Volume 39 Number 3 September 2015 ISSN 0350-5596

Informatica

An International Journal of Computing and Informatics

Special Issue:

MATCOS-13 conference

Guest Editors:

Andrej Brodnik

Gábor Galambos

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-13 conference"

It is passing almost two years since we successfully concluded *Middle-European Conference on Applied Theoretical Computer Science*, MATCOS-13 conference, that was held at the University of Primorska as a part of multiconference Information Society. The conference welcomed papers from a wide spectrum of topics, however with a clear emphasis on applicability of research of theoretical Computer Science. Moreover, out of 15 regular papers programme committee chairs invited seven papers to participate in a special issue of journal Informatica.

As the conference topics are colourful, so are the selected papers. However, all of them present two emphases of the conference – use of the theoretical Computer Science in practical problem solving. The selected papers are arranged from the most theoretical to the most practical. Therefore we start with graph related problems.

The first paper, *Barrier Resilience of Visibility Polygons*, is related to the visibility problem in Computational Geometry. The problem asks how to reach a target *t* from a source *s* in such a way that the least number of observers see your travel. In the second paper we dig even deeper into graph theory. The paper, *The Random Hypergraph Assignment Problem*, generalizes Parisi's proven conjecture on the expected optimal cost value of an assignment problem on a complete bipartite graph to a class of bipartite hypergraphs. The last among the graph related problems, *Strategic deployment in graphs*, studies a problem in a graph where a vertex weight represents the size of force needed to conquer the vertex, while an edge weight represents the minimum force needed to traverse it. The problem asks to minimize the initial size of force to traverse the whole graph. The fourth graph related paper, *Relaxations in Practical Clustering and Blockmodeling*, analyses different approaches to clustering in networks. The authors show that most of the practical approaches can be viewed as relaxations od four basic graph theoretic approaches. Finally, the last graph related problem, *Integer Programming Models for the Target Visitation Problem*, is also an optimization problem on graphs. Indeed, it is related to travelling salesman problem with additional preference to the order of visited sites. The authors present several integer programme solutions and study their suitability for branch-and-cut approaches.

The last two papers included in this special issue represent the other pole of the conference topic. They are much more practically oriented. In the first one, *Estimation of cervix cancer spatial distribution for brachytherapy applicator analysis*, the authors study a problem of applying radiation to the tissue. The problem is modelled in 3D space. If we started with computation geometry we also wrap up with a topic related to it. Namely, the last paper, *Detection of ground in point-clouds generated from stereo-pair images*, presents a novel approach to filter data to construct digital terrain models.

We wish you an interesting reading!

Andrej Brodnik
Gábor Galambos
Editors

Barrier Resilience of Visibility Polygons

Alexander Gilbers
Institute of Computer Science, University of Bonn 53113 Bonn, Germany
E-mail: agilbers@gmx.de

Keywords: computational geometry, barrier resilience, visibility

Received: June 21, 2014

We consider the problem of computing the Barrier Resilience of a set of Visibility Polygons inside a Polygon. We show that in simple polygons the problem is solvable in time linear in the number of edges. In polygons with holes the problem is APX-hard, so only for special cases can we provide polynomial time algorithms.

Povzetek: Prispevek analizira problematičnost prehoda poligona.

1 Introduction

Put yourself in a smuggler's shoes. You want to deliver some goods to a fixed destination but you do not want to be seen by many witnesses. Unfortunately, there is no way to your destination that is completely unobserved, nor can you conceal your goods. Perhaps you just want to minimize the number of witnesses or perhaps there is some number k of witnesses that still is acceptable.

It is not important to you, how often or how long the witnesses see you on your way. You only care for their number.

You are given a map of your city, in which your starting and your target point are marked as well as the positions of all the possible witnesses, see Figure 1. Can you compute the path that is seen by the minimum number of witnesses?

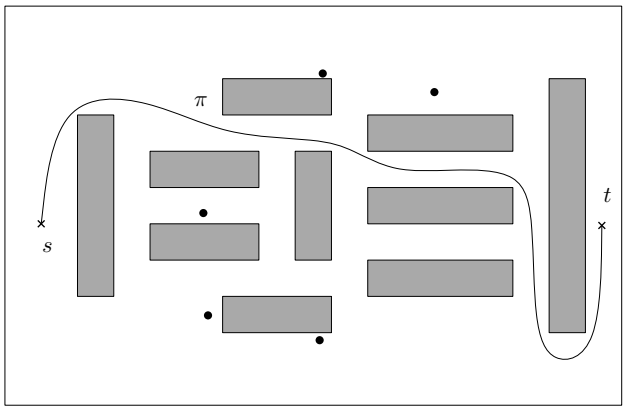


Figure 1: Path π from s to t is seen by two witnesses (black points).

This turns out to be a special case of the BARRIER RESILIENCE problem. Given a start point s and a target point t as well as the positions and ranges of n sensors that are designed to detect intruders, we want to find a path from s to t that minimizes the number of its witnesses (i.e. the sensors that detect the agent traveling on this path, see Figure 2 for an instance of the BARRIER RESILIENCE problem for disk sensors). We call an optimal path in this respect a *minimum witness path*.

This problem can be seen from two sides: On the one hand, it is a path planning problem. On the other hand, the minimum possible number of sensors that detect a path of the agent is an important parameter of the sensor network. It is called the *barrier resilience* of the network. sensor networks with a low barrier resilience are more error-prone than those with high barrier resilience. In the analysis of a sensor network that is designed to detect an intruder, the minimum witness path points to the network's weak spot. Therefore, to optimize sensor networks it would be very helpful to have an efficient method at hand to compute the barrier resilience of the network or, even better, a minimum witness path.

There are many different types of sensor networks conceivable. We here restrict our attention to the very natural case where the sensor regions are visibility domains.

In the following sections, we will show that we can find minimum witness paths in polynomial time in simple polygons and in polygons with one hole. On the other hand we prove that the BARRIER RESILIENCE problem for visibility polygons in polygons with holes is APX-hard. In particular, we get a stronger inapproximability factor than the hardness results known for line segments.

The results in this paper have also been presented in my thesis [9].

2 Related work

Finding minimum witness paths is related to several other tasks. Algorithms that are concerned with the search for shortest paths in polygons (see for example [10]) or minimum cost paths in graphs, where weights are assigned to the edges of the graph [6] are among the best-researched topics in the field of algorithms.

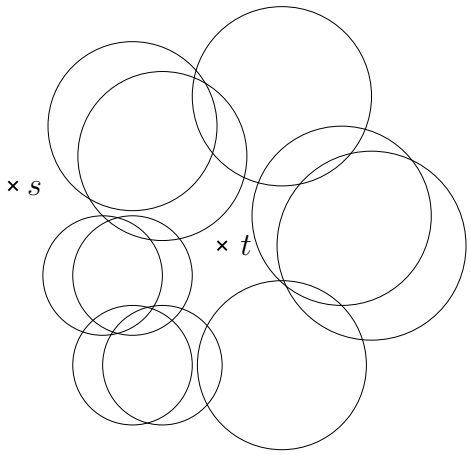


Figure 2: An instance of the BARRIER RESILIENCE problem for disks

While this paper is about getting somewhere without being seen by too many people, there are many works concerning itself with deploying guards or cameras so that everything of interest is seen at least once or at least a certain number of times. In this category fall the many variations of the art gallery problem, see for example [14].

Also problems that combine path planning questions with guarding problems have been examined. In the WATCHMAN ROUTE problem, introduced by Chin and Ntafos [5] the task is to find a shortest closed path π from a given starting point through a polygon P such that every point of P can be seen from some point of π. Since then, various versions of the WATCHMAN ROUTE problem have been defined. The one most strongly related to our problem is the ROBBER problem that was defined by Ntafos in 1990 [13]. Given a set of edges S and a set of threats T, a robber in a simple polygon P wants to find a shortest cycle from which he can see all of S, while not being seen by any of the threats. In this setting, the problem has got a solution only if there exists such a path outside the visibility polygons of the threats. Ntafos gives an algorithm that solves the problem in time $O(n^4 \log\log n)$.

In [7] Gewali et al. define a special case of the WEIGHTED REGIONS problem [12] and apply it to the following problem. Given a polygon with holes, a starting point s, a target point t and a set of k threats. Find the *least risk path* from s to t. The authors give an algorithm that computes a least risk path in time $O(k^4 n^4)$. The risk is measured by the total length of the subpaths that are inside the visibility polygon of some threat. Here lies the main difference to our model in which the cost of using a witnesses visibility region is fixed, no matter how often or how long the path traverses this region.

In 2005, in the environment of sensor networks Kumar et al. [11] introduce the notion of a *k-barrier coverage*. In their setting, somebody wants to cross a belt region over which a sensor network is deployed. The belt region is called *k-barrier covered* if every path that crosses the belt is detected by at least k sensors.

Bereg and Kirkpatrick [2] introduce the notion of *barrier resilience*: Given a collection of geometric objects that model the ranges of sensors and two points s, t in the plane, find the minimum number of objects one has to remove such that s and t are in the same component of the complement of the remaining objects. I.e. the barrier resilience is the maximum k such that the region is k-covered. They give an approximation algorithm for this problem when the sensor ranges are unit disks. Until today it is unknown if this original problem is NP-hard. In [1] Alt et al. show that the BARRIER RESILIENCE problem for line segments is APX-hard and they also define related problems. In [15] Tseng and Kirkpatrick strengthen the result to *unit* line segments. Gibson et al. [8] give an approximation algorithm for a path that visits multiple points and tries to avoid as many unit disks as possible. Chan and Kirkpatrick [4] give a 2-approximation algorithm for the case of Non-identical Disk Sensors.

One can also view the barrier resilience problem in a very abstract graph-theoretic setting where an agent wants to travel from some start vertex of a graph G to some target vertex. In this setting the barriers are arbitrary subsets of the edge set of G. The barriers can also be interpreted as colors that are assigned to the edges. This problem is then called the MINIMUM COLOR PATH problem. Carr et al. [3] show that unless P = NP, the optimal solution cannot be approximated to within a factor $O(2^{\log^{1-\delta(|C|)}|C|})$, where $|C|$ is the number of colors and $\delta(|C|) = \frac{1}{\log\log^a |C|}$, for any constant $a < 1/2$. In [16], Yuan et al. use the Minimum Color Path model to analyze reliability in mesh networks.

3 Minimum witness paths in simple polygons

In our first setting the starting point s and the target point t lie inside a simple polygon P, and we are given a finite set of witness points $W \subset P$. We want to find a path from s to t that is seen by as few as possible witnesses. Let us restate this formally.

Definition 1. Let a polygon P, two points $s, t \in P$ and a set of so-called *witness points* $W = \{w_1, \ldots, w_n\} \subset P$ be given. The *barrier resilience* of W is the minimum cardinality of a subset V of W such that there is an $s-t$-path in P that does not touch any visibility polygon of a point in $W \setminus V$. A path that attains this minimum is called a *minimum witness path*.

We use the usual notion of visibility inside simple polygons that is also illustrated in Figure 3.

Definition 2. Let P be a simple polygon. We say that $p_1 \in P$ *sees* $p_2 \in P$ iff the line segment $\overline{p_1 p_2}$ is a subset of P. We say that a witness point $w \in P$ *sees the path* π iff there is a point p on π that is seen by w.

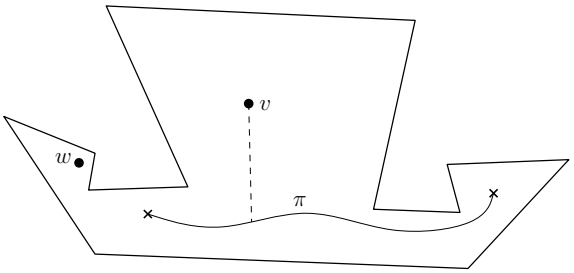


Figure 3: Path π is seen by witness v but not by witness w.

It turns out that in this setting one can find an optimal path very efficiently. The key insight is the following structural lemma.

Lemma 1. *Let P a simple polygon, points $s, t \in P$ and a witness point $w \in P$. If there is a path π in P from s to t that is not seen by w, then the shortest path from s to t in P is not seen by w.*

Before we prove the lemma, we draw the following conclusions that settle the problem for simple polygons.

Theorem 1. *Given a simple polygon P with n edges, two points $s, t \in P$ and a set of witness points $W \subset P$, the shortest path between s and t is an optimal solution to the minimum witness path problem.*

Proof. Let π' denote the shortest path from s to t. By Lemma 1, for every path π between s and t the set $W' = \{w \in W \mid w \text{ sees } \pi'\}$ is a subset of $W(\pi) = \{w \in W \mid w \text{ sees } \pi\}$ and consequently $|W'| \leq |W(\pi)|$. □

Corollary 1. *Given a simple polygon P with n edges, two points $s, t \in P$ and a set of witness points $W \subset P$, we can determine a minimum-witness path in time $O(n)$.*

Proof. The shortest path between two points inside a simple polygon with n edges can be computed in time $O(n)$ [10]. □

The proof of the lemma uses the simple topological structure of the polygon.

Proof of Lemma 1. Let π' be the shortest path between s and t and $w \in P$ a point that sees the point p on π'. If w sees s or t it obviously sees every path from s to t. Otherwise consider the line $L(w, p)$ through w and p.

The points w and p lie in the same connected component C of $L(w, p) \cap P$. Now $P \setminus C$ splits into at least two connected components. As π' is the shortest path, s and t lie in different components (otherwise π' could be shortened to a path that is completely contained in the common component of s and t).

It follows that every path from s to t must pass C and is therefore seen by w. □

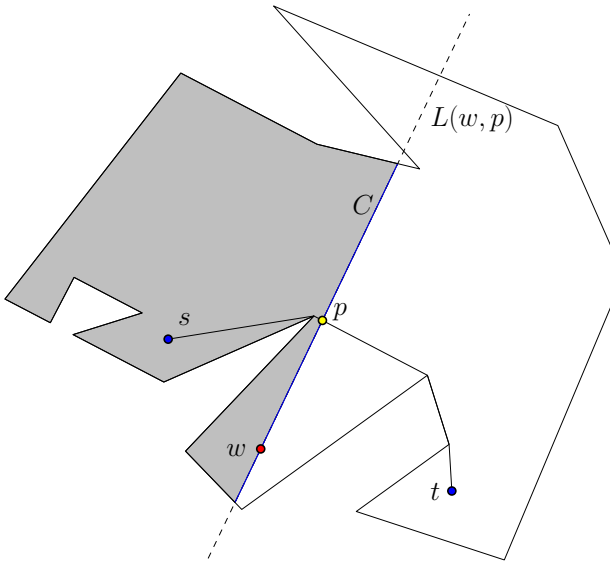


Figure 4: The connected component C of $L(w, p) \cap P$ that contains w and p splits P into two connected components, one containing s, the other containing t.

4 Polygons with holes

The next step is looking at polygons with holes. So now we have a simple polygon P' and a collection of simple polygons $H_1, \ldots, H_m$, called the holes, where every hole lies in the interior of P' and $H_i \cap H_j = \emptyset$ for all $1 \leq i < j \leq m$. The polygon with holes P then is defined to be $P = P' \setminus \bigcup_{i=1}^{m} \mathring{H}_i$, where $\mathring{H}_i$ denotes the topological interior of H_i. Let $|P|$ denote the total number of edges of P. Two points $p_1, p_2 \in P$ *see* each other if and only if the line segment $\overline{p_1 p_2}$ is completely contained in P.

Again we are given two points $s, t \in P$ and witnesses $w_1, \ldots, w_n \in P$ in general position, and we want to find a path π inside P from s to t minimizing the number of witnesses who can see π.

First we show that the problem is APX-hard by a reduction from Vertex Cover that provides a stronger factor than other hardness proofs in the context of barrier resilience.

Theorem 2. *Estimating the barrier resilience of a set of visibility polygons inside polygons with holes is APX-hard. In particular, unless $P = NP$, the barrier resilience of visibility polygons with holes cannot be approximated within a factor of* 1.3606*. If the Unique Games Conjecture is true, then the barrier resilience cannot be approximated within any constant factor better than* 2*.*

Proof. We show this by an approximation factor preserving reduction from Minimum Vertex Cover.

Let $G = (V, E)$ be an instance of vertex cover. Let $e_1, e_2, \ldots, e_m$ an enumeration of the edges, $v_1, v_2, \ldots v_n$ an enumeration of the vertices.

We now construct a polygon with holes P in the plane that contains a start point s, a target point t and n witness

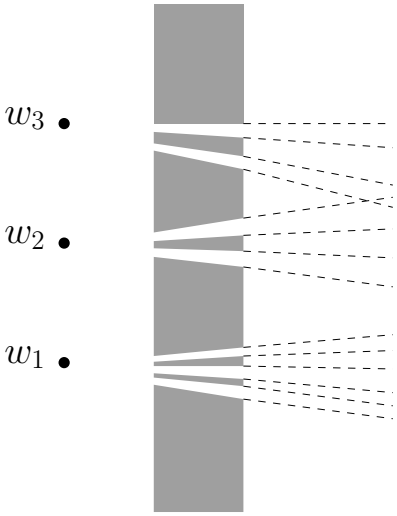


Figure 5: On the left side of the polygon there are only narrow slits between the holes through which the witnesses (which correspond to the vertices) can peek.

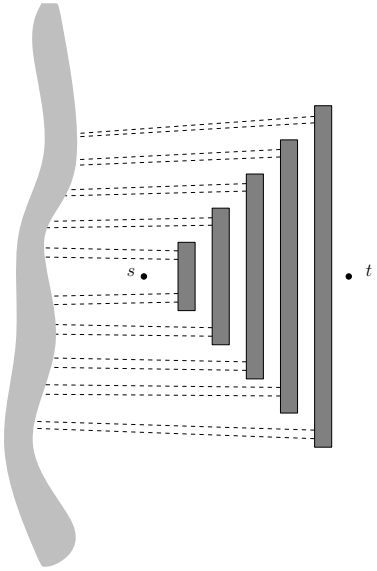


Figure 6: Far away on the right side portions of visibility regions hit the rectangles corresponding to edges.

points $w_1, \dots w_n$ such that every path from s to t in P corresponds to a vertex cover of G.

To this end we build a big surrounding rectangle $P' = [-2(m+n+1), m+2] \times [-m-n-1, m+n+1]$. We place the start point at the origin, $s = (0,0)$ and the target point at $t = (m+1, 0)$.
For every edge e_j in E, we add a thin rectangular hole $R_j = [j, j+0.5] \times [-j, j]$.
Then we place the witness points at $w_i = (-2(m+n), i - \lceil \frac{n}{2} \rceil)$. If v_k and v_l (with $k \leq l$) are the vertices incident to edge e_j we define $L(j) = w_k$, $H(j) = w_l$ to be the witnesses corresponding to the vertices with lower and with higher index, respectively. We also define $f \colon \{w_1, \dots, w_n\} \longrightarrow \{v_1, \dots, v_n\}$ to be the bijection that maps every w_i to v_i.

To construct the holes that model the vertex-edge incidences we proceed as follows:
We start with one rectangle
$Z = [-2(m+n)+0.5, -2(m+n)+1] \times [-m-n, m+n]$
and split it into $2m+1$ pieces.
For every edge e_j we define the two triangles

$$TH_j = \Delta(H(j), (j, j-0.25), (j, j-0.5))$$

and

$$TL_j = \Delta(L(j), (j, 0.25-j), (j, 0.5-j)).$$

Now we construct the $2m+1$ holes by simultaneously cutting the interiors of all these triangles out of Z. We set

$$Z' = Z \setminus \bigcup_{j=1}^{m} (\mathring{TH}_j \cup \mathring{LH}_j)$$

We add the connected components of Z' as holes to our scene.
By this construction every witness w_i sees a rectangle R_j iff the vertex v_i is incident to e_j.

We first notice that this reduction is clearly polynomial-time. The total number of edges of P is $12m+8$ and the number of points (witnesses and start/target) is n+2, each of which can easily be computed in polynomial time.

To see that every path from s to t that is seen by k witnesses corresponds to a vertex cover of G, observe the following: For every edge e_j the quadrilateral with corners $(j, 0.5-j), (j, j-0.5), H(j), L(j)$ contains s and does not contain t. Thus every path from s to t must cross one of its four sides. One of the sides is the edge of a hole that cannot be crossed. The other three sides are visibility segments of $L(j)$ and $H(j)$, respectively, and thus crossing them means to be seen by $L(j)$ or $H(j)$. Therefore, if π is a path from s to t that is seen by the set of witnesses $W(\pi)$ then the image of $W(\pi)$ under f is a vertex cover of G. As f is a bijection, the set of witnesses has the same cardinality as the resulting vertex cover.

On the other hand, if $C \subset V$ is a vertex cover of G we can construct a path from s to t with at most the same number of witnesses. From s we first go to the point $(1,0)$. Now we are on the boundary of R_1 that corresponds to edge e_1. By definition, $f(H(1))$ or $f(L(1))$ are in C. If $f(H(1))$ is in C, our path proceeds to $(1,1)$, crossing the visibility region of $H(1)$ (but no other visibility region), and then to $(1.5, 1)$. Otherwise, the path proceeds to $(1, -1)$ (crossing the visibility region of $L(1)$) and then to $(1.5, -1)$. In both cases, the next way point is $(2, 0)$.
We continue in this manner, getting, for every j, from $(j, 0)$ to $(j+1, 0)$ by crossing the visibility region of $H(j)$ if $f(H(j)) \in C$ and crossing the visibility region of $L(j)$ otherwise, until we reach t. The resulting set $W(\pi)$ of witnesses has at most as many elements as C.
It follows that an α-approximation for the BARRIER RESILIENCE problem yields an α-approximation for MINIMUM VERTEX COVER. □

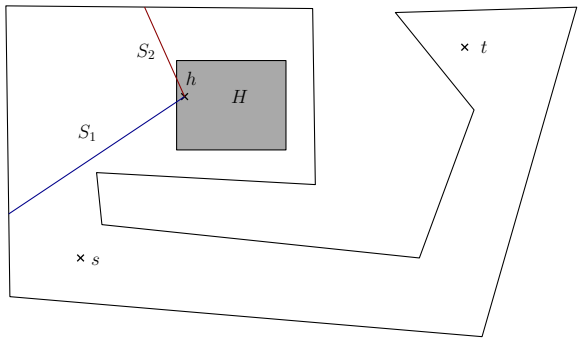


Figure 7: The removal of the segments S_1 and S_2 splits P into two connected components. s and t lie in the same connected component.

Next we show that in the case of one convex hole either one can ignore the hole (Lemma 2) or one can compute two paths, one of which is a minimum witness path (Theorem 3).

Lemma 2. *Let P be a polygon with one convex hole H, (i.e. $P = P' \setminus \mathring{H}$ for some simple polygon P' and a convex polygon $H \subset P$). Assume that for every point $h \in H$ and for every two line segments $S_1, S_2 \subset P \cup H$ that both have as one endpoint h and the other endpoint on $\partial(P \cup H)$, s and t lie in the same connected component of $(P \cup H) \setminus (S_1 \cup S_2)$. Then there is a unique shortest path from s to t in P and it is a minimum witness path.*

Proof. Take the shortest path π between s and t in $P' = P \cup H$. As this is a simple polygon, π is unique. By assumption, there is no point $h \in H$ and line segments $S_1, S_2 \subset P'$ that connect h to the boundary of P' such that s, t lie in different components of $P' \setminus (S_1 \cup S_2)$, see Figure 7. Then π does not intersect H.

Otherwise we could take a point $h \in \pi \cap H$ and draw a line segment $S \subset P'$ that crosses π in h and ends in the two points b_1, b_2 on the boundary of P'. Then setting $S_1 = \overline{b_1 h}$ and $S_2 = \overline{b_2 h}$ yields a contradiction to the assumption as s and t lie in different connected components of $P' \setminus (S_1 \cup S_2)$ (because the shortest path crosses the segment S exactly once).

Therefore, π is completely contained in P and is the unique shortest path between s and t. Now suppose, there was a path that was seen by less witnesses π'. Then there was in particular one witness w that sees π but not π'. Let p be a point on the path π that is seen by w. Let further S be the connected component of the intersection $L(w, p) \cap P'$ of the line through w and p with P' that contains p and S_1 be the connected component of $L(w, p) \cap P$ that contains p. If both endpoints of S_1 lay on the boundary of P' then s and t were in distinct components of $P \setminus S_1$. Then every path from s to t would have to cross S_1 and therefore be seen by w, a contradiction.

Thus, one of the endpoints must lie on the boundary of H, let us call this endpoint h. If we now set S_2 to be the topological closure of $S \setminus S_1$, then h, S_1, S_2 are as above and s, t are in different connected components of $P' \setminus (S_1 \cup S_2)$, a contradiction.

It follows, that there can be no path π from s to t and witness w such that w sees π but not π'. Thus the shortest path π is optimal. □

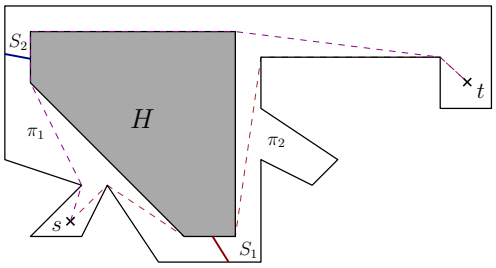


Figure 8: The removal of either S_1 or S_2 leaves polygons with unique shortest paths π_1, π_2 one of which is a minimum witness path

Theorem 3. *Let $P = P' \setminus \mathring{H}$ a polygon with one convex hole, s, t be start and target point, respectively. Let there be line segments S_1, S_2, each of them connecting a point on an edge (not a vertex) of H to a point on an edge (not a vertex) of the boundary of $P \cup H$, so that s and t lie in different connected components of $P \setminus (S_1 \cup S_2)$. Either the shortest path π_1 from s to t in $P \setminus S_1$ or the shortest path π_2 from s to t in $P \setminus S_2$ is a minimum witness path in P.*

Proof. Suppose none of them were optimal. Then there exist witnesses w_1, w_2 (possibly $w_1 = w_2$) and a path π', such that w_1, w_2 do not see π', but w_1 sees point p_1 on π_1 and w_2 sees p_2 on π_2. Let T_1 and T_2 denote the line segments from boundary to boundary of $P \cup H$ through w_1 and p_1 and through w_2 and p_2, respectively. The segments S_1 and T_1 together with H as well as S_2, T_2, H separate the points s and t. By the existence of π', T_1, T_2 and H together do not separate s and t. The connected component of s in $P \setminus (T_1 \cup T_2)$ is simply connected and contains t. As π' does not cross T_1, T_2 it crosses both S_1 and S_2. s and t lie in different components of $P \setminus (S_1 \cup S_2)$, so $(S_1 \cup S_2)$ is crossed an odd number of times. Now we can repeatedly replace subpaths between two crossings of the same segment S_i by the direct paths along the segment (this does not add witnesses) until only one crossing is left, contradicting the fact, that π' crosses S_1 and S_2. □

It follows that in this case the barrier resilience can be computed in polynomial time by computing S_1 and S_2 and then the respective shortest paths.
One can show that this also holds if P contains many convex holes that are strictly separated in a sense made precise below.

Theorem 4. *Let $P = P' \setminus \bigcup_{i=1}^{m} \mathring{H}_i$ a polygon with convex holes, $s, t \in P$, $W = \{w_1, \ldots, w_n\} \subset P$ a set of witness points. Let for every $i \neq j$ there be a line segment $S_{ij} \subset P$*

s.t. H_i and H_j lie in distinct connected components of $P' \setminus S_{ij}$ and S_{ij} is not seen by any witness $w \in W$. Then one can find a minimum witness path from s to t in polynomial time.

Proof. Let C_{ij} denote the connected component of $P' \setminus S_{ij}$ that contains H_i. Then for every $1 \leq i \leq m$, $C_i = \bigcap_{j \neq i} C_{ij}$ is a simple polygon, that contains H_i but no H_j for any other index $j \neq i$. For $j \neq i$ $C_i \cap C_j = \emptyset$. We now compute in $O(|P'|)$ time the shortest path π' from s to t in P'. The parts of π' outside $\bigcup_{i=1}^{m} C_i$ are already optimal by Theorem 3. To get the witness-minimal path π through P, we replace the parts of π' inside the C_i by witness-optimal paths, according to Lemma 2 or Theorem 3. The different parts do not affect each other. To this end let us call the point where π' enters C_i s_i and the point where it leaves C_i t_i. We then draw a segment B_1 from an arbitrary point on the boundary of H_i to the boundary of C_i. Then we compute in time $O(|P'| + m) = O(|P|)$ the shortest path π_i^1 from s_i to t_i in $C_i \setminus (\mathring{H}_i \cup B_1)$. We then choose a second segment B_2 from H_i to the boundary of C_i, that intersects π_i^1 (if such a segment exists; otherwise π_i^1 is optimal in C_i by Lemma 2) and compute in time $O(|P'| + m) = O(|P|)$ the shortest path π_i^2 from s_i to t_i in $C_i \setminus (\mathring{H}_i \cup B_2)$. We choose the path less seen by witnesses in P to replace the part of π' inside C_i. (Testing all possible path π_i^1 or π_i^2 with all possible witnesses can be done in total time $O(n|P|^2)$ after the construction of the witnesses' visibility polygons.)

By sewing together the thus computed parts we get a witness-minimal path π from s to t in P. The running time is dominated by the visibility tests that can be carried out in $O(n|P|^2)$ The computation of the polygons C_i can be computed in time $O(m^2|P'|)$. (Shoot a ray from H_i to find the boundary of C_i in $O(|P'|+m)$. Then follow the boundary, turning at every intersection. Testing for the intersections of the m many S_{ij} is in total time $O(m^2)$, following the boundary of $|P'|$ and testing if it meets a segment is in $O(m|P'|)$.) □

We note that as usual for fixed k the question if the barrier resilience is at most k is polynomially solvable by checking all k-element subsets of the set of visibility polygons of witnesses.

5 Future work

Finding more classes of polygons where the problem is polynomially solvable is one direction of future research. Introducing more sophisticated assumptions on the seperatedness of the sensors is another direction. There are also variations like weighted or mobile sensors waiting to be examined further. It would be interesting to know if the inapproximability result is tight, so probably the most important task is to design an approximation algorithm for the general case of polygons with arbitrarily many holes.

References

[1] Helmut Alt, Sergio Cabello, Panos Giannopoulos, and Christian Knauer. Minimum cell connection and separation in line segment arrangements. *CoRR*, abs/1104.4618, 2011.

[2] Sergey Bereg and David G. Kirkpatrick. Approximating barrier resilience in wireless sensor networks. In *ALGOSENSORS*, pages 29–40, 2009.

[3] Robert D. Carr, Srinivas Doddi, Goran Konjevod, and Madhav Marathe. On the red-blue set cover problem. In *Proceedings of the eleventh annual ACM-SIAM symposium on Discrete algorithms*, SODA '00, pages 345–353, Philadelphia, PA, USA, 2000. Society for Industrial and Applied Mathematics.

[4] David Yu Cheng Chan and David G. Kirkpatrick. Approximating barrier resilience for arrangements of non-identical disk sensors. In *ALGOSENSORS*, pages 42–53, 2012.

[5] Wei-Pang Chin and Simeon Ntafos. Optimum watchman routes. *Information Processing Letters*, 28:39–44, 1988.

[6] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[7] Laxmi Gewali, Alex Meng, Joseph S. B. Mitchell, and Simeon Ntafos. Path planning in 0/1/∞ weighted regions with applications. In *Symposium on Computational Geometry*, pages 266–278, 1988.

[8] Matt Gibson, Gaurav Kanade, and Kasturi R. Varadarajan. On isolating points using disks. In *ESA*, pages 61–69, 2011.

[9] Alexander Gilbers. *Visibility Domains and Complexity*. PhD thesis, Universität Bonn, 2014.

[10] Leonidas Guibas, John Hershberger, Daniel Leven, Micha Sharir, and Robert Tarjan. Linear-time algorithms for visibility and shortest path problems inside triangulated simple polygons. *Algorithmica*, 2:209–233, 1987.

[11] Santosh Kumar, Ten H Lai, and Anish Arora. Barrier coverage with wireless sensors. In *Proceedings of the 11th annual international conference on Mobile computing and networking*, pages 284–298. ACM, 2005.

[12] Joseph S.B. Mitchell and Christos H. Papadimitriou. The weighted region problem: finding shortest paths through a weighted planar subdivision. *Journal of the ACM*, 38(1):18–73, 1991.

[13] Simeon Ntafos. The robber problem. *Information Processing Letters*, 34:59–63, 1990.

[14] Joseph O'Rourke. *Art gallery theorems and algorithms*. Oxford University Press, New York, 1987.

[15] Kuan-Chieh Robert Tseng and David G. Kirkpatrick. On barrier resilience of sensor networks. In *ALGOSENSORS*, pages 130–144, 2011.

[16] Shengli Yuan, S. Varma, and J.P. Jue. Minimum-color path problems for reliability in mesh networks. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 4, pages 2658–2669 vol. 4, 2005.

The Random Hypergraph Assignment Problem

Ralf Borndörfer
Zuse Institute Berlin, Takustr. 7, 14195 Berlin, Germany
E-mail: borndoerfer@zib.de

Olga Heismann
Zuse Institute Berlin, Takustr. 7, 14195 Berlin, Germany
E-mail: heismann@zib.de

Keywords: assignment, hyperassignment, random, expected value

Received: July 13, 2014

Parisi's famous (proven) conjecture states that the expected cost of an optimal assignment in a complete bipartite graph on $n + n$ nodes with i. i. d. exponential edge costs with mean 1 is $\sum_{i=1}^{n} 1/i^2$, which converges to an asymptotic limit of $\pi^2/6$ as n tends to infinity. We consider a generalization of this question to complete "partitioned" bipartite hypergraphs $G_{2,n}$ that contain edges of size two and proper hyperedges of size four. We conjecture that for i. i. d. uniform hyperedge costs on $[0, 1]$ and i. i. d. exponential hyperedge costs with mean 1, optimal assignments expectedly contain half of the maximum possible number of proper hyperedges. We prove that under the assumption of this number of proper hyperedges the asymptotic expected minimum cost of a hyperassignment lies between 0.3718 and 1.8310 if hyperedge costs are i. i. d. exponential random variables with mean 1. We also consider an application-inspired cost function which favors proper hyperedges over edges by means of an edge penalty parameter p. We show how results for an arbitrary p can be deduced from results for $p = 0$.

Povzetek: V članku je opisana analiza komplektnosti dvojno povezanih grafov na osnovi razširitve Parisijevega izreka.

1 Introduction

A way to gain a better understanding of the structure of a combinatorial optimization problem is to analyze the optimal values of random instances. For the assignment problem, such results were conjectured after extensive computational experiments and then proven theoretically. In particular, the famous (proven) Conjectures of Mézard and Parisi [Mézard and Parisi, 1985] state that the expected optimal cost value of an assignment problem on a complete bipartite graph with i. i. d. uniform edge costs on $[0, 1]$ or i. i. d. exponential edge costs with mean 1 converges to $\frac{\pi^2}{6} = 1.6449\ldots$ if the number of vertices tends to infinity. The limit is equal for both distributions since it can be proven that only the density at 0 is relevant, which coincides for both distributions [Aldous, 1992]. For a survey on the *random assignment problem* and several of its variants, see [Krokhmal and Pardalos, 2009].

We consider a generalization of this setting to a class of *bipartite hypergraphs* in terms of what we call the random *hypergraph assignment problem* (HAP). This problem is an idealized version of vehicle rotation planning problems in long-distance passenger rail transport, see [Borndörfer et al., 2011] for further details and [Maróti, 2006] for a survey on railway vehicle rotation planning.

We will deal with HAPs in a special well-structured type of bipartite hypergraphs $G_{2,n}$, that contain on each side n "parts" of size 2 each. In this case, the HAP is already NP-hard [Borndörfer and Heismann, 2012] and therefore interesting to analyze. The hyperedge set of such a partitioned hypergraph $G_{2,n}$ consists only of edges of size 2 and proper hyperedges of size 4, and it has a structure that makes it easy to view a hyperassignment as a combination of two assignments, one consisting only of edges, and the other one consisting only of proper hyperedges (that can also be viewed as edges). Despite this simple general idea, however, combining the two assignments involves a choice over an exponential number of possibilities which is quite difficult to analyze. We will explain this in more detail in Section 2 after introducing the problem.

In Section 3, we conjecture that the expected number of proper hyperedges in an optimal solution of the random HAP on partitioned hypergraphs $G_{2,2n}$ with i. i. d. uniform random edge costs on $[0, 1]$ or i. i. d. exponential random edge costs with mean 1 is n. This conjecture is based on extensive computational results. Assuming that this conjecture holds, we can prove a lower bound of 0.3718 and an upper bound of 1.8310 for the expected value of a minimum cost hyperassignment in $G_{2,2n}$ for the exponential edge cost distribution and for vertex numbers tending to infinity. To achieve this, we first use a combinatorial argument to represent the bounds in terms of bounds for random assignments. Then, we compute these bounds using results

for the random assignment problem.

In hypergraph assignment problems that arise from practical applications, proper hyperedges represent unions of edges. Such hyperedges have costs that are smaller than the sum of the costs of the edges that they contain; these edges are considered to be similar and a solution with much similarity is desirable [Borndörfer et al., 2011]. We consider a setting with *regularity-rewarding* cost functions, in which the number of proper hyperedges in a solution and the optimal value of a random HAP in $G_{2,n}$ do not only depend on the number of vertices n but also on an edge *penalty parameter* p. We will show how the number of proper hyperedges and the value of an optimal solution for every p can be deduced from results for $p = 0$ in Section 4.

The paper ends in Section 5 with a discussion of the results.

A short conference version of this paper has already been published as [Heismann and Borndörfer, 2013].

2 The hypergraph assignment problem

We consider in this paper hypergraph assignment problems on a special type of bipartite hypergraphs.

Definition 2.1. Let $G_{2,n} = (U, V, E)$ be the *bipartite hypergraph* with *vertex sets*

$$U = \bigcup_{i=1}^{n} U_i, \qquad V = \bigcup_{i=1}^{n} V_i$$

with

$$U_i = \{u_i, u_i'\}, \qquad V_i = \{v_i, v_i'\}$$

and *hyperedge set* $E = E_1 \cup E_2$ where

$$E_1 = \{\{u, v\} : u \in U, v \in V\}$$

are the *edges* and

$$E_2 = \{U_i \cup V_j : i, j \in \{1, \dots, n\}\}$$

are the *proper hyperedges* of size 4. The sets U_i and V_i, $i \in \{1, \dots, n\}$ are called the *parts* on the U- and V-side, respectively.

For a visualization of such a hypergraph, see Figure 1.

Note that every hyperedge in $G_{2,n}$ connects a part on the U- and a part on the V-side. We remark that the HAP can be formulated in the same way for more general bipartite hypergraphs, with less structure and possibly containing hyperedges which contain more than four vertices, see [Borndörfer and Heismann, 2012].

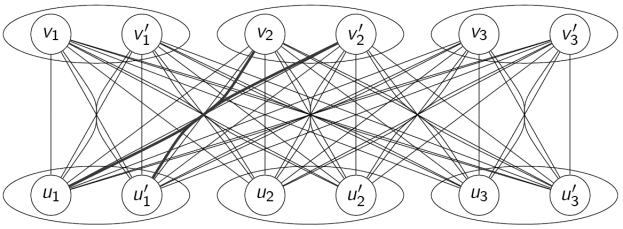


Figure 1: Visualization of the bipartite hypergraph $G_{2,3}$. The thick hyperedge is the proper hyperedge $U_1 \cup V_2 = \{u_1, u_1', v_2, v_2'\}$.

Definition 2.2. For a vertex subset $W \subseteq U \cup V$ we define the *incident hyperedges*

$$\delta(W) := \{e \in E : e \cap W \neq \emptyset, e \setminus W \neq \emptyset\}$$

to be the set of all hyperedges having at least one vertex in both W and $(U \cup V) \setminus W$.

A *hyperassignment* in $G_{2,n}$ is a subset $H \subseteq E$ of pairwise disjoint hyperedges that cover U and V, i. e., for all $e_1, e_2 \in H$, $e_1 \cap e_2 = \emptyset$, and $\bigcup H = U \cup V$. Given a cost function $c_E : E \to \mathbb{R}$, the cost of a hyperassignment is $\sum_{e \in H} c_E(e)$. The *hypergraph assignment problem* with input $(G_{2,n}, c_E)$ consists of finding a hyperassignment in $G_{2,n}$ of minimum cost w. r. t. c_E.

