.org/10.1145%2F2591062.2591167

[9] B. Isong, "A Methodology for Teaching Computer Programming: first year students' perspective," *International Journal of Modern Education and Computer Science(IJMECS)*, vol. 6, no. 9, p. 15, Sep. 2014.

[10] M. Hauswirth and A. Adamoli, "Teaching Java programming with the Informa clicker system," *Science of Computer Programming*, vol. 78, no. 5, pp. 499–520, May 2013.

https://doi.org/10.1016%2Fj.scico.2011.06.006

[11] G. Fesakis and K. Serafeim, "Influence of the Familiarization with 'Scratch' on Future Teachers' Opinions and Attitudes About Programming and ICT in Education," in *Proceedings of the 14th Annual ACM SIGCSE Conference on Innovation and Technology in Computer Science Education*, New York, NY, USA, 2009, pp. 258–262. https://doi.org/10.1145%2F1562877.1562957

[12] S. Mishra and S. Iyer, "Problem Posing Exercises (PPE): An Instructional Strategy for Learning of Complex Material in Introductory Programming Courses," in *2013 IEEE Fifth International Conference on Technology for Education (t4e 2013)*, 2013, pp. 151–158. https://doi.org/10.1109%2FT4E.2013.45

[13] A. Abdullah and T. Y. Yih, "Implementing Learning Contracts in a Computer Science Course as a Tool to Develop and Sustain Student Motivation to Learn," *Procedia - Social and Behavioral Sciences*, vol. 123, pp. 256–265, Mar. 2014. https://doi.org/10.1016%2Fj.sbspro.2014.01.1422

[14] J. M. Rodríguez Corral, A. Civit Balcells, A. Morgado Estévez, G. Jiménez Moreno, and M. J. Ferreiro Ramos, "A game-based approach to the teaching of object-oriented programming languages," *Computers & Education*, vol. 73, pp. 83–92, Apr. 2014. https://doi.org/10.1016%2Fj.compedu.2013.12.013

[15] H. Cigdem, "How Does Self-Regulation Affect Computer-Programming Achievement in a Blended Context?," *Contemporary Educational Technology*, vol. 6, no. 1, pp. 19–37, Mar. 2015.

[16] C.-H. Chen, K.-C. Wang, and Y.-H. Lin, "The Comparison of Solitary and Collaborative Modes of Game-based Learning on Students' Science Learning and Motivation.," *Educational Technology & Society*, vol. 18, no. 2, pp. 237–248, 2015.

[17] M. Maleko, D. Nandi, M. Hamilton, D. D'Souza, and J. Harland, "Facebook versus Blackboard for Supporting the Learning of Programming in a Fully Online Course: The Changing Face of Computing Education," in *2013 Learning and Teaching in Computing and Engineering*, 2013, pp. 83–89. https://doi.org/10.1109%2FLaTiCE.2013.31

[18] C. Z. Kertész, "Using GitHub in the classroom - a collaborative learning experience," in *2015 IEEE 21st International Symposium for Design and Technology in Electronic Packaging (SIITME)*, 2015, pp. 381–386. https://doi.org/10.1109%2FSIITME.2015.7342358

[19] T. Kasame, K. Pachoen, and A. Manit, "The use of engineering design concept for computer programming course: A model of blended learning environment," *Educational Research and Reviews*, vol. 11, no. 18, pp. 1757–1765, Sep. 2016. https://doi.org/10.5897%2FERR2016.2948

[20] G. Ion, A. Barrera-Corominas, and M. Tomàs-Folch, "Written peer-feedback to enhance students' current and future learning," *International Journal of Educational Technology in Higher Education*, vol. 13, no. 1, Dec. 2016. https://doi.org/10.1186%2Fs41239-016-0017-y

[21] A. Othman, A. Impes, C. Pislaru, and D. Wilson, "Virtual Lab for Teaching a Computer Programming Course," *Online Journal of Art and Design*, vol. 2, no. 4, pp. 42–54, 2014.

[22] G. Álvarez and L. Bassa, "ICTs and collaborative learning: a case study of a class blog for improving the writing skills of pre-university students," *RUSC. Universities and Knowledge Society Journal*, vol. 10, no. 2, p. 5, Jul. 2013. https://doi.org/10.7238%2Frusc.v10i2.1740

[23] J. Owolabi, O. A. Adedayo, A. O. Amao-Kehinde, and T. A. Olayanju, "Effects of Solo and Pair Programming Instructional Strategies on Students' Academic Achievement in Visual-Basic.Net Computer Programming Language," *GSTF Journal on Computing (JoC)*, vol. 3, no. 3, Aug. 2014.

[24] A.-B. Al-Azawei, "Evaluating the Effect of Arabic Engineering Students' Learning Styles in Blended Programming Courses," *Journal of Information Technology Education: Research*, vol. 15, pp. 109–130, 2016.