For bipartite hypergraphs $G_{2,n}$, the hypergraph assignment problem can be seen as a combination of two assignment problems. Namely, observe that for every hyperassignment H and every part U_i and V_i, $i \in \{1, \dots, n\}$, the set of incident hyperedges $\delta(U_i) \cap H$ and $\delta(V_i) \cap H$ consists either of one proper hyperedge or of two edges. If we decide for every part U_i and V_i whether the hyperassignment to be constructed is incident to one proper hyperedge or to two edges, we can restrict the hyperedge set of $G_{2,n}$ to

- the set of edges connecting pairs of vertices from the parts U_i, V_i that will be incident to edges—this is the first assignment problem, and
- the proper hyperedges $\{U_i \cup V_j\}$ for U_i and V_j that will be incident to proper hyperedges—viewing U_i and V_j as composite vertices and the hyperedges as edges connecting them—this is the second assignment problem.

Solving these two assignment problems independently produces the minimum cost hyperassignment subject to the fixed edge and hyperedge incidences.

The HAP in $G_{2,n}$ can thus be solved in two steps. The first step decides which parts U_i and V_i will be incident to proper hyperedges. Of course, we must chose the same number of parts on the U- and the V-side, equal to the number of proper hyperedges in the hyperassignment to be constructed; the other parts will be incident to edges. The second steps consists of solving the resulting two assignment problems stated above.

3 Expected optimal values for the random HAP with exponential or uniform costs

Predicting the optimal value of a random hypergraph assignment problem in $G_{2,n}$ involves a prediction of the number of proper hyperedges in an optimal solution. This number depends on how advantageous it is to choose a proper hyperedge instead of two edges (so that one has just one number adding to the cost instead of two) compared to the disadvantage of having less freedom (there are fewer possibilities to cover a single vertex with a proper hyperedge than with an edge) when searching for a hyperassignment with the least possible cost. We conjecture that one can expect that an optimal hypergraph assignment in $G_{2,n}$ contains half of the possible number of proper hyperedges.

Conjecture 3.1. The expected number of proper hyperedges in a minimum cost hyperassignment in $G_{2,2n}$ with cost function c_E such that all $c_E(e)$, $e \in E$ are i. i. d. exponential random variables with mean 1 or i. i. d. uniform random variables on $[0, 1]$ is n.

Table 1 backs this conjecture. It gives computational results for the random hypergraph assignment problem in the bipartite hypergraph $G_{2,n}$ with i. i. d. uniform random variables on $[0, 1]$ and i. i. d. exponential random variables with mean 1 as hyperedge costs. For every n, we report the mean value and the standard deviation of the optimal cost value and the number of proper hyperedges in the optimal solution for 1000 computations. The HAPs were solved as integer programs using CPLEX 12.5.

The first column (n) of Table 1 shows the number of parts on the U- and V-side. Columns 2 and 6 (opt. val.) give the mean optimal values. Their standard deviations can be seen in columns 3 and 7 (s. d.) for the two cost function distributions, respectively. Columns 4 and 8 (# pr. hy.) show the number of proper hyperedges in the optimal solutions found, columns 5 and 9 (s. d.) show their standard deviations. The important finding w. r. t. Conjecture 3.1 is that the values in columns 4 and 8 are about half the values in column 1 in each row.

The computational results also suggest that the expected optimal cost converges to a value around 1.05 for both distributions. Although for larger n more hyperedges are contained in a hyperassignment, the optimal value does not increase much. This can be intuitively explained by noting that for larger n there are also more possible hyperassignments to select from, and the chances to find a hyperassignment that has a low cost are therefore still good even if it will contain more hyperedges.

We will now compute a lower and upper bound on the expected value of a minimum cost hyperassignment in $G_{2,2n}$ with n proper hyperedges for the exponential distribution. To this end, we will use the following result: For a complete bipartite graph with vertex sets of size m and n and with i. i. d. exponential random variables with mean 1 as edge costs, the expected minimum value of the sum of k pairwise disjoint edges (this is called a partial assignment) is

$$E(m, n, k) := \sum_{\substack{i,j\geq 0 \\ i+j\leq k-1}} \frac{1}{(n-i)(m-j)}.$$

This result was conjectured in [Coppersmith and Sorkin, 1999] and first proved in [Linusson and Wästlund, 2004]. The latter paper also shows that for $m = n = k$ this term can be written as

$$E(n, n, n) = \sum_{i=1}^{n} \frac{1}{i^2}.$$

That this formula gives the expected value of a random assignment is Parisi's Conjecture.

Theorem 3.2. *Let* $\mathbf{E}$ *be the expected value of the minimum cost of a hyperassignment in* $G_{2,2n} = (U, V, E)$ *with exactly* n *proper hyperedges and cost function* c_E *with i. i. d. exponential random variables* $c_E(e)$ *with mean 1 for all* $e \in E$. *The following holds for* $n \to \infty$*:*

$$0.3718 < \mathbf{E} < 1.8310.$$

Proof. By definition,

$$E(n) := E(2n, 2n, n) = \sum_{\substack{i,j\geq 0 \\ i+j\leq n-1}} \frac{1}{(2n-i)(2n-j)}.$$

Using $E(n)$, we can bound the expected value of a hyperassignment in $G_{2,2n}$ with i. i. d. exponential random variables with mean 1 as hyperedge costs restricted to the hyperassignments with n proper hyperedges as follows.

For the lower bound, observe that in the best possible hyperassignment the selected n proper hyperedges can be only as good as the n pairwise disjoint proper hyperedges with the least possible cost sum in $G_{2,2n}$. Also, the selected $2n$ edges can be only as good as the $2n$ pairwise disjoint edges with the least possible cost sum in $G_{2,2n}$. Thus, $E(n) + E(2n)$ is a lower bound for $\mathbf{E}$.

On the other hand, choosing the n pairwise disjoint proper hyperedges with the least possible cost sum in $G_{2,2n}$ and finding the best possible edges for the remaining "unused" vertices leads to an upper bound of $E(n) + E(2n, 2n, 2n)$ for $\mathbf{E}$.

To transform the two-indexed sum describing $E(n)$ to a sum with only one index, we calculate the difference $D(n) := E(n + 1) - E(n)$ and use the recursive formula

$$E(n) = E(1) + \sum_{i=1}^{n-1} D(i) = \frac{1}{4} + \sum_{i=1}^{n-1} D(i). \qquad (1)$$

We get

$$\begin{aligned} D(n) &= E(n+1) - E(n) \\ &= E(2n+2, 2n+2, n+1) - E(2n, 2n, n) \end{aligned}$$

Table 1: Computational results for random hypergraph assignment problems in $G_{2,n}$ for i. i. d. uniform random variables on $[0, 1]$ or i. i. d. exponential random variables with mean 1 as hyperedge costs. The mean optimal values (column 2 and 6) and their standard deviations (column 3 and 7) are rounded to the third decimal place. The number of proper hyperedges in the optimal hyperassignments (column 4 and 8) and their standard deviations (column 5 and 9) are rounded to one decimal place. 1000 computations were done for each value of n and each distribution. The values in columns 4 and 8 are about half the value of column 1 in each row. This supports Conjecture 3.1.

	uniform on $[0, 1]$				exponential with mean 1			
n	opt. val.	s. d.	# pr. hy.	s. d.	opt. val.	s. d.	# pr. hy.	s. d.
10	0.943	0.177	5.5	2.0	1.019	0.206	5.3	2.0
20	1.006	0.136	10.4	2.8	1.039	0.141	10.4	2.8
30	1.018	0.109	15.5	3.4	1.049	0.117	15.3	3.4
40	1.037	0.096	20.7	4.0	1.045	0.097	20.5	3.9
50	1.036	0.085	25.8	4.4	1.054	0.085	25.4	4.3
60	1.044	0.078	31.0	4.8	1.050	0.080	30.6	4.7
70	1.041	0.074	35.8	4.9	1.053	0.079	35.6	5.1
80	1.044	0.070	40.9	5.4	1.054	0.069	40.6	5.4
90	1.044	0.066	45.9	5.8	1.053	0.066	45.9	5.8
100	1.047	0.061	50.9	6.3	1.057	0.063	50.6	6.3
110	1.047	0.058	56.3	6.3	1.054	0.060	56.1	6.4
120	1.048	0.057	61.1	6.6	1.052	0.056	61.1	6.7
130	1.051	0.055	66.4	7.1	1.054	0.053	66.3	6.9
140	1.053	0.054	71.6	7.4	1.053	0.051	71.3	7.1
150	1.051	0.053	76.0	7.7	1.051	0.050	76.2	7.5
160	1.048	0.049	81.6	7.4	1.054	0.048	81.2	7.6

$$= \sum_{\substack{i,j\geq 0\\ i+j\leq n}} \frac{1}{(2n+2-i)(2n+2-j)} - \sum_{\substack{i,j\geq 0\\ i+j\leq n-1}} \frac{1}{(2n-i)(2n-j)}.$$

Shifting the index of the first sum to get the same summand type in both sums yields

$$= \sum_{\substack{i,j\geq -2\\ i+j\leq n-4}} \frac{1}{(2n-i)(2n-j)} - \sum_{\substack{i,j\geq 0\\ i+j\leq n-1}} \frac{1}{(2n-i)(2n-j)}.$$

We now split the sums to sums with index range $i, j \geq 0$, $i + j \leq n - 4$ so that they can cancel. The remainder is as follows. For the first sum, it is used that it is symmetric in i and j. The term $\frac{(4n+3)^2}{4(n+1)^2(2n+1)^2}$ is the sum of the values where $-2 \leq i, j \leq -1$. This has to be subtracted from the first term as otherwise these values would be counted twice.

$$D(n) = 2 \cdot \sum_{\substack{-2\leq i\leq -1, j\geq -2\\ i+j\leq n-4}} \frac{1}{(2n-i)(2n-j)} - \frac{(4n+3)^2}{4(n+1)^2(2n+1)^2} - \sum_{\substack{i,j\geq 0\\ i+j=n-1}} \frac{1}{(2n-i)(2n-j)} - \sum_{\substack{i,j\geq 0\\ i+j=n-2}} \frac{1}{(2n-i)(2n-j)} - \sum_{\substack{i,j\geq 0\\ i+j=n-3}} \frac{1}{(2n-i)(2n-j)}.$$

Splitting the first sum into two parts with $i = -1$ and $i = -2$ and substituting j by $a - i$ where $i + j = a$ yields

$$\begin{aligned} D(n) = &\sum_{j=-2}^{n-3} \frac{2}{(2n+1)(2n-j)} \\ &+ \sum_{j=-2}^{n-2} \frac{2}{(2n+2)(2n-j)} \\ &- \frac{(4n+3)^2}{4(n+1)^2(2n+1)^2} \\ &- \sum_{i=0}^{n-1} \frac{1}{(2n-i)(n+1+i)} \\ &- \sum_{i=0}^{n-2} \frac{1}{(2n-i)(n+2+i)} \\ &- \sum_{i=0}^{n-3} \frac{1}{(2n-i)(n+3+i)}. \end{aligned}$$

Using the notation $H_n = \sum_{i=1}^{n} \frac{1}{i}$ for the n-th harmonic number and partial fraction decomposition to get denominators linear in n for the last two summations, we get

$$\begin{aligned}
D(n) = &\frac{2H_{2n+2} - 2H_{n+2}}{2n+1} + \frac{2H_{2n+2} - 2H_{n+1}}{2n+2} \\
&- \frac{(4n+3)^2}{4(n+1)^2(2n+1)^2} - \frac{2H_{2n} - 2H_n}{3n+1} \\
&- \frac{2H_{2n} - 2H_{n+1}}{3n+2} - \frac{2H_{2n} - 2H_{n+2}}{3n+3} \\
= &\frac{2H_{2n} + \frac{2}{2n+1} + \frac{2}{2n+2} - 2H_n - \frac{2}{n+1} - \frac{2}{n+2}}{2n+1} \\
&+ \frac{2H_{2n} + \frac{2}{2n+1} + \frac{2}{2n+2} - 2H_n - \frac{2}{n+1}}{2n+2} \\
&- \frac{(4n+3)^2}{4(n+1)^2(2n+1)^2} - \frac{2H_{2n} - 2H_n}{3n+1} \\
&- \frac{2H_{2n} - 2H_n - \frac{2}{n+1}}{3n+2} \\
&- \frac{2H_{2n} - 2H_n - \frac{2}{n+1} - \frac{2}{n+2}}{3n+3}.
\end{aligned}$$

Finally, simplification yields

$$\begin{aligned}
D(n) = &-(H_{2n} - H_n)\cdot \\
&\frac{9n^2 + 11n + 4}{3(n+1)(2n+1)(3n+1)(3n+2)} \\
&+ \frac{8n^2 + 13n + 6}{12(n+1)^2(2n+1)^2(3n+2)}.
\end{aligned}$$

To get bounds on $E(n)$ using Equation (1), we first use that

$$\begin{aligned}
\sum_{n=1}^{\infty} &\frac{8n^2 + 13n + 6}{12(n+1)^2(2n+1)^2(3n+2)} \\
&= -\frac{1}{4} - \frac{\pi}{\sqrt{3}} + \frac{\pi^2}{9} - \frac{10\ln(2)}{3} + \ln(27). \quad (2)
\end{aligned}$$

Then, observe that $H_{2n} - H_n$ is a non-negative number monotonically increasing with n. Also, this is an alternating harmonic number that for $n \to \infty$ converges to $\ln(2)$. For $n = 80$, $H_{2n} - H_n$ can be calculated and results in a fraction, which is > 0.69. Therefore, for $n \geq 80$,

$$0.69 < H_{2n} - H_n < \ln(2) \quad (3)$$

Now, computing the partial sum

$$\sum_{n=1}^{79} -(H_{2n} - H_n)\frac{9n^2 + 11n + 4}{3(n+1)(2n+1)(3n+1)(3n+2)}$$

exactly and the limes

$$\sum_{n=80}^{\infty} -(H_{2n} - H_n)\frac{9n^2 + 11n + 4}{3(n+1)(2n+1)(3n+1)(3n+2)}$$

after substituting for $H_{2n} - H_n$ the lower and upper bounds given by (3), Equations (1) and (2) yield

$$0.1859 < \lim_{n\to\infty} E(n) < 0.1860.$$

Thus, we get for the lower bound

$$\begin{aligned}
\lim_{n\to\infty} (E(n) + E(2n)) &= 2 \cdot \lim_{n\to\infty} E(n) \\
&> 2 \cdot 0.1859 \\
&= 0.3718
\end{aligned}$$

and for the upper bound

$$\begin{aligned}
\lim_{n\to\infty} (E(n) + E(2n, 2n, 2n)) &= \lim_{n\to\infty} E(n) \\
&\quad + \lim_{n\to\infty} E(2n, 2n, 2n) \\
&< 0.1860 + \frac{\pi^2}{6} \\
&< 1.8310.
\end{aligned}$$

□

We remark that the upper bound computed in Theorem 3.2 is greater than the expected optimal value of the random assignment problem $\frac{\pi^2}{6} = 1.6449\ldots$. We believe that it must be possible to reduce it, because moving from an assignment problem in a complete bipartite graph with $4n$ vertices on each side to a HAP in $G_{2,2n}$ adds more possibilities (still all assignments are feasible solutions but using hyperassignments with proper hyperedges gives additional ones). Indeed, it is clear that if we do not prescribe the number of proper hyperedges in an optimal solution, the expected optimal value of a hyperassignment in $G_{2,2n}$ will tend to some number $\leq \frac{\pi^2}{6}$. As already discussed, the computational results shown in Table 1 suggest that the correct number is some value around 1.05, much smaller than $\frac{\pi^2}{6}$.

4 Regularity rewarding costs

Hypergraph assignment problems arising from practical applications feature costs for proper hyperedges that depend on the costs of the edges that they contain. Indeed, proper hyperedges model a "reward" for choosing combinations of edges; in this way, one can model a so-called regularity of the solution [Borndörfer et al., 2011]. More precisely, one considers partitioned bipartite hypergraphs and wants to favor the simultaneous choice of a set of edges that connects all nodes in a certain part in U to all nodes in a certain part in V. To this purpose, one introduces a proper hyperedge that represents the union of such pairwise disjoint edges and that has a cost that is smaller than the sum of the edge costs. If different edge combinations result in the same hyperedge, the cost is inferred from the edge set with the minimum cost sum. Here is a more formal statement.

Definition 4.1. Let $G = (U, V, E)$ be a partitioned hypergraph. For $e \in E$, let

$$\mathcal{E}(e) := \{E' \subseteq E_1 : e_1 \cap e_2 = \emptyset \; \forall e_1, e_2 \in E' \text{ with } e_1 \neq e_2, \bigcup E' = e\}$$

be the set of all pairwise disjoint edge sets with union e.

For some *penalty* $p \geq 0$, we call a cost function $c_E^p : E \to \mathbb{R}$ *regularity-rewarding* if for all proper hyperedges $e \in E_2$,

$$c_E(e) = \min_{E' \in \mathcal{E}(e)} \left(\sum_{e' \in E'} c_E(e') - p \cdot |E'| \right).$$

The greater p, the more irregularity is punished and regularity rewarded. We remark that the cost of a hyperedge in a vehicle rotation planning model depends on several other parameters such as an additional irregularity penalty for hyperedges that are not inclusion-wise maximal [Borndörfer et al., 2011]. This is the reason why we call p a penalty and not a bonus or a reward.

A way to define a regularity-rewarding random cost function c_E^p is to draw a *random basic cost* r_e for each edge $e \in E_1$, e. g., from a uniform distribution on $[0, 1]$ or an exponential distribution with mean 1, and then to set

$$c_E^p(e) := \begin{cases} r_e + p & \text{if } e \text{ is an edge,} \\ \min_{E' \in \mathcal{E}(e)} \sum_{e' \in E'} r_{e'} & \text{if } e \text{ is a proper hyperedge.} \end{cases}$$

In the following, we will assume that c_E^p is structured in this way with arbitrary r_e.

For a given bipartite hypergraph $G_{2,n} = (U, V, E)$ and random basic costs r_e for the edges $e \in E_1$, we denote by $z(h, p)$ the minimal cost value of a hyperassignment with penalty p that contains exactly $0 \leq h \leq n$ proper hyperedges.

Obviously, the number of proper hyperedges and the value of an optimal solution will depend on p. If $p = 0$, there is no reward for choosing a proper hyperedge. For every solution using proper hyperedges, we can find a solution with the same value that contains only edges by replacing each proper hyperedge $\{u_i, u_i', v_j, v_j'\}$ by either the two edges $\{u_i, v_j\}, \{u_i', v_j'\}$ or the two edges $\{u_i, v_j'\}, \{u_i', v_j\}$ depending on which two edges have the lower cost sum. On the other hand, if p is very large, choosing edges for a solution becomes so disadvantageous that the number of proper hyperedges in an optimal solution will become very high.

Fortunately, knowledge about the case $p = 0$ gives information about all other penalties as the following theorem shows. Thus, we do not need to analyze random HAPs for regularity-rewarding cost functions separately for each penalty p.

For some random basic cost distribution, we denote by $Z(h)$ the expected value of $z(h, 0)$ with respect to this distribution. Although $z(h, 0)$ is defined only for integral h, we will view $Z(h)$ as a continuous, monotonically increasing, differentiable function on $[0, n]$. This will allow us to formulate our result in a much easier way than if we would have to replace the derivative by its discretization. We can require $Z(h)$ to be monotonically increasing, because $z(h, 0)$ is monotonically increasing with increasing h. The reason is that, as described above, using proper hyperedges in the solution cannot lead to smaller optimal values than using only edges in the case $p = 0$.

Theorem 4.2. *Consider the complete bipartite hypergraph $G_{2,n} = (U, V, E)$ and let r_e, $e \in E_1$ be random basic costs chosen from some random distribution. Denote by $h_d^1, \ldots h_d^k$ the solutions to the equation $Z'(h) = 2p$ and let*

$$h^* = \arg \min_{h \in \{0, h_d^1, \ldots, h_d^k, n\}} (Z(h) - (2n - 2h)p)$$

Then the expected number of proper hyperedges in an optimal solution to the HAP in $G_{2,n}$ w. r. t. c_E^p with basic random costs r_e is h^* and the expected optimal value of the random HAP is

$$Z(h^*) - (2n - 2h^*)p.$$

Proof. First, observe that

$$z(h, p) = z(h, 0) + (2n - 2h)p$$

holds since the cost of each hyperassignment H w. r. t. c_E^p is

$$\begin{aligned} c_E^p(H) &= \sum_{e \in E} c_E^p(e) \\ &= \sum_{e \in E_1} c_E^p(e) + \sum_{e \in E_2} c_E^p(e) \\ &= \sum_{e \in E_1} (r_e + p) + \sum_{e \in E_2} \min_{E' \in \mathcal{E}(e)} \sum_{e' \in E'} r_{e'} \\ &= \sum_{e \in E_1} r_e + |E_1 \cap H| p \\ &\quad + \sum_{e \in E_2} \min_{E' \in \mathcal{E}(e)} \sum_{e' \in E'} r_{e'} \\ &= \sum_{e \in E_1} r_e + (2n - 2|E_2 \cap H|)\, p \\ &\quad + \sum_{e \in E_2} \min_{E' \in \mathcal{E}(e)} \sum_{e' \in E'} r_{e'} \\ &= \sum_{e \in E} c_E^0(e) + (2n - 2|E_2 \cap H|)\, p \\ &= c_E^0(H) + (2n - 2|E_2 \cap H|)\, p. \end{aligned}$$

Since this holds for all random basic costs, it also holds for the expected value of all random basic cost distributions and we get

$$\mathbf{E}(z(h, p)) = Z(h) + (2n - 2h)p.$$

Its derivative is $Z'(h) - 2p$. A minimum of a differentiable function is attained either at the bounds or where the derivative is equal to zero, which proves the theorem. □

5 Discussion

In this paper, we have presented results on the expected minimum cost of the random hypergraph assignment problem for two types of cost functions.

For the first type, i. i. d. exponential random variables with mean 1 or i. i. d. uniform random variables on $[0, 1]$, we conjectured that the number of proper hyperedges in an optimal solution is expected to be n for the hypergraph $G_{2,2n}$, and showed computational results supporting this conjecture. Assuming this number of proper hyperedges in an optimal solution, we proved bounds on the expected optimal value for a vertex number tending to infinity. A proof of our conjecture as well as convergence results and either sharper bounds or an exact limit would be a natural continuation of our work towards a generalization of Mézard and Parisi's Conjecture. A first step is to extend the proof of our bounds to fixed numbers of hyperedges other than n by altering the computation.

For the second type of regularity-rewarding cost functions, we established a connection between results for different penalty values. This result could be extended by an analysis similar to that for the first cost function type in future.

All our results hold for complete partitioned hypergraphs $G_{2,n}$. A further line of research could try to extend these results to bipartite hypergraphs with larger part sizes or even bipartite hypergraphs that are not partitioned or/and not complete.

Our results show how to approach the random HAP using results for the random assignment problem. Probably approaches using more sophisticated probability-theoretical results are needed to understand more about the problem.

References

[Aldous, 1992] Aldous, D. (1992). Asymptotics in the random assignment problem. *Probability Theory and Related Fields*, 93:507–534.

[Borndörfer and Heismann, 2012] Borndörfer, R. and Heismann, O. (2012). The hypergraph assignment problem. Technical Report 12-14, ZIB.

[Borndörfer et al., 2011] Borndörfer, R., Reuther, M., Schlechte, T., and Weider, S. (2011). A Hypergraph Model for Railway Vehicle Rotation Planning. In Caprara, A. and Kontogiannis, S., editors, *11th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2011)*, volume 20 of *OpenAccess Series in Informatics (OASIcs)*, pages 146–155, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ZIB Report 11-36.

[Coppersmith and Sorkin, 1999] Coppersmith, D. and Sorkin, G. B. (1999). Constructive bounds and exact expectations for the random assignment problem. *Random Structures & Algorithms*, 15(2):113–144.

[Heismann and Borndörfer, 2013] Heismann, O. and Borndörfer, R. (2013). The random hypergraph assignment problem. In *Proceedings of the 16th International Multiconference INFORMATION SOCIETY – IS 2013*.

[Krokhmal and Pardalos, 2009] Krokhmal, P. A. and Pardalos, P. M. (2009). Random assignment problems. *European Journal of Operational Research*, 194(1):1–17.

[Linusson and Wästlund, 2004] Linusson, S. and Wästlund, J. (2004). A proof of Parisi's conjecture on the random assignment problem. *Probability Theory and Related Fields*, 128(3):419–440.

[Maróti, 2006] Maróti, G. (2006). *Operations research models for railway rolling stock planning*. PhD thesis, Technische Universiteit Eindhoven.

[Mézard and Parisi, 1985] Mézard, M. and Parisi, G. (1985). Replicas and optimization. *Journal de Physique Lettres*, 46:771–778.

Strategic Deployment in Graphs

Elmar Langetepe and Andreas Lenerz
University of Bonn, Department of Computer Science I, Germany

Bernd Brüggemann
FKIE, Fraunhofer-Institute, Germany

Keywords: deployment, networks, optimization, algorithms

Received: June 20, 2014

Conquerors of old (like, e.g., Alexander the Great or Ceasar) had to solve the following deployment problem. Sufficiently strong units had to be stationed at locations of strategic importance, and the moving forces had to be strong enough to advance to the next location. To the best of our knowledge we are the first to consider the (off-line) graph version of this problem. While being NP-hard for general graphs, for trees the minimum number of agents and an optimal deployment can be computed in optimal polynomial time. Moreover, the optimal solution for the minimum spanning tree of an arbitrary graph G results in a 2-approximation of the optimal solution for G.

Povzetek: Predlagana je izvirna rešitev za razvrstitev enot in premikanje na nove pozicije.

1 Introduction

Let $G = (V, E)$ be a graph with non-negative edge end vertex weights w_e and w_v, respectively. We want to minimize the number of agents needed to traverse the graph subject to the following conditions. If vertex v is visited for the first time, w_v agents must be left at v to cover it. An edge e can only be traversed by a force of at least w_e agents. Finally, all vertices should be covered. All agents start in a predefined start vertex $v_s \in V$. In general they can move in different groups. The problem is denoted as a *strategic deployment problem* of $G = (V, E)$.

The above rules can also easily be interpreted for modern non-military applications. For a given network we would like to rescue or repair the sites (vertices) by a predifined number of agents, whereas traversing along the routes (edges) requires some escorting service. The results presented here can also be applied to a problem of positioning mobile robots for guarding a given terrain; see also [3].

We deal with two variants regarding a *notification* at the end of the task. The variants are comparable to *routes* (round-trips) and *tours* (open paths) in traveling-salesman scenarios.

(Return) Finally some agents have to return to the start vertex and report the success of the whole operation.

(No-return) It suffices to fill the vertices as required, no agents have to return to the start vertex.

Reporting the success in the return variant means, that finally a set, M, of agents return to v_s and the union of *all* vertices visited by the members of M equals V.

We give an example for the no-return variant for the graph of Figure 1. It is important that the first visit of a vertex immediately binds some units of the agents for the control of the vertex. For start vertex $v_s = v_1$ at least 23 agents are required. We let the agents run in a single group. In the beginning one of the agents has to be placed immediately in v_1. Then we traverse edge e_1 of weight 1 with 22 agents from v_1 to v_2. Again, we have to place one agent immediately at v_2. We move from v_2 to v_3 along e_2 of weight 20 with 21 agents. After leaving one agent at v_3 we can still move back along edge e_2 (weight 20) from v_3 to v_2 with 20 agents. The vertex v_2 was already covered before. With 20 agents we now visit v_4 (by traversing e_1 (weight 1) and e_3 (weight 1), the vertex v_1 was already covered and can be passed without loss). We have to place one agent at v_4 and proceed with 19 agents along e_3 (weight 1), e_1 (weight 1) and e_4 (weight 7) to v_5 where we finally have to place 15 agents. 4 agents are not settled.
It can be shown that no other traversal requires less than

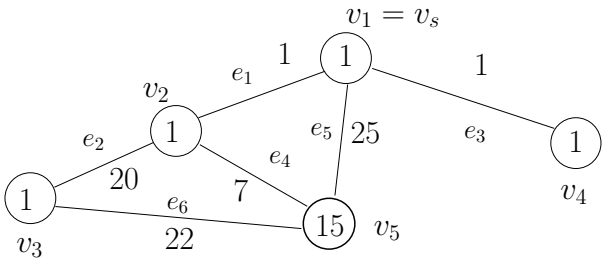


Figure 1: A graph with edge and vertex weights. If the agents have to start at the vertex v_1 an optimal deployment strategy requires 23 agents and visits the vertices and edges in a single group in the order (v_1, e_1, v_2, e_2, v_3, e_2, v_2, e_1, v_1, e_3, v_4, e_3, v_1, e_1, v_2, e_4, v_5). The traversal fulfills the demand on the vertices in the order v_1, v_2, v_3, v_4, v_5 by the first visits w.r.t. the above sequence. At the end 4 agents are not settled.

23 agents. By the results of Section 3 it turns out that the return variant solution has a different visiting order v_1, v_2, v_3, v_5, v_4 and requires 25 agents.

Although the computation of an efficient flow of some items or goods in a weighted network has a long tradition and has been considered under many different aspects the problem presented here cannot be covered by known (multi-agent) routing, network-flow or agent-traversal problems.

For example, in the classical *transportation network* problem there are source and sink nodes whose weights represent a supply or a demand, respectively. The weight of an edge represents the transportation cost along the edge. One would like to find a transshipment schedule of minimum cost that fulfils the demand of all sink nodes from the source nodes; see for example the monograph of [4] and the textbooks [10, 1] . The solutions of such problems are often based on linear programming methods for minimizing (linear) cost functions.

In a *packet routing* scenario for a given weighted network, m packet sets each consisting of s_i packets for $i = 1, 2, \dots m$ are located at m given source nodes. For each packet set a specified sink node is given. Here the edge weights represent an *upper bound* on the number of single packets that can be transported along the edge in one time step. One is for example interested in minimizing the so-called *makespan*, i.e., the time when the last packet arrives at its destination; see for example [13]. For a general overview see also the survey [9].

Similarily, in [11] the *multi-robot routing* problem considers a set of agents that has to be moved from their start locations to the target locations. For the movement between two locations a cost function is given and the goal is to minimize the path costs. Such multi-robot routing problems can be considered under many different constraints [16]. For the purpose of patrolling see the survey [14].

Additionally, *online multi-agent traversal* problems in discrete environments have attracted some attention. The problem of exploring an *unknown* graph by a set of k agents was considered for example in [5, 6]. Exploration means that at the end all vertices of the graph should have been visited. In this motion planning setting either the goal is to optimize the number of overall steps of the agents or to optimize the makespan, that is to minimize the time when the last vertex is visited.

Some other work has been done for k cooperative cleaners that move around in a grid-graph environment and have to *clean* each vertex in a contaminated environment; see [2, 17]. In this model the task is different from a simple exploration since after a while *contaminated* cells can reinfect cleaned cells. One is searching for strategies for a set of k agents that guarantee successful cleanings.

Our result shows that finding the minimum number of agents required for the strategic deployment problem is NP-hard for general graphs even if all vertex weights are equal to one. In Section 2 this is shown by a reduction from 3-Exact-Cover (3XC). The optimal number of agents for the minimum spanning tree (MST) of the graph G gives a 2-approximation for the graph itself; see Section 3. For weighted trees we can show that the optimal number of agents and a corresponding strategy for T can be computed in $\Theta(n \log n)$ time. Altogether, a 2-approximation for G can be computed efficiently. Additionally, some structural properties of the problem are given.

The problem definition gives rise to many further interesting extensions. For example, here we first consider an offline version with global communication, but also online versions with limited communication might be of some interest. Recently, we started to discuss the makespan or traversal time for a *given* optimal number of agents. See for example the masterthesis [12] supervised by the second author.

2 General graphs

We consider an edge- and vertex-weighted graph $G = (V, E)$. Let $v_s \in V$ denote the start vertex for the traversal of the agents. W.l.o.g. we can assume that G is connected and does not have multi-edges.

We allow that a traversal strategy subdivides the agents into groups that move separately for a while. A traversal strategy is a schedule for the agents. At any time step any agent decides to move along an outgoing edge of its current vertex towards another vertex or the agent stays in its current vertex. We assume that any edge can be traversed in one time step. *Long* connections can be easily modelled by placing intermediate vertices of weight 0 along the edge. Altogether, agent groups can arrive at some vertex v at the same time from different edges.

The schedule is called *valid* if the following condition hold. For the movements during a time step the number of agents that use a single edge has to exceed the edge weight w_e. After the movement for any vertex v that already has been visited by some agents, the number of agents that are located at v has to exceed the vertex weight v_e. From now on an *optimal deployment strategy* is a valid schedule that uses the minimum number of agents required.

Let $N := \sum_{v \in V} w_v$ denote the number of agents required for the vertices in total. Obviously, the maximum overall edge weight $w_{\max} := \max\{w_e | e \in E\}$ of the graph gives a simple upper bound for the additional agents (beyond N) used for edge traversals. This means that at most $w_{\max} + N$ agents will be required. With $w_{\max} + N$ agents one can for example use a DFS walk along the graph and let the agents run in a single group.

2.1 NP-hardness for general graphs

For showing that computing the optimal number of agents is NP-hard in general we make use of a reduction of the 3-Exact-Cover (3XC) problem. We give the proof for the no-return variant, first.

The problem 3-Exact-Cover (3XC) is given as follows. Given a finite ground set X of $3n$ items and a set $\mathbb{F}$ of sub-

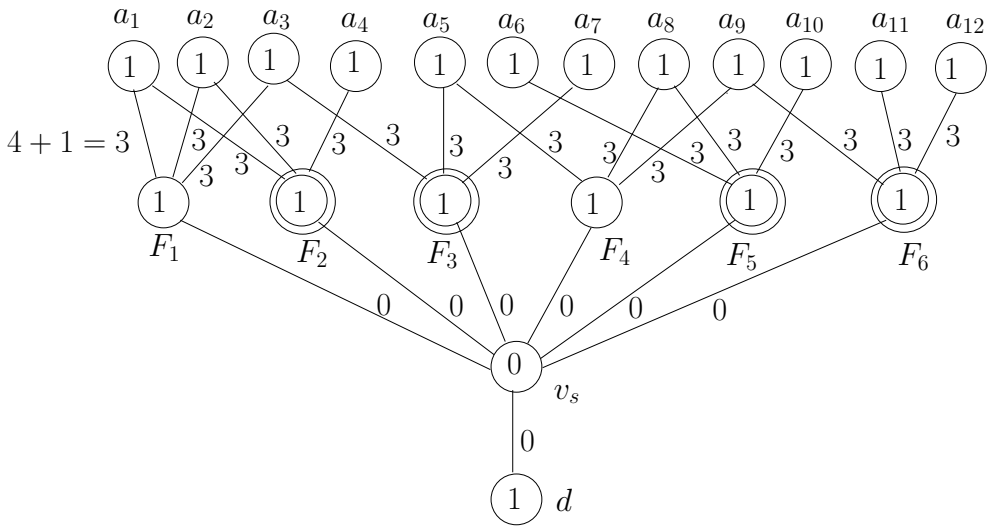


Figure 2: For $X = \{a_1, a_2, \ldots, a_{12}\}$ and the subsets $\mathbb{F} = \{F_1, F_2, \ldots, F_6\}$ with $F_1 = \{a_1, a_2, a_3\}$, $F_2 = \{a_1, a_2, a_4\}$, $F_3 = \{a_3, a_5, a_7\}$, $F_4 = \{a_5, a_8, a_9\}$, $F_5 = \{a_6, a_8, a_{10}\}$ and $F_6 = \{a_9, a_{11}, a_{12}\}$ there is an exact 3-cover with F_2, F_3, F_5 and F_6. For the start vertex v_s an optimal traversal strategy moves in a single group. We start with $3n + m + 1 = 19$ agents, first visit the vertices of F_2, F_3, F_5 and F_6 and cover all elements from there, visiting an element vertex last. After that $3n + n = 4n = 16$ agents have been placed and $m - n + 1 = 3$ still have to be placed including the dummy node. With this number of agents we can move back along the corresponding edge of weight $m - n + 1 = 3$ and place the remaining 3 agents.

sets of X so that any $F \in \mathbb{F}$ contains exactly 3 elements of X. The decision problem of 3XC is defined as follows: Does $\mathbb{F}$ contain an exact cover of X of size n? More precisely is there a subset $F_c \subseteq \mathbb{F}$ so that the collection F_c contains all elements of X and F_c consists of precisely n subsets, i.e. $|F_c| = n$. It was shown by Karp that this problem is NP-hard; see Garey and Johnsson[8].

Let us assume that such a problem is given. We define the following deployment problem for $(X, \mathbb{F})$. Let $X = \{a_1, a_2, \ldots, a_{3n}\}$. For any a_i there is an *element vertex* $v(X)_i$ of weight 1. Let $\mathbb{F}$ consists of $m \geq n$ subsets of size 3, say $\mathbb{F} = \{F_1, F_2, \ldots, F_m\}$. For any $F_j = \{a_{j_1}, a_{j_2}, a_{j_3}\}$ we define a *set vertex* $v(\mathbb{F})_j$ of weight 1 and we insert three edges $(v(\mathbb{F})_j, v(X)_{j_1})$, $(v(\mathbb{F})_j, v(X)_{j_2})$ and $(v(\mathbb{F})_j, v(X)_{j_3})$ each of weight $m - n + 1$. Additionally, we use a sink vertex v_s of weight $w_{v_s} = 0$ and insert m edges $(v_s, v(F)_j)$ from the sink to the set vertices of $\mathbb{F}$. All these edges get weight 0. Additionally, one dummy node d of weight $w_d = 1$ is added as well as an edge (v_s, d) of weight 0.

Figure 2 shows an example of the construction for the set $X = \{a_1, a_2, \ldots, a_{12}\}$ and the subsets $\mathbb{F} = \{F_1, F_2, \ldots, F_6\}$ with $F_1 = \{a_1, a_2, a_3\}$, $F_2 = \{a_1, a_2, a_4\}$, $F_3 = \{a_3, a_5, a_7\}$, $F_4 = \{a_5, a_8, a_9\}$, $F_5 = \{a_6, a_8, a_{10}\}$ and $F_6 = \{a_9, a_{11}, a_{12}\}$ with $m - n + 1 = 3$.

Starting from the sink node v_s we are asking whether there is an agent traversal schedule that requires exactly $N = 3n + m + 1$ agents. If there is such a traversal this is optimal (we have to fill all vertices). The following result holds. *If and only if $(X, \mathbb{F})$ has an exact 3-cover, the given strategic deployment problem can be solved with exactly $N = 3n + m + 1$ agents.*

Let us first assume that an exact 3-cover exists. In this case we start with $N = 3n + m + 1$ agents at v_s and let the agents run in a single group. First we successively visit the set vertices that build the cover and fill all $3n$ element vertices using $3n + n$ agents in total. More precisely, for the set vertices that build the cover we successively enter such a vertex from v_s, place one agent there and fill all three element vertices by moving back and forth along the corresponding edges. Then we move back to v_s and so on. At any such operation the set of agents is reduced by 4. Finally, when the last set vertex of the cover was visited, we end in the overall last element vertex. After fulfilling the demand there, we still have $N - 4n = 3n + m + 1 - 4n = m - n + 1$ agents for traveling back to v_s along the corresponding edges. Now we fill the remaining set vertices by successively moving forth and back from v_s along the edges of weight 0. Finally, with the last agent, we can visit and fill the dummy node.

Conversely, let us assume that there is no exact 3-cover for $(X, \mathbb{F})$ and we would like to solve the strategic deployment problem with $N = 3n + m + 1$ agents. At some point an optimal solution for the strategic deployment problem has to visit the last element vertex $v(X)_j$, starting from a set vertex $v(\mathbb{F})_i$. Let us assume that we are in $v(\mathbb{F})_i$ and would like to move to $v(X)_j$ now and $v(X)_j$ was not visited before. Since there was no exact 3-cover we have already visited strictly more than n set vertices at this point and exactly $3n - 1$ element vertices have been visited. This means at least $3n - 1 + n + 1 = 4n$ agents have been used.

Now we consider two cases. If the dummy node was already visited, starting with N agents we only have at most $3n + m + 1 - 4n - 1 = m - n$ agents for travelling toward

the last element vertex, this means that we require an additional agent beyond N for traversing the edge of weight $m-n+1$. If the dummy node was not visited before and we now decide to move to the last element vertex, we have to place one agent there. This means for travelling back from the last element vertex along some edge (at least the dummy must still be visited), we still require $m-n+1$ agents. Starting with N at the beginning at this stage only $3n+m+1-4n-1 = m-n$ are given. At least one additional agent beyond N is necessary for travelling back to the dummy node for filling this node.

Altogether, we can answer the 3-Exact-Cover decision problem by a polynomial reduction into a strategic deployment problem. The proof also works for the return variant, where at least one agent has to return to v_s, if we omit the dummy node, make use of $N := 3n+m$ and set the non-zero weights to $m-n$.

Theorem 1. *Computing the optimal number of agents for the strategic deployment problem of a general graph G is NP-hard.*

2.2 2-approximation by the MST

For a general graph $G=(V,E)$ we consider its minimum spanning tree (MST) and consider an optimal deployment strategy on the MST.

Lemma 1. *An optimal deployment strategy for the minimum spanning tree (MST) of a weighted graph $G=(V,E)$ gives a 2-approximation of the optimal deployment strategy of G itself.*

Proof: Let e be an edge of the MST of G with maximal weight w_e among all edges of the MST. It is simply the nature of the MST, that any traversal of the graph that visits all vertices, has to use an edge of weight at least w_e. The optimal deployment strategy has to traverse an egde of weight at least w_e and requires at least $k_{\text{opt}} \geq \max\{N, w_e\}$ agents. The optimal strategy for the MST approach requires at most $k_{\text{MST}} \leq w_e + N$ agents which gives $k_{\text{MST}} \leq 2k_{\text{opt}}$. □

2.3 Moving in a single group

In our model it is allowed that the agents run in different groups. For the computation of the optimal number of agents required, this is not necessary. Note that *group-splitting* strategies are necessary for minimizing the *completion time*. Recently, we also started to discuss such optimization criteria; see the masterthesis [12] supervised by the first author.

During the execution of the traversal there is a set of *settled* agents that already had to be placed at the visited vertices and a set of *non-settled* agents that still move around. We can show that the non-settled agents can always move in a single group. For simplicity we give a proof for trees.

Theorem 2. *For a given weighted tree T and the given minimal number of agents required, there is always a deployment strategy that lets all non-settled agents move in a single group.*

Proof: We can reorganize any optimal strategy accordingly, so that the same number of agents is sufficient.

Let us assume that at a vertex v a set of agents X is separated into two groups X_1 and X_2 and they separately explore disjoint parts T_1 and T_2 of the tree. Let w_{T_i} be the maximum edge weight of the edges traversed by the agents X_i in T_i, respectively. Clearly $|X_i| \geq w_{T_i}$ holds. Let $|X_1| \geq |X_2|$ hold and let X_2' be the set of non-settled agents of X_2 after the exploration of T_2. We can explore T_2 by $X = X_1 \cup X_2$ agents first, and we do not need the set X_2 there. $|X_1| \geq w_{T_2}$ means that we can move back with $X_1 \cup X_2'$ agents to v and start the exploration for T_1.

The argument can be applied successively for any split of a group. This also means that we can always collect all non-settled vertices in a single moving group. □

Note that the above Theorem also holds for general graphs G. The general proof requires some technical details because a single vertex might collect agents from different sources at the first visit. We omit the rather technical proof here.

Proposition 1. *For a given weighted graph G and the given minimal number of agents required, there is always a deployment strategy that lets all non-settled agents move in a single group.*

2.4 Counting the number of agents

From now on we only consider strategies where the non-settled agents always move in a single group. Before we proceed, we briefly explain how the number of agents can be computed for a strategy given by a sequence S of vertices and edges that are visited and crossed successively. A pseudocode is given in Algorithm 1. The simple counting procedure will be adapted for Algorithm 2 in Section 3.3 for counting the optimal number of agents efficiently.

For a sequence S of vertices and edges that are visited and crossed by a single group of agents the required number of agents can be computed as follows. We count the number of additional agents beyond N (where N is the overall sum of the vertex weights) in a variable add. In another variable curr we count the number of agents currently available. In the beginning add $:= 0$ and curr $:= N$ holds. A strategy successively crosses edges and visits vertices of the tree, this is given in the sequence S. We always choose the next element x (edge e or vertex v) out of the sequence. If we would like to cross an edge e, we check whether curr $\geq w_e$ holds. If not we set add $:=$ add $+ (w_e -$ curr$)$ and curr $:= w_e$ and can cross the edge now. If we visit a vertex v we similarily check whether curr $\geq w_v$ holds. If this is true, we set curr $:=$ curr $- w_v$. If this is not true, we set add $:=$ add $+ (w_v -$ curr$)$ and curr $:= 0$. In any case we set

Algorithm 1: Number of agents, for $G = (V, E)$ and given sequence S of vertices and edges.

$N := \sum_{v \in V} w_v$; curr $:= N$; add $:= 0$; $x :=$ first(S);
while $x \neq$ NIL **do**
  if x is an edge e **then**
    if curr $< w_e$ **then**
      add $:=$ add $+ (w_e -$ curr$)$; curr $:= w_e$;
    end if
  else if x is a vertex v **then**
    if curr $< w_v$ **then**
      add $:=$ add $+ (w_v -$ curr$)$; curr $:= 0$;
    else
      curr $:=$ curr $- w_v$;
    end if
    $w_v := 0$;
  end if
  $x :=$ next(S);
end while
RETURN $N +$ add

$w_v := 0$, the vertex is filled after the first visit. Obviously this simple algorithm counts the number of agents required in the number of traversal steps of the single group.

3 Optimal solutions for trees

Lemma 1 suggests that for a 2-approximation for a graph G, we can consider its MST. Thus, it makes sense to solve the problem efficiently for trees. Additionally, by Theorem 1 it suffices to consider strategies of single groups.

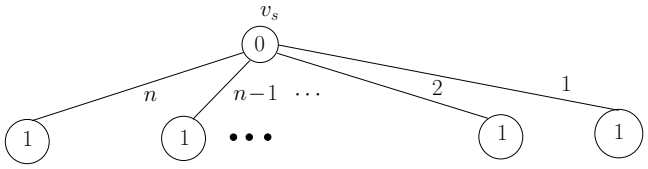


Figure 3: An optimal strategy that starts and ends in v_s has to visit the leafs with respect to the decreasing order of the edge weights. The minimal number of agents is $n+1$. Any other order will lead to at least one extra agent.

3.1 Computational lower bound

Let us first consider the tree in Figure 3 and the return variant. Obviously it is possible to use $n + 1$ agents and visit the edges in the decreasing order of the edge weights $n, n-1, \ldots, 1$. Any other order will increase the number of agents. If for example in the first step an edge of weight $k \neq n$ is visited, we have to leave one agent at the corresponding vertex. Since the edge of weight n still has to be visited and we have to return to the start, $n + 1$ agents in total will not be sufficient. So first the edge of weight n has to be visited. This argument can be applied successively.

Altogether, by the above example there seems to be a computational lower bound for trees with respect to sorting the edges by their weights. Since integer values can be sorted by bucket sort in linear time, such a lower bound can only be given for real edge and vertex weights. This seems to be a natural extension of our problem. We consider the transportation of sufficient material along an edge (condition 1.). Additionally, the demand of a vertex has to be fully satisfied before transportation can go on (condition 2.). How many material is required?

For a computational lower bound for trees we consider the Uniform-Gap problem. Let us assume that n unsorted real numbers $x_1, x_2, \ldots, x_n$ and an $\epsilon > 0$ are given. Is there a permutation $\pi : \{1, \ldots, n\} \to \{1, \ldots, n\}$ so that $x_{\pi(i-1)} = x_{\pi(i)} + \epsilon$ for $i = 2, \ldots, n$ holds? In the algebraic decision tree model this problem has computational time bound $\Omega(n \log n)$; see for example [15].

In Figure 3 we simply replace the vertex weights of 1 by ϵ and the n edge weights by $x_1, x_2, \ldots, x_n$. With the same arguments as before we conclude: If and only if the Uniform-Gap property holds, a unique optimal strategy has to visit the edges in a single group in the order of decreasing edge weights $x_{\pi(1)} > x_{\pi(2)} > \cdots > x_{\pi(n)}$ and requires an amount of $x_{\pi(1)} + \epsilon$ in total. Any other order will lead to at least one extra ϵ.

The same arguments can be applied to the no return variant by simple modifications. Only the vertex weight of the smallest x_j, say $x_{\pi(n)}$, is set to $x_{\pi(n)}$.

Lemma 2. *Computing an optimal deployment strategy for a tree of size n with positive real edge and vertex weights takes $\Omega(n \log n)$ computational time in the algebraic decision tree model.*

3.2 Collected subtrees

The proof of Lemma 2 suggests to visit the edges of the tree in the order of decreasing weights. For generalization we introduce the following notations for a tree T with root vertex v_s.

For every leaf b_l along the unique shortest path, $\Pi_{v_s}^{b_l}$, from the root v_s to b_l there is an edge $e(b_l)$ with weight $w_{e(b_l)}$, so that $w_{e(b_l)}$ is greater than or equal to any other edge weight along $\Pi_{v_s}^{b_l}$. Furthermore, we choose $e(b_l)$ so that it has the shortest edge-distance to the root among all edges with the same weight. Let $v(b_l)$ denote the vertex of $e(b_l)$ that is closer to the leaf b_l. Thus, every leaf b_l defines a unique path, T_{b_l}, from $v(b_l)$ to the leaf b_l with incoming edge $e(b_l)$ with edge weight $w_{e(b_l)}$. The edge $e(b_l)$ *dominates* the leaf b_l and also the path $T(b_l)$.

For example in Figure 4 we have $e(b_2) = e_5$ and $v(b_2) = v_3$, the path $T(b_2)$ from v_3 over v_5 to b_2 is dominated by the edge e_5 of weight 10.

If some paths $T_{b_{l_1}}, T_{b_{l_2}}, \ldots, T_{b_{l_m}}$ are dominated by the same edge e, we collect all those paths in a *collected subtree* denoted by $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$. The tree has unique root $v(b_{l_1})$ and is dominated by unique edge $e(b_{l_1})$.

For example, in Figure 4 for b_6 and b_7 we have $v(b_6) = v(b_7) = v_4$ and $e(b_6) = e(b_7) = e_7$ and $T(b_6, b_7)$ is given by the tree T_{v_4} that is dominated by edge e_7.

Altogether, for any tree T there is a unique set of disjoint collected subtrees (a path is a subtree as well) as uniquely defined above and we can sort them by the weight of its dominating edge. For the tree in Figure 4 we have disjoint subtrees $T(b_6, b_7)$, $T(b_2, b_3, b_4)$, $T(b_1)$, $T(b_5)$ and $T(b_0)$ in this order.

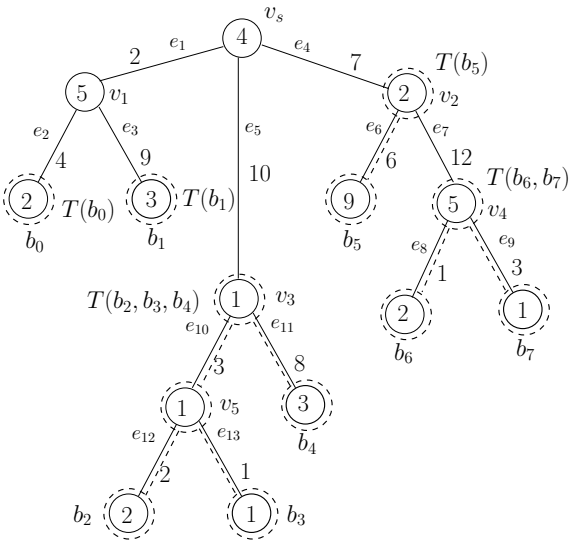


Figure 4: The optimal strategy with start and end vertex v_s visits, fully explores and leaves the collected subtrees $T(b_6, b_7)$, $T(b_2, b_3, b_4)$, $T(b_1)$, $T(b_5)$ and $T(b_0)$ in the order of the weights $w_{e_7} = 12$, $w_{e_5} = 10$, $w_{e_3} = 9$, $w_{e_4} = 7$ and $w_{e_2} = 4$ of the dominating edges.

3.3 Return variant for trees

We show that the collected subtrees can be visited in the order of the dominating edges.

Theorem 3. *An optimal deployment strategy that has to start and end at the same root vertex v_s of a tree T can visit the disjoint subtrees $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ in the decreasing order of the dominating edges.*

Any tree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ can be visited, fully explored in some order (for example by DFS) and left then.

An optimal visiting order of the leafs and the optimal number of agents required can be computed in $\Theta(n \log n)$ time for real edge and vertex weights and in optimal $\Theta(n)$ time for integer weights.

For the proof of the above Theorem we first show that we can reorganize any optimal strategy so that at first the tree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ with maximal dominating edge weight can be visited, fully explored and left, if the strategy does not end in this subtree (which is always true for the return variant). The number of agents required cannot increase. This argument can be applied successively. Therefore we formulate the statement in a more general fashion.

Lemma 3. *Let $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ be a subtree that is dominated by an edge e which has the greatest weight among all edges that dominate a subtree.*

Let S be an optimal deployment strategy that visits some vertex v_t last and let v_t be not a vertex inside the tree $T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})$. The strategy S can be reorganized so that first the tree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ can be visited, fully explored in any order and finally left then.

Proof: The tree $T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})$ rooted at $v(b_{l_1})$ and with maximal dominating edge weight $w_{e(b_{l_1})}$ does not contain another subtree $T(b_{k_1}, b_{k_2}, \ldots, T_{b_{k_n}})$. This means that $T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})$ is the full subtree $T_{v(b_{l_1})}$ of T rooted at $v(b_{l_1})$. Let $\text{Path}(v(b_{l_1})$ denote the number of agents that has to be settled along the unique path from v_s to the predecessor, $\text{pred}(v(b_{l_1}))$, of $v(b_{l_1})$.

Let us assume that an optimal strategy is given by a sequence S and let $S_v(i)$ denote the strategy that ends after the i-th visit of some vertex v in the sequence of S. Let $|S_v(i)|$ denote the number of settled agents and let $\text{curr}(S_v(i))$ denote the number of non-settled agents after the i-th visit of v. We would like to replace S by a sequence $S'S''$. If vertex $v(b_{l_1})$ is finally visited, say for the k-th time, in the sequence S, we require $\text{curr}(S_{v(b_{l_1})}(k)) \geq w_{e(b_{l_1})}$ and $|S_{v(b_{l_1})}(k)| \geq |T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})| + \text{Path}(v(b_{l_1})$ since the strategy ends at $v_t \notin T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})$. In the next step S will move back to $\text{pred}(v(b_{l_1}))$ along $e(b_{l_1})$ and in S the root $v(b_{l_1})$ of the tree $T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})$ and the edge $e(b_{l_1})$ will never be visited again.

If we consider a strategy S' that first visits $v(b_{l_1})$, fully explores $T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})$ by DFS and moves back to the start v_s by passing $e(b_{l_1})$, the minimal number of agents required for this movement is exactly $|T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})| + \text{Path}(v(b_{l_1}) + w_{e(b_{l_1})}$ with $w_{e(b_{l_1})}$ non-settled agents. With $\left(|S_{v(b_{l_1})}| - |T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})| - \text{Path}(v(b_{l_1})\right) + w_{e(b_{l_1})}$ agents we now start the whole sequence S again.

In the concatenation of S' and S, say $S'S$, the vertex $v(b_{l_1})$ is visited $k' = k + 2$ times for $|T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})| \geq 2$ and $k' = k+1$ times for $m = 1$ and $v(b_{l_1}) = T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})$.

After S' was executed for the remaining movement of $S'S_{v(b_{l_1})}(k')$ the portion $w_{e(b_{l_1})}$ of $\left(|S_{v(b_{l_1})}| - |T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})| - \text{Path}(v(b_{l_1})\right) + w_{e(b_{l_1})}$ allows us to cross all edges in $S'S_{v(b_{l_1})}(k')$ for *free*, because $w_{e(b_{l_1})}$ is the maximal weight in the tree.

Thus obviously $\text{curr}(S'S_{v(b_{l_1})}(k')) = \text{curr}(S_{v(b_{l_1})}(k))$ holds and $S'S$ and S require the same number of agents. In $S'S$ all visits of $T(b_{l_1}, b_{l_2}, \ldots, T_{b_{l_m}})$ made by S were useless because the tree was already completely filled by S'. Skipping all these visits in S, we obtain a sequence S'' and $S'S''$ has the desired structure. □

Proof (Theorem 3) The strategy of the single group has to return back to the start vertex v_s at the end. Therefore no

subtree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ contains the vertex v_s visited last. Let us assume that N_1 is the optimal number of agents required for T.

After the first application of Lemma 3 to the subtree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ with greatest incoming edge weight $w_{e(b_{l_1})}$ we can move with at least $w_{e(b_{l_1})}$ agents back to the root v_s without loss by the strategy S'. Let us assume that N'_1 agents return to the start.

We simple set all node weights along the path from v_s to $v(b_{l_1})$ to zero, cut off the fully explored subtree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ and obtain a tree T'. Note that the collected subtrees were disjoint and apart from $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ the remaining collected subtrees will be the same in T' and T. By induction on the number of the subtrees in the *remaining* problem T' we can visit the collected subtrees in the order of the dominating edge weights.

Note that the number of agents required for T' might be less than N'_1 because the weight $w_{e(b_{l_1})}$ was responsible for N'_1. This makes no difference in the argumentation.

We consider the running time. By a simple DFS walk of T, we compute the disjoint trees $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ implicitly by pointers to the root vertices $v(b_{l_1})$. For any vertex v, there is a pointer to its unique subtree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ and we compute the sum of the vertex weights for any subtree. This can be done in overall linear time. Finally, we can sort the trees by the order of the weights of the incoming edges in $O(n \log n)$ time for real weights and in $O(n)$ time for integer weights.

For computing the number of agents required, we make use of the following efficient procedure, similar to the algorithm indicated at the beginning of this Section. Any visited vertex will be marked. In the beginning let add $:= 0$ and curr $:= N$. Let $|T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})|$ denote the sum of the vertex weights of the corresponding tree. We successively *jump* to the vertices $v(b_{l_1})$ of the trees $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ by making use of the pointers. We mark $v(b_{l_1})$ and starting with the predecessor of $v(b_{l_1})$ we move backwards along the path from $v(b_{l_1})$ to the root v_s, until the first marked vertex is found. Unmarked vertices along this path are labeled as marked and the sum of the corresponding vertex weights is counted in a variable Path. Additionally, for any such vertex v that belongs to some other subtree $T(\ldots)$ we subtract the vertex weight w_v from $|T(\ldots)|$, this part of $T(\ldots)$ is already visited.

Now we set curr $:=$ curr $-$ $(|T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})| +$ Path$)$. If curr $< w_e$ holds, we set add $:=$ add $+ (w_e -$ curr$)$ and curr $:= w_e$ as before. Then we turn over to the next tree. Obviously with this procedure we compute the optimal number of agents in linear time, any vertex is marked only once. A pseudocode is presented in Algorithm 2. □

We present an example of the execution of Algorithm 2. For example in Figure 4 we have $N := 41$ and first jump to the root v_4 of $T(b_6, b_7)$, we have $|T(b_6, b_7)| = 8$. Then we count the 6 agents along the path from v_4 back to v_s and mark the vertices v_2 and v_s as visited. This gives curr $:= 41 - (8 + 6) = 27$, which is greater than $w_{e_7} = 12$. Additionally, for v_2 we subtract 2 from $|T(b_5)|$ which gives $|T(b_5)| = 9$. Now we jump to the root v_3 of $T(b_2, b_3, b_4)$ with $|T(b_2, b_3, b_4)| = 8$. Moving from v_3 back to v_s to the first unmarked vertex just gives no step. No agents are counted along this path. Therefore curr $:= 27-(8+0) = 19$ and curr $> w_{e_5} = 10$. Next we jump to the root b_1 of $T(b_1)$ of size $|T(b_1)| = 3$. Moving back to the root we count the weight 5 of the unvisited vertex v_1 (which will be marked now). Note that v_1 does not belong to a subtree $T(\ldots)$. We have curr $:= 19 - (3 + 5) = 11$. Now we jump to the root v_2 of $T(b_5)$ of current size $|T(b_5)| = 9$. Therefore curr $:= 11-(9+0) = 2$ which is now smaller than $w_{e_4} = 7$. This gives add $:=$ add $+ (w_e -$ curr$) = 0 + (7 - 2) = 5$ and curr $:= w_e = 7$. Finally we jump to $b_0 = T(b_0)$ and have curr $:= 7 - (2 - 0) = 5$ which is greater than w_{e_2}. Altogether 5 additional agent can move back to v_s and $N +$ add $= 46$ agents are required in total.

Algorithm 2: Return variant. Number of agents for $T = (V, E)$. Roots $v(b_{l_1})$ of trees $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ are given by pointers in a list L in the order of dominating edge weights. NIL is the predecessor of root v_s.

```
N := ∑_{v∈V} w_v; curr := N; add := 0;
while L ≠ ∅ do
    v(b_{l_1}) := first(L); deleteFirst(L);
    Mark v(b_{l_1});
    Path := 0; pathv := pred(v(b_{l_1}));
    while pathv not marked and pathv ≠ NIL do
        Path := Path + w_pathv;
        if pathv belongs to T(...) then
            |T(...)| := |T(...)| − w_pathv
        end if
        Mark pathv; pathv := pred(pathv);
    end while
    curr := curr − (|T(b_{l_1}, b_{l_2}, ..., b_{l_m})| + Path).
    if curr < w_e then
        add := add + (w_e − curr); curr := w_e;
    end if
end while
RETURN N + add
```

3.4 Lower bound for traversal steps

It is easy to see that although the number of agents required and the visiting order of the leafs can be computed subquadratic optimal time, the number of traversal steps for a tree could be in $\Omega(n^2)$; see the example in Figure 5. In this example the strategy with the minimal number of agents is unique and the agents have to run in a single group.

3.5 No-return variant

Finally, we discuss the more difficult task of the no return variant. In this case for an optimal solution not all collected

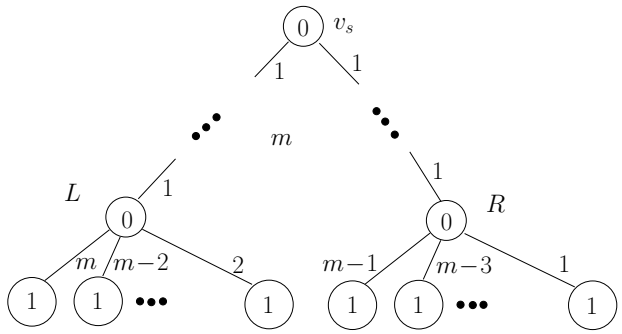


Figure 5: An optimal deployment strategy for the tree with $3m$ edges requires $m+1$ agents and successively moves from L to R beyond v_s in total $\Omega\left(\frac{m}{2}\right)$ times. Thus $\Omega(m^2)$ steps are required.

subtrees will be visited in the order of the decreasing dominating edge weights.

For example a strategy for the no-return in Figure 4 that visits the collected subtrees $T(b_6, b_7)$, $T(b_2, b_3, b_4)$, $T(b_1)$, $T(b_5)$ and $T(b_0)$ in the order of the weights $w_{e_7} = 12$, $w_{e_5} = 10$, $w_{e_3} = 9$, $w_{e_4} = 7$ and $w_{e_2} = 4$ of the dominating edges requires 46 agents even if we do not finally move back to the start vertex. As shown at the end of Section 3.3 we required 5 additional agents for leaving $T(b_5)$, entering and leaving $T(b_0)$ afterwards requires no more additional agents.

In the no return variant, we can assume that any strategy ends in a leaf, because the last vertex that will be served has to be a leaf. This also means that it is reasonable to enter a collected subtree, which will not be left any more. In the example above we simply change the order of the last two subtrees. If we enter the collected subtrees in the order $T(b_6, b_7)$, $T(b_2, b_3, b_4)$, $T(b_1)$, $T(b_0)$ and $T(b_5)$ and $T(b_5)$ is not left at the end, we end the strategy in b_5 (no-return) and require exactly $N = 41$ agents, which is optimal.

Theorem 4. *For a weighted tree T with given root v_s and non-fixed end vertex we can compute an optimal visiting order of the leafs and the number of agents required in amortized time $O(n \log n)$.*

For the proof of the above statement we first characterize the structure of an optimal strategy. Obviously we can assume that a strategy that need not return to the start will end in a leaf. Let us first assume that the final leaf, b_t, is already known or given. As indicated for the example above, the *final* collected subtree will break the order of the collected subtrees in an optimal solution. This behaviour holds recursively.

Lemma 4. *An optimal traversal strategy that has to visit the leaf b_t last can be computed as follows: Let $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ be the collected subtree of T that contains b_t.*

1. *First, all collected subtrees $T(b_{q_1}, b_{q_2}, \ldots, b_{q_o})$ of the tree T with dominating edge weight greater than $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ are successively visited and fully explored (each by DFS) and left in the decreasing order of the weights of the dominating edges.*
2. *Then, the remaining collected subtrees that do not contain b_t are visited in an arbitrary order (for example by DFS).*
3. *Finally, the collected subtree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ that contains b_t is visited. Here we recursively apply the same strategy to the subtree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$. That is, we build a list of collected subtrees for the tree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ and recursively visit the collected subtrees by steps 1. and 2. so that the collected subtree that contains b_t is recursively visited last in step 3. again.*

Proof: The precondition of the Theorem says that there is an optimal strategy given by a sequence S of visited vertices and edges so that the strategy ends in the leaf b_t. Let $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ be the collected subtree of T that contains b_t and let $w_{e(b_t)}$ be the corresponding dominating edge weight. So $b_t \in \{b_{l_1}, b_{l_2}, \ldots, b_{l_m}\}$ and $v(b_t)$ is the root of $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$. Similarily as in the proof of Lemma 3 we would like to reorganize S as required in the Lemma.

For the trees $T(b_{q_1}, b_{q_2}, \ldots, b_{q_o})$ with dominating edge weight greater than $w_{e(b_t)}$ we can successively apply Lemma 3. So we reorganize S is this way by a sequence S' that finally moves the agents back to the start vertex v_s. Then we apply the sequence S again but skip the visits of all collected subtrees already fully visited by S' before. This show step 1. of the Theorem.

This gives an overall sequence $S'S''$ with the same number of agents and S'' does only visit collected subtrees of T with dominating edges weight smaller than or equal to $w_{e(b_t)}$. Furthermore, S'' also ends in b_t.

The collected subtree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ with weight $w_{e(b_t)}$ does not contain any collected subtree with weight smaller than or equal to $w_{e(b_t)}$. At some point in S'' the vertex $v(b_t)$ is visited for the last time, say for the k-th time, by a movement from the predecessor $\text{pred}(v(b_t)$ of $v(b_t)$ by passing the edge of weight $w_{e(b_t)}$. At least $w_{e(b_t)}$ agents are still required for this step. At this moment all subtrees different from $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ and edge weight smaller than or equal to $w_{e(b_t)}$ habe been visited since the strategy ends in $b_t \in \{b_{l_1}, b_{l_2}, \ldots, b_{l_m}\}$.

Since $w_{e(b_t)}$ agents are required for the final movement along $e(b_t)$ there will be no loss of agents, if we postone all movements into $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ in S'' first and then finally solve the problem in $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ optimally. For the subtrees different from $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ and edge weight smaller than or equal to $w_{e(b_t)}$ we only require the agents that have to be placed there, since at least $w_{e(b_t)}$ non-settled agents will be always present. Therefore we can also decide to visit the subtrees different from $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ and edge weight smaller than or equal to $w_{e(b_t)}$ in an arbitrary order (for example by DFS). This gives step 2. of the Theorem.

Finally, we arrive at $v(b_t)$ and $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ and would like to end in the leaf b_t. By induction on the height of the trees the tree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ can be handled in the same way. That is, we build a list of collected subtrees for the tree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ itself and recursively visit the collected subtrees by steps 1. and 2. so that the collected subtree that contains b_t is recursively visited last in step 3. again. □

The remaining task is to efficiently find the best leaf b_t where the overall optimal strategy ends. The above Lemma states that we should be able to start the algorithm recursively at the root of a collected subtree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ that contains b_t. For v_s a list, L, of the collected subtrees for T is given and for finding an optimal strategy we have to compute the corresponding lists of collected subtrees for all trees $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ in L recursively.

Figure 6 shows an example. In this setting let us for example consider the case that we would like to compute an optimal visiting order so that the strategy has to end in the leaf b_2. Since b_2 is in $T(b_2, b_3, b_4)$ in the list of v_s in Figure 6 by the above Lemma in step 1. we first visit the tree $T(b_6, b_7)$ of dominating edge weight greater than the dominating edge weight of $T(b_2, b_3, b_4)$. Then we visit $T(b_1)$, $T(b_5)$ and $T(b_0)$ in step 2. After that in step 3. we recursively start the algorithm in $T(b_2, b_3, b_4)$. Here at v_3 the list of collected sutrees contains $T(b_4)$ and $T(b_2, b_3)$ and by the above recursive algorithm in step 1. we first visit $T(b_4)$. There is no tree for step 2. and we recursively enter $T(b_2, b_3)$ at v_5 in step 3. Here for step 1. there is no subtree and we enter the tree $T(b_3)$ in step 2. until finally we recursively end in $T(b_2)$ in step 3. Here the algorithm ends. Note that in this example b_2 is not the overall optimal final leaf.

If we simply apply the given algorithm for any leaf and compare the given results (number of agents required) we require $O(n^2 \log n)$ computational time. For efficiency, we compute the required information in a single step and check the value for the different leafs successively. It can be shown that in such a way the best leaf b_t and the overall optimal strategy can be computed in amortized $O(n \log n)$ time.

Finally, we give a proof for Theorem 4 by the following discussion. We would like to compute the lists of the collected subtrees $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ recursively. More precisely, for the root v_s of a full tree T with leafs $\{b_1, b_2, \ldots, b_n\}$ we obtain a list, denoted by $T(b_1, b_2, \ldots, b_n)$, of the collected subtrees of T with respect to the decreasing order of the dominating edge weights as introduced in Section 3.2.

The elements of the list are pointers to the roots of the collected subtrees $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$. For any such root $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ of a subtree in the list $T(b_1, b_2, \ldots, b_r)$ we recursively would like to compute the corresponding list of collected subtrees recursively; see Figure 6 for an example.

Additionally, for any considered collected subtree $T(b_{k_1}, b_{k_2}, \ldots, b_{k_r})$ that belongs to the pointer list of

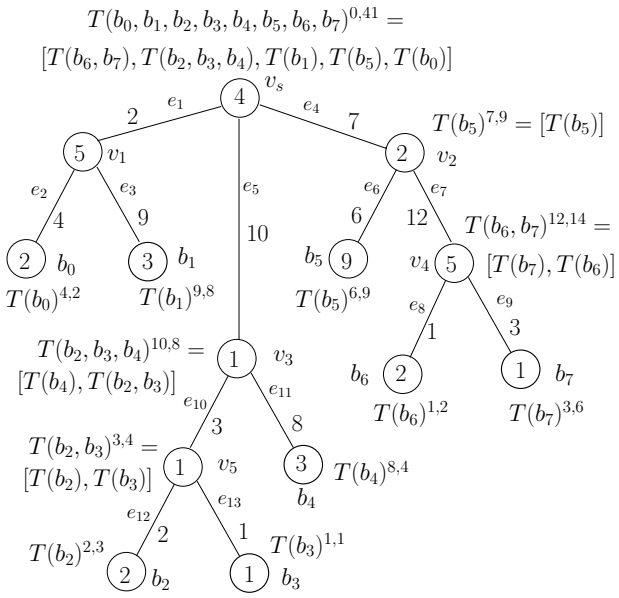


Figure 6: All information required can be computed recursively from bottom to top in amortized $O(n \log n)$ time.

$T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ we store a pair of integers x, y at the corresponding root of $T(b_{k_1}, b_{k_2}, \ldots, b_{k_r})$; see Figure 6. Here x denotes the weight of the dominating edge. The value y denotes the size of $|T(b_{k_1}, b_{k_2}, \ldots, b_{k_r})| +$ Path), if we recursively start the optimal tree algorithm in the root $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$; see Algorithm 2. This means that y denotes the size of the collected subtree *and* the sum of the weights along the path back from $T(b_{k_1}, b_{k_2}, \ldots, b_{k_r})$ to the root $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ if $T(b_{k_1}, b_{k_2}, \ldots, b_{k_r})$, if $T(b_{k_1}, b_{k_2}, \ldots, b_{k_r})$ is the first entry of the list $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ and therefore has maximal weight.

The list of subtrees at the root v_s of T is denoted by $T(b_1, b_2, \ldots, b_r)^{x,y}$ and obtains the values $x := 0$ (no incoming edge) and $y := N$ (the sum of the overall vertex weights). We can show that all information can be computed efficiently from bottom to top and finally also allows us to compute an overall optimal strategy.

For the overall construction of all pointer lists $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ we internally make use of Fibonacci heaps [7]. The corresponding heap for a vertex v always contains *all* collected subtrees of the leafs of T_v. The collected subtree list for the vertex v itself might be empty; see for example that vertex v_1 does not root a set of collected subtree. In the following the list of pointers to collected subtrees is denoted by $[\ldots]$ and the internal heaps are denoted by $(\ldots)$.

The subtrees in the heap are also given by pointers. But the heap is sorted by *increasing* dominating edge weights. Note, that we have two different structures. Occasionally a final subtree for a vertex with a list of pointers for the collected subtrees (in decreasing order) and the internal heaps with a collection of all collected subtrees (in increasing or-

der) have the same elements.

With the help of the heaps we successively compute and store the final collected subtree lists for the vertices. We start the computations on the leafs of the tree. For a single leaf b_l the heap $(T(b_l)^{x,y})$ and the subtree $T(b_l)^{x,y}$ represent exactly the same. The value x of $T(b_l)^{x,y}$ is given by the edge weight of the leaf. The value y of $T(b_l)^{x,y}$ will be computed recursively, it is initialized by the vertex weight of the leaf. For example, in Figure 6 for b_7 and b_6 we first have $T(b_6)^{1,2}$ and $T(b_7)^{3,1}$, representing both the heaps and the subtrees.

Let us assume that the heaps for the child nodes of an internal node v already have been computed and v is a branching vertex with incoming edge weight w_e. We have to add the node weight of v to the value y of one of the subtrees in the heap. We simply additionally store the subtree with greatest weight among the branches. Thus in constant tim we add the node weight of the branching vertex to the value y of a subtree with greatest weight. Then we unify the heaps of the children. They are given in the *increasing* order of the dominating edges weights. This can be done in time proportional to the number of child nodes of v. For example, at v_4 in Figure 6 first we increase the y-element of the subtree $T(b_7)^{3,1}$ in the heap by the vertex weight 5 of v_4 which gives $T(b_7)^{3,6}$. Then we unify $(T(b_6)^{1,2})$ and $(T(b_7)^{3,6})$ to a heap $(T(b_6)^{1,2}, T(b_7)^{3,6})$. For convenience in the heap we attach the values x and y directly to the pointer of the subtree.

Now, for branching vertex v by using the new unified heap we find, delete and collect the subtrees with minimal incoming edge weight as long as the weights are smaller than or equal to the weight w_e. If there is no such tree, we do not have to build a new collected subtree at this vertex and also the heap remains unchanged. If there are some subtrees that have incoming edge weight smaller than or equal to w_e the pointers to all these subtrees will build a new collected subtree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ with x-value w_e at the node v. Additionally, the pointers to the corresponding subtrees of $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ can easily be ordered with *increasing* weights since we have deleted them out of the heap starting with the smallest weights. Additionally, we sum up the values y of the deleted subtrees. Finally, we have computed the collected subtree $T(b_{l_1}, b_{l_2}, \ldots, b_{l_m})$ and its information x, y at node v. At the end a new subtree is also inserted into the fibonacci heap of the vertex v for future unions and computations.

For example in Figure 6 for the just computed heap $(T(b_6)^{1,2}, T(b_7)^{3,6})$ at vertex v_4 we delete and collect the subtrees $T(b_6)^{1,2}$ and $T(b_7)^{3,6}$ out of the heap because the weight $w_{e_7} = 12$ dominates both weights 1 and 3. This gives a new subtree $T(b_7, b_6)^{12,8} = [T(b_7), T(b_6)]$ at v_4 and also a heap $(T(b_7, b_6)^{12,8})$.

Note, that sometimes no new subtree is build if no tree is deleted out of the heap because the weight of the incoming edge is less than the current weights. Or it might happen that only a single tree of the heap is collected and gets a new dominating edge. In this case also no subtree is deleted out of the heap. We have a single subtree with the same leafs as before but with a different dominating edge. We do not build a a collected subtree for the vertex at this moment, the insertion of such subtrees at the corresponding vertex is postponed.

For example for the vertex v_2 with incoming edge weight 7 in Figure 6 we have already computed the heaps $(T(b_5)^{6,9})$ and $(T(b_6, b_7)^{12,8})$ of the subtrees. Now the vertex weight 2 of v_2 is added to the y-value of $T(b_6, b_7)^{12,8}$ which gives $T(b_6, b_7)^{12,10}$ for this subtree. Then we unify the heaps to $(T(b_5)^{6,9}, T(b_6, b_7)^{12,10})$. Now with respect to the incoming edges weight 7 only the first tree in the heap is collected to a subtree and this subtree gives the list for vertex v_2. The heap of the vertex v_2 now reads $(T(b_5)^{7,9}, T(b_7, b_6)^{12,10})$ and the collected subtree is $T(b_5)^{7,9} = [T(b_5)]$.

Finally, we arrive the root vertex v_s and all subtrees of the heap are inserted into the list of collected subtrees for the root.

The delete operation for the heaps requires amortized $O(\log m)$ time for a heap of size m and subsumes any other operation. Any delete operation leads to a collection of subtrees, therefore at most $O(n)$ delete operation will occur. Altogether all subtrees and its pointer lists and the values x and y can be computed in amortized $O(n \log n)$ time.

The remaining task is that we use the information of the subtrees for calculating the optimal visiting order of the leafs in overall $O(n \log n)$ time. Here Algorithm 2 will be used as a subroutine.

As already mentioned we only have to fix the leaf b_t visited last. We proceed as follows. An optimal strategy ends in a given collected subtree with some dominating edge weight w_e. The strategy visits and explores the remaining trees in the order of the dominating edges weights.

Let us assume that on the top level the collected subtrees are ordered by the weights $w_{e_1} \leq w_{e_2} \leq \ldots \leq w_{e_j}$. Therefore by the given information and with Algorithm 2 for any i we can successively compute the number of additional agents required for any successive order $w_{e_{i+1}} \leq w_{e_{i+2}} \leq \ldots \leq w_{e_j}$ and by the y-values we can also compute the number of agents required for the trees of the weights $w_{e_1} \leq w_{e_2} \leq \ldots \leq w_{e_{i-1}}$. The number of agents required for the final tree of weight w_{e_i} and the best final leaf stems from recursion. With this informations the number of agents can be computed. This can be done in overall linear time $O(j)$ for any i.

The overall number of collected subtrees in the construction is linear for the following reason. We start with n subtrees at the leafs. If this subtree appears again in some list (not in the heap), either it has been collected together with some others or it builds a subtree for its own (changing dominance of a single tree). If it was collected, it will never appear for its own again on the path to the root. If it is a single subtree of that node, no other subtree appears in the list at this node. Thus for the $O(n)$ nodes we have $O(n)$ collected subtrees in the lists total.

From i to $i+1$ only a constant number of additional calculations have to be made. By induction this can recursively be done for the subtree dominated by w_{e_i} as well. Therefore we can use the given information for computing the optimal strategy in overall linear time $O(n)$ if the collected subtrees are given recursively.

4 Conclusion

We introduce a novel traversal problem in weighted graphs that models security or occupation constraints and gives rise to many further extensions and modifications. The problem discussed here is NP-hard in general and can be solved efficiently for trees in $\Theta(n \log n)$ where some machinery is necessary. This also gives a 2-approximation for a general graph by the MST.

References

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, Englewood Cliffs, NJ, 1993.

[2] Th. Beckmann, R. Klein, D. Kriesel, and E. Langetepe. Ant-sweep: a decentral strategy for cooperative cleaning in expanding domains. In *Symposium on Computational Geometry*, pages 287–288, 2011.

[3] Bernd Brüggemann, Elmar Langetepe, Andreas Lenerz, and Dirk Schulz. From a multi-robot global plan to single-robot actions. In *ICINCO (2)*, pages 419–422, 2012.

[4] V. Chvátal. *Linear Programming*. W. H. Freeman, New York, NY, 1983.

[5] M. Dynia, J. Łopuszański, and Ch. Schindelhauer. Why robots need maps. In *SIROCCO '07: Proc. 14th Colloq. on Structural Information an Communication Complexity*, LNCS, pages 37–46. Springer, 2007.

[6] P. Fraigniaud, L. Gasieniec, D. R. Kowalski, and A. Pelc. Collective tree exploration. *Networks*, 43(3):166–177, 2006.

[7] M. L. Fredman and R. E. Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *J. ACM*, 34:596–615, 1987.

[8] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, New York, NY, 1979.