[25] B. Estácio *et al.*, "Evaluating Collaborative Practices in Acquiring Programming Skills: Findings of a Controlled Experiment," in *2015 29th Brazilian Symposium on Software Engineering*, 2015, pp. 150–159. https://doi.org/10.1109%2FSBES.2015.24

[26] S. Hamid, J. Waycott, S. Kurnia, and S. Chang, "Understanding students' perceptions of the benefits of online social networking use for teaching and learning," *The Internet and Higher Education*, vol. 26, pp. 1–9, 2015. https://doi.org/10.1016%2Fj.iheduc.2015.02.004

[27] S. M. Alyami and A. M. Alagab, "The Difference in Learning Strategies in Virtual Learning Environment and Their Effect on Academic Achievement and Learning Satisfaction for Distance Teaching & Training Program Students," in *2013 Fourth International Conference on e-Learning "Best Practices in Management, Design and Development of e-Courses: Standards of Excellence and Creativity,"* 2013, pp. 102–112. https://doi.org/10.1109%2FECONF.2013.40

[28] A. Azemi, M. Bodek, and G. Chinn, "Teaching an introductory programming course using hybrid e-learning approach," in *2013 IEEE Frontiers in Education Conference (FIE)*, 2013, pp. 1911–1913. https://doi.org/10.1109%2FFIE.2013.6685168

[29] M. E. Muuro, R. O. Oboko, and P. W. Wagacha, "Evaluation of Intelligent Grouping Based on Learners' Collaboration Competence Level in Online Collaborative Learning Environment," *The International Review of Research in Open and Distributed Learning*, vol. 17, no. 2, Mar. 2016. https://doi.org/10.19173%2Firrodl.v17i2.2066

[30] J. Dwarika and M. R. R. de Villiers, "Use of the Alice Visual Environment in Teaching and

Learning Object-oriented Programming," in *Proceedings of the 2015 Annual Research Conference on South African Institute of Computer Scientists and Information Technologists*, New York, NY, USA, 2015, pp. 14:1–14:10. https://doi.org/10.1145%2F2815782.2815815

[31] Y. Hayashi, K. I. Fukamachi, and H. Komatsugawa, "Collaborative Learning in Computer Programming Courses That Adopted the Flipped Classroom," in *2015 International Conference on Learning and Teaching in Computing and Engineering*, 2015, pp. 209–212. https://doi.org/10.1109%2FLaTiCE.2015.43

[32] N. Funabiki, Y. Korenaga, Y. Matsushima, T. Nakanishi, and K. Watanabe, "An Online Fill-in-the-Blank Problem Function for Learning Reserved Words in Java Programming Education," in *2012 26th International Conference on Advanced Information Networking and Applications Workshops*, 2012, pp. 375–380. https://doi.org/10.1109%2FWAINA.2012.64

[33] H. Jonsson, "Using flipped classroom, peer discussion, and just-in-time teaching to increase learning in a programming course," in *2015 IEEE Frontiers in Education Conference (FIE)*, 2015, pp. 1–9. https://doi.org/10.1109%2FFIE.2015.7344221

[34] M. Husain, N. Tarannum, and N. Patil, "Teaching programming course elective: A new teaching and learning experience," in *2013 IEEE International Conference in MOOC, Innovation and Technology in Education (MITE)*, 2013, pp. 275–279. https://doi.org/10.1109%2FMITE.2013.6756349

[35] F. Silvestre, P. Vidal, and J. Broisin, "Tsaap-Notes – An Open Micro-blogging Tool for Collaborative Notetaking during Face-to-Face Lectures," in *2014 IEEE 14th International Conference on Advanced Learning Technologies*, 2014, pp. 39–43. https://doi.org/10.1109%2FICALT.2014.22

[36] S. L. Wismath and D. Orr, "Collaborative Learning in Problem Solving: A Case Study in Metacognitive Learning," *Canadian Journal for the Scholarship of Teaching and Learning*, vol. 6, no. 3, 2015. https://doi.org/10.5206%2Fcjsotl-rcacea.2015.3.10

[37] J. Olivier, "Blended learning in a first-year language class: Evaluating the acceptance of an interactive learning environment," *ResearchGate*, vol. 37, no. 2, Oct. 2016. https://doi.org/10.4102%2Flit.v37i2.1288

[38] M.-I. Dascalu, C.-N. Bodea, A. Moldoveanu, A. Mohora, M. Lytras, and P. O. de Pablos, "A recommender agent based on learning styles for better virtual collaborative learning experiences," *Computers in Human Behavior*, vol. 45, pp. 243–253, Apr. 2015. https://doi.org/10.1016%2Fj.chb.2014.12.027

[39] G. Temperman, B. De Lièvre, P. Bossaert, and J. De Stercke, "Effets de la distribution de rôles selon le style d'apprentissage dans un environnement collaboratif à distance," in *TICE 2010*, Nancy, France, 2010, p. 6.

[40] J. Andriessen, M. Baker, and D. Suthers, Eds., Arguing to Learn: Confronting Cognitions in Computer-Supported Collaborative Learning Environments. Springer Netherlands, 2003.

[41] A. Zagalsky, J. Feliciano, M.-A. Storey, Y. Zhao, and W. Wang, "The Emergence of GitHub As a Collaborative Platform for Education," in *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, New York, NY, USA, 2015, pp. 1906–1917. https://doi.org/10.1145%2F2675133.2675284

[42] F. J. Rodríguez, K. M. Price, and K. E. Boyer, "Exploring the Pair Programming Process: Characteristics of Effective Collaboration," in *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, New York, NY, USA, 2017, pp. 507–512. https://doi.org/10.1145%2F3017680.3017748

[43] P. Dillenbourg, Over-scripting CSCL: The risks of blending collaborative learning with instructional design. Heerlen, Open Universiteit Nederland, 2002.

[44] C. Armstrong, "Catalyzing Collaborative Learning: How Automated Task Distribution May Prompt Students to Collaborate," *E-Learning and Digital Media*, vol. 7, no. 4, pp. 407–415, Dec. 2010. https://doi.org/10.2304%2Felea.2010.7.4.407

[45] K. Prahl, "Best Practices for the Think-Pair-Share Active-Learning Technique," *The American Biology Teacher*, vol. 79, no. 1, pp. 3–8, Dec. 2016. https://doi.org/10.1525%2Fabt.2017.79.1.3

[46] I. N. Umar and T. H. Hui, "Learning Style, Metaphor and Pair Programming: Do they Influence Performance?," *Procedia - Social and Behavioral Sciences*, vol. 46, pp. 5603–5609, 2012. https://doi.org/10.1016%2Fj.sbspro.2012.06.482

[47] S. Gross and N. Pinkwart, "Towards an Integrative Learning Environment for Java Programming," in *2015 IEEE 15th International Conference on Advanced Learning Technologies*, 2015, pp. 24–28. https://doi.org/10.1109%2FICALT.2015.75

[48] B. J. Parra, "Learning strategies and styles as a basis for building personal learning environments," *International Journal of Educational Technology in Higher Education*, vol. 13, no. 1, Dec. 2016. https://doi.org/10.1186%2Fs41239-016-0008-z

[49] G.-H. Hwang, B. Chen, and C.-W. Huang, "Development and Effectiveness Analysis of a Personalized Ubiquitous Multi-Device Certification Tutoring System Based on Bloom's Taxonomy of Educational Objectives," *Journal of Educational Technology & Society*, vol. 19, no. 1, pp. 223–236, 2016.

[50] A. H. Seyal, Y. S. Mey, M. H. Matusin, N. H. Siau, and A. A. Rahman, "Understanding Students Learning Style and Their Performance in Computer Programming Course: Evidence from Bruneian Technical Institution of Higher Learning," *International Journal of Computer Theory and Engineering*, vol. 7, no. 3, p. 241, 2015.

[51] C. Ferraris, A. Lejeune, L. Vignollet, and J.-P. David, "Modélisation de scénarios pédagogiques collaboratifs." 28-Jun-2005.

[52] G. Paquette, F. Crevier, C. Aubin, J. Rocheleau, C. Paquin, and M. Léonard, "Méthode d'ingénierie d'un système d'apprentissage," *Revue Informations In Cognito*, vol. 8, p. 1997, 1997.

[53] C. Yáñez-Aldecoa, A. Okada, and R. Palau, "New learning scenarios for the 21st century related to Education, Culture and Technology," *RUSC. Universities and Knowledge Society Journal*, vol. 12, no. 2, p. 87, Apr. 2015. https://doi.org/10.7238%2Frusc.v12i2.2454

[54] J.-P. Pernin and A. Lejeune, "Dispositifs d'apprentissage instrumentés par les technologies : vers une ingénierie centrée sur les scénarios," presented at the Technologies de l'Information et de la Connaissance dans l'Enseignement Supérieur et de l'Industrie, 2004, pp. 407–414.

[55] W. Debabi and T. Bensebaa, "Using serious game to enhance learning and teaching algorithmic," *Journal of e-Learning and Knowledge Society*, vol. 12, no. 2, 2016.

[56] N. Mamat and N. Yusof, "Learning Style in a Personalized Collaborative Learning Framework," *ResearchGate*, vol. 103, pp. 586–594, Nov. 2013. https://doi.org/10.1016%2Fj.sbspro.2013.10.376

[57] A. R. Pacios Lozano and G. Bueno de la Fuente, "Teamwork and leadership in a virtual learning environment," *RUSC. Universities and Knowledge Society Journal*, vol. 10, no. 2, p. 112, Jul. 2013. https://doi.org/10.7238%2Frusc.v10i2.1452

# 7 Appendix (A)

| | Absolutely Yes | May be | I do not | Not at all |
|---|---|---|---|---|
| Do you know the E-learning platforms | 27.27% | 22.73% | 20.45% | 29.55% |
| Do you think that the online courses help you in your learning | 31.82% | 34.09% | 15.91% | 18.18% |
| | Course format | Reading list | Online discussion | Online tests |
| Do you think that the content of these courses must be changed and / or must contain other things | 75% | 18.18% | 59.09% | 43.18% |
| | No course | 1-2 | 2-4 | + 4 |
| How many courses have you studied in ICT | 9.09% | 81.82% | 4.55% | 4.55% |

Table 3: Students' responds performance for part 2 questions.

| | Very important | Important | Not important | Not important at all |
|---|---|---|---|---|
| Collaboration within the group | 36.11% | 58 .33% | 5.56% | 0.00% |
| | Very effective | Effective | Not effective | Not effective at all |
| Coordination within the group:<br>1. Tasks' assignment | 27.78% | 58 .33% | 13.89% | 0.00% |
| 2. Time management | 8.33% | 27.78% | 63.89% | 0.00% |
| 3. Partition into subgroups | 25.00% | 41.67% | 33.33% | 0.00% |
| | Very important | Important | Not important | Not important at all |
| Communication within the group:<br>1. Listening among the groups' members | 33.33% | 58 .33% | 8.33% | 0.00% |
| 2. Fast execution of tasks | 36.11% | 61.11% | 2.78% | 0.00% |
| 3. Exchanges of ideas, tasks | 16.67% | 27.78% | 55.56% | 0.00% |

Table 4: Performance for part 4 questions students' responds.

| | Totally agree | Agree | Disagree | Strongly disagree |
|---|---|---|---|---|
| I consider the students' number per group suitable. | 47.22% | 25.00% | 27.78% | 0.00% |
| I get along with my partners in the same group. | 33.33% | 47.22% | 19.44% | 0.00% |
| When the group meets a conflict case, it is generally easily to overcome. | 16.67% | 52.78% | 30.56% | 0.00% |
| I prefer choosing my group's mate | 80.56% | 19.44% | 0.00% | 0.00% |
| I prefer students' same academic level in each group | 8.33% | 27.78% | 61.11% | 2.78% |
| I prefer students' different academic level in each group | 16.67% | 58.33% | 22.22% | 2.78% |