[9] Miltos D. Grammatikakis, D. Frank Hsu, Miro Kraetzl, and Jop F. Sibeyn. Packet routing in fixed-connection networks: A survey. *Journal of Parallel and Distributed Computing*, 54(2):77 – 132, 1998.

[10] Bernhard Korte and Jens Vygen. *Combinatorial Optimization: Theory and Algorithms*. Springer Publishing Company, Incorporated, 4th edition, 2007.

[11] Michail G. Lagoudakis, Evangelos Markakis, David Kempe, Pinar Keskinocak, Anton Kleywegt, Sven Koenig, Craig Tovey, Adam Meyerson, and Sonal Jain. Auction-based multi-robot routing. In *Proceedings of Robotics: Science and Systems*, Cambridge, USA, June 2005.

[12] Simone Lehmann. Graphtraversierungen mit Nebenbedingungen. Masterthesis, Rheinische Friedrich-Wilhelms-Universität Bonn, 2012.

[13] Britta Peis, Martin Skutella, and Andreas Wiese. Packet routing: Complexity and algorithms. In *WAOA 2009*, number 5893 in LNCS, pages 217–228. Springer-Verlag, 2009.

[14] David Portugal and Rui P. Rocha. A survey on multi-robot patrolling algorithms. In Luis M. Camarinha-Matos, editor, *DoCEIS*, volume 349 of *IFIP Advances in Information and Communication Technology*, pages 139–146. Springer, 2011.

[15] F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, New York, NY, 1985.

[16] Alexander V. Sadovsky, Damek Davis, and Douglas R. Isaacson. Optimal routing and control of multiple agents moving in a transportation network and subject to an arrival schedule and separation constraints. In *No. NASA/TM–2012–216032*, 2010.

[17] I. A. Wagner, Y. Altshuler, V. Yanovski, and A. M. Bruckstein. Cooperative cleaners: A study in ant robotics. *The Int. J. Robot. Research*, 27:127–151, 2008.

Relaxations in Practical Clustering and Blockmodeling

Stefan Wiesberg and Gerhard Reinelt
Institut für Informatik, Universität Heidelberg
Im Neuenheimer Feld 368, 69120 Heidelberg, Germany
E-mail: stefan.wiesberg/gerhard.reinelt@informatik.uni-heidelberg.de

Keywords: network analysis, clustering, blockmodeling

Received: June 21, 2014

Network analysts try to explain the structure of complex networks by the partitioning of their nodes into groups. These groups are either required to be dense (clustering) or to contain vertices of equivalent positions (blockmodeling). However, there is a variety of definitions and quality measures to achieve the groupings. In surveys, only few mathematical connections between the various definitions are mentioned. In this paper, we show that most of the definitions used in practice can be seen as certain relaxations of four basic graph theoretical definitions. The theory holds for both clustering and blockmodeling. It can be used as the basis of a methodological analysis of different practical approaches.

Povzetek: Pri razdeljevanju omrežij na podskupine so pristopi opredeljeni kot eni od štirih teoretičnih skupin.

1 Introduction

The structure of large networks is usually not comprehensible to the human beholder. Especially, if the network has not been designed by a human architect, but rather evolved over time in a complex (natural) process. Examples for such networks are social (friendship, mailing, scientific collaboration, advice giving), economic (trading between countries or companies), chemical (protein-protein reactions), biological (food chain), or internet link networks. Nevertheless, researchers in these fields use the networks to gain insight into their structural makeup. To this end, a first step is most often the reduction of the network's complexity with the help of algorithms. A common approach is to reduce the high number of nodes in the network. The idea of *blockmodeling approaches* is to group the nodes such that the number of groups is much lower than the number of nodes. The grouping is done in a way that leads to patterns in the network's links. We distinguish two kinds of patterns: Patterns of link *density* (Section 1.1) and of link *existence* (Section 1.2). An example for patterns of link density is given in Figure 1: On the left-hand side, we see a random drawing of a graph $G = (V, E)$. In the center, we see a partition of V into four groups A, B, C, D, indicated by four different colors, such that a density pattern becomes apparent. Densely connected are the group pairs AB, BD, DD, CD, CA, sparsely connected are AA, BB, CC, AD, BC. Note that we use a merely intuitive definition of density here for motivational reasons; strict mathematical definitions will be introduced subsequently.

Before we explain the patterns of link density, we formalize a vertex grouping of a graph G with vertex set V and edge set E as a vertex coloring. This is possible since every vertex coloring $\phi : V \to [c]$, where $[c] = \{1, 2, \ldots, c\}$, naturally defines a partition of V into the color classes. W.l.o.g. we assume that ϕ is surjective, i.e., all c colors are used. In this paper, we assume that our network is given as an undirected graph $G = (V, E)$. More general cases, in which there are weights (on the arcs or on group pairs) or multiple arc types are not treated here.

1.1 Patterns of link density

The goal of the grouping which is discussed in this section is to group the vertices in a way such that for each pair of groups, there are either very *many* or very *few* links between the groups. In other words, we search for a *pattern of link density* in the network.

Once such a pattern has been found, the network's complexity has been reduced in the following sense: One can now shrink the groups to single vertices, and connect two such vertices by an edge if the corresponding groups were densely connected prior to the shrinking. The shrunk graph for the example in Figure 1 is depicted on the right-hand side of the figure.

Let us formalize the *pattern* notion. Given a coloring ϕ, the pattern specifies for each pair of color groups whether they are interpreted to be densely or sparsely connected. A pattern is usually notated as a binary square matrix I. Its dimension is the number of groups. An entry I_{AB} is 1 if groups A and B are interpreted to be densely connected, and 0 if they are interpreted to be sparsely connected. The matrix I representing the pattern is usually called *image matrix*. It is symmetric as the network graph is undirected. The graph whose adjacency matrix is the image matrix is called *image graph*. Figure 1 (right) shows the image graph to the density pattern described in the caption text of the

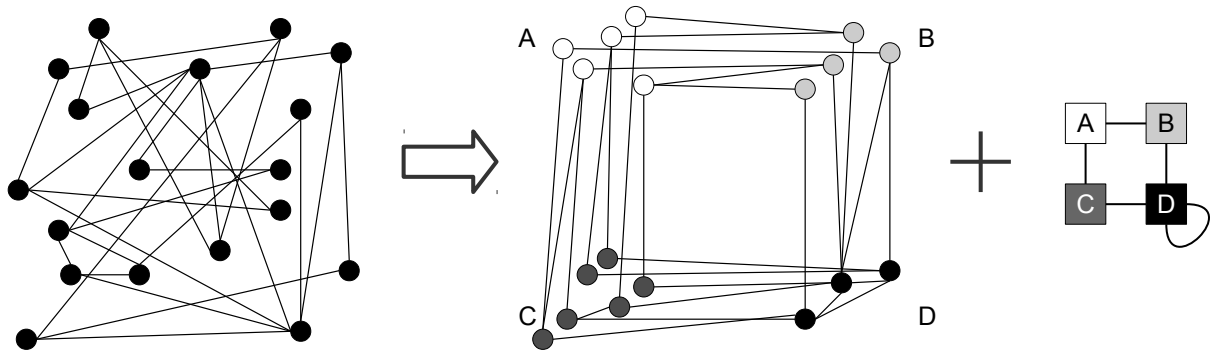


Figure 1: An exemplary density pattern.

figure. Note that the image graph can be seen as a simplification of the network structure: There is an edge in the image graph wherever there are many edges in the original network, and no edge wherever there are only few edges.

For a given network graph G, one is hence interested in a coloring ϕ of the vertices together with a density pattern. Such a pair (ϕ, I) of a coloring ϕ and its interpretation, an image matrix I of appropriate dimension, is called a *blockmodel*. The process of computing a good blockmodel for a given network is sometimes called *blockmodeling*.

1.2 Patterns of link existence

Density patterns imply that for the vertices in a group A, it holds that either all of them have very many or all of them have very few links to the vertices in a group B. In a pattern of link *existence*, however, one demands that either many vertices in group A have *at least one* link into group B or almost no vertex in group A has a link to group B. Analogously to the density pattern case, we can define an image matrix. It encodes for each pair of groups which of the two cases are interpreted to exist in the given coloring. The image graph then visualizes a pattern of connectivity. If an edge exists between groups A and B, then almost every vertex in A is connected to B, and vice versa. Otherwise, the groups are almost disconnected.

1.3 Fixing patterns

To find a suitable number of groups is generally part of the blockmodeling process. In practical blockmodeling, however, it is sometimes set a priori to a small fixed value. Moreover, the whole pattern is sometimes fixed a priori. The blockmodeling then reduces to the search for a coloring which matches the given pattern best. This is useful to test whether an assumed pattern actually exists in the network. A prominent example of pattern fixing is the *clustering problem*. Here, one searches for density patterns. The number c of groups is fixed to a small value and the image matrix is fixed to the $c \times c$ identity matrix. The blockmodeling hence consists of the search for a coloring with c colors such that the color groups themselves are dense, whereas their interconnections are sparse. In case that c is not fixed, the family of all identity matrices is considered as the set of feasible patterns.

1.4 Outline of the paper

Literature shows a large variety in practical blockmodeling approaches. Not only are they distinct in the way they use a priori fixings, they also differ in the ways they measure the quality of a given blockmodel for a given network. Usually, the search for clusters, link density and link existence patterns are treated separately. There are separate methods and publications for each of the three problem types.

In this paper, we present a new classification of the approaches. This classification holds for all (non-stochastic) clustering and blockmodeling approaches which quantify the quality of blockmodels and are reported in the following survey books: *Social Network Analysis* by Wasserman and Faust [16], *Network Analysis* by Brandes and Erlebach [6] (except *conductance*), and *Community Detection in Graphs* by Fortunato [10].

The search for an ideal blockmodel can usually be formulated as a graph coloring problem. By our classification, we show that the practical approaches can be seen as methods to optimize very specific relaxations of these problems. They are the same in clustering, link density and link existence patterns search.

Section 2 presents the graph coloring problems, which are relaxed in practical approaches. Section 3 explains the three types of relaxations that are used. Each type is illustrated with practical examples from the survey books. Finally, Section 4 summarizes and gives an outlook on applications of the classification.

2 Ideal blockmodels

In this section, we define *ideal* blockmodels of link density and existence. In an *ideal* blockmodel (ϕ, I) for link density, *all* links exist in the dense parts and *no* links exist in the sparse ones. In an ideal blockmodel for link existence, either *all* or *no* vertex in A have a neighbor in B.

Ideal *colorings* can be similarly defined. The reason is that in an ideal blockmodel (ϕ, I), the image matrix I can be directly constructed from ϕ: The entry I_{AB} is 0 if and only if there is no edge from A to B. We hence call a coloring ϕ *ideal* if the blockmodel (ϕ, I) is ideal, where I is constructed as explained.

There are three graph theoretical definitions of ideal colorings. They will be presented in the next three subsections.

2.1 The subgraph definition

In ideal blockmodels for density patterns, certain subgraphs are either complete or empty. These subgraphs can be defined as follows. Given a coloring ϕ, there is one such subgraph $G_{\phi,A,B}$ for every pair (A, B) of colors. It is obtained from G by deleting all vertices but the ones colored with A or B and deleting all edges but those connecting an A-colored with a B-colored vertex. $G_{\phi,A,B}$ is hence bipartite for $A \neq B$. Note that all of these subgraphs are edge disjoint. A similar observation can be made for ideal link existence blockmodels: That all vertices in A have at least one neighbor in B, and vice versa, is equivalent to the statement that $G_{\phi,A,B}$ contains no isolated vertices.

We have seen that *clustering* is a special case of link density, where the image matrix is a priori fixed. However, there is a common variant of clustering. It only requires the color groups to be dense, but does *not* require their interconnections to be sparse. In other words, only the diagonal image matrix entries are given. We include this variant into our classification scheme as it is widely used. It corresponds to Part (i) of the following definition of ideal colorings. Part (ii) defines ideal link density and Part (iii) ideal link existence colorings. See Figure 2 for examples.

Definition 1. *Given a graph G, a c-coloring $\phi : V \to [c]$ of its vertex set is*

(i) an ideal clique c-coloring*, if for all $A \in [c]$, the graph $G_{\phi,A,A}$ is complete.*

(ii) an ideal structural c-coloring*, if for all color pairs $A, B \in [c]$, the graph $G_{\phi,A,B}$ is either empty or a complete (complete bipartite for $A \neq B$) graph.*

(iii) an ideal regular c-coloring*, if for all color pairs $A, B \in [c]$, the graph $G_{\phi,A,B}$ is either empty or contains no isolated vertices.*

2.2 The node pair definition

We have seen that ideal colorings can be defined by subgraph characterizations. Alternatively, they can be defined by properties of same-colored vertices. In a clique c-coloring, every two vertices with the same color are connected by an edge. In a structural c-coloring, two vertices with the same color have exactly the same neighboring vertices in G. In a regular c-coloring, two vertices with the same color have exactly the same colors in their neighborhoods. Let $N(u)$ denote the set of vertices that are adjacent to vertex u. The following definition is hence equivalent to the subgraph definition above. See Lorrain and White [12] for details.

Definition 2. *Given a graph G, a c-coloring $\phi : V \to [c]$ of its vertex set is an*

(i) ideal clique c-coloring*, if for all $u, v \in V$ with $\phi(u) = \phi(v)$: $uv \in E$.*

(ii) ideal structural c-coloring*, if for all $u, v \in V$ with $\phi(u) = \phi(v)$: $N(u) \setminus \{v\} = N(v) \setminus \{u\}$.*

(iii) ideal regular c-coloring*, if for all $u, v \in V$ with $\phi(u) = \phi(v)$: $\{\phi(w) \mid w \in V, uw \in E\} = \{\phi(w) \mid w \in V, vw \in E\}$.*

2.3 The single node definition

A definition from the perspective of *single* vertex is only possible with respect to a fixed image matrix I. In this case, the following single node definition is equivalent to the two definitions above.

Definition 3. *Given a graph G and a $c \times c$ image matrix I, a c-coloring $\phi : V \to [c]$ of G's vertex set is*

(i) an ideal clique c-coloring*, if for all $u \in V$: u is adjacent to all $v \in V$ with $\phi(v) = \phi(u)$.*

(ii) an ideal structural c-coloring w. r. t. I*, if for all $u \in V$ and all $C \in [c]$: u is adjacent to all $v \in V$ with $\phi(v) = C$ if $I_{\phi(u)C} = 1$, and to no $v \in V$ with $\phi(v) = C$ if $I_{\phi(u)C} = 0$.*

(iii) an ideal regular c-coloring w. r. t. I*, if for all $u \in V$ and all $C \in [c]$: u is adjacent to at least one $v \in V$ with $\phi(v) = C$ if $I_{\phi(u)C} = 1$, and to no $v \in V$ with $\phi(v) = C$ if $I_{\phi(u)C} = 0$.*

3 Relaxations

For a given graph G, one can theoretically compute a coloring from Definition 1 or 2 to obtain an ideal coloring (and thus ideal blockmodel). However, this is usually not done in practice. In Section 3.1, we list some common reasons for this decision. In Section 3.2, we show that the approaches used in practice can be interpreted as the solution of an optimization problem on a relaxed problem definition.

3.1 Reasons for relaxations

There are several reasons for the use of relaxations instead of directly searching for ideal blockmodels. We list four of them.

1. *Non-existence of solutions.* An ideal coloring might only exist if a large number of colors is used.

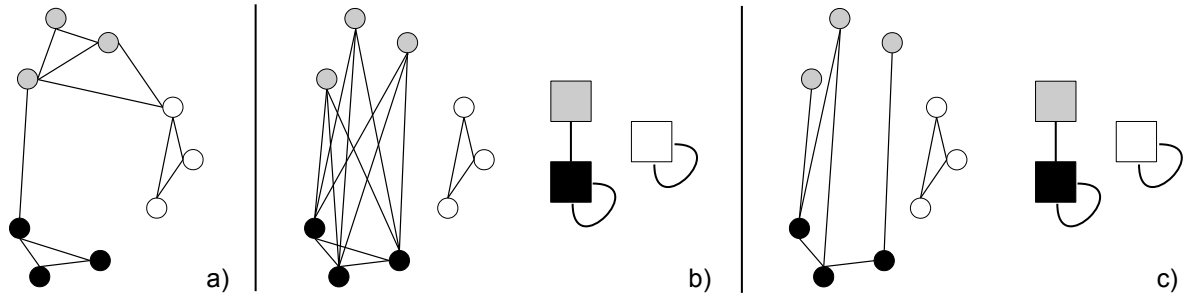


Figure 2: Ideal clique (a), structural (b), and regular (c) 3-colorings. In b) and c), the corresponding image graph is depicted.

2. *Real-world modeling reasons.* The definition might be too restrictive for the application at hand. For example, the graph theoretical definition of *clique* might be too strict to describe friendship cliques in social networks, where some edges can be missing.

3. *Involvement of statistics.* The relaxations allow to define statistically profound criteria for the quality of colorings, instead of the purely graph theoretical ones.

4. *Robustness against measuring errors.* The extraction of graphs from complex networks can be erroneous, especially in biological or chemical networks. However, a regular coloring on a graph can turn non-regular by the deletion or addition of one single edge. Relaxations are hence useful to limit the influence of these errors on the colorings.

3.2 General relaxation

In this section, we show how ideal blockmodels are relaxed in practice. Denote by $CC(c, G)$ the set of all clique c-colorings of the vertices of G. Analogously, we define $SC(c, G)$ and $RC(c, G)$ for structural and regular c-colorings. As a shorthand, we simply write $X(G)$ in a statement which holds for any fixed type (CC, SC, RC) and any fixed number c of colors. Practitioners, often implicitly, enlarge the set $X(G)$ of feasible colorings to a set $X_L(G) \supseteq X(G)$ and assign a penalty value $p(\phi) \geq 0$ to each member ϕ of the enlarged set $X_L(G)$. Afterwards, they solve the optimization problem of finding a coloring ϕ^* in $X_L(G)$ with the minimum penalty value $p(\phi^*)$. We now show that this is usually done in the following way: The set $X(G)$ of feasible colorings is enlarged by dropping some of the requirements in the definition of X. Furthermore, the penalty function p is not arbitrary, but measures the degree of violation against the *dropped* requirements. The optimization problem to be solved is thus:

(MIN-P) Given the set $X_L(G)$ and the penalty function $p : X_L(G) \to \mathbb{R}_0^+$, find a $\phi^* \in X_L(G)$ which minimizes p.

That is, among the colorings satisfying the non-dropped requirements, find the one which violates the dropped requirements to the least possible extent. As a convention, a penalty value of 0 is assigned to the colorings in $X(G)$, as they do not violate any dropped requirements (*compatibility requirement*, see Doreian et. al. [9]). Hence, a coloring satisfying the original definition X is always an optimum solution to (MIN-P).

We will now classify literature by the type of relaxation used. As we are considering the relaxation of ideal colorings, three types of relaxations come to mind: The relaxation of the coloring definition, of the node pair ideality definition and the subgraph ideality definition. Indeed, these possibilities are widely used. In Section 3.3, we will look at the cases where the general definition of coloring is relaxed. Sections 3.4 and 3.5 treat the ideality definition relaxations respectively.

3.3 Coloring relaxations

In Definition 1 and 2 for ideal colorings, the definition of "coloring" itself can be relaxed. If we use the binary variables x_{vA} to express whether vertex v is colored with A ($x_{vA} = 1$) or not ($x_{vA} = 0$), the requirement "to be a c-coloring" can be decomposed into the following sub requirements:

$$\sum_{A=1}^{c} x_{vA} = 1 \quad \text{for all } v \in V, \tag{1}$$

$$\sum_{v \in V} x_{vA} \geq 1 \quad \text{for all } A \in [c], \tag{2}$$

$$0 \leq x_{vA} \leq 1 \quad \text{for all } v \in V, A \in [c], \tag{3}$$

$$x_{vA} \in \mathbb{Z} \quad \text{for all } v \in V, A \in [c]. \tag{4}$$

Example (Fuzzy Colorings.) In some applications, it is meaningful for a vertex to get several colors at the same time. E.g., a person might be a member of several clubs. In this case, requirement (1) is dropped. Alternatively, a vertex might be allowed to consist of color fractions that sum up to 1, such as 50% red, 30% green and 20% blue. In this case, requirement (4) is dropped. One speaks of *fuzzy colorings* or *partitions* in both of these cases of relaxation. Usually, there is no penalty for a vertex to have more than one color at the same time. That is, the penalty function p is usually constant with respect to the coloring requirements.

Example (Number of Colors.) For many applications, a good choice for the number of colors is not a priori known

and hence not fixed to a certain value c. That is, the requirement that c colors must be used is dropped. As small numbers are usually more suitable for interpretation, the penalty function p might be defined to assign each coloring ϕ the number of colors used by ϕ. The lower the number of colors, the less the amount of penalty. As an example, the algorithm CATREGE [4] solves (MIN-P) for such a p and X=RC. I.e., given a graph, it finds a regular c-coloring with the smallest possible c.

3.4 Single node and node pair relaxations

In single node relaxations, the properties for a single vertex to contribute to an ideal coloring are relaxed. As we have seen in Definition 3, single node definitions are only possible if the image matrix I is fixed. An example are the nodal degree relaxations for clusterings, i.e., for Part (i).

Example (Nodal Degree Relaxations.) Seidman and Foster [15] relax the requirement that every vertex must be adjacent to all other vertices of the same color by the requirement that every vertex can be non-adjacent to at most k other vertices of the same color. In an ideal coloring, the resulting subgraphs are hence not cliques, but so-called k-*plexes*. Usually, the relaxation is not penalized. That is, p is constant, say $p \equiv 0$. The search for an ideal blockmodel is hence simply the search for a partition of the vertices into k-plexes. Instead of k-plexes, the similar k-cores are sometimes used.

We now turn to the more common node *pair* relaxations. Here, the properties for same-colored vertex pairs in Definition 2 are relaxed. Two forms of p are most commonly used, which will be explained by the following two examples: p is either constant or decomposable over the set of all vertex pairs.

Example (Sociometric Cliques.) Alba [1] finds the graph theoretical definition of *clique* to be not perfectly appropriate to describe friendship (or sociometric) cliques in social networks. He thus relaxes its definition to so-called n-cliques. Here, two same-colored vertices do not need to be connected by an edge. They need to be connected by a path of length at most n, which relaxes the edge connection requirement. If no penalties are introduced, the problem (MIN-P) merely consists in the search for *any* partition into n-cliques. Similar to the n-clique are the n-clan and n-club relaxations [13].

We now treat a second common type of node pair relaxation: The *vertex similarity approaches*. The idea is to consider for each vertex pair separately, whether it should be same-colored or not. In this special case of (MIN-P), the penalty function p can thus be decomposed over all vertex pairs, i.e., $p(\phi) = \sum_{u,v \in V} p_{uv}\delta(\phi(u), \phi(v))$. Here, $p_{uv} \geq 0$ are real numbers and δ denotes the Kronecker function. It is 1 if $\phi(u) = \phi(v)$ and 0 otherwise. In literature, the numbers p_{uv} are often called *(dis)similarity values*. The relaxation technique of using such a decomposable function is called *indirect blockmodeling approach* by Doreian et al. [9].

Example (Structural Equivalence.) For X=SC, several functions p of the above form have been proposed. This propositions were made indirectly by a specification of the values p_{uv}. They quantify how much a coloring violates this dropped requirement, that is, to quantify how similar two vertices are with respect to common neighbors. See Leicht, Holme, and Newman [11] for an overview on these functions.

3.5 Subgraph relaxations

In subgraph relaxations, the requirements of Definition 1 for ideal blockmodels are relaxed.

Assume a practitioner is interested in regular 4-colorings on a given graph $G = (V, E)$. However, such a coloring does not exist on G. It is then reasonable to consider a 4-coloring ϕ to be a good solution, if it is not regular on G, but turns regular if G is changed by a very small amount. Following this idea, the best 4-coloring is the one that requires the lowest amount of changes in G in order to become regular. Possible changes are usually the deletion and addition of edges. That is, requirements of the forms "$uv \in E$" and "$uv \notin E$" are dropped. If they are penalized by the function p, then the coloring ϕ^* which requires the lowest amount of edge changes in G will be the optimal solution to (MIN-P).

In order to define a suitable penalty function p, we first need to define a function d to measure the amount of edge changes. More precisely, d measures the distance of two graphs $G = (V, E)$ and $H = (V, F)$ on the same vertex set V. A simple but common exemplary form of such a d is given by

$$d(G, H) = \sum_{u,v \in V, u \neq v} |A(G)_{u,v} - A(H)_{u,v}|, \qquad (5)$$

where A denotes the adjacency matrix of the graph. The function counts the number of different entries in the adjacency matrices of G and H. More complex distance functions are discussed below. The function d measures the distance of G to a single graph H. We can also measure its distance to a set of graphs $\mathbf{H}$, by defining the distance $d(G, \mathbf{H})$ as the distance of G to its closest element in $\mathbf{H}$. That is,

$$d(G, \mathbf{H}) := \min_{H \in \mathbf{H}} d(G, H).$$

To measure how much G has to be changed, it is compared to sets of ideal graphs $\mathcal{H}(\phi)$, on which ϕ perfectly satisfies the requirements. In our example, $\mathcal{H}(\phi)$ is defined such that ϕ is 4-regular on all $H \in \mathcal{H}(\phi)$. The penalty function for (MIN-P) is hence $p(\phi) = d(G, \mathcal{H}(\phi))$.
We now give more details on this procedure. First, we will see how ideal graphs $\mathcal{H}(\phi)$ can be defined. Then, we give an overview on the distance functions $d(G, H)$ which are used in practice. Afterwards, a common variant of this procedure is discussed, which does not relax G, but several subgraphs of G simultaneously. We close by some examples on how graph relaxation is used in literature.

Ideal, Worst and Average Graphs

Given an ideal coloring definition X (for example CC, SC, RC), a graph $G = (V, E)$ and a coloring ϕ of its vertices, the set $\mathcal{H}(\phi)$ of ideal graphs can be naturally defined. It is the set of all graphs H with the same vertex set as G, such that ϕ is an X-coloring on H. Definition 1 gives a characterization of these graphs. In the case of clustering, i. e., X = CC, the ideal graphs are those in which vertices of the same color induce complete graphs. Note that for every $\phi : V \to [c]$, the set $\mathcal{H}(\phi)$ is non-empty.

Alternatively, one can define $\mathcal{H}(\phi)$ to be the set of *worst* graphs instead of ideal ones. Worst graphs can be easily defined for CC and SC. This is because their subgraph characterization in Definition 1 use empty and complete graphs only. As "being empty" and "being complete" are opposite extremes, one can define worst graphs by interchanging the words "empty" and "complete" in the definition. E. g., in a worst graph for clustering (CC) no cluster contains any edges. If worst graphs are used, the distance of the closest graph to G needs to be maximized instead of minimized.

A third alternative has been used for CC and SC: G is compared to average graphs. For clustering, the subgraphs are hence neither empty nor complete, but have an average density. The distance of G to the average graphs $\mathcal{H}(\phi)$ can then be positive or negative, depending on whether G is worse (sparser) or better (denser) than average. The same holds hence for the penalty function. It is usually used as a reward function $\overline{p}$: The farther G is from average in the positive direction, the larger $\overline{p}$ is, and the better ϕ is.

Overview on Distance Functions

We already stated the most simple distance function to measure the distance between two graphs on the same vertex set:

$$d(G, H) = \sum_{u,v \in V, u \neq v} |A(G)_{u,v} - A(H)_{u,v}|,$$

It counts the number of edges to be added or deleted (changed) in G to obtain the ideal graph H. See Figure 3 for an example for structural 3-colorings (X = SC). The distance $d(G, \mathcal{H}(\phi))$ of the depicted coloring ϕ of the drawn graph G is 3. The reason is that 3 changes are at least necessary to obtain a structural 3-coloring: Add two edges from gray to black and delete one edge within white. Hence, the penalty value for this coloring is $p(\phi) = 3$.

If G is compared to *average* graphs, the absolute value function is a problem. Here, we want to distinguish whether G is worse or better than average. Hence, the following function is more suitable in this case.

$$d(G, H) = \sum_{u,v \in V, u \neq v} (A(G)_{u,v} - A(H)_{u,v}). \quad (6)$$

The adjacency matrix of H is possibly weighted, as average graphs usually do not have binary edge weights.

There is a third function for the case that *vertices* are relaxed instead of edges. More precisely, if requirements

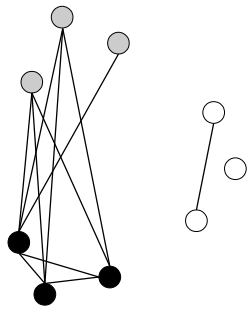


Figure 3: Example for the distance function (5) when applied to a structural 3-coloring problem.

of the form "$v \in V$" are relaxed. Note that the opposite requirement "$v \notin V$" is never relaxed, as the addition of vertices cannot contribute to the transformation of G into an ideal graph. For every coloring ϕ of the vertices in $G = (V, E)$, G is compared to a set of ideal graphs $\mathcal{H}(\phi)$. Every such graph $H = (V_H, E_H)$ in $\mathcal{H}(\phi)$ has a vertex subset $V_H \subseteq V$ and the edge set $E_H = E(V_H)$. That is, H can be obtained from G by deleting vertices together with their incident edges. A distance function needs to measure the amount of vertices to be deleted to transform G into H.

$$d(G, H) = |V(G)| - |V(H)|. \quad (7)$$

Beside these linear functions, several non-polynomial functions have been proposed. Being derived from general statistical matrix correlation measures, they can be used to compare the adjacency matrices of G and H. See Wasserman and Faust [16] or Arabie et al. [2] for an overview.

Combining Subgraph Penalties

In Definition 1, the ideal coloring conditions are formulated as requirements for the subgraphs $G_{\phi,A,B}$ of G. In the widely used *direct blockmodeling approach*, these subgraphs are relaxed separately. That is, there is a separate penalty value for each subgraph. However, the same distance function d is used for each subgraph. Whether the separate relaxations of the subgraphs is equivalent to the relaxation of G itself depends on the choice of d. In direct blockmodeling, we have single penalty values $p_{AB}(\phi) = d(G_{\phi,A,B}, \mathbf{H}_{\phi,A,B})$ for the subgraphs. They need to be combined to a total penalty value $p(\phi)$. In most cases, the p_{AB} are simply summed up:

$$p(\phi) = \sum_{A,B \in [c]} p_{AB}(\phi). \quad (8)$$

For clustering (X=CC), the sum runs clearly only over those (A, B) with $A = B$. If scaling is used, the factor is usually $1/m_{AB}$, where m_{AB} is the number of possible edges in the subgraph $G_{\phi,A,B}$. More precisely, $m_{AB} = |A| \cdot |B|$ if $A \neq B$, $m_{AA} = |A| \cdot (|A| - 1)$, and

$$p(\phi) = \sum_{A,B \in [c]} \frac{1}{m_{AB}} \cdot p_{AB}(\phi). \quad (9)$$

In some approaches, the squares of the penalties are summed up instead. This mostly occurs in so-called χ^2 *approaches*.

$$p(\phi) = \sum_{A,B \in [c]} (p_{AB}(\phi))^2. \tag{10}$$

Besides the above scaling factor, a second one can be used here. The distance of $G_{\phi,A,B}$ to $H_{\phi,A,B}$ can be seen in relation to the maximum distance $d^{\max}_{\phi,A,B}$ of any graph, on the same vertex set, to $H_{\phi,A,B}$.

$$p(\phi) = \sum_{A,B \in [c]} m_{AB} \cdot \left(\frac{p_{AB}(\phi)}{d^{\max}_{\phi,A,B}} \right)^2. \tag{11}$$

Examples

We now give some examples on how this kind of relaxation is used in literature, either for coloring type CC, SC, or RC. For each example, we need to specify the following modeling choices:

- Whether ideal, worst, or average graphs are used (and how average is defined).
- Whether edges or vertices are relaxed.
- How $p(\phi)$ is combined from the $p_{AB}(\phi)$.

Example (Cluster Performance). The *performance* of a clustering counts the number of missing edges within the clusters and adds the number of existing edges between the clusters. It is hence a measure for the clustering special case of $X = SC$. According to our classification, ideal graphs are used, edges are relaxed, and $p(\phi)$ is simply the sum of the $p_{AB}(\phi)$.

Example (Maximal Cluster Density.) A basic measure for the quality of a clustering (X=CC) on $G = (V, E)$ is the sum over all *intra-cluster densities* $\delta_{int}(V_i)$. They give the proportion of actual edges to theoretically possible edges within the i-th cluster:

$$\delta_{int}(V_i) = \frac{\text{\# internal edges of } V_i}{|V_i|(|V_i|-1)/2}.$$

The search for a coloring ϕ^* with maximum total intra-cluster density is a (MIN-P) problem. Ideal graphs are used, edges are relaxed, and the penalty values $p_{AB}(\phi)$ are linearly combined by Formula (9).

Example (Maximal Structural Density.) Wasserman and Faust explain a simple measure for structural colorings in their survey [16]. It is a generalization of the preceding example from clique to structural colorings. For each pair A, B of colors, they sum up the values $|I_{AB} - \Delta_{AB}|$. Here, I denotes the image matrix and Δ_{AB} denotes the density. The density is defined as the number of edges from A-colored to B-colored vertices, divided by the maximum possible number m_{AB} of such edges. Hence, ideal graphs are used, edges are relaxed, and the penalties $p_{AB}(\phi)$ are linearly combined by formula (9).

Example (Newman-Girvan-Modularity.) Newman and Girvan [14] present a well-known relaxation for clustering. They choose $H(\phi)$ to contain average graphs. More precisely, $H(\phi)$ consists of exactly one graph $H = (V, F)$. The edge weight of $uv \in F$ is $deg(u)deg(v)/2|E|$. This is precisely the probability of the edge to exist in a random graph with the same degree distribution as G. For this reason, H can be interpreted as the average graph w.r.t. to the degree distribution of G. Hence, average graphs are used, edges are relaxed, and the penalties $p_{AB}(\phi)$ are simply summed up (Formula (8)).

Note that $p(\phi)/2|E|$ is called the *modularity* of ϕ. The factor $1/2|E|$ is however constant and can thus be ignored in the solution of (MIN-P). Other so-called *Newman-like modularities* can be modeled analogously.

Example (Berkowitz-Carrington-Heil Index.) The index [8] is designed for structural colorings (X=SC). It compares G to an average graph H. The user is asked to specify an average density α from the interval between 0 and 1. H is then the complete graph with edge weights all α, letting its density equal α. The distance function d is (5), hence the most simple one. It is applied on subgraphs. Since the index is a χ^2 approach, the function $p(\phi)$ is composed as in (11).

Example (Vertex Relaxation.) Batagelj et. al. [3] relax vertices for regular colorings (X=RE). They use ideal graphs, relax vertices, and simply sum up the penalties $p_{AB}(\phi)$. However, they restrict the natural set $\mathcal{H}(\phi)$ of ideal graphs by allowing only those $H \in \mathcal{H}(\phi)$ for which it holds that whenever there is an edge $uv \in E$ and u is not in V_H, then v cannot be in V_H either. An optimization heuristic for this function is implemented in UCINET [5]. Brusco and Steinley [7] present an exact optimization algorithm based on an integer programming model.

4 Summary and conclusions

We present a classification for clustering and blockmodeling approaches used in practice. We show that these approaches are based on relaxations of graph theoretical coloring definitions. Basically, there are only three types of relaxations. The classification unifies link density pattern (including clustering) and link existence pattern approaches and shows the connections between them.

An obvious drawback of such a theory about used approaches is clearly its invalidity as soon as new kinds of approaches are invented. Furthermore, it does not yet cover approaches which penalize blockmodels in which the colors groups do not have similar sizes. An example is the *conductance approach* for clusterings. On the one hand, the function p minimizes the number of edges for $I_{AB} = 0$, which is a classical subgraph relaxation approach. On the other hand, p also minimizes size differences between the vertex groups. To classify this approach, the requirement for same group sizes needs to be added to the ideality definitions, such that a deviation can be penalized. We did not

include it as most approaches deal with this requirement indirectly: They exclude blockmodels with largely differing group sizes from the set $X_L(G)$ of feasible blockmodels.

However, we also see two kinds of practical benefits. First, the classification can be used to think about the "missing" approaches. For example, approaches which use average graphs usually compare G to a single average graph H, whose edge weights are fractional. This choice seems to be arbitrary, as one could also use a whole set $\mathcal{H}(\phi)$ of unweighted average graphs for the comparison to G. The latter idea is standard if ideal instead of average graphs are used. Second, the question which approach is the most suitable one for a given network can now be answered stepwise: Are ideal or average graphs more suitable, should edges or vertices be relaxed, should node pairs or subgraphs be relaxed, how should subgraph penalties be combined, etc.?

References

[1] R. D. Alba. A graph-theoretic definition of a sociometric clique. *Journal of Mathematical Sociology*, 3(1):113–126, 1973.

[2] P. Arabie, S. A. Boorman, and P. R. Levitt. Constructing blockmodels: How and why. *Journal of Mathematical Psychology*, 17(1):21–63, 1978.

[3] V. Batagelj, P. Doreian, and A. Ferligoj. An optimizational approach to regular equivalence. *Social Networks*, 14(1):121–135, 1992.

[4] S. P. Borgatti and M. G. Everett. Two algorithms for computing regular equivalence. *Social Networks*, 15(4):361–376, 1993.

[5] S. P. Borgatti, M. G. Everett, and L. C. Freeman. Ucinet for windows: Software for social network analysis. 2002.

[6] U. Brandes and J. Lerner. Structural similarity: spectral methods for relaxed blockmodeling. *Journal of Classification*, 27(3):279–306, 2010.

[7] M. J. Brusco and D. Steinley. Integer programs for one-and two-mode blockmodeling based on prespecified image matrices for structural and regular equivalence. *Journal of Mathematical Psychology*, 53(6):577–585, 2009.

[8] P. J. Carrington, G. H. Heil, and S. D. Berkowitz. A goodness-of-fit index for blockmodels. *Social Networks*, 2(3):219–234, 1980.

[9] P. Doreian, V. Batagelj, and A. Ferligoj. *Generalized blockmodeling*, volume 25. Cambridge University Press, 2005.

[10] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75–174, 2009.

[11] E. Leicht, P. Holme, and M. Newman. Vertex similarity in networks. *Physical Review E*, 73(2):026120, 2006.

[12] F. Lorrain and H. C. White. Structural equivalence of individuals in social networks. *The Journal of Mathematical Sociology*, 1(1):49–80, 1971.

[13] R. J. Mokken. Cliques, clubs and clans. *Quality and Quantity*, 13:161–173, 1979.

[14] M. E. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113, 2004.

[15] S. Seidman and B. Foster. A graph-theoretic generalization of the clique concept. *Journal of Mathematical Sociology*, 6:139–154, 1978.

[16] S. Wasserman and K. Faust. *Social network analysis: Methods and applications*, volume 8. Cambridge university press, 1994.

Inter Programming Models for the Target Visitation Problem

Achim Hildenbrandt and Gerhard Reinelt
Institut für Informatik, Universität Heidelberg, Im Neuenheimer Feld 368
D-69120 Heidelberg, Germany

Keywords: target visitation problem, programming models

Received: July 25, 2014

The target visitation problem (TVP) is concerned with finding a route to visit a set of targets starting from and returning to some base. In addition to the distance traveled a tour is evaluated by taking also preferences into account which address the sequence in which the targets are visited. The problem thus is a combination of two well-known combinatorial optimization problems: the traveling salesman and the linear ordering problem. In this paper we point out some polyhedral properties and develop a branch-and-cut algorithm for solving the TVP to optimality. Some computational results are presented.

Povzetek: Prispevek obravnava iskanje poti v grafu, kjer je potrebno obiskati več ciljev v najboljšem vrstnem redu.

1 Introduction

Let $D_{n+1} = (V_{n+1}, A_{n+1})$ be the complete digraph on $n+1$ nodes where we set $V_{n+1} = \{0, 1, \ldots, n\}$. Furthermore let two types of arc weights be defined: weights d_{ij} (*distances*) for every arc (i, j), $0 \leq i, j \leq n$, and weights p_{ij} (*preferences*) associated with every arc (i, j), $1 \leq i, j \leq n$. The *target visitation problem (TVP)* consists of finding a Hamiltonian tour starting at node 0 visiting all remaining nodes (called *targets*) exactly once in some order and returning to node 0. Every tour can be represented by a permutation π of $\{1, 2, \ldots, n\}$ where $\pi(i) = j$ if target j is visited as i-th target. For convenience we also define $\pi(0) = 0$ and $\pi(n+1) = 0$.