Table 5: Performance for part 5 questions students' responds.

| | Totally agree | Agree | Disagree | Strongly disagree |
|---|---|---|---|---|
| Does the distribution of roles facilitate the realization of collaborative work? | 27.78% | 72.22% | 0.00% | 0.00% |
| What criteria could strengthen the group work? | 44.44% | 41.67% | 47.22% | 100% |
| Do roles change according to the nature of the work requested? | 25.00% | 55.56% | 16.67% | 2.78% |
| I got involved significantly in my role | 41.67% | 47.22% | 16.67% | 0.00% |
| How do you evaluate the involvement of your classmates in their roles? | 19.44% | 58.33% | 25.00% | 0.00% |

Table 6: Some students' responds performance for part 6 questions.

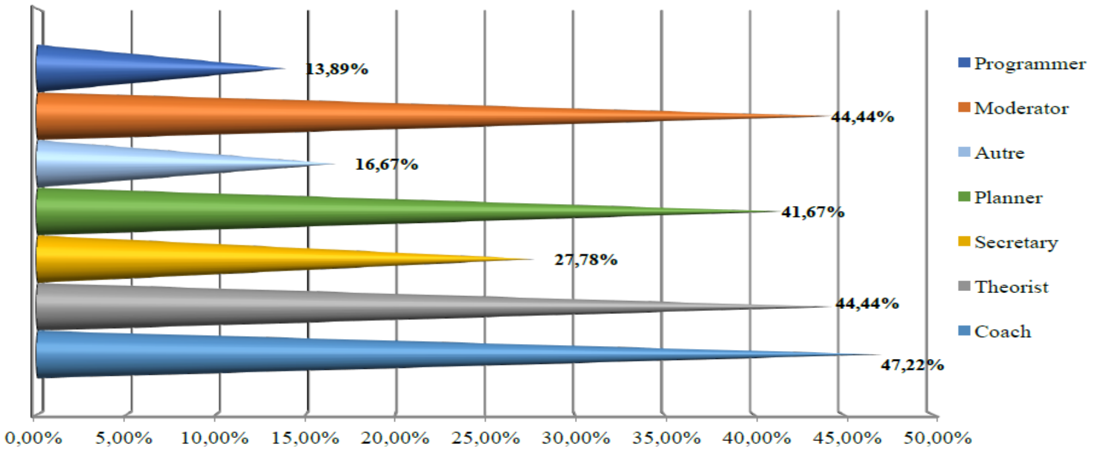


Figure 5: Roles' percentage held by students.

https://doi.org/10.31449/inf.v43i1.2657

# Microworlds with Different Pedagogical Approaches in Introductory Programming Learning: Effects in Programming Knowledge and Logical Reasoning

Joana Martinho Costa
Instituto Universitário de Lisboa (ISCTE-IUL), ISTAR-IUL, Lisboa, Portugal
E-mail: joana.martinho.costa@gmail.com

**Thesis summary**

**Keywords:** microworlds, introductory programming learning, 4C/ID model, expository teaching

**Received:** January 21, 2019

*This paper presents a summary of doctoral thesis that evaluates the using of Alice microworld combined with different pedagogical approaches to learn introductory programming, specifically in programming knowledge and logical reasoning variables. The results suggest that when microworlds are combined with the 4C-ID instructional model there are positive effects in learning to program a computer, but they have no impact if an expository method is used. However, given the limited number of sampled subjects in this thesis, this topic should be further researched. It is necessary to understand to what extent, measurable, the use of certain teaching methods with or without association with microworlds, influence the results of students in knowledge and logical reasoning.*

*Povzetek: Opisano je doktorsko delo na področju poučevanja osnovnih programskih veščin.*

## 1 Introduction

The scientific community has been developing several proposals to overcome the inherent difficulties in the initial learning of computer programming. One of the solutions is the use of microworlds. Faced with this problem, some solutions have been developed over the past few years to help overcome student difficulties. One of the proposed solutions is composed of the use of microworlds, such as Alice. This software aims to help and motivate students to learn how to program and was created about 15 years ago. Since then, it has been updated and several versions have emerged [1].

The Alice software is considered a microworld for programming because it respects the characteristics of Papert's microworld [2], namely: i) the possibility of creating simple examples in the microworld itself, in this case programming examples; ii) there should be no obstacles in the manipulation of the microworld, with the possibility of creating activities, games and art within the microworld; and iii) the content needed for learning should be able to be defined within the microworld.

The present research organized into a thesis [3] with four sequential chapters, had the objective of understanding the effects of using Alice microworld, in programming learning when associated with certain pedagogical methods, specifically in the programming knowledge and in the logical reasoning, with a focus on vocational education at Portugal.

## 2 Methodology

The research process began with the typification of the Alice software into microworld in order to extend the thesis conclusion to other microworlds.

The next step was conducted a systematic literature review and meta-analysis in order to understand the effects of Alice in programming learning on previous studies [4]. The combination of studies indicated promising results in the field (d = 0.541), but presented some weaknesses, where we highlight: (i) significant variation in data collection instruments, (ii) and the design of the studies, which do not specify the teaching strategies used to teach computer programming. In addition, there was no analysis of whether the use this microworld has a different impact has when students differ in cognitive and social characteristics, as these variables were also not isolated in any of the studies considered in the meta-analysis.

Therefore, we pursued the research by building a valid and reliable instrument, in Portuguese language, to evaluate the initial learning of programming in quasi-experimental studies [5]. This instrument was constituted by multiple-choice and development questions and the difficulty level of the items was based on Bloom's Taxonomy. The test was subsequently validated by a group of specialists and submitted to a pilot-group. This process concluded that the instrument is valid and reliable (alfa = 0.837).

Lastly, we developed two quasi-experimental studies.

In the first one [6], the experimental group (n = 11) used programming concepts with the Alice software and pseudocode, while the control group (n = 11) learned the same concepts without resorting to software. Both were taught using expository teaching, the most commonly used method in Portugal [7]. In the first experimental study, based on the characteristics of a quasi-experimental design, we defined the following procedures:

1) Data collection on the characteristics of the participants, application of a survey about the interest in programming and application of the Echelle Collective de Développement Logique (ECDL) [8], a scale that classifies the student thinking according to the five stages of Jean Piaget's theory on cognitive thinking.;
2) Experimental treatment: Introductory programming learning using an expository teaching combined with Alice microworld in the experimental group and the same conventional model without Alice, for four weeks with seven hours of programming per week
3) Application of the created test [5] after experimental treatment to both groups.

In the second study [9], computer program knowledge and development of logical reasoning were compared between two groups, which differed in logical reasoning and socioeconomic status, after they learned programming with Alice microworld associated to the Four Component Instructional Design Model (4C-ID). We followed next procedures:

1) Data collection on the characteristics of the participants, application of a questionnaire on the interest of programming and application of the ECDL [7];
2) Experimental treatment: Initial programming learning using the 4C/ID instructional model combined with Alice software, for eight weeks with 30 classes of 50 minutes, in both groups;
3) After experimental treatment, application of the knowledge test and the ECDL to both groups.

## 3 Results

In the first study, the average score in the programming test, applied after the experimental treatment when covaried with the average score on the Logical Development Scale, applied before to the experimental treatment, was positive in both groups with no statistically significant difference between them (F(1,19)=0.984; p > 0.05). When the ECDL covariable is maintained at its average value (11.95), the group that used Alice obtained a mean result in the programming test (6.073 +- 0.834) lower than the group that exclusively used pseudocode (7,290 +- 0.834). The analysis of planned contrasts between the experimental and control group reveals that the differences between the approaches (Alice with pseudocode or exclusive pseudocode) are not significant (t(19) = 0.992; p > 0.05).

In the second study, the average score in the programming test, applied after the experimental treatment, when covaried with the average score on the Logical Development Scale (ECDL) prior to experimental treatment was positive in both groups, with no statistically significant difference between them (p > 0.05). Total ECDL score before and after experimental treatment differed positively in the control group (n = 8; p < 0.05) and experimental group, although here the difference was not statistically significant (n = 6; p > 0.05). The experimental group in this second study with lower socioeconomic status obtained the highest computer test average among the four groups in the two studies.

## 4 Conclusions and future work

This study aimed to understand the effects of the use of microworlds, when associated with different teaching methodologies, in the introductory programming learning. To answer this question, we used Alice microworld in quasi-experimental work.

The results suggested that when Alice microworld is combined with the 4C-ID instructional model there are positive effects in learning to program a computer, but this software have no impact if an expository method is used. However, given the limited number of sampled subjects, this topic should be further researched.

Finally, we suggest conducting experimental research that continues the work initiated in this thesis. It is necessary to understand the extent, measurable, the use of certain teaching methods with or without association with microworlds, influence the results of students in knowledge and logical reasoning. In future research, the same measuring instruments present in this study should be used, in groups with similar characteristics, in order to compare the results obtained with those described in this thesis.

## References

[1] Alice Project. (2018). Alice – Tell Stories. Build Games. Learn to Program. Retrieved from http://www.alice.org.

[2] Papert, S. (1985). Different visions of logo. Computers in the schools. *Interdisciplinary Journal of Practice, Theory, and Applied Research,* 2(2–3), 3–8.

[3] Costa, J.M. (2019). *Utilização do micromundo Alice na aprendizagem da programação: Efeitos sobre os conhecimentos e o raciocínio lógico* (Unpublished doctoral dissertation). Institute of Education, University of Lisbon, Lisbon, Portugal.

[4] Costa, J. M., & Miranda, G. L. (2017). Relation between Alice software and programming learning: A systematic review of the literature and meta-analysis. *British Journal of Educational Technology, 48*(6), 1464–1474. doi: https://doi.org/10.1111/bjet.12496

[5] Costa, J. M., & Miranda, G. L. (2017). Desenvolvimento e validação de uma prova de avaliação das competências iniciais de programação. *RISTI - Revista Ibérica de Sistemas e Tecnologias de Informação*, 25, 66-81. https://dx.doi.org/10.17013/risti.25.66-81

[6] Costa, J. M. (2018). Using Alice Software with the Expository Method: a pilot study. *International Journal of Engineering and Technology*, 10(6),

1681-1686. doi: https://doi.org/10.21817/ijet/2018/v10i6/181006046

[7] Azevedo, J., & Portela (2014, November 29). Como se define o contexto de uma escola e aquilo que se espera dela? *O Público*. Retrieved from https://www.publico.pt/2014/11/29/sociedade/analise/contexto-e-valor-esperado-do-contexto-1677516

[8] Hornemann, J. (1974). Influence du contenu sur la résolution de problèmes logiques. *Enfance*, 27, 45-64.