So we are essentially looking for a traveling salesman tour, but for the TVP the profit of a tour depends on the two weights. Namely, the value of a tour is the sum of pairwise preferences between the targets corresponding to their visiting sequence minus the sum of distances traveled, i.e., it is calculated as

$$\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} p_{\pi(i)\pi(j)} - \sum_{i=0}^{n} d_{\pi(i)\pi(i+1)},$$

and the task is to find a tour of maximum value. So we have a multicriteria objective function.

The TVP was introduced in [4] and combines two classical combinatorial optimization problems: the *asymmetric traveling salesman problem (ATSP)* asking for a shortest Hamiltonian tour and the *linear ordering problem (LOP)* which is to find an acyclic tournament of maximum weight. (There is an obvious 1–1 correspondence between acyclic tournaments and linear orders of the node). Computational results of a genetic algorithm for problem instances with up to 16 targets have been reported in [1]. The original application of the TVP is the planning of routes for UAVs (unarmed aerial vehicles). But there is a wide field of applications, e.g. the delivery of relief supplies or any other routing problem where additional preferences should be considered (town cleaning, snow-plowing service, etc.).

Obviously, the TVP is NP-hard because it contains the traveling salesman problem ($p \equiv 0$) and the linear ordering problem ($d \equiv 0$) as special cases.

In this paper we present first polyhedral results for the TVP and develop an algorithm for solving it to optimality. In section 2 we introduce an integer programming model. Section 3 discusses some structural properties of the associated polytope. A branch-and-cut algorithm based on these results is described in section 4. The algorithm is then applied to a set of benchmark problems and the computational results are presented in section 5. A few remarks conclude the paper.

2 An integer programming model for the TVP

For convenience we first transform the problem to a Hamiltonian path problem and also get rid of the special base node. This transformation is well-known for the ATSP [7] and can be adapted for the TVP as follows.

The key idea is to exploit the fact that each tour has to start at the base and return to it and that no preferences are to be taken into account for the base. In the *TVP-path model* we leave out this node and just search for a Hamiltonian path which visits all targets exactly once.

Following [7] we make the following modifications.

(i) Transform the distance matrix by setting $d'_{ij} = d_{ij} - d_{i0} - d_{0j}$, for all pairs i and j of nodes, $1 \leq i, j \leq n$ $i \neq j$.

(ii) Change the computation of the distance part of the objective function to

$$\sum_{i=1}^{n-1} d'_{\pi(i)\pi(i+1)} - \sum_{i=1}^{n} d_{i0} - \sum_{i=1}^{n} d_{0i}.$$

The preferences are not affected by this change. From now on we consider the TVP as finding a Hamiltonian path in the complete digraph $D_n = (V_n, A_n)$ with additional preference costs to be taken into account. The path is described by a permutation π of $\{1, \ldots, n\}$ where $\pi(k)$ is the node at position k.

We introduce two types of variables. The sequence in which the targets are visited is represented by binary *ATSP variables* x_{ij} for $1 \leq i, j \leq n,\ i \neq j$, with the interpretation

$$x_{ij} := \begin{cases} 1 & \text{if } i = \pi(k) \text{ and } j = \pi(k+1) \\ & \text{for some } 1 \leq k \leq n-1, \\ 0 & \text{otherwise.} \end{cases}$$

The fact that some target i is visited before some target j is modeled with binary *LOP variables* w_{ij} for $1 \leq i, j \leq n,\ i \neq j$, with the definition

$$w_{ij} := \begin{cases} 1 & \text{if } i = \pi(k) \text{ and } j = \pi(l) \\ & \text{for some } 1 \leq k < l \leq n, \\ 0 & \text{otherwise.} \end{cases}$$

An obvious idea for obtaining an IP model of the TVP is to combine well-known IP formulations for the ATSP and the LOP. This combination gives the following integer programming model.

$$\max \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} p_{ij} w_{ij} - \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} d_{ij} x_{ij} \tag{1}$$

$$\sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} x_{ij} = n - 1, \tag{2}$$

$$\sum_{i \in S} \sum_{\substack{j \in S \\ j \neq i}} x_{ij} \leq |S| - 1,$$

$$S \subseteq \{1, \ldots, n\}, 2 \leq |S| \leq n-1, \tag{3}$$

$$\sum_{\substack{i=1 \\ i \neq j}}^{n} x_{ij} \leq 1, \qquad 1 \leq j \leq n, \tag{4}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{n} x_{ij} \leq 1, \qquad 1 \leq i \leq n, \tag{5}$$

$$w_{ij} + w_{jk} + w_{ki} \leq 2, \qquad 1 \leq i, j, k \leq n,\ i \neq j \neq k, \tag{6}$$

$$w_{ji} + w_{ij} = 1, \qquad 1 \leq i, j \leq n,\ i \neq j \tag{7}$$

$$x_{ij} - w_{ij} \leq 0, \qquad 1 \leq i, j \leq n,\ i \neq j \tag{8}$$

$$x_{ij} \in \{0, 1\}, \quad 1 \leq i, j \leq n,\ i \neq j \tag{9}$$

$$w_{ij} \in \{0, 1\}, \quad 1 \leq i, j \leq n,\ i \neq j \tag{10}$$

Constraints (2)–(5) model the directed Hamiltonian paths where inequalities (3) are the *subtour elimination constraints*. Acyclic tournaments are modelled by the *3-dicycle inequalities* (6) and the *tournament equations* (7). Inequalities (8) connect the solutions of both problems. Together with the integrality conditions (9) and (10) this obviously constitutes a 0/1 model of the TVP.

At first want to prove the correctness of the model.

Lemma 1. *The model presented in (1) - (10) is a correct IP model for the TVP).*

Proof. At first we have to prove that every feasible solution fulfills the model. Since (2) - (5), (9) is a well known model for the ATSP and (6) - (7), (10) is a well known model for the LOP it is sufficient to show that the values of x_{ij} do match with the values w_{ij} or in equal that both types of variables describe the same TVP-path. To assure this it is sufficient to prove the following two facts:

a) $x_{ij} = 1 \Rightarrow w_{ij} = 1$

b) $w_{ij} = 1 \Rightarrow i$ must be visited before j in the path describe by the x-variables

Because (8) must be fulfilled a) is obvious. To prove b) we assume j is visited before i in the path. That means the are existing indizees $k_1, \ldots, k_l$ so that $j, k_1, \ldots, k_n, i$ is a part of the path. So it follows that $x_{j,k_1} = x_{k_1,k_2} = \cdots = x_{k_l,i} = 1$. With a) we get that $w_{j,k_1} = w_{k_1,k_2} = \cdots = w_{k_l,i} = 1$. Because of (6) and (7) we can than iteratively conclude that $w_{j,k_2} = 1,\ w_{j,k_3} = 1, \ldots, w_{j,i} = 1$. But this is a contradiction to our assumption.

It remains to show that every feasible solution of (1) - (10) is a correct TVP-path. It is clear that every feasible integer solution must induce a feasible linear ordering and a feasible TSP tour. Because of the facts we mentioned above it is clear that the two feasible solutions must match which each other.

□

As an interesting fact we note that the subtour elimination constraints are actually not needed. If (w, x) satisfies (2) and (4)–(10), but not all inequalities (3) then there is some subtour on $k \geq 2$ nodes. W.l.o.g. we can assume that the node set is $\{1, 2, \ldots, k\}$ and the subtour is given as $\{(1,2), (2,3), \ldots, (k-1, k), (k, 1)\}$. Hence $x_{12} = x_{23} = ... = x_{k-1,k} = x_{k1} = 1$, implying because of (8) that $w_{12} = w_{23} = ... = w_{k-1,k} = w_{k1} = 1$. This is a contradiction to the requirement that the w-variables represent an acyclic tournament.

So we can eliminate the exponentially many constraints (3) and obtain a TVP formulation with a polynomial number (cubic in n) of constraints.

For our algorithm it will be useful to calculate the position of a node i in the path. This can easily be done using

the LOP variables. The value $n - \sum_{\substack{j=1 \\ j \not= i}}^{n} w_{ij}$ gives the position of node i.

Note that because of the tournament equations we can substitute an LOP variable $w_{ij}, j > i$, by $1 - w_{ji}$. Now the 3-dicycle inequalities are turned into $1 \geq w_{ij} + w_{jk} - w_{ik} \geq 0$ for all $1 \leq i < j < k \leq n$ and the part of the objective function for the LOP variables reads $\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} [(p_{ij} - p_{ji})w_{ij} + p_{ji}]$.

2.1 Extended formulation of the basic model

The use of extended formulations is a common technique with is used to strengthen the LP Formulation of a combinatorial optimization problem. The key idea of this approach is to add new variables and constraints to a given IP formulation so that the gap between the solution of the LP relaxation and the optimal integral solution becomes much smaller.

In the case of TVP we can obtain an extended formulation for the TVP by adding three-indexed variables, which are a generalization of the linear ordering variables to the standard model. In detail this new variables w_{ijk} are than defined as follows:

$$w_{ijk} := \begin{cases} 1 & \text{if } i = \pi(a), j = \pi(b) \text{ and } k = \pi(c) \\ & \text{for some } 1 \leq a < b < c \leq n-1\ , \\ 0 & \text{otherwise.} \end{cases}$$

So as one see this new type of variables is a straight forward extension of the w_{ij}-variables. In the objective function we assign zero coefficients to the new variables. In order to extend our standard model we also need to introduce two new classes of constraints to make sure that the solution of the new variables match with the old $x_i j$ and $w_i j$ variables. In detail the extended formulation looks a follows:

$$\max \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \not= i}}^{n} p_{ij} w_{ij} - \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \not= i}}^{n} d_{ij} x_{ij} \tag{11}$$

$$s.t.$$

$$\sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \not= i}}^{n} x_{ij} = n - 1, \tag{12}$$

$$\sum_{i=1}^{n} x_{ij} \leq 1,\ 1 \leq j \leq n, \tag{13}$$

$$\sum_{j=1}^{n} x_{ij} \leq 1,\ 1 \leq i \leq n, \tag{14}$$

$$\sum_{i \in S} \sum_{\substack{j \in S \\ j \not= i}} x_{ij} \leq |S| - 1,$$

$$S \subseteq \{1, \ldots, n\}, 2 \leq |S| \leq n-1, \tag{15}$$

$$w_{ij} + w_{jk} + w_{ki} \leq 2,$$

$$1 \leq i, j, k \leq n,\ i < j, i < k j \neq k, \tag{16}$$

$$w_{ij} + w_{jik} + w_{jki} + w_{kji} = 1$$

$$1 \leq i, j, k \leq n, i < j \tag{17}$$

$$x_{ij} - w_{ijk} - w_{kij} \leq 0\ 1 \leq i, j, k \leq n, i < j \tag{18}$$

$$x_{ij} - w_{ij} \leq 0,\ 1 \leq i, j \leq n, \tag{19}$$

$$x_{ij} \in \{0, 1\},\ 1 \leq i, j \leq n, \tag{20}$$

$$w_{ij} \in \{0, 1\},\ 1 \leq i, j \leq n. \tag{21}$$

$$w_{ijk} \in \{0, 1\},\ 1 \leq i, j, k \leq n. \tag{22}$$

3 The edge-node-formulation

The key idea of the next model is to combine the w and x variables of the -Model to new three index variables which states the relation between a node n and an fixed edge (i, j). More precisely we define:

$$w_{ij}^{k} := \begin{cases} 1 & \text{if } k = \pi(a), i = \pi(b) \text{ and } j = \pi(b+1) \\ & \text{for some } 1 \leq a < b \leq n-1\ , \\ 0 & \text{otherwise.} \end{cases}$$

and analogously :

$$w_{k}^{ij} := \begin{cases} 1 & \text{if } i = \pi(a), j = \pi(a+1) \text{ and } k = \pi(b) \\ & \text{for some } 1 \leq a < b \leq\ , \\ 0 & \text{otherwise.} \end{cases}$$

A first IP-Model can then be develop out of the basic model by transforming the inequalities/equations to inequalities/equations with the new variables:

$$max \sum_{i=1}^{n} \sum_{j=1, i \not= j}^{n} p_{ij} (\sum_{m=1, m \not= j}^{n} w_{mj}^{i}) + \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} (w_{ij}^{\Omega} + w_{\Omega}^{ij}) \tag{23}$$

$$s.t.$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} (w_{ij}^{\Omega} + w_{\Omega}^{ij}) = n - 1 \tag{24}$$

$$\sum_{i=1}^{n} (w_{ij}^{\Omega} + w_{\Omega}^{ij}) \leq 1\ \ j \in V \tag{25}$$

$$\sum_{j=1}^{n} (w_{ij}^{\Omega} + w_{\Omega}^{ij}) \leq 1\ \ i \in V \tag{26}$$

$$\sum_{l=1}^{n} w_{lj}^{i} + \sum_{l=1}^{n} w_{lk}^{j} + \sum_{l=1}^{n} w_{li}^{k} + (w_{ik}^{\Omega} + w_{\Omega}^{ik}) \leq 2\ \ i, j, k \in V \tag{27}$$

(28)

Please note that that $\Omega \in V$ and it could be chosen arbitrarily for each summand in (23) -(26) but $\neq j$ and $\neq i$. It is the same in (27) but here Ω must not be equal i or k

4 Three distance model

Another idea for constructing an IP-model for the TVP has been made by Prof. E. Fernandez from the UPC Barcelona. The key idea of this approach is the use of distance variables. In Detail we define Variables z_{ij}^t which describe the fact whether there is as path of length t between i an j or not. More formally we state:

$$z_{ij}^t = \begin{cases} 1 & \text{if } i \text{ the solution contains a path with } t \text{ arcs} \\ & \text{from } i \text{ to } j, \\ 0 & \text{otherwise.} \end{cases}$$

The advantage of this model is that we only have to deal here with one type of variables. Since we are not longer distinguish between distance and ordering variables we have to adjust the coefficients in the following way:

$$w_{ij}^t = \begin{cases} c_{ij} - d_{ij} & \text{if } t = 1, \\ c_{ij} & \text{otherwise.} \end{cases}$$

With are now able to formulated a TVP model with distance variables:

The formulation is:

$$\max \sum_{i \in N} \sum_{j \in N \setminus \{i\}} \sum_{t \in N \setminus \{n\}} \overline{w}_{ij}^t z_{ij}^t - \sum_{i \in N} (d_{i0} + d_{0i}) \quad (29)$$

$$\sum_{i=1}^{n} z_{ij}^1 \leq 1 \qquad j \in N, \quad (30)$$

$$\sum_{j=1}^{n} z_{ij}^1 \leq 1 \qquad i \in N, \quad (31)$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} z_{ij}^k = n - k \qquad k \in V \quad (32)$$

$$z_{ij}^{t_1} + z_{jk}^{t_2} \leq z_{ik}^{t_1+t_2} + 1,$$
$$i, j, k \in N, t_1, t_2 \in N \setminus \{n\},$$
$$i \neq j, j \neq k, i \neq k, t_1 + t_2 < n, \quad (33)$$

$$\sum_{t=1}^{n-1} z_{ij}^t + z_{ji}^t = 1 \qquad i, j \in N, i \neq j, \quad (34)$$

$$z_{ij}^t \in \{0, 1\} \qquad i, j \in N, i \neq j, t \in N \setminus \{n\}. \quad (35)$$

Also we only have one Type of variable now the $z_{i,j}^1$ variables still play a special role, for example in the objective function. On the other hand we again have a cubic number of variables

5 Conclusions

The target visitation problem turned out to be a very difficult and therefore challenging problem. The present paper gives some first results. More research is needed. An improvement of the simple heuristic used here can be accomplished along well-known lines. It is more interesting to find ways for improving the upper bound. The IP model already seems to be at its limits for fairly small problem instances unless some additional insight into the polyhedral structure can be obtained. Alternate optimization approaches like branch-and-bound with combinatorial bounds, dynamic or semidefinite programming should be devised and their limits should be explored. Furthermore it should be investigated further how the balance between the distance and the preference part of the object function influences the difficulty of problem instances.

References

[1] A. ARULSELVAN, C.W. COMMANDER and P.M. PARDALOS: *A hybrid genetic algorithm for the target visitation problem*, Technical report, University of Florida.

[2] T. CHRISTOF and A. Loebel: *PORTA: POlyhedron Representation Transformation Algorithm, Version 1.4*, http://comopt.ifi.uni-heidelberg.de/software/PORTA/index.html.

[3] M. GRÖTSCHEL, M. JÜNGER and G. REINELT: *Facets of the linear ordering polytope*, Math. Programming 33 (1985), 43–60.

[4] D.A. GRUNDEL and D.E. JEFFCOAT: Formulation and solution of the target visitation problem, In: Proceedings of the AIAA 1st Intelligent Systems Technical Conference, 2004.

[5] A. HILDENBRAND: *Benchmark instances for the TVP*, http://comopt.ifi.uni-heidelberg.de/people/hildenbrandt/TVP/template.html.

[6] M. JÜNGER and S. THIENEL: *The ABACUS System for branch-and-cut-and-price algorithms in integer programming and combinatorial optimization*, Software: Practice and Experience, 30 (2000), 1325–1352.

[7] M. QUEYRANNE and Y. WANG: *Hamiltonian path and symmetric travelling salesman polytopes*, Math. Programming 58 (1993), 89–110.

[8] G. REINELT: *TSPLIB – Benchmark instances for the TSP*, http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/.

Cervix Cancer Spatial Modelling for Brachytherapy Applicator Analysis

Peter Rogelj
University of Primorska, Faculty of Mathematics, Natural Sciences and Information Technologies
Glagoljaška 8, 6000 Koper, Slovenia
E-mail: peter.rogelj@upr.si

Muhamed Baraković
University of Verona, Department of Biotechnology
Ca' Vignal 1, Strada Le Grazie 15, 37134 Verona, Italia
E-mail: muhamed.barakovic@studenti.univr.it

Keywords: cervix cancer, spatial distribution, PCA, BT applicators

Received: June 18, 2014

Standard applicators for cervix cancer brachytherapy (BT) do not always enable a sufficient radiation dose coverage of the target structure (HR-CTV). The aim of this study was to develop methodology for building models of the BT target from a cohort of cervix cancer patients, which would enable BT applicator testing. In this paper we propose two model types, a spatial distribution model and a principal component model. Each of them can be built from data of several patients that includes medical images of arbitrary resolution and modality supplemented with delineations of HR-CTV structure, reconstructed applicator structure and eventual organs at risk (OAR) structures. The spatial distribution model is a static model providing probability distribution of the target in the applicator coordinate system, and as such provides information of the target region that applicators must be able to cover. The principal component model provides information of the target spatial variability described by only a few parameters. It can be used to predict specific extreme situations in the scope of sufficient applicator radiation dose coverage in the target structure as well as radiation dose avoidance in OARs. The results are generated 3D images that can be imported into existent BT planning systems for further BT applicator analysis and eventual improvements.

Povzetek: Razvita sta dva modela za analizo aplikatorjev v brahiradioterapiji.

1 Introduction

Applicators for cervix cancer brachytherapy (BT) enable cancer treatment that in comparison with external beam radiotherapy (EBRT) provides better radiation coverage of the high risk clinical target volume (HR-CTV) and better avoidance of organs at risk [1]. During the last decade remarkable progress has been made in radiotherapy, including cervix cancer BT [2]. Standard BT applicators for cervix cancer, as shown in Fig. 1, however still do not always enable a sufficient radiation dose coverage of the target, especially in cases of locally advanced cervical cancer. Improvements are searched in the direction of incorporating additional application needles. A development of new applicators that would enable better target dose coverage requires knowledge of cervix cancer spatial distribution and variation. Furthermore, as the applicators should be able to avoid organs at risk, the information of their variability should also be taken into account. In this work we aimed to develop methodology to obtain this information statistically using available data of past and present cervix cancer patients. The information required from each patient includes BT planning medical image, delineated HR-CTV structure, reconstructed BT applicator structure, and organs at risk (OAR) structures. HR-CTV and OAR structures are in each 3D image delineated on each image slice, wherever the specific structure is present and, thus, available as a set of closed planar contours. BT applicators are reconstructed such that an applicator model is positioned in the 3D image inside the BT planning system. The applicator models consist of a ring, applicator tandem and needles, which are reconstructed independently. The actual position of the applicator is evident from the position of the applicator ring. Because the applicator must be positioned directly to the cervix and because the purpose of the models is applicator analysis, the spatial distribution and variation must be defined in the applicator coordinate system.

The significance of tumor distribution depend on the tumor type. It can help in development of tumor treatment and biopsy strategies and techniques [3, 4]. In the case of the cervical cancer it is also important due to BT applicator design.

Representation of 3D structures by sets of closed planar contours is not convenient for further spatial analysis. Other representations can be used instead, e.g., by tensors [5], Gaussian random spheres [6], signed distance maps [7] and others. However, due to eventual high complexity of BT target structures, we have selected the most common

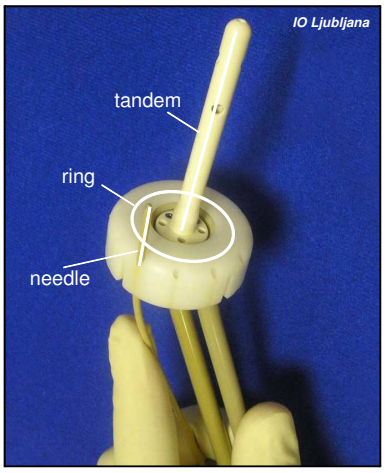


Figure 1: An image of a standard BT cervix cancer applicator with indicated parts: tandem, ring and optional needles. Image courtesy of Institute of Oncology Ljubljana.

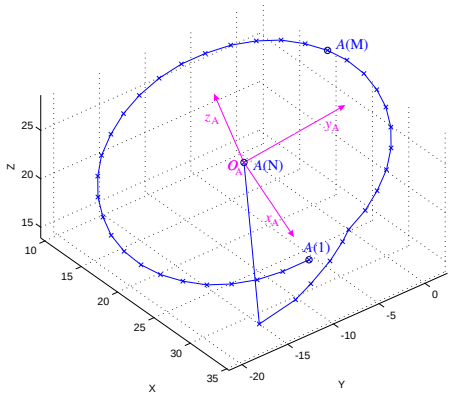


Figure 2: The applicator coordinate system is defined according to the applicator ring structure contour, with origin in the applicator ring center defined by the last point of the contour $A(N)$, xy plane in the ring plane with x axis pointing towards the contour starting point $A(1)$ and z axis in the direction of the tandem.

representation of structures by binary images.

In the following sections our approach to model the spatial configuration of cervix cancer is described first. We describe the proposed methods of constructing the spatial distribution model and the principal component model. Then we show some test results; for the spatial distribution model based on real patient data, while the principal component model is illustrated using synthetic data. We conclude with discussion that includes the analysis of provided benefits and limitations.

2 Methods

Our approach to build spatial models of cervix cancer consists of the following processing steps that are described below: data input, applicator coordinate system definition, structure processing, modelling and data export.

2.1 Data input

The input data for building the models consists of patient medical data sets that comprise all the required information of each patient, i.e., a 3D medical image, delineated HR-CTV structure, OAR structures and a reconstructed applicator structure. This data is typically provided in the DICOM file format, which can be imported using DICOM libraries, e.g., the GDCM library [8], or by Matlab using Image processing toolbox. Medical images are needed only to obtain the image configuration, i.e., transformation of image coordinate system according to the patient coordinate system and image slice positions, which are required to correctly interpret the structures. Structures are given in a form of structure sets that include all the structure data required for BT treatment. The target structure (HR-CTV), OAR structures and the applicator ring structure can be identified from all the structures according to their names that must be known in advance for each individual data set; the structure naming is not standardized. The target and OAR structures are defined as sets of closed planar contours, i.e., each contour is positioned on one slice of the corresponding 3D image. The applicator ring structures are described with a single open nonplanar contour. All contours are defined in the patient coordinate system.

2.2 Applicator coordinate system definition

Because the spatial configuration of cervix cancer needs to be defined according to the applicator perspective, an applicator coordinate system needs to be defined. The applicator reconstruction [9, 10] is performed on radiotherapy planning systems by importing predefined geometry structures. The applicator consists of tandem, ring and eventual additional needles, see Figure 1, which are all reconstructed independently. The ring structure, when inserted, tightly fits to the cervix anatomy, and provides a good base for defining the applicator coordinate system. Different applicator types may have different ring diameter, may be described with different number of contour points, however in practice the point ordering is always the same. For the illustration see Fig. 2. We propose that the applicator coordinate system is defined with origin in the ring center (the last point of the contour), xy plane in the ring plane, x axis in the direction towards ring contour starting point and z axis in the direction of the tandem.

The transformation that defines the applicator coordinate system according to the patient coordinate system can be for each applicator type computed from its ring contour coordinates. Actually, only three noncollinear contour points are required to compute the applicator coordinate system transformation $\mathbf{T}_A$, i.e., the last point, $A(N)$ that is positioned at the ring center as the coordinate system origin, $A(1)$ that defines the applicator coordinate system x axis, and any other point in the applicator contour circumference, $A(M)$ for defining the xy plane. The procedure is

the following:

$$O_A = A(N) \quad (1)$$
$$V_1 = A(1) - O_A \quad (2)$$
$$V_2 = A(M) - O_A \quad (3)$$
$$V_z = \frac{V_1 \times V_2}{\|V_1 \times V_2\|} \quad (4)$$
$$V_y = \frac{V_z \times V_1}{\|V_z \times V_1\|} \quad (5)$$
$$V_x = \frac{V_y \times V_z}{\|V_y \times V_z\|} \quad (6)$$
$$\mathbf{T}_A = \begin{bmatrix} V_x(x) & V_y(x) & V_z(x) & O_A(x) \\ V_x(y) & V_y(y) & V_z(y) & O_A(y) \\ V_x(z) & V_y(z) & V_z(z) & O_A(z) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

where, V and O are three dimensional vectors with components x, y, and z, such that O_A represents applicator coordinate system origin while V_x, V_y, and V_z are applicator coordinate system axes. Vector (cross) products assured coordinate axis perpendicularity, defining a Cartesian coordinate system.

The obtained transformation matrix $\mathbf{T}_A$ is needed for transforming BT structures to the applicator coordinate system in which the cervix cancer needs to be modelled.

2.3 Structure processing

For each image the corresponding BT target structure and OAR structures must be mapped into the applicator coordinate system. These structures are created by drawing contours on individual image slices and are provided as point sequences in the patient coordinate system. Such vector definition of structures is difficult to process statistically in coordinate system that is not parallel to the coordinate system of the originating image. Our solution is to present the structures in bitmap instead of vector format and process them as 3D (binary) images with voxel values 1 representing regions inside structures and 0 representing the surrounding. The approach is illustrated in Figure 3. The binary images cover the same region as the original medical image, except that, they may have different resolution in x and y image direction to control discretization error and data size. The resolution in z image direction must remain unchanged in order to preserve location of slices on which contours are defined.

The process of converting certain structure into a binary image starts with mapping the structure to the coordinate system of the original image. The transformation $\mathbf{T}_I$, form patient to image coordinate system can be obtained from image meta information, i.e., from DICOM tags *Image Position Patient* (0020,0032) and *Image Orientation Patient* (0020,0037). Thus, each contour C of structure S gets defined in its image coordinate system as C_I:

$$C_I = \mathbf{T}_I^{-1} C. \quad (8)$$

All points of the the same contour gets equal image coordinate z_I that is equivalent to the position of the image slice on which the contour was defined. The obtained structure can, as such, be drawn to the binary image, contour by contour. The process initiates by initializing all voxel values of the binary structure image to 0, followed by drawing the contours by checking each slice voxel if it is positioned inside of a polygon of contour points. Voxels inside the polygon gets negated to correctly interpret even complex structure shapes, e.g., shapes that include holes. Binary structure images enable further data integration towards spatial cervix cancer models.

To integrate the structure binary images of all patients into a spatial model, they all need to be mapped into the common applicator coordinate system (A), because patient coordinate systems (P) and image coordinate systems (I) are specific for each study/patient. Transformation between the coordinate systems are illustrated in Fig. 4. The data defined in image coordinate system (I) can be transformed to the applicator coordinate system (A) through the patient coordinate system (P) using transformation $\mathbf{T}_{IA}$:

$$\mathbf{T}_{IA} = \mathbf{T}_A^{-1} \mathbf{T}_I. \quad (9)$$

Structure binary images do not differ only according to their coordinate systems, but also according to image size and voxel size (resolution). For further analysis they need to be unified. The target region of interest and required precision define configuration of the resulting model (image) size and voxel size. All binary images must be resampled into this common spatial configuration. We recommend resampling by linear interpolation in reverse direction such that intensity corresponding to each voxel in the model configuration is interpolated from voxel intensities in the binary image. Note, that linear interpolation transforms a binary image into an image with real voxel values in interval $[0, 1]$. The result of structure processing is, therefore, a set of structure images S_A in a common applicator coordinate system and with common size and voxel size, i.e., each structure of each patient results in one structure image with common applicator (ring) position:

$$S_A = interp(\mathbf{T}_{IA}^{-1} S), \quad (10)$$

Where S denotes a structure binary image in the coordimate system of the original image, $interp$ a linear image interpolation, and S_A a structure image in the applicator coordinate system.

2.4 Spatial distribution model

The purpose of the spatial distribution model is to provide an overview of BT target spatial extent. It is given in a form of a spatial distribution image D, i.e., an image of the region of interest whose voxel values denote probability of voxel being inside the BT target region. It is obtained from images of the HR-CTV structure by averaging:

$$D = \frac{1}{n} \sum_{p=1}^{P} S_{A,p,HR-CTV}, \quad (11)$$

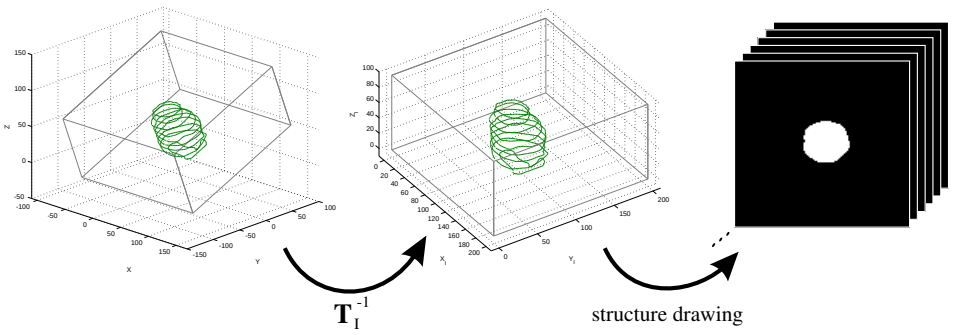


Figure 3: Illustration of structure processing: first, contours provided in the patient coordinate system (left) are transformed to the image coordinate system (center). Then, contours are drawn on image slices in 2D, which resulted in a 3D binary image of the structure (right).

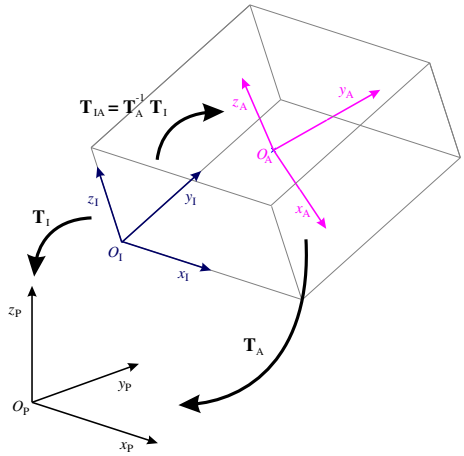


Figure 4: Illustration of patient (P), image (I) and applicator (A) coordinate systems and their transformations: $\mathbf{T}_{IA} = \mathbf{T}_A^{-1}\mathbf{T}_I$.

where $S_{A,p,HR-CTV}$ represents HR-CTV structure data of p-th patient resampled into an applicator coordinate system, and P is a total number of patients included in the analysis.

2.5 Principal component model

The principal component model provides information of the BT target spatial variability expressed by only a small number of parameters. The general idea is to be able to reconstruct any target configuration, i.e., position and extent of HR-CTV as well as OAR structures, by correctly setting the model parameters. As such the principal component model can be used to predict various target configurations, e.g., extreme situations in the scope of sufficient applicator radiation dose coverage in the target structure as well as radiation dose avoidance in OARs. Such situations may be crucial for testing real applicator efficiency. The principal component model tends to extract a minimal set of orthogonal components of spatial variations in the region of interest using the principal component analysis (PCA). PCA projects the data into a lower dimensional linear space such that the variance of the projected data is maximized, or equivalently, it is the linear projection that minimizes the mean squared distance between the data points and their projections. PCA provides a full set of components that enable perfect data reconstruction, however, it also orders the components according to their importance, i.e., according to their contribution to the data description. It turns out that majority of the components have low importance and only a small error is made when only a few most important components are used. In this case the important components can be computed more efficiently using singular value docomposition (SVD) [11].

Our input data for the PCA analysis of the BT target are the HR-CTV structure images in the applicator coordinate system $S_{A,p,HR-CTV}$. The data of each image is reordered into a row vector and joined for all the patients into a matrix $\mathbf{X}_{P\times L}$, with L being the number of pixels in the image. Then the mean vector $\overline{\mathbf{X}}$ is computed and subtracted from each data row to obtain the matrix $\mathbf{X}_0$ representing the zero-mean data variation. Here, the mean vector $\overline{\mathbf{X}}$ corresponds to the reordered data of the spatial distribution model D. SVD decomposes $\mathbf{X}_0$ into three matrices; matrix $\mathbf{V}$ with orthogonal columns that represent principal components, diagonal matrix $\mathbf{S}$ with singular values that represent importance of the components, and matrix $\mathbf{U}$ providing component weights for reconstructing the input data:

$$\mathbf{X}_0 = \mathbf{U}\mathbf{S}\mathbf{V}^T \tag{12}$$

The efficient SVD implementations, e.g., Matlab *svds* function, enable computation of only a given number of principal components R, and as such provide approximate solutions:

$$\mathbf{X}_{0\,P\times L} \approx \mathbf{U}_{P\times R}\mathbf{S}_{R\times R}\mathbf{V}^T_{L\times R} \tag{13}$$

The obtained matrices $\mathbf{S}$ and $\mathbf{V}$ represent a principal component model of the HR-CTV structure, such that HR-CTV structure of any patient can be represented with the R components, i.e., the columns of $\mathbf{V}$, with weights:

$$\mathbf{U}' = \mathbf{X}'_0\mathbf{V}\mathbf{S}^{-1} \tag{14}$$

where $\mathbf{X}'_0 = \mathbf{X}' - \overline{\mathbf{X}}$ represents deviation of the data from the average. Similarly, BT target data can also be simulated

by manual setting of component weights in $\mathbf{U}$, following equation (13) and adding the mean vector $\overline{\mathbf{X}}$. Component weights form a low dimensional linear space with a certain region around the origin that corresponds to realistic data variation. The limits of this realistic subspace can be estimated by analyzing large amount of data, i.e., large number of patients. Values at the border of the realistic subspace can be used in $\mathbf{U}$ to simulate specific extreme situations suitable for BT applicator analysis.

Testing of BT applicators on their ability to radiate the HR-CTV regions may be biased, as applicators should also be able to avoid radiation of OAR structures. It is important that HR-CTV and OAR structures cannot overlap. This property can be used to simultaneously model both structure types , i.e., HR-CTV as well as OAR structures, without increasing the amount of data in the PCA analysis. For this purpose the input vector $\mathbf{X}$ must be constructed from all the structure images, such that positive values represent target regions and negative values represent OAR regions:

$$\mathbf{X} = \mathbf{X}_{\text{HR-CTV}} - \sum \mathbf{X}_{\text{OAR}} \tag{15}$$

Here, $\mathbf{X}_{\text{HR-CTV}}$ and $\mathbf{X}_{\text{OAR}}$ are constructed from structure images with reordering into row vectors as described earlier, using HR-CTV and available OAR structures. Typically, OAR structures include bladder, rectum and sigmoid colon.

The simulated or reconstructed data that results from the principal component model, as well as principal components themselves, can be reordered back into 3D images. Due to interpolations and approximations the reconstructed structures are not presented only with values 1, -1 and 0 for target structures, OAR structures and surrounding respectively. Consequently, we recommend completing the reconstruction procedure with thresholding using thresholds -0.5 and 0.5.

2.6 Data export

The resulting images, i.e., an image of the spatial distribution model and simulated or reconstructed BT target configurations can be used in BT planning systems, e.g., Brachyvision©, for further analysis. BT planning systems include functionalities that enable radiation simulations using different radiation plans and can be used to test the efficiency of different applicators. To enable this procedures the images shall be exported to DICOM image format, which can be done using DICOM libraries, e.g. GDCM library [8], or Matlab image processing toolbox.

3 Results

The proposed methods have been tested on real and simulated data. First, the spatial distribution model was tested by a group of brachytherapy experts [12] using data of 264 patients from Institute Oncology Ljubljana, Medical University of Vienna and Aarhus University Hospital. The obtained cervix cancer distribution model was named a

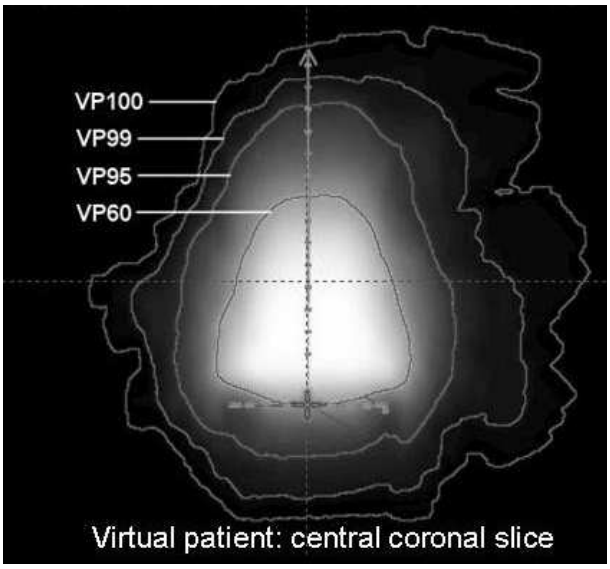


Figure 5: Illustration of the cervix cancer spatial distribution representing a virtual patient, the central coronal slice, as obtained in further research using the proposed method [12]. Image courtesy of Institute of Oncology Ljubljana.

virtual patient (VP). Imported to the BT planning system isosurfaces that connect voxels with the same values were created and labeled as percentage of encompassed voxels. VPn was defined as VP subvolume, encompassed by the n% isosurface, see the illustration in Figure 5. The obtained VP data was used for analysis and development of BT applicators for cervix cancer [12]. It was found out, that standard tandem and ring (T&R) applicator enables adequate treatment of VP60 subvolume, additional needles parallel to tandem extend adequate treatment to VP95 and additional oblique needles, inserted at points, angles and depths extend adequate treatment to VP99 subvolume. The principal component model was not built for this dataset, such that applicators were tested only for their general capability of radiating the HR-CTV, not considering the capability of avoiding the radiation of OAR structures.

The principal component model was tested using a simulated dataset that we have created for this purpose. Note that the simulated structure images presented here do not realistically simulate the BT target configuration, however enable illustration of the concept and testing of its suitability for creating a realistic model.

The simulated data was generated using four random parameters where three of them were used to simulate variability of the HR-CTV structure and the additional one for the variability of one OAR structure. The HR-CTV structure was simulated as an ellipsoid with the three parameters representing the semi-axes lengths while its center was always in the applicator coordinate system origin. The OAR region was simulated as a sphere with the given parameter representing its radius, while its center was defined such that the distance between the edges of BT target and the OAR structure was constant. For the illustration see Figure 6.

A principal component model was generated from a

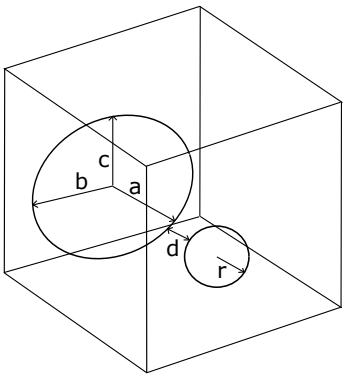


Figure 6: Illustration of the simulated dataset configuration. The HR-CTV was simulated with an ellipsoid and one additional OAR structure as a sphere with a constant distance d from the HR-CTV.

dataset of 400 simulated 3D images with $100 \times 100 \times 50$ voxels. The simulation parameters were selected randomly in the following ranges: $a \in [40, 73]$, $b \in [35, 59]$, $c \in [35, 49]$, $r \in [15, 20]$ and $d = 5$. The computation of the principal component model was restricted to 11 principal components. The mean image $\overline{\mathbf{X}}$ and the components are illustrated in Figure 7. The singular values that represent the distribution of the dataset's energy among the principal components indicate that the component energy gradually decreases with the component number, see Figure 8. However, although not all of the energy was considered, the reconstructed images did not differ considerably from the images from the training set as shown in Figure 9, where a randomly selected input structure image is compared with its reconstructed approximations obtained using three and eleven principal components. We can notice minor differences even when reconstructing from three components only.