[9] Costa, J. M., & Miranda, G. L. (2019). Using Alice Software with 4C-ID Model: Effects in Programming Knowledge and Logical Reasoning. Informatics in Education, 18(1), 1-15.
doi: https://doi.org/10.15388/infedu.2019.01.

# JOŽEF STEFAN INSTITUTE

*Jožef Stefan (1835-1893) was one of the most prominent physicists of the 19th century. Born to Slovene parents, he obtained his Ph.D. at Vienna University, where he was later Director of the Physics Institute, Vice-President of the Vienna Academy of Sciences and a member of several scientific institutions in Europe. Stefan explored many areas in hydrodynamics, optics, acoustics, electricity, magnetism and the kinetic theory of gases. Among other things, he originated the law that the total radiation from a black body is proportional to the 4th power of its absolute temperature, known as the Stefan–Boltzmann law.*

The Jožef Stefan Institute (JSI) is the leading independent scientific research institution in Slovenia, covering a broad spectrum of fundamental and applied research in the fields of physics, chemistry and biochemistry, electronics and information science, nuclear science technology, energy research and environmental science.

The Jožef Stefan Institute (JSI) is a research organisation for pure and applied research in the natural sciences and technology. Both are closely interconnected in research departments composed of different task teams. Emphasis in basic research is given to the development and education of young scientists, while applied research and development serve for the transfer of advanced knowledge, contributing to the development of the national economy and society in general.

At present the Institute, with a total of about 900 staff, has 700 researchers, about 250 of whom are postgraduates, around 500 of whom have doctorates (Ph.D.), and around 200 of whom have permanent professorships or temporary teaching assignments at the Universities.

In view of its activities and status, the JSI plays the role of a national institute, complementing the role of the universities and bridging the gap between basic science and applications.

Research at the JSI includes the following major fields: physics; chemistry; electronics, informatics and computer sciences; biochemistry; ecology; reactor technology; applied mathematics. Most of the activities are more or less closely connected to information sciences, in particular computer sciences, artificial intelligence, language and speech technologies, computer-aided design, computer architectures, biocybernetics and robotics, computer automation and control, professional electronics, digital communications and networks, and applied mathematics.

The Institute is located in Ljubljana, the capital of the independent state of S**love**nia (or S♡nia). The capital today is considered a crossroad between East, West and Mediterranean Europe, offering excellent productive capabilities and solid business opportunities, with strong international connections. Ljubljana is connected to important centers such as Prague, Budapest, Vienna, Zagreb, Milan, Rome, Monaco, Nice, Bern and Munich, all within a radius of 600 km.

From the Jožef Stefan Institute, the Technology park “Ljubljana” has been proposed as part of the national strategy for technological development to foster synergies between research and industry, to promote joint ventures between university bodies, research institutes and innovative industry, to act as an incubator for high-tech initiatives and to accelerate the development cycle of innovative products.

Part of the Institute was reorganized into several high-tech units supported by and connected within the Technology park at the Jožef Stefan Institute, established as the beginning of a regional Technology park "Ljubljana". The project was developed at a particularly historical moment, characterized by the process of state reorganisation, privatisation and private initiative. The national Technology Park is a shareholding company hosting an independent venture-capital institution.

The promoters and operational entities of the project are the Republic of Slovenia, Ministry of Higher Education, Science and Technology and the Jožef Stefan Institute. The framework of the operation also includes the University of Ljubljana, the National Institute of Chemistry, the Institute for Electronics and Vacuum Technology and the Institute for Materials and Construction Research among others. In addition, the project is supported by the Ministry of the Economy, the National Chamber of Economy and the City of Ljubljana.

Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
Tel.:+386 1 4773 900, Fax.:+386 1 251 93 85
WWW: http://www.ijs.si
E-mail: matjaz.gams@ijs.si
Public relations: Polona Strnad

# INFORMATICA

## AN INTERNATIONAL JOURNAL OF COMPUTING AND INFORMATICS

## INVITATION, COOPERATION

### Submissions and Refereeing

Please register as an author and submit a manuscript at: http://www.informatica.si. At least two referees outside the author's country will examine it, and they are invited to make as many remarks as possible from typing errors to global philosophical disagreements. The chosen editor will send the author the obtained reviews. If the paper is accepted, the editor will also send an email to the managing editor. The executive board will inform the author that the paper has been accepted, and the author will send the paper to the managing editor. The paper will be published within one year of receipt of email with the text in Informatica MS Word format or Informatica LaTeX format and figures in .eps format. Style and examples of papers can be obtained from http://www.informatica.si. Opinions, news, calls for conferences, calls for papers, etc. should be sent directly to the managing editor.

## SUBSCRIPTION

Please, complete the order form and send it to Dr. Drago Torkar, Informatica, Institut Jožef Stefan, Jamova 39, 1000 Ljubljana, Slovenia. E-mail: drago.torkar@ijs.si

Since 1977, Informatica has been a major Slovenian scientific journal of computing and informatics, including telecommunications, automation and other related areas. In its 16th year (more than twentyfive years ago) it became truly international, although it still remains connected to Central Europe. The basic aim of Informatica is to impose intellectual values (science, engineering) in a distributed organisation.

Informatica is a journal primarily covering intelligent systems in the European computer science, informatics and cognitive community; scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the Refereeing Board.

Informatica web edition is free of charge and accessible at http://www.informatica.si.
Informatica print edition is free of charge for major scientific, educational and governmental institutions. Others should subscribe.

## Informatica WWW:

## http://www.informatica.si/

**Referees from 2008 on:**

A. Abraham, S. Abraham, R. Accornero, A. Adhikari, R. Ahmad, G. Alvarez, N. Anciaux, R. Arora, I. Awan, J. Azimi, C. Badica, Z. Balogh, S. Banerjee, G. Barbier, A. Baruzzo, B. Batagelj, T. Beaubouef, N. Beaulieu, M. ter Beek, P. Bellavista, K. Bilal, S. Bishop, J. Bodlaj, M. Bohanec, D. Bolme, Z. Bonikowski, B. Bošković, M. Botta, P. Brazdil, J. Brest, J. Brichau, A. Brodnik, D. Brown, I. Bruha, M. Bruynooghe, W. Buntine, D.D. Burdescu, J. Buys, X. Cai, Y. Cai, J.C. Cano, T. Cao, J.-V. Capella-Hernández, N. Carver, M. Cavazza, R. Ceylan, A. Chebotko, I. Chekalov, J. Chen, L.-M. Cheng, G. Chiola, Y.-C. Chiou, I. Chorbev, S.R. Choudhary, S.S.M. Chow, K.R. Chowdhury, V. Christlein, W. Chu, L. Chung, M. Ciglarič, J.-N. Colin, V. Cortellessa, J. Cui, P. Cui, Z. Cui, D. Cutting, A. Cuzzocrea, V. Cvjetkovic, J. Cypryjanski, L. Čehovin, D. Čerepnalkoski, I. Ćosić, G. Daniele, G. Danoy, M. Dash, S. Datt, A. Datta, M.-Y. Day, F. Debili, C.J. Debono, J. Dedič, P. Degano, A. Dekdouk, H. Demirel, B. Demoen, S. Dendamrongvit, T. Deng, A. Derezinska, J. Dezert, G. Dias, I. Dimitrovski, S. Dobrišek, Q. Dou, J. Doumen, E. Dovgan, B. Dragovich, D. Drajic, O. Drbohlav, M. Drole, J. Dujmović, O. Ebers, J. Eder, S. Elaluf-Calderwood, E. Engström, U. riza Erturk, A. Farago, C. Fei, L. Feng, Y.X. Feng, B. Filipič, I. Fister, I. Fister Jr., D. Fišer, A. Flores, V.A. Fomichov, S. Forli, A. Freitas, J. Fridrich, S. Friedman, C. Fu, X. Fu, T. Fujimoto, G. Fung, S. Gabrielli, D. Galindo, A. Gambarara, M. Gams, M. Ganzha, J. Garbajosa, R. Gennari, G. Georgeson, N. Gligorić, S. Goel, G.H. Gonnet, D.S. Goodsell, S. Gordillo, J. Gore, M. Grčar, M. Grgurović, D. Grosse, Z.-H. Guan, D. Gubiani, M. Guid, C. Guo, B. Gupta, M. Gusev, M. Hahsler, Z. Haiping, A. Hameed, C. Hamzaçebi, Q.-L. Han, H. Hanping, T. Härder, J.N. Hatzopoulos, S. Hazelhurst, K. Hempstalk, J.M.G. Hidalgo, J. Hodgson, M. Holbl, M.P. Hong, G. Howells, M. Hu, J. Hyvärinen, D. Ienco, B. Ionescu, R. Irfan, N. Jaisankar, D. Jakobović, K. Jassem, I. Jawhar, Y. Jia, T. Jin, I. Jureta, Đ. Juričić, S. K, S. Kalajdziski, Y. Kalantidis, B. Kaluža, D. Kanellopoulos, R. Kapoor, D. Karapetyan, A. Kassler, D.S. Katz, A. Kaveh, S.U. Khan, M. Khattak, V. Khomenko, E.S. Khorasani, I. Kitanovski, D. Kocev, J. Kocijan, J. Kollár, A. Kontostathis, P. Korošec, A. Koschmider, D. Košir, J. Kovač, A. Krajnc, M. Krevs, J. Krogstie, P. Krsek, M. Kubat, M. Kukar, A. Kulis, A.P.S. Kumar, H. Kwaśnicka, W.K. Lai, C.-S. Laih, K.-Y. Lam, N. Landwehr, J. Lanir, A. Lavrov, M. Layouni, G. Leban, A. Lee, Y.-C. Lee, U. Legat, A. Leonardis, G. Li, G.-Z. Li, J. Li, X. Li, X. Li, Y. Li, Y. Li, S. Lian, L. Liao, C. Lim, J.-C. Lin, H. Liu, J. Liu, P. Liu, X. Liu, X. Liu, F. Logist, S. Loskovska, H. Lu, Z. Lu, X. Luo, M. Luštrek, I.V. Lyustig, S.A. Madani, M. Mahoney, S.U.R. Malik, Y. Marinakis, D. Marinčič, J. Marques-Silva, A. Martin, D. Marwede, M. Matijašević, T. Matsui, L. McMillan, A. McPherson, A. McPherson, Z. Meng, M.C. Mihaescu, V. Milea, N. Min-Allah, E. Minisci, V. Mišić, A.-H. Mogos, P. Mohapatra, D.D. Monica, A. Montanari, A. Moroni, J. Mosegaard, M. Moškon, L. de M. Mourelle, H. Moustafa, M. Možina, M. Mrak, Y. Mu, J. Mula, D. Nagamalai, M. Di Natale, A. Navarra, P. Navrat, N. Nedjah, R. Nejabati, W. Ng, Z. Ni, E.S. Nielsen, O. Nouali, F. Novak, B. Novikov, P. Nurmi, D. Obrul, B. Oliboni, X. Pan, M. Pančur, W. Pang, G. Papa, M. Paprzycki, M. Paralič, B.-K. Park, P. Patel, T.B. Pedersen, Z. Peng, R.G. Pensa, J. Perš, D. Petcu, B. Petelin, M. Petkovšek, D. Pevec, M. Pičulin, R. Piltaver, E. Pirogova, V. Podpečan, M. Polo, V. Pomponiu, E. Popescu, D. Poshyvanyk, B. Potočnik, R.J. Povinelli, S.R.M. Prasanna, K. Pripužić, G. Puppis, H. Qian, Y. Qian, L. Qiao, C. Qin, J. Que, J.-J. Quisquater, C. Rafe, S. Rahimi, V. Rajkovič, D. Raković, J. Ramaekers, J. Ramon, R. Ravnik, Y. Reddy, W. Reimche, H. Rezankova, D. Rispoli, B. Ristevski, B. Robič, J.A. Rodriguez-Aguilar, P. Rohatgi, W. Rossak, I. Rožanc, J. Rupnik, S.B. Sadkhan, K. Saeed, M. Saeki, K.S.M. Sahari, C. Sakharwade, E. Sakkopoulos, P. Sala, M.H. Samadzadeh, J.S. Sandhu, P. Scaglioso, V. Schau, W. Schempp, J. Seberry, A. Senanayake, M. Senobari, T.C. Seong, S. Shamala, c. shi, Z. Shi, L. Shiguo, N. Shilov, Z.-E.H. Slimane, F. Smith, H. Sneed, P. Sokolowski, T. Song, A. Soppera, A. Sorniotti, M. Stajdohar, L. Stanescu, D. Strnad, X. Sun, L. Šajn, R. Šenkeřík, M.R. Šikonja, J. Šilc, I. Škrjanc, T. Štajner, B. Šter, V. Štruc, H. Takizawa, C. Talcott, N. Tomasev, D. Torkar, S. Torrente, M. Trampuš, C. Tranoris, K. Trojacanec, M. Tschierschke, F. De Turck, J. Twycross, N. Tziritas, W. Vanhoof, P. Vateekul, L.A. Vese, A. Visconti, B. Vlaovič, V. Vojisavljević, M. Vozalis, P. Vračar, V. Vranić, C.-H. Wang, H. Wang, H. Wang, H. Wang, S. Wang, X.-F. Wang, X. Wang, Y. Wang, A. Wasilewska, S. Wenzel, V. Wickramasinghe, J. Wong, S. Wrobel, K. Wrona, B. Wu, L. Xiang, Y. Xiang, D. Xiao, F. Xie, L. Xie, Z. Xing, H. Yang, X. Yang, N.Y. Yen, C. Yong-Sheng, J.J. You, G. Yu, X. Zabulis, A. Zainal, A. Zamuda, M. Zand, Z. Zhang, Z. Zhao, D. Zheng, J. Zheng, X. Zheng, Z.-H. Zhou, F. Zhuang, A. Zimmermann, M.J. Zuo, B. Zupan, M. Zuqiang, B. Žalik, J. Žižka,