If we observe the component weights (the values of matrix U), we can see that they are spread over a limited PCA subspace, see Figure 10, which corresponds to valid structure images. According to the shape of the subspace, we can conclude that component weights of valid images are not fully independent, although the components are orthogonal. By selecting weights manually, additional structure images can be simulated. If the selected weights are from the subspace of valid structure images, the simulated images follow the concepts of the input dataset, else the results may include major deviations as demonstrated in Figure 11.

The possibility to simulate structure images and have control over its validity offers good opportunity to generate specific synthetic images of the BT target region that represent extreme situations for BT applicator testing. In that case the principal component model should be created from real patient data and the test cases selected at the border of the populated PCA subregion.

The realistic model has not been created, yet, however we are looking forward to create it in collaboration with medical institutions that maintain large databases of their cervix cancer patients.

4 Discussion and conclusion

Cancer spatial distributions must be considered whenever cancer treatment tools and procedures are being developed. Unfortunately, statistical analysis of spatial distributions related to specific organs is in general tedious due to difficulties defining the reference coordinate systems because of their complex shapes and their high variability. In the specific case of cervix cancer the organ geometry enables unambiguous coordinate system definition that agrees with the applicator ring structure. Analysis of other cancer types would require definition of analysis coordinate system according to organ geometry and data integration performed by image registration with a reference or atlas [13]. Similarly, image registration has already been used for analyzing interfraction variation of high dose regions of OARs [14], and could be extended to intersubject analysis of cancer distributions.

The spatial distribution model provides useful information about target region that needs to be radiated, and has already been used for development of novel applicator types [12]. However, this model does not consider OARs and difficulties of restricting radiation dose in these structures. If a distribution model would be made for OARs as well, it would most probably overlap with the cancer distribution model due to closeness of some OAR structures to HR-CTV and due to anatomical variability. Better applicator testing must, therefore, take into account the BT target variability, e.g., by testing on diverse specific target configurations, which can correspond to real patients or obtained by modelling. The proposed principal component model has advantages over using the real patients' data, because of the established control over the specificity of the cases, a possibility to simulate the non-existent cases and depersonalization.

The limitation of the principal model is in its high computational cost. Computation of all the PCA components would require enormous amount of memory, only the $\mathbf{V}$ matrix would have the size of 500k $\times$ 500k elements (assuming $2 \times 2 \times 2$ mm voxel size), which in float data format requires 1TB of memory. Using the SVD approach with computation of the most important components only, drastically reduces the memory requirements; in our simulated case matrix $\mathbf{V}$ occupied only 22MB. Such reduction of components is possible due to final thresholding, which is applicable due to binary nature of the structures. When the computational cost remains a problem, high efficient PCA solutions [15] or alternative structure representations could be used.

A principal component model of real cervix cancer has not been made, yet. A large number of patient datasets is required and in contrast to the spatial distribution model the OAR structures must be included. The preparation of such data is tedious due to non-standardized structure naming.

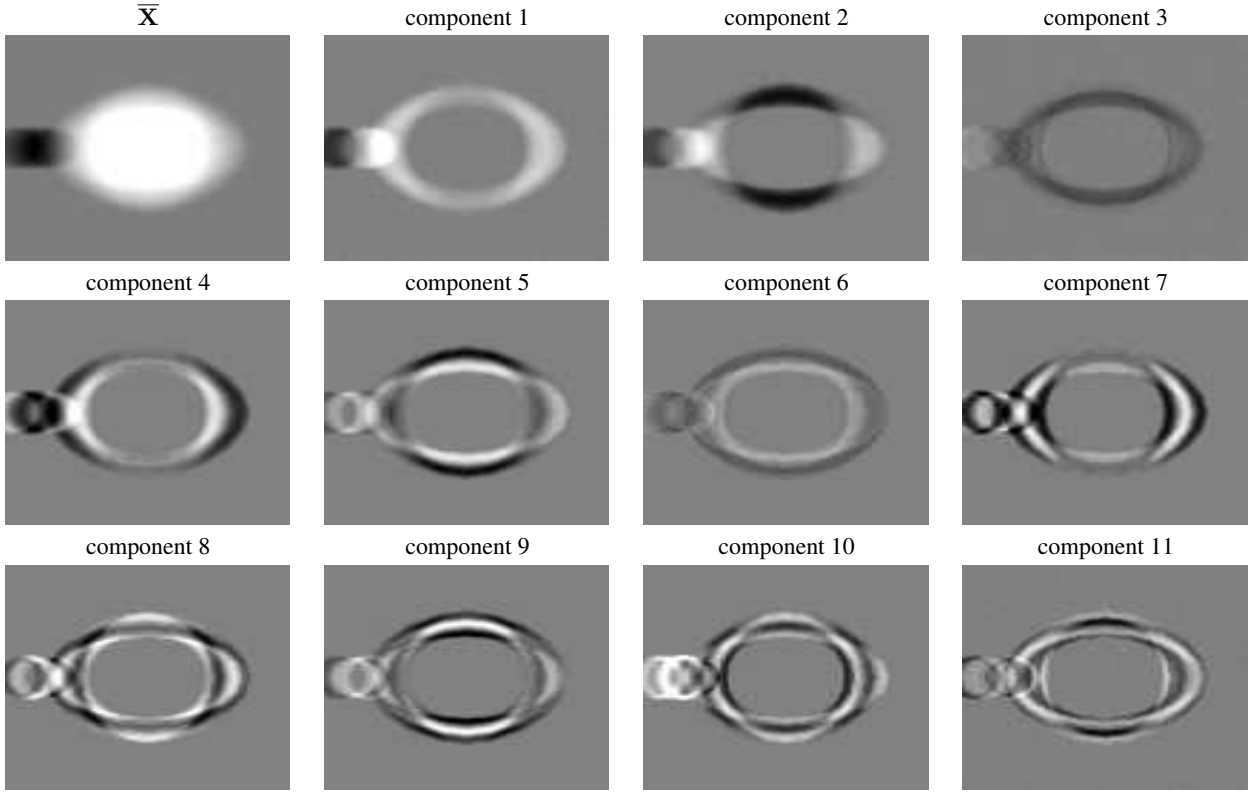


Figure 7: Components of the simulated dataset (central slices only). The mean image $\overline{\mathbf{X}}$ is presented in a scale from -1 (black) to +1 (white) and components with a scale from -0.01 (black) to +0.01 (white).

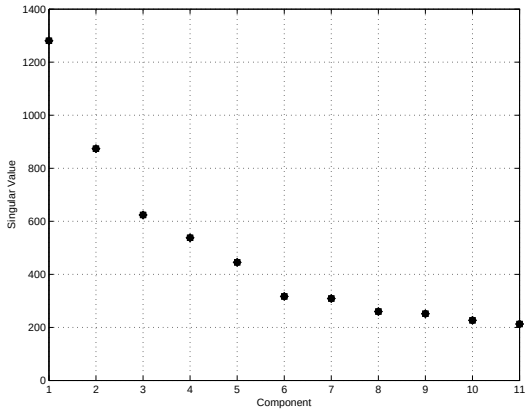


Figure 8: Singular values corresponding to the first 11 components; singular values represent the distribution of the dataset's energy.

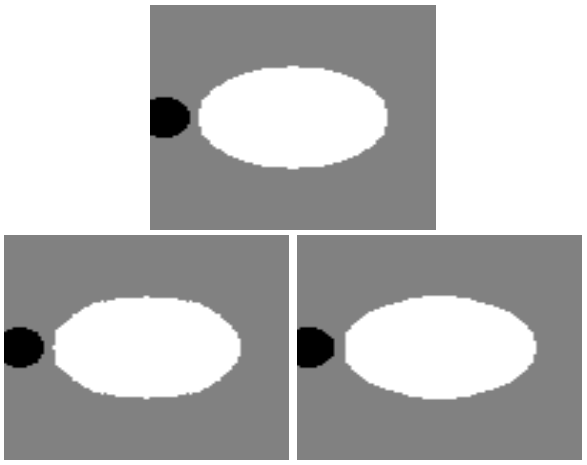


Figure 9: The central slice of an input structure image (top) and its reconstruction using 3 and 11 components (bottom left and right respectively).

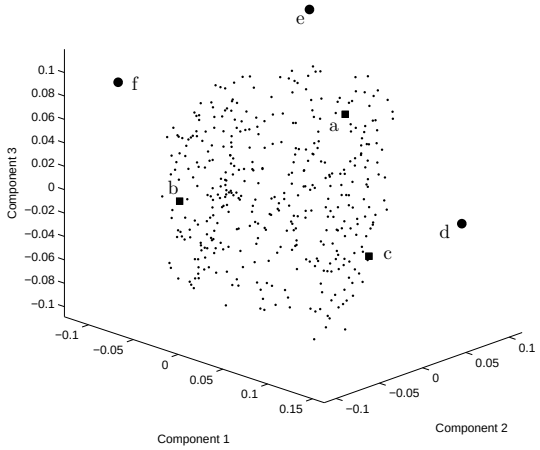


Figure 10: Weights for the first three components of the simulated structure images. The small dots correspond to images from the input dataset, squares and large dots represent selected values inside and outside the populated subspace for further simulations.

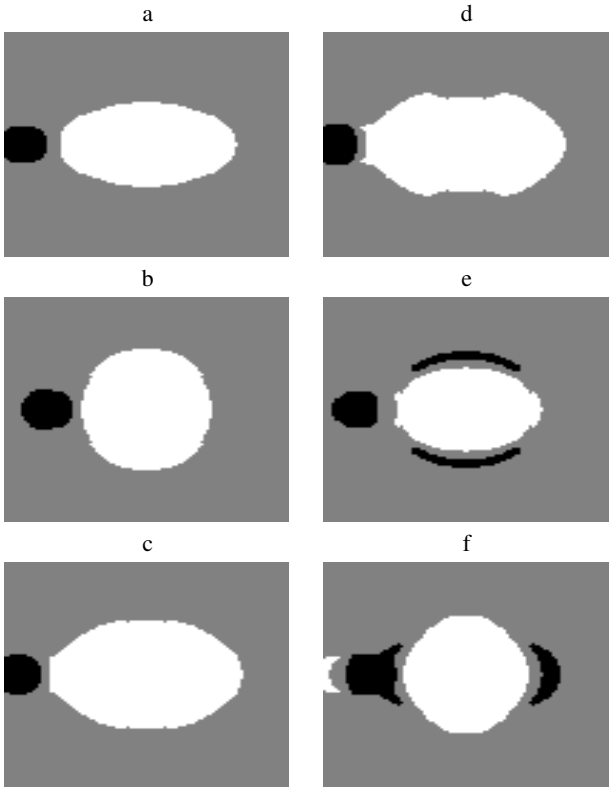


Figure 11: Central slices of simulated structure images using selected component weights from the subspace of valid structure images (left) and from other parts of the PCA space (right).

However the benefits of such dataset are not only in the support of applicator development, but also in outcomes of further statistical analysis that could support clinical process, e.g., structure delineation or radiation planing, as well as making of clinical decisions.

To conclude, it may be widely accepted that reducing dose at organs of risk is difficult without reducing dose at large tumors [16], we believe that applicator improvements based on spatial modelling could provide better alternatives.

References

[1] R. Potter, "Image-guided brachytherapy sets benchmarks in advanced radiotherapy." *Radiother Oncol*, vol. 91, no. 2, pp. 141–146, May 2009.

[2] A. H. Sadozye and N. Reed, "A review of recent developments in image-guided radiation therapy in cervix cancer." *Curr Oncol Rep*, vol. 14, no. 6, pp. 519–526, Dec 2012.

[3] M. B. Opell, J. Zeng, J. J. Bauer, R. R. Connelly, W. Zhang, I. A. Sesterhenn, S. K. Mun, J. W. Moul, and J. H. Lynch, "Investigating the distribution of prostate cancer using three-dimensional computer simulation." *Prostate Cancer Prostatic Dis*, vol. 5, no. 3, pp. 204–208, 2002.

[4] Y. Ou, D. Shen, J. Zeng, L. Sun, J. Moul, and C. Davatzikos, "Sampling the spatial patterns of cancer: optimized biopsy procedures for estimating prostate cancer volume and gleason score." *Med Image Anal*, vol. 13, no. 4, pp. 609–620, Aug 2009.

[5] R. Xu and Y.-W. Chen, "Appearance models for medical volumes with few samples by generalized 3d-pca," in *Neural Information Processing*, ser. Lecture Notes in Computer Science, M. Ishikawa, K. Doya, H. Miyamoto, and T. Yamakawa, Eds. Springer Berlin Heidelberg, 2008, vol. 4984, pp. 821–830.

[6] R. C. Conceicao, M. O'Halloran, E. Jones, and G. M., "Investigation of classifiers for early-stage breast cancer based on radar target signatures," *Progress In Electromagnetics Research*, vol. 105, pp. 295–311, 2010.

[7] K. M. Pohl, S. K. Warfield, R. Kikinis, W. L. Grimson, and W. M. Wells, "Coupling statistical segmentation and pca shape modeling," in *Medical Image Computing and Computer-Assisted Intervention MICCAI 2004*, ser. Lecture Notes in Computer Science, C. Barillot, D. Haynor, and P. Hellier, Eds. Springer Berlin Heidelberg, 2004, vol. 3216, pp. 151–159.

[8] "GDCM library home page (version 1.x)." [Online]. Available: http://www.creatis.insa-lyon.fr/software/public/Gdcm/

[9] D. Berger, J. Dimopoulos, R. Potter, and C. Kirisits, "Direct reconstruction of the vienna applicator on mr

images." *Radiother Oncol*, vol. 93, no. 2, pp. 347–351, Nov 2009.

[10] S. Haack, S. K. Nielsen, J. C. Lindegaard, J. Gelineck, and K. Tanderup, "Applicator reconstruction in mri 3d image-based dose planning of brachytherapy for cervical cancer." *Radiother Oncol*, vol. 91, no. 2, pp. 187–193, May 2009.

[11] M. E. Wall, A. Rechtsteiner, and L. M. Rocha, "Singular Value Decomposition and Principal Component Analysis," *ArXiv Physics e-prints*, Aug. 2002.

[12] P. Petric, R. Hudej, P. Rogelj, J. Lindegaard, K. Tanderup, C. Kirisits, D. Berger, J. C. A. Dimopoulos, and R. Potter, "Frequency-distribution mapping of hr ctv in cervix cancer : possibilities and limitations of existent and prototype applicators." *Radiother. oncol.*, vol. 96, suppl. 1, p. S70, 2010.

[13] K. Diaz, B. Castaneda, M. L. Montero, J. Yao, J. Joseph, D. Rubens, and K. J. Parker, "Analysis of the spatial distribution of prostate cancer obtained from histopathological images," in *Proc. SPIE 8676, Medical Imaging 2013: Digital Pathology, 86760V*, March 2013.

[14] J. Swamidas, U. Mahantshetty, D. Deshpande, and S. Shrivastava, "Inter-fraction variation of high dose regions of oars in mr image based cervix brachytherapy using rigid registration," *Medical Physics*, vol. 39, no. 6, p. 3802, JUN 2012.

[15] V. Zipunnikov, B. Caffo, D. M. Yousem, C. Davatzikos, B. S. Schwartz, and C. Crainiceanu, "Multilevel functional principal component analysis for high-dimensional data," *Journal of Computational and Graphical Statistics*, vol. 20, no. 4, pp. 852–873, 2011.

[16] R. Kim, A. F. Dragovic, and S. Shen, "Spatial distribution of hot spots for organs at risk with respect to the applicator: 3-d image-guided treatment planning of brachytherapy for cervical cancer," *International Journal of Radiation Oncology, Biology, Physics*, vol. 81, no. 2, S, p. S466, 2011.

Detection of Ground in Point-clouds Generated from Stereo-pair Images

Domen Mongus and Borut Žalik
University of Maribor, Faculty of Electrical Engineering and Computer Science
E-mail: domen.mongus@um.si and borut.zalik@um.si, http://gemma.uni-mb.si

Keywords: digital terrain model, mathematical morphology, Θ-mapping

Received: June 24, 2014

This paper proposes a new approach for constructing digital terrain models (DTM) from the point-clouds generated from airborne stereo-pair images. The method uses data decomposition based on the differential attribute profiles and Θ-mapping for the extraction of the most-contrasted connected-components. Their filtering is achieved based on multicriterion threshold function. The method is evaluated by comparing the output DTM with the reference Light Detection and Ranging data (LiDAR).

Povzetek: V članku predstavljamo novo metodo za konstrukcijo digitalnega modela reliefa iz oblaka točk, ki so generirani iz stereo-parov zračnih fotografij. Metoda uporablja podatkovno dekompozicijo na osnovi diferencialnih atributnih profilov in Θ-mapiranja, s katerima doseže zaznavo najbolj kontrastnih povezanih komponent. Razpoznavo točk terena dosežemo z večkriterijskim pragovnim filtriranjem. Metode je evaluirana s primerjavo z digitalnim modelom reliefa, ustvarjenim iz podatkov LiDAR.

1 Introduction

Digital terrain models (DTMs) are essential part of various spatial analysis, geographic applications, and virtual reality systems [19, 6, 14]. In recent years, a considerable effort has been directed towards developing efficient approaches for accurate DTM generation.

When considering DTM generation from point-clouds, the most often used approaches can, according to the literature, be classified as slope-based, linear prediction-based, and morphological methods [20, 9]. Slope-based methods [18, 21] achieve point-filtering by comparing the gradients between neighbouring points. Consequentially, they have difficulties filtering points on step slopes and tend to smooth terrain undulations [20, 9]. Linear prediction-based methods, on the other hand, have difficulties filtering small and low objects as they rely on rough surface approximation to establish a liner prediction of the terrain [8, 2]. Actual filtering is usually achieved by observing the points' residuals from the predicted surface. Preservation of sharp terrain details (e.g. ridges) can, therefore, be exposed as another weakness [20, 9]. By applying operations of mathematical morphology [5, 11, 4, 16], morphological filters proved to be fairly resistant to previously exposed drawbacks. However, they are severely dependent on the definition of the structuring element, as large objects (e.g. buildings) cannot be removed using a small structuring element, whilst large structuring element tends to flatten terrain details (e.g. mountain peaks) [20, 9, 5]. Several attempts have been proposed for optimal definition of a structuring element, the most efficient of which are based on a multi-scale filtering. A set of filters of different scales is used for this purpose and different threshold values are usually defined for each of them. A progressive filtering was proposed by Chen et al. [5], where thresholding is applied on height differences achieved by each filter. On the other hand, Mongus and Žalik [11] proposed data-filtering by iterating thin-plate splines towards the ground, where resolution is increased at each iteration by including points, filtered according to their residuals from the previously estimated surface. This, so-called, hierarchical multiresolution filtering has recently been improved by Chen et al. [4]. Pingel et al. [16] have, on the other hand, based their approach on the slope estimation achieved by linearly increasing filtering scale. Since all of these methods are adopted for processing high-resolution point-clouds containing vast amounts of points (e.g. LiDAR data), iterative approaches may not always be appropriate. Mongus and Žalik have [12] proposed an efficient multiscale approach that avoids iterations by using attribute filters based on the max-tree data structure. Although the method proves efficient when filtering LiDAR data, its accuracy is not guaranteed when filtering low-resolution point-clouds (such as those generated from stereo-pair images) since it is based on the standard deviation of point heights.

This paper presents a new method for estimation of digital terrain model from point-clouds generated from airborne stereo image pairs. By considering $\Theta-$mapping, the proposed method is an extension of [12], where a different set of attributes is used for the filtering. Section 2 explains theoretical foundation of connected operators from mathematical morphology that allows their efficient estimation. The method is explained in Section 3. Section 4 gives the results, whilst Section 5 concludes the paper.

2 Theoretical background

Let $g : E \rightarrow \mathbb{R}$ be a regular grid, where $E \subset \mathbb{Z}^2$ and $p \in E$ is a grid point. Consider a level-set $E_l \subset E$ given by the hight-level l as $E_l = \{p \mid g[p] = l\}$. A connected component from E_l is named a flat-zone of g. A filter that acts on flat-zones rather than individual grid-points is named a connected operator [17]. A connected operator can eider remove a flat-zone (by merging it with some other flat-zones) or leave it perfectly preserved, but it cannot brake it. If the decision about which flat-zones to merge is based on some of their attributes, this type of operator is named an attribute filter [1]. Consider a set of all thresholded sets $T = \{T_l\}$ of g, each obtained by

$$T_l = \{p \mid g[p] \geq l\}. \tag{1}$$

A peak connected component $C_l^k \in T_l$ is defined by its height level l and its component-at-level index k . Let an attribute function $A(C_l^k)$ that estimates a particular attribute of C_l^k, e.g. its area, diameter, or bounding-box. For simplicity, let A be increasing, thus, satisfying the condition $C_{l_1}^{k_1} \subseteq C_{l_2}^{k_2} \rightarrow A(C_{l_2}^{k_2}) \leq A(C_{l_2}^{k_2})$. An attribute filter γ_a^A acting on g is at a particular point p defined by

$$\gamma_a^A(s)[p] = \bigvee\{l \mid p \in C_l^k,\ A(C_l^k) \geq a\}, \tag{2}$$

where $\bigvee$ is supremum (i.e. the upper-bound). In other words, an attribute filter γ_a^A removes all the peak connected components not satisfying an attribute threshold condition a by assigning to each point p the maximal height-level at which it still belongs to a peak connected component C_l^k with $A(C_l^k) \geq a$. Since $\forall g, \gamma_a^A(g) \leq g$, γ_a^A is an anti-extensive morphological filter named attribute opening. Its dual, an attribute closing ϕ_a^A, is defined as $\gamma_a^A(g) = -\phi_a^A(-g)$.

A decomposition named DAP or differential attribute profile Δ has recently been proposed by Ouzounis et. al. [15]. Δ is based on progressive content reduction by filtering g at an increasing scale. Consider an ordered set of attribute thresholds $\mathbf{a} = \{a_i\}$, where $i \in [0, I]$ and $a_{i-1} < a_i$, Δ is obtained by

$$\Delta_{\mathbf{a}}^A(g) = \{\gamma_{a_{i-1}}^A(g) - \gamma_{a_i}^A(g)\}, \tag{3}$$

where $i \in [1, I]$. Thus, $\Delta_{\mathbf{a}}^A(g)$ is an $I-$long response vector registering the differences introduced by each particular $\gamma_{a_i}^A$, whilst $\gamma_{a_I}^A(g)$ is a grid residual.

Recently, Mongus and Žalik [12] proposed $\Theta-$mapping that registers the most-contrasted connected-components and estimates their arbitrary attributes by observing characteristic values contained in $\Delta_{\mathbf{a}}^A$. Namely, $\Theta-$mapping estimates the most-contrasted connected-components from g by registering the maximal responses from $\Delta_{\mathbf{a}}^A(s)$ and filtering scale at which they are obtained. Formally, $\Theta(g, A, \mathbf{a}) : g \rightarrow (g', g^\circ)$, is at p given by

$$g'[p] = \bigvee \Delta_{\mathbf{a}}^A(g)[p], \tag{4}$$

$$g^\circ[p] = \bigwedge i \mid \gamma_{a_{i-1}}^A(g)[p] - \gamma_{a_i}^A(g)[p] = g'[p], \tag{5}$$

where $\bigwedge$ is infinum (i.e. the lower-bound). Consider a set of peak connected-components $C^p = \{C_l^p\}$ containing a point p, i.e. $C_l^p = C_l^k \mid p \in C_l^k$. The most-contrasted connected-component C_{max}^p with the respect to the given $\Delta_{\mathbf{a}}^A(g)$ is identified by

$$max = \bigvee l \mid a_{g^\circ[p]-1} \leq A(C_l^p), \tag{6}$$

where max defines the height-level of the most-contrasted connected-component. Note that possibly no response was obtained at a given p, meaning that the corresponding peak connected-components are not in contrast against their surroundings and, therefore, belong to the grid residual, i.e. background. In any case, an arbitrary attribute of C_{max}^p can then be measured and used as an attribute in multicriterion threshold definition.

3 Ground extraction from point-clouds

The proposed method generates a digital terrain models from point-clouds obtained by stereo-pair images in the following tree steps:

- **Initialization** is the first step of the method were input point-cloud is sampled into a grid,
- **Point filtering** is performed in the space of the most-contrasted connected-components obtained by Θ-mapping, and
- **Construction of DTM** is the final step of the method, where removed points are interpolated.

Each of these steps is discussed in continuation.

3.1 Initialization

In order to apply morphological operators on point-clouds, points are firstly sub-sampled into a regular grid g. The resolution of the grid R_g is defined by the point-density D_L as $R_g = 1.0/D_L$. When a particular grid-cell contains more than one point, the hight level of the grid-point is defined by the lowest one since it has the highest probability of being a ground point. On the other hand, interpolation is used in order to estimate the hight levels of the undefined grid-points $g[p^*] = UNDEF$, obtained when there are no points contained within the corresponding grid-cells. In our case, the height level at p^* is estimated using inverse distance weighting (IDW) as [10]

$$g[p_n^*] = \frac{\sum_{p_n \in W^{p_n^*}} g[p_n]\ d_{p_n}^{-r}}{\sum_{p_n \in W^{p_n^*}} d_{p_n}^{-r}}, \tag{7}$$

where p_n is a grid-point from the neighbourhood $W^{p_n^*}$ of p_n^*, and d_{p_n} is the Euclidean distance between p_n^* and

p_n. Parameter r defines the smoothness of the interpolation. According to the evaluation of the spatial interpolation methods described in [3], accurate results are obtained when $W^{p^*_n}$ contains at least three closest points and $r = 2$.

3.2 Ground filtering

In order to achieve extraction of the most-contrasted connected-components, the underlying definition of DAPs is given first. In compliance with demanded increasing property of the attribute used for grid decomposition, the proposed method constructs DAPs according to the area of the contained peak connected components A. An area threshold vector **a** is given as

$$\mathbf{a} = \{20.0 * i\}, \tag{8}$$

where $i \in [0, I]$. Note that **a** is given in square-metres, thus, its definition should be adjusted when the input point-cloud is not georeferenced. In any case, the following attributes of the most-contrasted connected-components are estimated by Θ-mapping:

- g' describes the height difference or residual of the most-contrasted connected-component from its background and is estimated by eq. 4,
- g° describes the area of the most-contrasted connected-components according to eq. 5,
- g^c is a function describing shape-compactness of the most-contrasted connected-components and is estimated based on a well-known distance transformation as [13]

$$g^c[p] = \frac{A(C^p_{max})}{9\pi * \overline{DT}(C^p_{max})}, \tag{9}$$

where $\overline{DT}(C^p_{max})$ is a function that estimates the average distance of a grid-point contained within C^p_{max} to the closest background point.

After g', g°, and g^c are estimated, a set of ground grid-points G is recognized with a multicriterion threshold function given by

$$G = \{p \mid g'[p] \leq t^R, g^\circ[p] \leq t^S, g^c[p] \leq t^C\}, \tag{10}$$

where t^R, t^S, and t^C are residual, size, and compactness thresholds, respectively.

3.3 DTM construction

In the final step of the method, DTM is contracted by interpolating the heights of the non-ground points $NG = E/G$ using IDW, as given by eq. 7. However, using $r = 2$ may not always be appropriate as it may produce some sharp unnatural terrain features. Additional smoothing is, therefore, performed based on morphological opening γ_w, where w is a structuring element. In our case, final DTM is obtained by

$$DTM[p] = \begin{cases} g[p] & ; \quad g[p] - \gamma_w(g) \leq R_g/2.0 \\ \gamma_w(g)[p] & ; \quad otherwise \end{cases} \tag{11}$$

where w is box-shaped structuring element of size 5×5.

4 Results

In order to evaluate the method, a point-cloud has been generated from georeferenced stereo-pair image as proposed in [7] with approximately 17.000 points. Average point spacing was below $3.1m$ and average absolute height error was $5.3m$ in comparison to the reference data (see Fig. 1a). The reference data was acquired with LiDAR technology. The referenced point-cloud contained more than 1.6 millions of points with average point-spacing below than $0.25m$ and average absolute height error below $0.1m$ (see Fig. 1c).

The reference DTM was obtained with [12] and was used for the evaluation of the proposed method (see Figs. 1b and d). The results show that the proposed method is capable of removing important portion of noise as the average absolute difference of DTMs was lower than the average error of the point-clouds. Namely, the error is reduced to $4.8m$. However, significant portion of DTM's details is missing due to the lower point-cloud resolution.

5 Conclusion

The paper proposes a new method for estimation of DTMs from point-clouds, generated by stereo-pair areal images. The method determines non-ground regions by estimating their geometrical characteristics, namely their sizes, shape compactness, and height differences from the background. As confirmed by the results, Θ-mapping provides sufficient solution for this purpose as great majority of errors were introduced by interpolation and lower data accuracy in comparison to LiDAR data.

6 Acknowledgments

This work was supported by the Slovenian Research Agency under grants $L2-3650$ and $P2-0041$. This paper was produced within the framework of the operation entitled "Centre of Open innovation and Research UM". The operation is co-funded by the European Regional Development Fund and conducted within the framework of the Operational Programme for Strengthening Regional Development Potentials for the period $2007-2013$, development priority 1: "Competitiveness of companies and research excellence", priority axis 1.1: "Encouraging competitive potential of enterprises and research excellence".

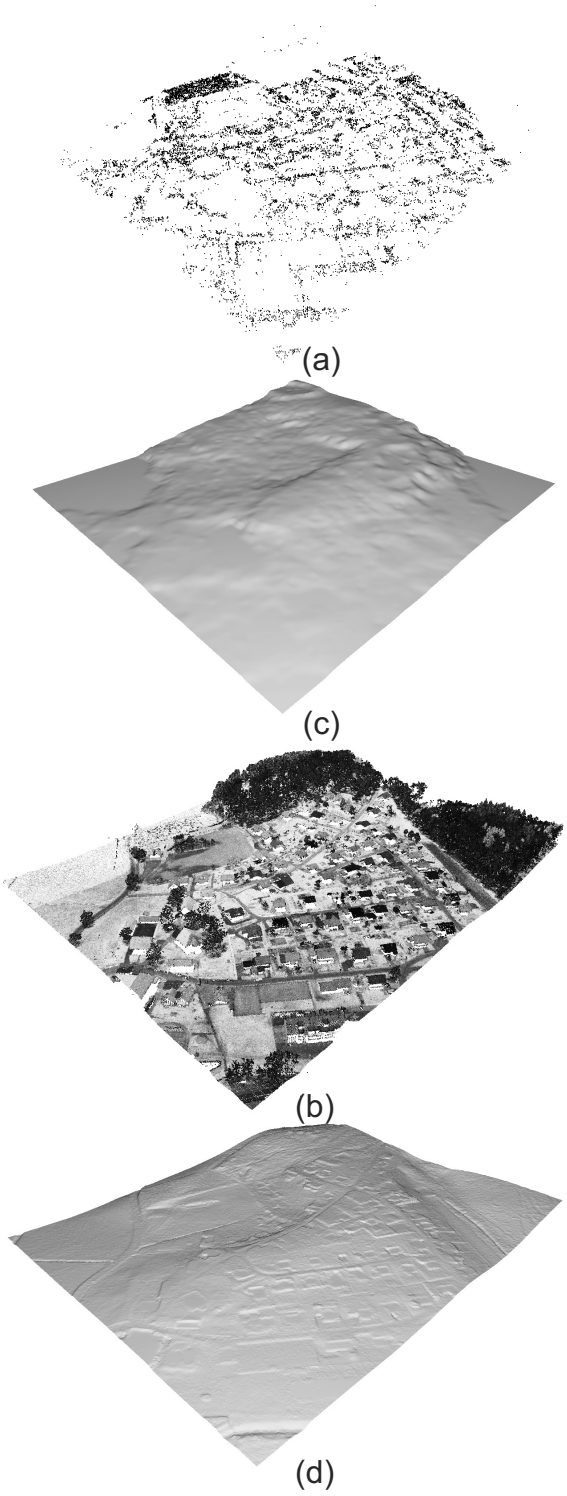


Figure 1: Estimation of DTM from (a and b) stereo-pair images and (c and d) the reference LiDAR data.

Acknowledgement

Acknowledgement text.

References

[1] E. Breen and R. Jones. Attribute openings, thinnings and granulometries. *Computer Vision and Image Understanding*, 64(3):377–389, 1996.

[2] M. A. Brovelli, M. Cannata, and U. M. Longoni. LiDAR data filtering and DTM interpolation within GRASS. *Transactions in GIS*, 8(2):155–174, 2004.

[3] V. Chaplot, F. Darboux, H. Bourennane, S. Leguédois, N. Silvera, and K. Phachomphon. Accuracy of interpolation techniques for the derivation of digital elevation models in relation to landform types and data density. *Geomorphology*, 77 (1-2):126–141, 2006.

[4] C. Chen, Y. Li, W. Li, and H. Dai. A multiresolution hierarchical classification algorithm for filtering airborne LiDAR data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 82:1–9, 2013.

[5] Q. Chen, P. Gong, D. Baldocchi, and G. Xie. Filtering airborne laser scanning data with morphological methods. *Photogrammetric Engineering & Remote Sensing*, 73(2):175–185, 2007.

[6] R. Dinuls, G. Erins, A. Lorencs, I. Mednieks, and J. Sinica-Sinavskis. Tree species identification in mixed baltic forest using LiDAR and multispectral data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 5(2):594–603, 2012.

[7] M. Eineder, N. Adam, R. Bamler, N. Yague-Martinez, and H. Breit. Spaceborne spotlight SAR interferometry with TerraSAR-X. *IEEE Transactions on Geoscience and Remote Sensing*, 47 (5):1524–1535, 2009.

[8] H. S. Lee and N. Younan. DTM extraction of LiDAR returns via adaptive processing. *IEEE Transactions on Geoscience and Remote Sensing*, 41(9):2063–2069, 2003.

[9] X. Liu. Airborne LiDAR for DEM generation: some critical issues. *Progress in Physical Geography*, 32(1):31–49, 2008.

[10] C. D. Lloyd. *Local Models for Spatial Analysis (2nd ed.)*. CRC Press, 2010.

[11] D. Mongus and B. Žalik. Parameter-free ground filtering of LiDAR data for automatic DTM generation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66 (1):1–12, 2012.

[12] D. Mongus and B. Žalik. Computationally efficient method for the generation of a digital terrain model from airborne LiDAR data using connected operators. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, In press:1–12, 2013.

[13] R. S. Montero and E. Bribiesca. State of the art of compactness and circularity measures. *International Mathematical Forum*, 4 (25-28):1305–1335, 2009.

[14] A. O. Onojeghuo and G. A. Blackburn. Characterising reedbeds using LiDAR data: Potential and limitations. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, (In press):1–7, 2012.

[15] G. K. Ouzounis, M. Pesaresi, and P. Soille. Differential area profiles: Decomposition properties and efficient computation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1533–1548, 2012.

[16] T. J. Pingel, K. C. Clarke, and W. A. McBride. An improved simple morphological filter for the terrain classification of airborne LIDAR data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 77:21–30, 2013.

[17] P. Salembier and M. H. Wilkinson. Connected operators: A review of region-based morphological image processing techniques. *IEEE Signal Processing Magazine*, 136 (6):136–157, 2009.

[18] J. Shan and A. Sampath. Urban DEM generation from raw LiDAR data: A labeling algorithm and its performance. *Photogrammetric Engineering & Remote Sensing*, 71(2):217–222, 2005.

[19] B. Sirmacek, H. Taubenbock, P. Reinartz, and M. Ehlers. Performance evaluation for 3-D city model generation of six different DSMs from air- and spaceborne sensors. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 5(1):59–70, 2012.

[20] G. Sithole and G. Vosselman. Experimental comparison of filter algorithms for bare earth extraction from airborne laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 59(1-2):85–101, 2004.

[21] C. Wang and Y. Tseng. DEM generation from airborne LiDAR data by an adaptive dualdirectional slope filter. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38(7B):628–632, 2010.

IJCAI 2015 – The Worst Best Ever

Matjaz Gams,
Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia
E-mail: matjaz.gams@ijs.si

Keywords: conference report

Received: August 5, 2015

Editorial

In 2015 the International Joint Conference on Artificial Intelligence (IJCAI) was held in Argentina from July 25th to July 31st. As a result of all the AI hype in the media – and in particular about the exaggerated danger of it ending the human race – the IJCAI was destined to succeed. But was this really so?

With around 30% more submitted papers than at the previous IJCAI in China, and with an absolute record in terms of the number of papers, the conference was a true success. Also, the quality of the papers and the invited lecturers remained superior to all other AI conferences, including rivals like the American AAAI or the European ECCAI. The technical papers showed progress in their particular technical fields, while the invited papers provided a broad overview of the field and the major achievements. Hardly any of the participants would object to these claims.

Interestingly, the number of accepted papers from China slightly surpassed the figure for the USA, and was proclaimed as the no. 1 country in the world - obviously (the EU not being regarded as a single country). Otherwise, the EU would be seen as the leading AI region in the world. One should also keep in mind that China is becoming the world's largest economy, while the EU is a close third. But here it is important to remember that China has a four-times larger population than the USA, and nearly three times that of the EU.

The progress and achievements in AI are not only evident to AI professionals, i.e., researchers and developers, but also to the public in several ways. For example, more and more companies are intensively investing in AI and, besides hiring AI staff, also provide substantial incomes. There is hardly any major IT or web-related company that is not using several AI methods. In practically all areas of computer and mobile use, AI is introducing new functionalities.

To demonstrate this concept, one could start with music, as was performed live at the opening of the IJCAI. Anybody from the audience could select a note in the musical scores, a pianist reproduced the selected music from that note on, and within a second or two the computer recognized the composition, displayed the current bar and began following the performance in real-time. Fascinating! Congratulations to MusicCompanion, the Austrian Gerhard Widmer and his team. This example nicely indicates the new orientations and achievements of modern AI, i.e., everybody is familiar with the possibility of computers finding any specific text on a disk or in a book, large directories or large databases. To achieve the performance of MusicCompanion, the sounds of a piano, or any other form of audio, have to be transformed into a computer-readable form, and then a proper program can find its location in the databases of properly represented music events – from text search to music search in just a couple of years. There was no major breakthrough in computer or artificial intelligence, "only" very clever programming, good AI and several years of hard work. Similarly, the list of such successful applications applied to real-world problems goes on and on.

The Computer and Thought Award recipient Ariel Procaccia presented his achievements in three areas. The first was in kidney exchange (human-organ transplantation), where AI enables a significantly better exchange due to its mechanisms: intelligent fight with combinatorial explosion, successful introduction of time and the ability to rapidly incorporate modifications. This and other achievements have been widely published in the USA and the world's media. A similar case exists with security scenarios, where the task is to organize patrols in such a way as to optimize the defence against attacks. Millind Tambe is recognized as the pioneer in this area. With modifications, these AI methods are being deployed at various national institutions. The third area is fair division, e.g., of a heritage. A computer program finds the best solutions given user preferences. The optimality is not guaranteed theoretically, yet it is very often achieved in reality. This and several other AI programs are available through the web.

The list of other prize recipients includes Barbara Grosz (Research Excellence in NLP and Multi-Agent Cooperation), Bart Selman (John McCarthy Award for Taming Complexity in General Inference Mechanisms) and Anthony G. Cohn (Donald E. Walker Distinguished Service Award for Contributions in AI Societies and Journals).

There were so many interesting papers and presentations that any IJCAI scientific reporter has to rely on some personal bias. For example, Christof Koch presented Consciousness in Biological and Artificial Brains. First, he described the necessary conditions for (human-type) consciousness, and then based on reported human cases provided proofs that one needs no feelings, no self-consciousness or several parts of the brain to be conscious. The minimum that is needed is approximately one hemisphere of the human brain, with the

corresponding neocortex of a live human. For example, the cerebellum, sensing or other brain or neuronal parts were found to be missing, but with the human still being conscious. From this minimal functionality Koch presented five axioms and the related consequences, and finally proposed a formula for computing the amount of consciousness, producing a numerical value as the output. According to his presentation, the formula for consciousness relies on the number of nodes and the complexity of the neural network's architecture with recursive loops and modules. As a consequence, current computers probably do not enable a physical realisation of consciousness. In his opinion, current computers therefore have zero consciousness, while animals have some and humans are at the top of the list. Future computers will sooner or later achieve consciousness, but the architecture will have to be physically different, not just different in terms of simulation. In the same way as a computer simulation of a black hole does not bend the time-space, a computer simulation of consciousness cannot produce true consciousness. These concepts might be hard to grasp, but just consider that a prototype of a car moves in a real world, while a simulation of a car does not move in the real world. Similar relations are valid for computers, the mind and consciousness. In any case, these revelations provide several new possibilities for thought and research experiments in AI and the cognitive sciences.

The panel "Rethinking the Turing Test" was based on the need for another, better test of computer intelligence. The majority of speakers revealed weaknesses of the Turing Test (TT): the test is about human psychology, TT competitions are about tricks designed to deceive an average interrogator with mistakes, deviations, etc., not helping to develop new AI methods. To measure actual AI progress, another test is needed, e.g., the Winograd schema, with current scores of about 70 (they did not mention that for random replies, the score should be 50). There are other tests, like Captcha, that can distinguish software agents from humans on the web because the software is not able to read blurred text. The Lovelace test is based on creativity capabilities. Another test deals with impossible stories or pictures the computers have problems with. All the panellists agreed that improvements beyond the TT are needed. The most damaging opinion was provided by a Russian child, who asked "What would our world look like without the TT?"

Maybe the only interesting confirmation of the TT was that no program has ever passed the Turing Test and that the media reports about such an event in 2014 were not well founded: "The 65-year-old iconic Turing Test was passed for the very first time by the computer programme Eugene Goostman during a Turing Test in 2014, held at the renowned Royal Society in London on Saturday. 'Eugene' simulates a 13-year-old boy and was developed in Saint Petersburg, Russia." These statements were later dismantled in a scientific paper and other publications. The trick was in misguiding the interrogators by telling them that the computer was a 13-year-old boy from the Ukraine in order to explain the odd responses. Looking at Eugene's replies it seems quite naïve that any human fell for it, but then again looking at situations when I was pickpocketed it seemed quite silly as well. It is important to realise that neither of the two situations had much in common with the true nature of the TT.

It seems that Stephen Muggleton was the only person objecting to new tests: "Cheetahs run faster than humans, but are not more intelligent. Similarly, these new tests will not measure intelligence."

My opinion is that there is nothing wrong with proposing new tests; however, there is no need to misinterpret the TT, one of the major scientific tests. The true concept of the TT is far beyond the understanding of the mass media. For example, consider that the students during one of my classes are instructed to propose a couple of questions that would reveal the true identity of a human/computer. Even though they are warned at the lectures that these questions will be asked at the exam, the students find these issues hard indeed. They typically prepare several questions in advance, but stall when the task is slightly modified or directed to enable perceiving a true understanding.

The catch is that computers have zero true understanding, semantics, and zero top human mental performance. If a new statement is formed and the next question is about the meaning of the previous sentence, computers fail miserably. The TT therefore correctly shows that current computers possess no true intelligence, as does Koch's equation for consciousness. Any test showing current consciousness or other top human mental property would clearly be bogus. Designing such a test would be a matter for witchcraft, not a matter for science.

And the question about what a world would be like without the TT – this is similar to several other scientific comprehensions and laws, e.g., about the existence of black holes or that there are other planets in our universe. Without this knowledge we would not properly understand the world around us. For example, without the TT we might think that we already have human-smart computers and would have no need to design novel AI systems. Then we would be like a civilization designing better and better springs in an attempt to travel the universe, not realising that we need something entirely different.

There have been several areas of significant AI progress recently, among them the smart assistants like Siri or Android Assistant. At the IJCAI, several new systems were presented, for example, for helping the elderly at home alone with a specialized knowledge of certain needs and tasks. For example, a Chinese robot, Vivian, has knowledge about travel, tickets, banking transactions, etc. Facebook presented cognitive computing as the new paradigm, consisting of learning for personalised information, and not forgetting IBM, which was involved in several new AI research activities, including previous victories over humans in chess and Jeopardy, and which produces several patents per day. Companies like Microsoft, and in particular Google, presented their AI achievements. of which several are

well popularized. The "Deep machine learning" term is often related to recurrent neural networks, but has progressed into another generation of machine learning, the one that renders ML repositories like Weka a kind of obsolete. The shift has moved from users using specific algorithms towards user declaring the task and the AI system chooses the best methods and parameters to fulfil the task. And is it not Auro-Weka which searches for optimal setting of the particular algorithm; it is also dealing with the algorithms and parameters and not with the task itself. Another less well known event is the Alibaba success story.

The Chinese company Alibaba was established as a private company with one owner in 1999. Their goal then and now remains the same: making business easy anywhere. Currently, they are the world leader in many parameters, e.g., in the number of registered users, the number of transactions per day, etc. One of their products is Taobao, the 20th most often visited webpage with 800 million products available for purchase. Several of their services are based on AI systems, mainly on machine learning. Among them is the dynamic-pricing, credit-scoring system and smart logistics. The credit system is totally autonomous – based on data about a particular customer it proposes a well-augmented amount of credit or rejects it. According to reports, the system will become the new standard credit system in China.

What about the media hype that concerns AI destroying the human race? As could probably be expected, no technical paper and no workshop or tutorial focusing on that issue took place at IJCAI2015. However, a couple of authors mentioned it during their presentations. For example, Ariel Pocacca, the first awarded speaker, rhetorically asked where the media sees a correlation between the AI contributions and the dangers to humanity. He presented three systems that are clearly beneficial to humanity, e.g., the first one about improved kidney exchange. On the other hand, the discussion about autonomous killing machines was one of the central debates in the non-technical presentations.

An open letter, calling for a ban on offensive autonomous weapons, signed by Stuart Russell, Nils Nilsson, Barbara Grosz, Tom Mitchell, Toby Walsh (contact person) and 1000 others was released to the press on the first day of IJCAI 2015.

Russel Norvig in his lecture asked the audience what a good AI researcher should do in a situation when the media reports the potential dangers of AI. While the researchers at the IJCAI present world-class achievements that actually move AI and humanity forward, the media remains ignorant of these true achievements and propagates the ungrounded fears proclaimed by non-AIers. His advice is to relax, as such mortal fears follow any major new technology, while the media propagates the most attractive views without understanding whether there is indeed any cause for concern. As long as the field is progressing, the salaries increase and the number of students grows, we have no reason to complain. AI research is more successful than ever, consistently finding ever better transitions from scientific research into the real world. influencing our everyday lives along the way. In his words: "We are faster and faster moving towards the greatest event in human history". That, of course, is the singularity point, where truly intelligent programs and machines will propel human civilisation into a new era.

At the panel organized by the conference chair Michael Woolridge (Oxford), the invited participants expressed rather diverging opinions. The announced panellist list of Maria Gini (Minnesota), Barbara Grosz (Harvard), Francesca Rossi (Padua), Stuart Russell (Berkeley), Manuela Veloso (CMU) with the agenda "who speaks in the name of AI" was soon distracted by the ban initiative. "We have a bewildering array of different organisations at the national and international levels representing us (AAAI, IJCAI, ECCAI, PRICAI, KR, etc.), with very little coordination or communication between them. Researchers in distinct sub-fields often work in their closed worlds, unaware of the work that is going on in other sub-fields of AI, and the development of the field is hindered by endless fragmentation…, the lack of any authoritative voice for AI creates a vacuum, where ill-informed speculation about the potential of AI is rife, and attention-seeking claims in the popular press receive unwarranted attention, with nobody in a position to speak for the field, and to give an authoritative, informed, and balanced response."

It's perhaps worth mentioning that there was no "authoritative, informed, and balanced response" from the panel as well. One panellist claimed that the fear hype is providing negative implications for AI because people will associate negative feelings with AI. Another panellist claimed that AI should not get involved in any discussions about such mass-media issues or anything that involves politics of any kind, yet, the woman issue was often reintroduced. The initiative by Russel and Toby Walsh to submit a petition to the UN to ban autonomous killing machines, i.e., without a human decision in the loop, was also greeted with several remarks. Yet, overall, the majority was not against the petition, rather the way it was presented. The IJCAI, after all, should be a democratic institution, with the participants as the voters. If a group of individuals stands out that may be because they want to draw attention to themselves – as was mentioned in one of the remarks. Another objection was that a strong scientific discussion is needed before any petition. In summary, while the debate was not convergent, the overall impression was that the majority of researchers favour the ban petition, albeit with more elaborated procedures.

Despite initial problems, the AI community is wakening up. Indeed, it seems strange that other professions play a key role regarding AI-related issues. Probably, the discussions at IJCAI prompted activities in other societies. For example, in Slovenia some scientific societies already supported the ban and voting is going on in several more. In a reasonable time, the Slovenian societies will submit their support for the ban to the national government and also to the UN.

According to the title of this editorial, the worst issues of the best IJCAI ever should also be mentioned. The purpose is clear – by highlighting potential

improvements, the next IJCAI could be even better. For one thing, the initiatives to come to an agreement about how to proceed with the ban petition and the agreement about how to propagate the voices of the AI researchers were at a rather basic level, without an actual procedure being determined. Yes, there is an agreed need, but proper formal procedures are needed as well.

Second, the organization at the hotel was peculiar at best. During the first lecture of the conference, renovation was going on in the next room, with banging and drilling, sometimes rendering the audio presentation incomprehensible. Third, the air conditioning in the rooms was kind of random – in some halls it was as cold as in the ice age and in others there was practically no air conditioning, leading to jungle conditions. That said, the Argentinian local organization proved to be very friendly and supportive.

Worst of all, due to there sometimes being as many as nine events in parallel, it was often impossible to visit the most interesting lectures. Whoever constructed such a program, putting in parallel several of the most relevant presentations, would certainly need to reconsider the scheduling next time. Or maybe an "IJCAI lemon" should be delivered with respect to this issue, in order to prevent a repetition in New York, the venue of the next IJCAI.

For those of you that are not familiar with the so-called "pig-style" events, one should attend the IJCAI cocktail party after the opening ceremony. The term comes from the analogy with pigs messing at a scarce food source, and later eating in a crowd standing and bumping into each other. The definition was fulfilled. One might argue that the same was true at the regular coffee-breaks, but there were no rows in front of the coffee and water stands. However, the lack of any food, even modest cookies, left a miserable impression, especially having in mind the unavailability of the published invited and awarded lectures and the enormous conference fee, which approached €1000 for the workshops, tutorials and the main conference. Surely, it was an enormous effort to organize together so many relevant events, but the room for improvement is quite large.

At the same time there were several improvements compared to the previous conference. For communicating AI matters to the media, the IJCAI ran a daily press conference, livestreamed to the world every day. This was a first for the IJCAI. The proceedings were published on the web two weeks in advance, the program was on schedule, and a large majority of the lectures were comprehensibly presented. The number of contributions is increasing, leading to the singularity at best, or something like that in the worst case. The IJCAI2015 had plenty of lectures to listen to and achievements to admire.

Regarding the representatives from Slovenia (remember, Informatica is also supported by the Slovenian SLAIS), there were just two issues worth mentioning. We co-organized one workshop, and at an ambient-intelligence (AmI) tutorial Juan Augusto described the PhD of Hristijan Gjoreski as one of the major events in ambient intelligence. Gjoreski designed an AmI-related version of a general random forest algorithm, improving the accuracy at statistical levels, at worst, and enormously for some demanding AmI tasks. The reason why Slovenia is lagging behind its usual achievements at the IJCAI is that we still attend the ECCAI because of European relations (to see and communicate with Muggleton, de Raedt, de Mantaras and other key scientists and institutions in the EU), while the national evaluation procedures render all IJCAI achievements (papers and their citations) as practically irrelevant. A couple of years before it was possible to publish a paper in a good journal from the IJCAI conference, but now the conference publication itself prevents similar journal publication, and in the national evaluation system a major achievement gets no acknowledgement. Hopefully, publications like this will help change bureaucratic irregularities.

In a conclusion, while AI is no doubt a current and long-time success story and a future hope for humanity, there is a lot of space for improvements, e.g., by organizing procedures to transmit AI opinion to the media. In all scenarios, staying open-minded, democratic and first of all true scientists and developers, it is up to the AI society to fulfil its own prophecy and not the mass media's fears. It should be a classic case of a self-fulfilling prophecy!

Fast Heuristics for Large Instances of the Euclidean Bounded Diameter Minimum Spanning Tree Problem

C. Patvardhan and V. Prem Prakash
Faculty of Engineering, Dayalbagh Educational Institute, Agra - 282005, India
E-mail: cpatvardhan@gmail.com, vpremprakash@acm.org

A. Srivastav
Institut für Informatik, Christian-Albrechts-Universität zu Kiel, Kiel, Germany
E-mail: asr@informatik.uni-kiel.de

Keywords: heuristic, euclidean, bounded diameter minimum spanning tree, constrained diameter, greedy

Received: January 29, 2015

Given a connected, undirected graph G = (V, E) on n = $|V|$ vertices, an integer bound D $\geq$ 2 and non-zero edge weights associated with each edge e $\in$ E, a bounded diameter minimum spanning tree (BDMST) on G is defined as a spanning tree T $\subseteq$ E on G of minimum edge cost w(T) = $\sum$ w(e), $\forall$ e $\in$ T and tree diameter no greater than D. The Euclidean BDMST Problem aims to find the minimum cost BDMST on graphs whose vertices are points in Euclidean space and whose edge weights are the Euclidean distances between the corresponding vertices. The problem of computing BDMSTs is known to be NP-Hard for $4 \leq D < n-1$, where D the diameter bound. Furthermore, the problem is known to be hard to approximate. Heuristics are extant in the literature which build low cost, diameter-constrained spanning trees in $O(n^3)$ time. This paper presents some fast and effective heuristic strategies for the Euclidean BDMST Problem and compares their performance with that of the best known existing heuristics. Two of the proposed heuristics run in $O(n^2\sqrt{n})$ time and another faster heuristic runs in $O(n^2)$, thereby allowing them to quickly build low cost BDSTs on larger sized problems than have been attempted hitherto. The proposed heuristics are shown to perform better over a wide range of benchmark instances used in the literature for the Euclidean BDMST Problem. Further, a new test suite of much larger problem sizes than attempted hitherto in the literature is designed and results presented.

Povzetek: Podan je hevristični postopek za hitro gradnjo minimalnega prekrivnega drevesa.

1 Introduction

Given a connected, weighted, undirected graph G and a positive integer D, the Bounded-Diameter Minimum Spanning Tree (BDMST) problem seeks a low cost spanning tree from amongst all spanning trees of G containing paths with no more than D edges. Formally, a bounded-diameter spanning tree (BDST) is a spanning tree T $\subseteq$ E on G = (V, E), whose diameter is no greater than D. The BDMST problem aims to find a bounded diameter spanning tree of minimum cost w(T) = $\sum$ w(e), $\forall\ e \in T$. Restricting the problem to Euclidean graphs (where vertices are points in Euclidean space and edge weights are the Euclidean distances between pairs of vertices) gives rise to the Euclidean BDMST Problem [1]. The problem finds application in several domains, ranging from distributed mutual exclusion [2] to wire-based communication network design [3] and data compression for information retrieval [4]. An important application also occurs in reducing the source-sink delays and total wire length in VLSI routing. Barring the special cases of D = 2, D = 3, D = $n-1$, and all edge weights being the same, the BDMST Problem is known to be NP-Hard [5]. Furthermore, the problem is also known to be hard to approximate; it has been shown that no polynomial time approximation algorithm can be guaranteed to find a solution whose cost is within log (n) of the optimum [6].

An exact algorithm for the BDMST Problem is given by Achuthan and Caccetta in [7]. This is improved by Achutan et al. [8], wherein a branch and bound framework is given which utilizes different branching rules and simple heuristics. Gouveia and Magnanti [9], give several variants of multi-commodity flow (MCF) formulations for the BDMST problem which achieve extremely tight LP bounds (within 1% of the optimal solution for almost all benchmarks tested). However, this approach has been able to solve BDMST instances of up to only 60 node graphs to optimality. In general, the exponential time complexity of exact algorithms allows them to solve only very small problem instances; this motivates the search for fast heuristics and meta-heuristic techniques which can approximate low cost BDSTs on much larger problem sizes within reasonable time.

Several meta-heuristics are given in the literature that evolve BDMSTs on larger problem instances, including an ant colony algorithm [18], evolutionary algorithms

[20],[21],[22] and a recent learning automata-based algorithm [23].

Many of the best known existing heuristics for the BDMST problem are based on a greedy, Prim's [19] algorithm-like strategy. Each of these heuristics works well under certain conditions in the Euclidean BDMST case, for instance when the range of diameter bounds is restricted to a small range or when the diameter bound is very small. Further, none of the existing heuristics is suitable for working on very large sized problems, as they require too much computation time for building BDSTs on large problem instances. A key goal of the research presented in this paper has thus been to develop fast and robust heuristics that would build low cost BDSTs on very large problem sizes. The extant heuristics in the literature are briefly discussed as follows.

Ayman Abdalla and Narsing Deo describe in [10], several construction heuristics for the BDMST problem, including a Prim's [19] algorithm–based heuristic called the one-time tree construction (OTTC) heuristic that runs in O(n^4) time and produces low cost BDSTs when the diameter bound D is small. Abdalla and Deo also give two iterative refinement algorithms that start with an unconstrained MST and iteratively decrease the length of long paths until the diameter constraint is satisfied. The center-based tree construction (CBTC) heuristic given by Julstrom in [11] performs better than OTTC both in terms of solution quality and running time (it requires O(n) time less than OTTC) by constructing the BDST as a height-restricted tree rooted at a central node (or two nodes if the diameter is odd). A randomized tree construction heuristic (RTC) is also given in [11] wherein each next node to be appended to the BDST is chosen at random and appended greedily.

The RTC and CBTC heuristics are improved further by Singh and Gupta [12] and Singh and Saxena [13]. The improved heuristics given in [12] are called RGH-I and CBTC-I, which try to improve RTC and CBTC in terms of both speed and solution quality. In particular, for each vertex v (excluding the root and vertices adjacent to the root) the heuristics search for tree vertices of lower depth than v to which v can be appended at lesser cost. Singh and Saxena [13] improve these heuristics further and demonstrate their effectiveness on a standard set of problem instances used widely in the literature.

A hierarchical clustering-based heuristic for the Euclidean BDMST problem is given by Gruber and Raidl [14], which obtains low cost BDSTs when the diameter constraint is very small.

The Center-based Least Sum-of-Costs (CBLSoC) heuristic given by Patvardhan et al.[15] builds a low cost BDST by repeatedly appending the non-tree vertex with the lowest mean cost to all the remaining non-tree vertices in the graph. This heuristic is run starting from every graph vertex and returns the lowest cost BDST obtained. It has a running time of O(n^3) and performs competitively vis-a-vis the other heuristics. Parallel versions of the CBTC, RTC and CBLSoC heuristics are given in Patvardhan et al.[16] and their performance compared over a comprehensive set of Euclidean and random benchmark graphs.

This paper presents some fast heuristics for the Euclidean BDMST problem. The first of these is a variant of the CBLSoC heuristic [15]. This heuristic, called CBLSoC-lite, produces BDSTs with comparable/better (lower) costs as compared to existing heuristics on a wide range of benchmarks. Two other "Quadrant-Centers based" heuristics try to construct an effective backbone of a small number of low height nodes appended to the tree via relatively longer edges, and then build the rest of the BDST. The heuristics presented in this work typically take less time to build a low cost BDST vis-à-vis extant heuristics. This allows them to handle much larger problem sizes than attempted hitherto by any other heuristic. Their performance is demonstrated on a test suite of completely connected Euclidean graphs having up to 10,000 vertices.

Subsequent sections of the paper are organized as follows: section 2 discusses three well known heuristics for the problem (OTTC, CBTC and RTC), section 3 describes CBLSoC and the proposed CBLSoC-lite heuristic, and Section 4 presents the proposed "Quadrant centers-based" heuristic strategy. Computational results obtained on benchmark problem instances and other larger randomly generated graphs are presented and summarized in section 5, and concluding remarks are made in section 6.

2 Some well known heuristics

This section presents several well known heuristics for the BDMST Problem and summarizes their key characteristics.

2.1 One-time Tree Construction (OTTC) Heuristic

One-Time Tree Construction (OTTC) given by Abdalla et al. [10] is a greedy heuristic that computes the diameter of the spanning tree at each step and ensures that the incoming vertex does not violate the diameter constraint. In order to obtain a low cost BDST, the OTTC algorithm is run starting from every vertex of the graph. For each vertex that it starts from, OTTC repeatedly appends to the growing MST the lowest-cost edge that appends a new vertex to the tree without violating the diameter bound. Adding each new vertex also involves updating the path lengths and eccentricities of the tree vertices, requiring at most O(n^2) time. This is done n-1 times in each run, so the algorithm has a total running time of O(n^3). As the algorithm is run starting once from each graph vertex (i.e., totally n times), the total time complexity is O(n^4). Pseudo-code for this heuristic is given in listing 1.

Listing 1: OTTC heuristic

```
V[.] ← set of graph vertices
S[.] ← set of tree vertices, initially empty
for each v ∈ V do
    T ← { }
    S ← { }
    S ← S ∪ {v}
    while |T| ≠ (n − 1) do
        Choose x ∈ V \ S and u ∈ S such that
        cost(u, x) is minimal ∀ u ∈ S and diameter
        constraint D is not violated
        T ← T ∪ (u, x)
        S ← S ∪ {x}
    if cost(T) < cost(bestTree) then
        bestTree ← T
return bestTree
```

Listing 2: CBTC heuristic

```
U[.] ← set of unconnected graph vertices
C[.] ← set of tree nodes with depth < ⌊D/2⌋ for each
vertex v0 ∈ V do
    Set v0 as root
    U ← U \ {v0}
    C ← {v0}
    depth[v0] = 0
    if D is odd then
        Choose vertex v1 ∈ U such that cost(v0, v1) is
        minimal
        U ← U \ {v1}
        C ← {v1}
        depth[v1] = 0
        T ← T ∪ (v0, v1)
    while U ≠ {} do
        Find u ∈ C and v ∈ U such that cost(u,v) is
        minimal
        T ← T ∪ (u, v)
        U ← U \ {v}
        depth[v] ← depth[u] + 1
        if (depth[v]<⌊D/2⌋) then
            C ← C ∪ {v}
return the tree with lowest cost out of all trees
generated above
```

2.2 Center-based tree construction (CBTC) heuristic

In a tree with diameter D, no vertex is more than D/2 hops or edges from the root vertex of the tree [17]. This motivates a faster Prim's algorithm-based heuristic called Center-Based Tree Construction (CBTC) heuristic [11], which improves OTTC by building the BDST from the tree's center, keeping track of the depth of each tree node and ensuring that no node depth exceeds $\lfloor D/2 \rfloor$. This heuristic avoids the task of repeatedly computing the tree diameter before appending each node, and returns the lowest cost BDST obtained over n runs, each starting from a different graph vertex, thereby bringing down the total running time to O(n^3). Pseudo-code for the CBTC heuristic is given in listing 2.

2.3 Randomized tree construction (RTC) heuristic

In the Randomized Tree Construction Heuristic (RTC), the center of the spanning tree is chosen at the outset as one vertex (if D is even) or two connected vertices (if D is odd) randomly selected from the set of graph nodes. Each next vertex is then chosen at random and connected to the tree greedily such that the inclusion does not yield a tree of diameter greater than the diameter bound D. Building the BDST requires repeating this process through $n - 1$ iterations. As before, this process is repeated n times, and the lowest cost BDST is returned. Hence the total running time of this heuristic is O(n^3). Pseudo-code for this heuristic is given in listing 3.

2.4 Some Other Heuristics

Singh and Gupta [12] improve the CBTC and RTC heuristics in two ways. Firstly, after building a BDST using either construction heuristic, it is checked for each vertex v∈V in the BDST whose depth is greater than 1 (essentially covering all vertices that are not either the root(s) or the vertices immediately connected to the root(s)) whether it can be reattached to a BDST vertex of depth less than depth[v] via a lower cost edge. Secondly, the heuristics maintain a sorted cost matrix and searching for a low cost edge to append a vertex v to the BDST in the $n/10$ elements of the cost matrix row entry for the vertex v. These two heuristics are further improved in Singh and Saxena [13], where they are called RGH+HT and CBTC+HT respectively, by allowing a sub-tree rooted at v to be connected to any vertex of the tree irrespective of its depth, provided the cost is reduced and the feasibility of the resulting BDST is retained. Gruber and Raidl [14] use agglomerative hierarchical clustering to guide the creation of an effective BDST backbone and transform the resulting dendogram structure into a height-restricted clustering that satisfies the diameter constraint. The heuristic then uses either a greedy heuristic or one of two dynamic programming (DP) approaches to identify a good root node within each cluster. The first DP approach restricts the search space of root nodes of a cluster to the root nodes of sub-clusters, while the second approximates optimal cluster roots using a correction value for estimating the cost of connecting each graph vertex as a leaf node of the BDST. The dynamic programming approaches take $O(H.|V|^2)$ and $O(|V|.|E| + H.|V|^2)$ time, when D is even and odd respectively, for computing the roots of clusters, where $H - 1$ is the tree height. The re-

Listing 3: RTC heuristic

```
1  U[.] ← set of unconnected graph vertices
2  C[.] ← set of tree nodes with depth < ⌊D/2⌋ for n
   times do
3      Set as root, a random vertex v_0 ∈ U
4      U ← U \ {v_0}
5      C ← {v_0}
6      depth[v_0] = 0
7      if D is odd then
8          Choose another random vertex v_1 ∈ U
9          U ← U \ {v_1}
10         C ← C ∪ {v_1}
11         depth[v_1] = 0
12         T ← T ∪ (v_0, v_1)
13     while U ≠ {} do
14         Choose a random vertex v ∈ U
15         Find u ∈ C such that cost(u,v) is minimal
16         T ← T ∪ (u, v)
17         U ← U \ {v}
18         depth[v] ← depth[u] + 1
19         if (depth[v]<⌊D/2⌋) then
20             C ← C ∪ {v}
21 return the tree with lowest cost out of all trees
   generated above
```

sults of the heuristic are further improved using a variable neighborhood descent (VND) given in [18].

2.5 Discussion

A major drawback of the OTTC and CBTC heuristics is that they always use low cost edges to build the tree, thus necessitating the later vertices to be appended to the tree through higher cost edges. This often results in BDSTs with larger costs, especially when the diameter bound is small. One way to overcome this is to select each node randomly and then append it to the tree greedily, as is done by the RTC heuristic. Possibly due to the random nature of node selection, the initial "backbone" of the BDST often turns out better in RTC than in the OTTC and CBTC heuristics, leading to lower cost BDSTs when the diameter bounds are small. However, the performance of the RTC heuristic rapidly deteriorates as the diameter bound is increased, as discussed in section 5. The heuristics of Singh and Saxena [13] produce lower cost BDSTs than CBTC and RTC. The Hierarchical Clustering-based heuristic produces very low cost BDSTs, but is seen to be effective only when diameter bounds are small.

3 The CBLSoC-lite and CBLSoC Heuristics

The Center-based Least Sum-of-Costs *Lite* (CBLSoC-lite) heuristic starts by selecting as root the vertex (or two vertices, in case of odd diameter) with lowest mean cost to all other graph vertices. Thereafter, each new graph vertex selected has the lowest sum of costs to all the remaining graph vertices. This vertex is then appended to the tree greedily via the lowest cost edge that does not violate the diameter bound. The heuristic uses a center-based approach, building the BDST from the tree's center, keeping track of the depth of each tree node and ensuring that no node depth exceeds $\lfloor D/2 \rfloor$. This preempts the need for dynamically computing the spanning tree's diameter at each step and results in a total computational time of $O(n^2)$ for the heuristic. Pseudo-code for the heuristic is given in listing 4. The CBLSoC heuristic iteratively builds BDSTs starting once from each graph vertex and returns the lowest cost BDST thus obtained. This results in further improvements in BDST costs, but incurs an additional overhead of O(n), and hence a total running time of O(n^3) for the heuristic.

Listing 4: CBLSOC-LITE heuristic

```
1  U[.] ← Adjacency matrix containing edge weights of
   the graph G
2  C[.] ← Set of unconnected graph vertices
3  Choose u∈V such that sum of entries in row u of adj.
   matrix A is minimal
4  U ← U \ {u}
5  if D is odd then
6      choose v ∈ V \ {u} such that sum of entries in row
       v of A is minimal
7  T ← T ∪ (u,v) U ← U\{v}
8  while U ≠ {} do
9      Choose x∈V \ T such that ∑ cost(x,y) is minimal
       ∀ y∈V\T,y≠x and diameter constraint is not
       violated
10     Choose u such that cost (u, x) is minimal, ∀ u ∈ T
11     T ← T ∪ (u, x)
12     U ← U \ {x}
13 return T
```

The CBLoC-lite and CBLSoC heuristics produce better (lower) cost BDSTs in comparison to the OTTC, CBTC and RTC heuristics, as shown in section 5 on a large number of benchmark problems.

4 *Quadrant Centers*-based Heuristics

As discussed in section 2, the greediness inherent in the OTTC and CBTC heuristics causes the backbone of the growing BDST to be typically constituted of short edges, thus forcing several nodes to be appended using long edges and thereby increasing the total cost. The relatively less

greedy, center-based LSoC heuristics discussed in section 3 mitigate this problem to some extent, as shown in the experimental results presented in section 5.

Another group of heuristics is proposed which start by empirically fixing the tree center and adding a few graph vertices to the tree, thereby building a backbone comprising of a small number of nodes connected to the tree via relatively longer edges. The remaining nodes are then appended to the BDST either greedily or by using the CBLSoC heuristic.

Listing 5: QCH-GREEDY heuristic

```
1  Choose v_0 ∈ V | { ∑ cost(v_0, x) is minimal ∀ x ∈ V, x ≠ v_0 }
2  U ← {v_0}
3  if D is odd then
4      Choose v_1 ∈ U | {cost(v_0, v_1) is minimal, ∀ v_1 ∈ U }
5      T ← T ∪ (v_0, v_1)
6      U ← U ∪ {v_1}
7  for qsize = 2 to √n do
8      for i = 1 to qsize do
9          Choose p ∈ quadrant i|{ ∑ cost(p,j) is minimal, where p, j ∈ U, j ∈ quadrant i and j ≠ p }
10         T ← T ∪ (k, p), where cost(k, p) is minimal of all tree vertices k
11         U ← U \ {p}
12     Append each remaining node x∈U to T greedily via the lowest cost edge such that depth[x] ≤ ⌊ D/2 ⌋
13     for each vertex i ∈ T do
14         ∀ j ∈ T such that depth[j] < ⌊ D/2 ⌋, if cost(i,j) < cost (i, parent[i]), replace (i,parent[i]) with (i,j) in the BDST
15     bestT ← lowest cost BDST found so far
16 return bestT
```

The Euclidean problem domain is widely modeled in the literature as a set of real points normally distributed in the unit square. Using this model, the proposed heuristics build a tree "backbone" by first choosing the root node of the BDST using the CBLSoC heuristic. Specifically, the node with the lowest mean cost to all other graph nodes is added to the BDST and chosen as the central, or root vertex. If (the diameter) D is even, then this single node serves as the root of the BDST. If D is odd, then the non-tree node with the lowest mean cost to the remaining graph vertices is also selected and added to the tree via the lowest cost edge; the sub-graph comprising these two nodes and the edge joining them is now considered as the center of the BDST. The remaining graph vertices are segregated into the "quadrants" of a uniform MxM matrix in the unit square, $2 \leq M \leq \sqrt{N}$ (each element of this matrix is termed a quadrant, for want of a more suitable expression). Within each quadrant, the node with lowest mean cost to all other nodes within the same quadrant is set as a tree backbone node of depth 1. The remaining vertices are then appended to the tree either greedily (this is called the QCH-Greedy heuristic) or using the CBLSoC heuristic in selecting each next vertex to append to the tree (this is called QCH-LSoC heuristic), while ensuring that the diameter constraint is not violated.

Listing 6: QCH-LSOC heuristic

```
1  Choose v_0 ∈ V | { ∑ cost(v_0, x) is minimal ∀ x ∈ V, x ≠ v_0 }
2  U ← {v_0}
3  if D is odd then
4      Choose v_1 ∈ U | {cost(v_0, v_1) is minimal, ∀ v_1 ∈ U}
5      T ← T ∪ (v_0, v_1)
6      U ← U ∪ {v_1}
7  for qsize = 2 to √n do
8      for i = 1 to qsize do
9          Choose p ∈ quadrant i|{ ∑ cost(p,j) is minimal, where p, j ∈ U, j ∈ quadrant i and j ≠ p }
10         T ← T ∪ (k, p), where cost(k, p) is minimal of all tree vertices k
11         U ← U \ {p}
12     for remaining vertices in U do
13         Choose k ∈ U | { ∑ cost(k,j) ∀ j∈U, j≠k} is minimal
14         T ← T ∪ (k,x), where (k,x) is the lowest cost edge s.t. depth[k] ≤ ⌊ D/2 ⌋
15         U ← U \ {p}
16     for each vertex i ∈ T do
17         ∀ j ∈ T such that depth[j] < ⌊ D/2 ⌋, if cost(i,j) < cost (i, parent[i]), replace (i,parent[i]) with (i,j) in the BDST
18     bestT ← lowest cost BDST found so far
19 return bestT
```

Pseudo-code for these two heuristics is given in listings 5 and 6 respectively.

Both the heuristics attempt to find an effective backbone by varying the number of quadrants up to n (where n is the input size) and building the backbone accordingly. The heuristics return the lowest cost BDST obtained by this procedure.

Setting up the backbone of the BDST requires O(n) time in the greedy QCH heuristic (QCH-Greedy) and $O(n^2)$ time in the CBLSoC-based variant (QCH-LSoC). In the greedy variant, the process of greedily appending each node to the BDST requires O(n) time, resulting in a total running time of $O(n^2)$. Running the heuristic up to $\sqrt{n}$ times in the worst case gives a total running time of $O(n^2\sqrt{n})$.

In the LSoC-based variant, identifying the non-tree node with the lowest mean cost to all other non-tree nodes can

be achieved in O(n) time when we keep track of the mean costs from each such node to all other such nodes and update in linear time, and appending it greedily to the BDST would also require linear time. Thus the total running time for the LSoC-based QCH heuristic is also O($n^2\sqrt{n}$). In practice, the heuristics terminate when there is no further improvement in cost as compared to the previous iteration. Both heuristics attempt to reattach each node of the BDST (excepting the root nodes) at a lower cost, if possible, in a simple post-processing step that requires O(n^2) time, which does not result in any change in the overall time complexity, and slightly improves the tree cost in several cases.

5 Comparison of performance on benchmarks

The Euclidean Steiner Problem data sets given in Beasley's OR-Library[1] have been used extensively in the literature for benchmarking heuristics and algorithms for the BDMST Problem. These instances comprise of vertices placed at random in the unit square, fifteen instances of each size for graphs of up to 1000 vertices. Julstrom [9] uses an enhanced test suite of Euclidean problem instances that augments the OR-Library instances with randomly generated Euclidean graphs, fifteen each of 100, 250, 500 and 1000 vertices, whose edge weights are, as before, the Euclidean distances between (randomly generated) points in the unit square. Another test suite of larger Euclidean problem instances comprising of thirty randomly generated Euclidean graphs of 1500, 2500, 5000 and 10,000 vertices was developed by the authors for comparing the performance of the heuristics presented in this work on larger problem instances. These problems are referred to in this paper as large problem instances, to differentiate from the enhanced test suite of standard sized problems given by Julstrom [9].

The heuristics presented in this paper were tested on the thirty "standard" problem instances of 100, 250, 500 and 1000 vertex graphs provided in the enhanced benchmark suite of [9], totaling 120 completely connected Euclidean graphs, and the mean (X) and standard deviation (SD) of tree costs, and mean CPU times (t_{avg}) were obtained for each node size. The results obtained on the enhanced benchmark suite of standard problem instances for the OTTC, CBTC, RTC, CBLSoC-lite, CBLSoC and proposed QCH heuristics are given in table 1.

The heuristics were also tested on the thirty "large" problem instances of 1500, 2500, 5000 and 10000 vertex graphs; the mean and standard deviation of tree costs and mean CPU times obtained for all the heuristics are given in table 2. Results for the OTTC heuristic were not computed for larger sized problems because it takes too much computational time, as is obvious from the times given in table 1 for 1000 vertex graphs. In any case OTTC always performs worse than the CBTC heuristic.

The values used for D (the diameter bound) in all the tests were always less than the smallest diameter of an unconstrained MST on each set of graphs. The mean CPU times quoted in table 1 for the OTTC heuristic were obtained in [9] on a Pentium IV, 2.53 GHz processor with 1 GB memory. All the other heuristics were implemented in C on a Dell Precision T-5500 Workstation with 12 Intel Xeon 2.4-Gigahertz processor cores and 11 GB RAM running Red Hat Enterprise Linux 6.

The proposed heuristics were compared in terms of lowest and mean BDST costs obtained and computation time vis-a-vis the improved heuristics of Singh et al. [13] on Euclidean problem instances in table 3 and the hierarchical clustering-based heuristics of Gruber and Raidl [1] in table 4.

The mean BDST costs for the CBTC, RTC, CBLSOC-lite and CBLSoC heuristics given in table 1 show that the CBLSoC-lite and CBLSoC heuristics outperform OTTC on all problem instances and produce lower mean costs vis-à-vis the CBTC heuristic on most instances.

The RTC heuristic produces relatively lower cost trees, but this is only when the diameter bound is very small; as the diameter bound is increased the lowest cost BDSTs are the ones produced by CBLSoC-lite and CBLSoC. In order to understand this behavior, we observe that the OTTC and CBTC heuristics always greedily append to the tree, the node with the lowest cost to the tree. As a result the tree backbone ends up comprising of a small number of relatively short edges, forcing many of the remaining graph vertices to be appended via longer edges in order to maintain the diameter bound, resulting in higher tree costs. In a sense, the inherent greediness of the heuristic adversely affects its performance.

The RTC heuristic, possibly due to its randomized node selection approach, has a much better chance of building a tree backbone close to clusters of nodes, several of which might then be appended to the backbone using short edges. When D is small, it usually returns trees with lower costs than any of the other heuristics (OTTC, CBTC or CBLSoC-lite). However, as the diameter bound is increased, the RTC policy of always choosing the next node to append in a random manner leads to several poor choices, thereby contributing adversely to the overall BDST cost. The heuristic fails to produce any improvements in BDST costs with as the diameter bound is relaxed and is eventually surpassed in performance by the other heuristics.

The CBLSoC-lite heuristic is relatively less greedy in the sense that the next node chosen to be appended to the tree is always the one with the lowest mean cost to all remaining nodes in the tree; this node need not necessarily be the node with the lowest cost to the tree. The performance of this heuristic, especially in terms of speedup, is significant. For instance, table 1 shows that while the OTTC and RTC average about 173 seconds and 15 seconds respectively for computing BDMSTs on the 1000 node instances,

[1]Maintained by J.E. Beasley, Department of Mathematical Sciences, Brunel University, UK. (http://people.brunel.ac.uk/ mastjjb/orlib/files)

the CBLSoC-lite heuristic takes about 0.03 seconds on average for problems of the same size, and computes BDMSTs with mean costs that are usually better. The CBLSoC heuristic takes O(n^3) time but produces lower cost BDSTs than CBLSoC-lite.

The QCH heuristics start by trying to fix up a "good" backbone for the BDST. The greedy variant, QCH-greedy incorporates the greedy selection strategy used by OTTC, CBTC and RTC; the other proposed variant use the node selection strategy followed by the CBLSoC heuristic. The QCH heuristics produce low cost BDSTs in general, as can be seen from the mean tree costs given in table 1.

The two QCH variants obtain competitive mean tree costs, with the QCH-greedy heuristic producing slightly better BDSTs on larger diameter bounds. The BDST costs obtained by the QCH heuristics are lower than that obtained by the RTC heuristic on all problem instances, more so when the diameter bounds are small. Both heuristics produce significantly lower cost BDSTs vis-à-vis CBLSoC and CBLSoC-lite when the diameter is small, and give competitive results on larger diameter bounds; the QCH variants perform better than OTTC and CBTC on all problem instances and for almost all diameter constraints.

On the large Euclidean problem instances, the computational time requirements of OTTC, CBTC, RTC and CBLSoC heuristics rapidly become prohibitively high (table 2), whereas the CBLSoC-lite and QCH heuristics are still able to quickly compute low cost BDSTs.

On 2500 vertex graphs for example, CBLSoC-lite computes lower cost BDSTs than CBTC on all except the smallest diameter bound, in less than 1/1000th of the time taken by CBTC (0.22 seconds for CBLSoC-lite versus 257.29 seconds for CBTC on the third 2500 vertex instance in table 2). On the same instance, the QCH-greedy heuristic computes the lowest cost BDST of all the heuristics in about 0.42 seconds.

RTC produces the lowest cost BDST of all the heuristics in instance each of 1500 and 2500 vertex graphs, and that too only on the smallest diameter bound. Furthermore it fails to improve tree costs as the diameter constraint is progressively increased. By contrast, CBLSoC-lite takes less than one second to build low cost BDSTs on benchmark graphs of the same size and outperforms CBTC and RTC as D is increased. Even on completely connected graphs of 10,000 vertices, the heuristic computes BDSTs in less than 5 seconds on average (the t_{avg} column in table 2).