# *Informatica*

## An International Journal of Computing and Informatics

Web edition of Informatica may be accessed at: http://www.informatica.si.

**Subscription Information** Informatica (ISSN 0350-5596) is published four times a year in Spring, Summer, Autumn, and Winter (4 issues per year) by the Slovene Society Informatika, Litostrojska cesta 54, 1000 Ljubljana, Slovenia.
The subscription rate for 2019 (Volume 43) is
– 60 EUR for institutions,
– 30 EUR for individuals, and
– 15 EUR for students
Claims for missing issues will be honored free of charge within six months after the publication date of the issue.

Typesetting: Borut Žnidar, borut.znidar@gmail.com.
Printing: ABO grafika d.o.o., Ob železnici 16, 1000 Ljubljana.

Orders may be placed by email (drago.torkar@ijs.si), telephone (+386 1 477 3900) or fax (+386 1 251 93 85). The payment should be made to our bank account no.: 02083-0013014662 at NLB d.d., 1520 Ljubljana, Trg republike 2, Slovenija, IBAN no.: SI56020830013014662, SWIFT Code: LJBASI2X.

Informatica is published by Slovene Society Informatika (president Niko Schlamberger) in cooperation with the following societies (and contact persons):
Slovene Society for Pattern Recognition (Vitomir Štruc)
Slovenian Artificial Intelligence Society (Mitja Luštrek)
Cognitive Science Society (Olga Markič)
Slovenian Society of Mathematicians, Physicists and Astronomers (Marej Brešar)
Automatic Control Society of Slovenia (Nenad Muškinja)
Slovenian Association of Technical and Natural Sciences / Engineering Academy of Slovenia (Mark Pleško)
ACM Slovenia (Borut Žalik)

Informatica is financially supported by the Slovenian research agency from the Call for co-financing of scientific periodical publications.

Informatica is surveyed by: ACM Digital Library, Citeseer, COBISS, Compendex, Computer & Information Systems Abstracts, Computer Database, Computer Science Index, Current Mathematical Publications, DBLP Computer Science Bibliography, Directory of Open Access Journals, InfoTrac OneFile, Inspec, Linguistic and Language Behaviour Abstracts, Mathematical Reviews, MatSciNet, MatSci on SilverPlatter, Scopus, Zentralblatt Math

Volume 43 Number 1 March 2019 ISSN 0350-5596

# Informatica

## An International Journal of Computing and Informatics

Informatica **43** (2019) Number 1, pp. 1–149