As already observed with the standard sized Euclidean problems, the QCH heuristics obtain the lowest cost among the heuristics being compared. In particular, the QCH-LSoC heuristics almost always produce the lowest cost trees on smaller diameter bounds, with the QCH-greedy heuristic obtaining the lowest costs BDSTs on larger diameter constraints for most of the large problem instances. It is worth noting that the CBLSoC-lite and the two QCH heuristics compute low cost BDSTs in a fraction of the computation time taken by the other heuristics. This is illustrated in figure 1.

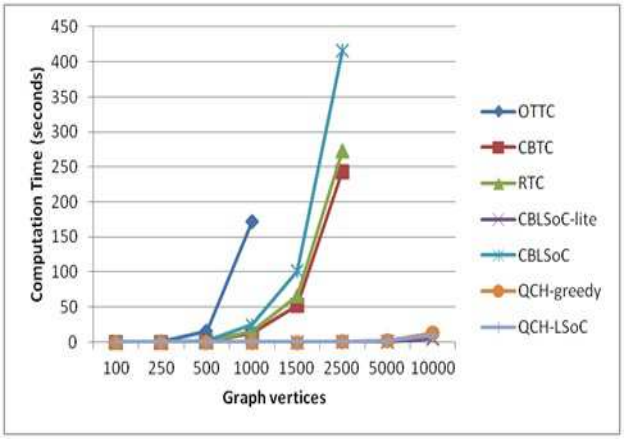


Figure 1: *Comparison of computational time taken by the heuristics*

The proposed heuristics are also compared with the improved heuristics of Singh and Saxena [13] in table 3 and with Gruber and Raidl's hierarchical clustering-based heuristic strategy in table 4. Both works provide computational results for small diameter bounds only on problem instances of up to 1000 vertex graphs.

Singh and Saxena [13] give the results obtained by their improved heuristics on the first five instances of the Beasley Euclidean Steiner Problem data sets for 50, 100, 250, 500 and 1000 vertex graphs, with diameter bounds of 5, 10, 15, 20 and 25 respectively.

Table 3 gives the lowest, mean and standard deviation (as applicable) of BDST costs obtained by these two heuristics in [13] vis-à-vis the proposed heuristics on Beasley's Euclidean problems. As the table shows, the faster CBLSoC-lite heuristic outperforms the CBTC+HT heuristic on smaller sized problems (50 and 100 node instances), and running this heuristic starting from each graph vertex (the CBLSoC heuristic) produces much better BDSTs than the CBTC+HT heuristic on all problem instances. When the diameter bound is very small, the lowest cost BDSTs are returned by the RGH+HT heuristic, which improves the results obtained by the RTC heuristic on these problem instances by about 8.26% on average. However, no further results on larger diameter bounds or on larger problem sizes are given in the literature for these heuristics. Further, the RGH+HT builds on the RTC heuristic, which has already been shown to perform poorly upon increasing the diameter bound on a wide range of benchmarks (tables 1 and 2). On the other hand, the CBLSoC and QCH heuristics are quite effective on larger diameter bounds and problem instances, as already demonstrated on Julstrom's enhanced test suite and the large problem instances.

Instances		OTTC			CBTC			RTC			CBLSoC-lite			CBLSoC			QCH-Greedy			QCH-LSoC		
n	D	X	SD	t	X	SD	t	X	SD	t	X	SD	t	X	SD	t	X	SD	t	X	SD	t
100	5	29.38	1.71	0.07	26.48	1.51	0.01	15.39	0.66	0.01	27.64	1.97	**<0.0001**	22.33	1.43	0.01	**13.76**	0.49	**0.0007**	**13.76**	0.49	**0.0007**
	10	18.43	1.86	0.07	15.59	1.28	0.01	9.76	0.29	0.01	16.61	1.84	**<0.0001**	12.54	0.96	0.01	9.53	0.35	**0.0003**	**9.43**	0.37	**0.0003**
	15	12.84	1.47	0.07	10.95	1	0.01	9.23	0.27	0.01	10.45	0.99	**<0.0001**	9.15	0.57	0.01	**8.47**	0.28	**0.0003**	8.63	0.28	**0.0003**
	25	8.06	0.59	0.07	7.69	0.34	0.02	9.16	0.25	0.01	7.42	0.3	**<0.0001**	**7.32**	0.24	0.01	7.89	0.25	**0.0003**	8.21	0.25	**0.0003**
250	10	58.2	5.38	1.18	49.07	2.99	0.1	16.84	0.31	0.08	53.82	5.3	**0.0013**	30.91	1.8	0.21	16.75	0.35	**0.0037**	**16.68**	0.48	**0.004**
	15	41.59	4.03	1.22	36.44	3.15	0.1	15.32	0.22	0.11	38.22	6.46	**0.0013**	21.79	1.79	0.23	**14.58**	0.28	**0.0037**	14.92	0.27	**0.0037**
	20	32.55	3.86	1.26	26.17	2.36	0.11	15.02	0.22	0.12	24.84	3.53	**0.0013**	17.84	1.13	0.21	**13.5**	0.31	**0.003**	13.9	0.42	**0.003**
	40	14.43	2.11	1.36	12.51	0.83	0.13	14.98	0.23	0.12	11.56	0.23	**0.0013**	**11.47**	0.2	0.2	12.17	0.38	**0.0027**	12.57	0.2	**0.0027**
500	15	106.87	5.03	12.69	94.38	5.56	0.82	22.21	0.33	0.85	100.67	6.51	**0.0053**	44.39	3.09	2.03	**22.64**	0.48	**0.0173**	22.7	0.51	**0.018**
	30	58.52	7.1	14.65	46.51	4.13	0.94	21.42	0.34	1.21	40.31	6.13	**0.006**	25.33	2.01	2.02	**18.23**	0.43	**0.013**	18.95	0.61	**0.0123**
	45	32.23	5.66	16.66	25.63	2.18	1.04	21.45	0.36	1.22	19.2	1.45	**0.0057**	17.8	0.78	2.19	**16.7**	0.48	**0.011**	18.02	0.69	**0.0117**
	60	20.33	3.46	17.64	17.72	0.92	1.16	21.42	0.34	1.03	16.13	0.32	**0.0053**	**16.03**	0.29	1.97	16.31	0.45	**0.011**	17.21	0.3	**0.0113**
1000	20	217.71	9.48	150.06	195.96	7.97	10.71	31.2	0.24	13.55	206.63	13.02	**0.0283**	51.65	1.7	23.66	**30.45**	0.52	**0.0797**	31.32	0.43	**0.0837**
	40	124.21	17.33	167.91	99.57	7.86	11.56	30.81	0.26	14.36	84.21	8.48	**0.031**	33.22	2.93	24.49	**25.23**	0.48	**0.0657**	26.51	0.64	**0.0503**
	60	69.83	12.2	183.21	50.97	5.24	12.94	30.81	0.26	16.06	31.72	3	**0.0323**	27.71	1.68	25.3	**23.49**	0.62	**0.053**	25.78	0.86	**0.0577**
	100	28.95	4.14	189.33	23.41	0.78	13.7	30.81	0.26	16.68	22.59	0.19	**0.0293**	**22.57**	0.19	24.61	22.59	0.5	**0.0527**	23.85	0.2	**0.0463**

Table 1: Results obtained on standard benchmark instances, thirty each of 100, 250, 500 and 1000 node graphs for the OTTC, CBTC, RTC, CBLSoC-lite, CBLSoC, QCH-Greedy and QCH-LSoC heuristics

Instances		CBTC			RTC			CBLSoC-lite			CBLSoC			QCH-Greedy			QCH-LSoC		
n	D	x	SD	t	x	SD	t	x	SD	t	x	SD	t	x	SD	t	x	SD	t
1500	30	119.73	14.19	45.03	**27.41**	0.4	58.44	235.92	50.2	**0.06**	41.6	4.53	97.71	35.19	1.3	**0.21**	33.03	1.33	**0.19**
	65	48.58	5.88	50.03	27.42	0.4	73.17	50.51	20.06	**0.06**	25.7	2.22	102.03	25.06	0.97	**0.17**	**24.64**	0.78	**0.12**
	100	30.09	2.86	56.02	27.39	0.4	60.76	23.81	2.2	**0.06**	21.73	0.69	104.28	**22.04**	0.97	**0.13**	22.71	0.73	**0.12**
	135	24.26	1.19	58.72	27.42	0.4	72.32	20.99	0.76	**0.06**	20.45	0.42	103.89	**20.84**	0.56	**0.12**	22.36	0.6	**0.11**
2500	40	182.05	18.47	208.39	**35.67**	0.38	273.07	345.22	60.71	**0.21**	50.42	5.61	395.42	43.78	2.26	**0.68**	40.71	1.85	**0.62**
	80	70.96	8.9	234.53	35.67	0.38	272.33	63.88	19.39	**0.22**	34.15	3.25	413.27	32.86	1.68	**0.52**	**31.69**	0.97	**0.47**
	120	43.31	4.63	257.29	35.67	0.38	272.53	31.98	3.68	**0.22**	29	2.05	416.9	**28.54**	1.02	**0.42**	29.65	1.42	**0.43**
	160	33.93	2.62	275.09	35.67	0.38	273.98	27.62	0.77	**0.22**	26.63	0.42	439.48	**26.69**	0.85	**0.38**	28.86	0.8	**0.42**
5000	50	-	-	-	-	-	-	800.67	138.84	**1.01**	-	-	-	63.72	1.85	**3.44**	**57.73**	2.72	**3.02**
	100	-	-	-	-	-	-	175.78	50.3	**1.07**	-	-	-	48.06	1.6	**2.72**	**45.23**	1.14	**2.47**
	150	-	-	-	-	-	-	47.35	6.22	**1.05**	-	-	-	**41.55**	1.58	**2.22**	41.76	1.39	**1.92**
	200	-	-	-	-	-	-	40.34	3.49	**1.06**	-	-	-	**38.49**	1.12	**1.9**	40.43	1.12	**1.96**
10000	60	-	-	-	-	-	-	1690.2	226.63	**4.76**	-	-	-	99.62	6.85	**15.61**	**88.74**	5.89	**12.46**
	120	-	-	-	-	-	-	404.44	139.71	**4.96**	-	-	-	71.45	5.47	**13.15**	**67.98**	2.45	**12.07**
	180	-	-	-	-	-	-	89.15	23.88	**5.01**	-	-	-	62.88	6.39	**11.31**	**60.63**	2.44	**9.76**
	240	-	-	-	-	-	-	60.88	4.44	**5.01**	-	-	-	**57.34**	2.56	**10.27**	58.17	2.93	**8.51**

Table 2: Results obtained on large Euclidean benchmark instances, thirty each of 1500, 2500, 5000 and 10000 node graphs for the CBTC, RTC, CBLSoC-lite, CBLSoC, QCH-Greedy and QCH-LSoC heuristics

Instances		CBTC+HT			CBLSoC-lite	CBLSoC			RTC			RGH+HT			QCH-Greedy	QCH-LSoC
N (D)	No.	Best	Mean	SD	Best	Best	Mean	SD	Best	Mean	SD	Best	Mean	SD	Best	Best
50 (5)	1	13.28	21.8	5.33	11.83	**10.94**	**13.34**	**1.42**	9.34	12.82	2.48	**8.53**	12.56	2.14	8.63	8.63
	2	13.19	19.23	3.73	13.79	**11.75**	**12.86**	**0.61**	8.98	11.56	1.56	8.74	11.39	1.48	**8.26**	**8.26**
	3	11.59	19.06	3.82	11.59	**10.3**	**12.01**	**0.64**	8.76	11.54	1.9	**8.28**	10.66	1.21	8.33	8.33
	4	10.78	16.79	3.65	11.29	**9.83**	**10.96**	**0.52**	7.47	10.57	1.66	**7.54**	9.8	1.52	7.93	7.93
	5	12.31	18.3	3.28	12.28	**10.55**	**12.4**	**1.13**	8.79	10.91	1.61	**8.59**	10.49	1.48	8.84	8.84
100 (10)	1	17.34	28.6	7.09	14.55	**12.54**	**14.96**	**1.18**	9.35	10.77	0.81	8.88	9.96	0.72	9.74	**8.87**
	2	14.17	26.56	6.33	17.54	**12.12**	**13.97**	**1.07**	9.41	10.8	0.81	**8.68**	10.16	1	9.76	8.87
	3	15.75	29.28	7.86	20.57	**13.5**	**17.25**	**2.58**	9.75	11.25	0.9	9.25	10.46	0.71	9.79	**9.21**
	4	14.9	28.48	7.93	18.09	**12.96**	**15.69**	**1.45**	9.55	11.03	0.89	**8.95**	10.35	0.85	9.31	9.44
	5	12.82	29.18	7.88	14.81	**12.68**	**14.36**	**1**	9.78	11.36	1.06	**9.09**	10.65	0.93	9.77	9.99
250 (15)	1	37.64	71.63	20.16	46.63	**17.7**	**34.51**	**6.2**	15.14	16.51	0.69	**14.04**	15.08	0.49	14.39	15.12
	2	28.9	74.73	19.46	30.76	**20.6**	**27.89**	**3.27**	15.2	16.33	0.67	**14.11**	14.99	0.48	14.24	14.6
	3	27.31	69.67	18.66	32.57	**22.18**	**28.36**	**3.09**	15.08	16.19	0.56	**13.8**	14.86	0.47	14.89	14.74
	4	29.42	75.44	19.86	32.97	**21.53**	**25.62**	**1.83**	15.49	16.77	0.62	**14.24**	15.38	0.48	14.68	15.19
	5	35.66	70.66	17.85	36.71	**21.63**	**29.46**	**3.7**	15.42	16.53	0.58	**14.11**	15.1	0.48	14.8	14.65
500 (20)	1	48.18	148.07	40.65	75.99	**27.73**	**50.84**	**13.53**	21.72	22.86	0.51	**19.39**	20.4	0.43	19.49	21.09
	2	60.15	146.37	40.38	63.69	**28.18**	**47.13**	**8.42**	21.46	22.52	0.46	**19.09**	20.17	0.42	19.75	20.89
	3	45.49	149.61	40.86	66.52	**33.92**	**47.88**	**8.9**	21.51	22.78	0.5	**19.42**	20.41	0.41	20.27	19.97
	4	63	148.34	40.22	68.69	**29.46**	**49.4**	**8.93**	21.82	22.85	0.47	**19.41**	20.46	0.46	19.58	21.88
	5	41.77	146.8	42.73	71.13	**28.09**	**50.97**	**13.2**	21.37	22.52	0.51	**18.86**	20.05	0.44	19.75	19.91
1000 (25)	1	90.01	321.07	84.9	158.4	**43.36**	**90.12**	**27.06**	30.97	32.19	0.41	**27.22**	28.26	0.43	29.53	29.75
	2	95.83	318.59	83.48	171.84	**41.54**	**90.73**	**33.04**	30.9	32.05	0.42	**27.08**	28.12	0.41	28.15	29.32
	3	94.02	312.7	85.72	150.01	**41.71**	**87.47**	**25.65**	30.69	31.77	0.42	**26.8**	27.83	0.4	28.85	29.03
	4	81.39	317.02	83.17	165.94	**44.75**	**90.59**	**28.04**	30.93	32.18	0.43	**27.05**	28.21	0.4	29.59	30.2
	5	70.55	318.52	81.37	167.49	**43.89**	**90.81**	**28.86**	30.85	31.93	0.42	**26.5**	27.91	0.42	27.68	28.88

Table 3: Results obtained on the Beasley Euclidean benchmarks, five instances each of 50, 100, 250, 500 and 1000 node graphs for the CBTC+HT, CBLSoC-lite, CBLSoC, RTC, RGH+HT, QCH-Greedy and QCH-LSoC heuristics

n=1000	CBTC		RTC		CdA		CdB				QCH-Greedy)			QCH-LSoC		
D	Mean	SD	mean	SD	mean	SD	mean	SD	t_{max}	(s)	mean	SD	t_{avg}	mean	SD	t_{avg}
(Even D)																
4	329.0261	6.02	146.4919	3.88	68.3241	0.72	**68.3226**	0.7	2.54	0.09	68.65	0.53	0.11	68.64	0.53	0.1547
6	306.2655	9.02	80.8636	2.4	47.4045	4.85	**47.1702**	4.61	4.55	0.49	54.17	0.57	0.0973	50.13	0.55	0.1247
8	288.3842	7.52	53.2535	1.33	37.0706	1.35	**36.9408**	1.34	5.92	0.42	46.3	0.65	0.088	42.94	0.53	0.108
10	266.3665	9.01	41.1201	0.68	33.546	0.67	**33.3408**	0.66	6.79	0.42	41.4	0.59	0.0933	39.28	0.26	0.104
12	250.0016	8.01	35.759	0.47	32.2571	0.48	**31.9561**	0.44	7.11	0.33	37.81	0.63	0.0767	36.68	0.4	0.1027
14	237.1403	6.28	33.3644	0.3	31.379	0.37	**31.0176**	0.33	7	0.64	35.36	0.47	0.0767	34.85	0.55	0.0987
16	224.3123	5.72	32.1965	0.24	30.7937	0.33	**30.4287**	0.29	7.2	0.72	33.22	0.42	0.074	33.4	0.67	0.0887
18	210.9872	7.63	31.5826	0.24	30.5182	0.29	**30.1348**	0.27	7.32	0.81	31.8	0.46	0.066	32.28	0.34	0.094
20	197.1772	7.99	31.2682	0.22	30.3116	0.31	30.0384	0.28	7.57	0.76	**30.41**	0.46	0.0647	31.27	0.43	0.084
22	183.0157	8.03	31.0864	0.22	30.2344	0.3	30.0739	0.28	8.56	0.98	**29.48**	0.35	0.0653	30.66	0.49	0.0747
24	172.8251	10.59	30.9921	0.23	30.0202	0.23	30.1603	0.27	8.28	1.41	**28.99**	0.53	0.066	29.92	0.55	0.072
(Odd D)																
5	241.3032	5.09	117.3238	2.22	62.2867	0.76	**62.0646**	0.67	24.59	2.02	68.08	0.43	0.1113	68.08	0.44	0.156
7	222.1441	4.5	67.7577	1.31	46.7291	3.92	**46.4112**	3.73	27.94	1.79	53.76	0.58	0.098	49.81	0.55	0.1267
9	204.6141	6	47.3168	0.85	37.0224	1.25	**36.8904**	1.27	18.27	1.68	46.04	0.67	0.0953	42.64	0.48	0.1093
11	189.7513	4.62	38.4754	0.5	33.414	0.7	**33.1749**	0.66	13.97	0.71	41.16	0.58	0.088	39.13	0.36	0.106
13	175.7382	4.23	34.5154	0.32	32.1094	0.43	**31.8041**	0.41	12.79	1.17	37.58	0.65	0.08	36.42	0.36	0.106
15	163.1926	4.31	32.7069	0.25	31.2654	0.35	**30.8941**	0.32	11.03	1.27	35.16	0.52	0.0767	34.68	0.53	0.0973
17	149.9852	5.14	31.8467	0.23	30.7699	0.33	**30.3664**	0.3	8.93	0.94	33.06	0.43	0.0727	33.22	0.64	0.09
19	139.973	4.32	31.4048	0.21	30.535	0.29	**30.0837**	0.27	7.91	1.08	31.67	0.49	0.0707	32.13	0.38	0.0853
21	128.183	4.9	31.1697	0.23	30.3017	0.3	**30.0384**	0.27	7.6	0.71	30.25	0.44	0.0647	31.12	0.34	0.0853
23	119.5551	4.46	31.0421	0.22	30.0627	0.24	30.1166	0.31	6.96	0.81	**29.34**	0.36	0.0653	30.59	0.52	0.08
25	110.6725	4.39	30.9772	0.23	29.945	0.21	30.1393	0.24	6.68	0.89	**28.87**	0.54	0.0713	29.78	0.52	0.0707

Table 4: Results obtained on the Beasley Euclidean benchmark instances: thirty independent runs on 1000 node graphs for CBTC, RTC, the Cluster-based Heuristic with two different cluster root assignment strategies (Cd^A and Cd^B), one run each of QCH-Greedy and QCH-LSoC heuristics

The QCH heuristics produce low cost BDSTs when the diameter bound is small, giving results that are competitive with RTC+HT. For example, on the results obtained for up to 1000 vertex graphs in table 3, the lower cost BDST of the two QCH Heuristics is no worse than 3.25% on average, with the QCH-Greedy and QCH-LSoC heuristics producing slightly lower cost BDSTs than RGH+HT on three instances. The QCH heuristics also do well when D is increased, cf. tables 1 and 2.

Gruber and Raidl [1] present the results obtained for their hierarchical clustering based heuristic on very small diameter bounds, averaged over all fifteen instances of 1000 vertex graphs from Beasley's Euclidean Steiner data set. Table 4 gives the results obtained by the proposed heuristics for the same set of diameter bounds and problem instances.

The maximum running times mentioned in the table for Gruber et al.'s heuristic are given in [1] and are as such not directly comparable with the mean computational time quoted for the proposed heuristics, as they were obtained on systems with different configurations.

The mean BDST costs given for CBTC, RTC and the two variants of the Hierarchical Clustering heuristic (Cd^A and Cd^B) were obtained over thirty independent runs on the fifteen 1000 vertex Euclidean graph instances. The parameters t_{max} and [s] represent the average and corresponding standard deviation (SD) over all instances, of the maximum running time of Cd^A and Cd^B.

The mean BDST costs and SD given for the two QCH heuristics represent the mean and SD obtained from a single run of the heuristics on each instance. Gruber et al. [1] also use a variable neighborhood descent strategy to further improve the results obtained by the hierarchical clustering-based heuristic; this is shown to work well only on low values of D (for instance when D is less than 14 on the 1000 vertex graphs).

The Hierarchical Clustering heuristic outperforms the CBTC and RTC heuristics by a wide margin; with increasing D, this margin is seen to narrow down. The CBLSoC heuristic returns higher cost BDSTs as compared to the RTC heuristic on small diameter bounds (tables 1 and 2), and is hence not tested in this case. The QCH heuristics, on the other hand, still give good results, performing much better than CBTC and RTC, and, as the value of D is increased further, outperforming Cd^B on the last five diameter bounds considered in the tests (table 4). Even on very tight diameter bounds, the QCH heuristics perform well: for example, the gap in solution quality for the smallest diameter bound considered in the test (D=4) is less than 0.5%.

6 Conclusions

The Euclidean Bounded Diameter Minimum Spanning Tree problem is to find a minimum spanning tree whose diameter does not exceed a specified number of edges, in the domain of graphs whose vertices are points in two dimensional space and edges are the Euclidean distances between vertices. The problem is known to be NP-hard, and hard to approximate, which motivates the search for effective heuristic strategies that are able to quickly find low cost BDSTs. This paper presents some simple fast and effective heuristic strategies and compares their performance with that of several extant heuristics for this problem over a wide range of benchmark problems, including a test suite of very large Euclidean dense graphs. One of the proposed heuristic approaches, called CBLSoC-lite, uses a less greedy node selection policy as compared to the OTTC and CBTC heuristics and builds low cost BDSTs in time that is atleast $O(n)$ faster than any of the extant heuristics. Running this heuristic starting from each graph node and returning the lowest cost BDST so obtained requires O(n^3) time but leads to better BDSTs. The other heuristic strategy starts with an empirically fixed tree "backbone" and appends the remaining nodes using either a greedy or CBLSoC-based node selection policy.

The heuristics presented in this work are classified in figure 2 into two categories, those that work on "standard" sized problems and those that are also able to solve large problem instances in reasonable time, and then ranked in increasing order of mean tree costs obtained as the diameter bounds go from small/tight to large/relaxed. Heuristics that perform competitively in a particular range share the same rank.

Problem Size	Diameter Bound	Heuristics in increasing order of mean tree costs
Standard	Small	1. Cd^B 2. Cd^A 3. RGH+HT/RTC/QCH-Greedy 4. QCH-LSoC 5. CBLSoC 6. CBLSoC-lite 7. CBTC+HT/CBTC 8. OTTC
	Large	1. CBLSoC 2. CBLSoC-lite/QCH-Greedy 3. QCH-LSoC 4. CBTC 5. OTTC 6. RTC
Large	Small	1. QCH-LSoC/RTC 2. QCH-Greedy 3. CBLSoC (limited range due to higher computation time) 4. CBLSoC-*lite*
	Large	1. QCH-Greedy 2. QCH-LSoC 3. CBLSoC-*lite*

Figure 2: *Performance-based ranking of the heuristics*

The OTTC heuristic produces spanning trees with larger costs, because it always uses low cost edges to build the tree, thus necessitating the later vertices to be appended to the tree through higher cost edges. As computational results show, this drawback is especially obvious when the diameter bound is small. The CBTC heuristic is faster and obtains lower cost BDSTs, but it also uses the same greedy premise as OTTC and hence suffers from the same drawbacks.

The CBLSoC heuristic is shown to perform better than the OTTC and CBTC heuristics on all of the benchmark instances used in this work. The CBLSoC-lite variant, which

has a running time of O(n2), outperforms OTTC on every instance and produces lower mean costs vis-à-vis the CBTC heuristic on several instances. On problem instances with small diameter bounds, the randomized heuristic RTC outperforms the other extant heuristics, but this does not hold true when the diameter bound is increased. The improved RGH+HT and CBTC+HT heuristics are able to improve the solution quality as compared to RTC and CBTC, but they retain the drawbacks inherent in both these heuristics, thus rendering the RTC variant unsuitable for larger diameter bounds, and the CBTC variant unsuitable for low diameter bounds. The hierarchical clustering-based heuristic returns the lowest cost BDSTs on standard problems for very small diameter bounds, but its performance worsens with increasing diameter bound.

The proposed QCH heuristics compare favorably with RTC on low diameter bounds, and generally do better than all the other heuristics as the diameter constraint is relaxed. Furthermore, the lower running time requirements of the CBLSoC-lite heuristic and the QCH heuristics means that they can be used effectively for solving much larger problems than have been hitherto attempted.

Acknowledgements

The authors are extremely grateful to the Revered Chairman, Advisory Committee on Education, Dayalbagh for the continuous guidance in all their endeavors. The authors also thank Prof. Bryant A. Julstrom, St. Cloud State University, Minnesota, USA for providing the extended suite of problem instances for the BDMST problem.

References

[1] Gruber, M. and Raidl, G.R. (2009) Solving the Euclidean Bounded Diameter Minimum Spanning Tree Problem by Clustering-Based (Meta-) heuristics. *In: R. Moreno-Diaz et al., (Eds.), EUROCAST 2009, LNCS 5717*, Springer, pp. 665-672.

[2] Raymond, K. (1989) A tree-based algorithm for distributed mutual exclusion, *ACM Transactions on Computer Systems*, ACM, Vol. 7 (1), pp. 61-77.

[3] Bala, K., Petropoulos, K., Stern, T.E. (1993) Multicasting in a linear lightwave network, *IEEE INFOCOM'93*, IEEE Press, pp. 1350-1358.

[4] Bookstein, A., Klein, S.T. (1996) Compression of correlated bit-vectors, *Information Systems*, Elsevier, Vol. 16(4), pp. 110-118.

[5] Garey, M.R, Johnson, D. S. (1979) *Computers and Intractibility: A Guide to the Theory of NP-Completeness*, W.H. Freeman, New York.

[6] Kortsarz, G., Peleg, D. (1997) Approximating shallow-light trees, *Eighth ACM-SIAM Symposium on Discrete Algorithms*, pp. 103-110.

[7] Achuthan N.R. and Caccetta L. (1992) Minimum weight spanning trees with bounded diameter, *Australasian Journal of Combinatorics*, University of Queensland Press, Vol. 5, pp. 261–276.

[8] Achuthan N. R., Caccetta L., Caccetta P., and Geelen J. F. (1994) Computational methods for the diameter restricted minimum weight spanning tree problem, *Australasian Journal of Combinatorics*, University of Queensland Press, vol. 10, pp. 51-71.

[9] Gouveia L. and Magnanti T.L. (2003) Network flow models for designing diameter constrained minimum spanning and Steiner trees, *Networks*, Wiley, Vol. 41, no. 3, pp. 159–173.

[10] Deo N., and Abdalla A. (2000) Computing a Diameter-Constrained Minimum Spanning Tree in Parallel, *Italian Conference on Algorithms and Complexity*, CIAC-2000, Springer, LNCS1767, pp. 17–31.

[11] Julstrom B.A. (2009) Greedy heuristics for the bounded diameter minimum spanning tree problem, *Journal of Experimental Algorithmics*, ACM, vol. 14, no. 1, pp. 1-14, February 2009.

[12] Singh A. and Gupta A.K. (2007) Improved heuristics for the bounded-diameter minimum spanning tree problem, *Soft Computing – A Fusion of Foundations, Methodologies and Applications*, Springer, vol. 11, no. 10, pp. 911–921.

[13] Singh A. and Saxena R. (2009) Solving bounded diameter minimum spanning tree problem with improved heuristics, *ADCOM 2009*, pp. 90-95.

[14] Gruber M. and Raidl G.R. (2009) Exploiting hierarchical clustering for finding bounded diameter minimum spanning trees on Euclidean instances, *Genetic and Evolutionary Computation Conference*, Springer, Montreal, Canada, pp. 263-270.

[15] Patvardhan C. and Prakash V. P. (2009) Novel Deterministic Heuristics for Building Minimum Spanning Trees with Constrained Diameter, *Pattern Recognition and Machine Intelligence*, Springer-Verlag, LNCS 5909, pp. 68-73, December 2009.

[16] Patvardhan, C.; Prakash, V.P.; Srivastav, A. (2014) Parallel heuristics for the bounded diameter minimum spanning tree problem, *India Conference (INDICON), 2014 Annual IEEE*, IEEE Press, pp.1-5, 11-13 Dec. 2014. (doi: 10.1109/INDICON.2014.7030575)

[17] Handler, G. Y. (1978) Minimax location of a facility in an undirected graph, *Transportation Science*, INFORMS, pp. 287–293, 7, 1978.

[18] Gruber M., Van Hemert J., and Raidl G. R. (2006) Neighborhood searches for the bounded diameter minimum spanning tree problem embedded in a VNS,

EA, and ACO, *M. Keijzer et al., Eds., Proceedings of Genetic and Evolutionary Computation Conference 2006*, Springer-Verlag, vol. 2, pp. 1187-1194.

[19] Prim, R.C. (year) Shortest connection networks and some generalizations, *Bell System Technical Journal*, vol. 36, pp. 1389-1401.

[20] Raidl G.R. and Julstrom B.A. (2003) Greedy heuristics and an evolutionary algorithm for the bounded diameter minimum spanning tree problem, *ACM Symposium on Applied Computing*, ACM Press, pp. 747-752.

[21] Julstrom B.A. and Raidl G.R. (2003) A permutation-coded evolutionary algorithm for the bounded-diameter minimum spanning tree problem, *Genetic and Evolutionary Computation Conference's Workshops Proceedings, Workshop on Analysis and Design of Representations*, pp. 2–7.

[22] Binh H. and Nghia N. (2009) New multiparent recombination in genetic algorithm for solving bounded diameter minimum spanning tree problem, *First Asian Conference on Intelligent Information and Database Systems*, pp. 283-288.

[23] Javad Akbari Torkestani (2012) An adaptive heuristic for the bounded diameter minimum spanning tree problem, *Soft Computing – A Fusion of Foundations, Methodologies and Applications*, Springer, vol. 16, no. 11, pp. 1977-1988.

*MWELex – MWE Lexica of Croatian, Slovene and Serbian Extracted from Parsed Corpora

Nikola Ljubešić
University of Zagreb, Faculty of Humanities and Social Sciences, Ivana Lučića 3
E-mail: nikola.ljubesic@ffzg.hr, http://nlp.ffzg.hr/

Kaja Dobrovoljc
Trojina, Institute for Applied Slovene Studies, Dunajska 116, SI-1000 Ljubljana
E-mail: kaja.dobrovoljc@trojina.si

Darja Fišer
Faculty of Arts, Aškerčeva 2, SI-1000 Ljubljana
E-mail: darja.fiser@ff.uni-lj.si

Keywords: Slovenian, English, Croatian, multilingual lexical repository

Received: May 1, 2015

*The paper presents *MWELex, a multilingual lexical of Croatian, Slovene and Serbian multi-word expressions that were extracted from parsed corpora. The lexica were built with the custom-built DepMWEx tool which uses dependency syntactic patterns to identify MWE candidates in parse trees. The extracted MWE candidates are subsequently scored by co-occurrence and organized by headwords producing a resource of 23 to 48 thousand headwords and 3.2 to 12 million MWE candidates per language. Similarly, precision over specific syntactic patterns varies greatly, 0.167-0.859 for Croatian, 0.158-1.00 for Slovene. The possible extension of the tool is demonstrated on a simplistic distributional-based extraction of non-transparent MWEs and cross-lingual linking of the extracted lexicons.*

*Povzetek: V prispevku predstavimo večjezični leksikon *MWELex, ki vsebuje hrvaške, slovenske in srbske večbesedne zveze, ki smo jih izluščili iz skladenjsko označenih korpusov. Leksikon smo zgradili s pomočjo lastnega orodja DepMWEx, ki za prepoznavanje kandidatov večbesednih zvez v odvisnostnih drevesih uporablja odvisnostne skladenjske vzorce, jih rangira in organizira glede na jedrno besedo. Leksikon vsebuje med 23 in 48 jedrnih besed in med 3,2 in 12 milijonov večbesednih zvez. Možnosti razširitve orodja pokažemo s pomočjo preprostega, na načelih distribucijske semantike temelječega luščenja večjezičnih netransparentnih večbesednih zvez iz izluščenega večjezičnega leksikona.*

1 Introduction

Multiword expressions (MWEs) are an important part of the lexicon of a language. There are various estimates on the number and therefore importance of MWEs in languages, but most claims point to the direction that the number of MWEs in a speaker's lexicon is of the same order of magnitude as the number of single words [Baldwin and Kim, 2010].

There are two basic approaches to identifying MWEs in corpora: the symbolic approach, which relies on describing MWEs through patterns on various grammatical levels, and the statistical approach, which relies on co-occurrence statistics [Sag et al., 2001]. Most approaches take the middle road by defining filters through the symbolic approach and rank the candidates passing the symbolic filters by the statistical approach.

The two most frequently used grammatical levels used for describing MWEs are the one of morphosyntax and syntax [Baldwin and Kim, 2010]. While morphosyntactic patterns [Church et al., 1991, Clear, 1993] are much more used since they have already yielded satisfactory results, there is a number of approaches that use the syntactic grammatical level as well [Seretan et al., 2003, Martens and Vandeghinste, 2010, Bejček et al., 2013].

In this paper we describe an approach that relies on syntactic patterns to identify MWE candidates. Our main argument for using the syntactic grammatical level is that on languages with partially free word order, such as Slavic languages, morphosyntactic patterns often have to rely on hacks, like allowing up to n non-content words between fixed words or classes, thereby keeping the precision under control while at the same time trying not to loose too much recall. Still, a significant amount of recall is lost since often only the most frequent order of constituents of an MWE is taken into account.

On the other hand, an argument against using syntax for describing MWEs is the precision of the syntactic analysis which is around 80% for well-resourced Slavic languages while morphosyntactic description of well resourced Slavic

languages regularly passes the 90% bar.

Most approaches that use the syntactic grammar layer for extracting MWEs, like [Pecina and Schlesinger, 2006] and the recently added feature in the well-known SketchEngine [Kilgarriff et al., 2004], take into account only MWEs consisting of two nodes, therefore missing the big opportunity syntax offers in defining much more complex patterns that could not be defined on the morphosyntactic level at all.

Until now, there have been no efforts in producing large-scale MWE resources for Croatian, Serbian or Slovene. The first experiments in Croatian include [Tadić and Šojat, 2003] who use PoS filtering, lemmatization and mutual information to identify candidate terms as a preprocessing step for terminological work, [Delač et al., 2009] who experiment on a Croatian legislative corpus while developing the TermeX tool for collocation extraction and [Pinnis et al., 2012] who use the CollTerm tool, part of the ACCURAT toolkit, for term extraction as the first step in producing multilingual terminological resources. All these approaches use morphosyntactic patterns for identifying candidates and do not produce any resources. The only resource for Croatian that does rely on syntactic relations is the distributional memory DM.HR [Šnajder et al., 2013], whose primary goal is distributional modeling of meaning.

A detailed account of the lexicographic treatment of corpus-based phraseology is given by Gantar [Gantar and Peterlin, 2006]. A comprehensive linguistic analysis of the potential and limitations of pattern-based extraction of MWE from a reference corpus was performed by Arhar [Arhar Holdt, 2011]. Semi-automatic procedures to extract MWEs for the Slovene Lexical Database have been proposed by Kosem et al. [Kosem et al., 2013a] while Krek and Dobrovoljc [Krek and Dobrovoljc, 2014] have conducted a pilot study in which they compare the performance of word-sketch-based vs. parser-based collocation extraction.

In this paper we describe a custom-based tool that enables writing complex dependency syntactic patterns for identifying MWE candidates and the resulting recall-oriented MWE resource obtained by applying the tool to parsed corpora of Croatian, Slovene and Serbian. As no such lexicon currently exists for the three languages included in the experiment presented in this paper, and because it is unrealistic to expect heavy investment in similar resources in the near future, our goal is to build a universal resource that will be useful in a wide range of HLT (human language technologies) applications as well as to professional language service providers and the general public. We therefore aim to strike a balance between recall and precision, giving a slight preference to recall in the hope that, on the one hand, human users can deal with the errors efficiently, and applications on the other can resort to post-processing steps in order to mitigate negative effects of noise in the resource.

The paper is structured as follows: in the next section we describe the DepMWEx tool used in building the resource, in Section 3 we describe the resource in numbers and give its initial evaluation, in Section 4 we discuss further possibilities like calculating semantic transparency and taking a multilingual approach, and conclude the paper in Section 5.

2 The DepMWEx tool

Our DepMWEx (Dependency Multiword Extractor) tool[1] consists of a Python module (defining the Tree and Node classes) and Python scripts that, given a grammar and a dependency parsed corpus, produce a list of strongest collocates for each headword.

2.1 The grammar

The grammar consists of a set of grammatical relations, each of which can be described with one or more pattern trees.

Patterns trees are hierarchical structures in which each node contains a boolean function. This function defines the criterion that a node in the parse tree of a sentence must satisfy in order to fill up that node. An example of a pattern tree, corresponding to the MWE *tražiti rupu u zakonu* (literally "search for a hole in the law"), which will be our working example in this section, is given in Figure 1. This pattern tree describes parse subtrees that have a predicate as the main verb which has a direct object and a prepositional phrase attached to it. The framed nodes represent headwords, e.g. *tražiti rupu u zakonu*, to which the MWEs will be added, namely *tražiti#Vm*, *rupa#Nc* and *zakon#Nc*.

The expressiveness of the formalism is substantial, allowing for boolean functions in specific nodes to include restrictions not only on the value of a specific node, but the remaining nodes in the pattern tree as well. One example of using this level of expressiveness is the restriction of the agreement in gender, number and case between nouns and their modifiers, which is a common linguistic phenomenon.

Another example where this level of expressiveness is exploited is the phenomenon in all three languages used in this experiment where nouns with numeral modifiers take the genitive case and not the semantically intended accusative case (semantically encoding the patient, beneficiary etc.) such as in the Croatian example *Podučavam studente* (accusative case, "I teach students") and *Podučavam pet studenata* (genitive case, "I teach five students").

2.2 Grammatical relation naming

The name of the grammatical relation of our MWE example is "gbz sbz4 u sbz6", which is a notation adopted from the Slovene Sketch grammar [Kosem et al., 2013b]. That grammar is defined over morphosyntactic patterns, and, for reasons of compatibility, all three grammars used in this experiment are based on that notation. The acronym denotes

[1] https://github.com/nljubesi/depmwex

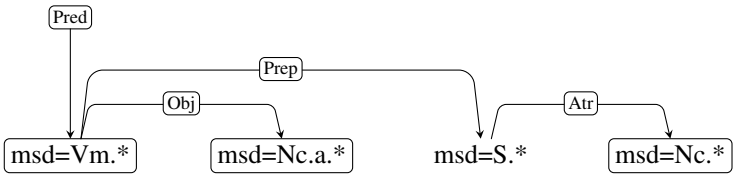


Figure 1: An example of the pattern tree corresponding to the Croatian MWE *tražiti rupu u zakonu*, *raditi račun bez konobara* (literally "to write the check without the waiter"), *raditi od buhe slona* (literally "make an elephant out of a fly", "overexaggerate") etc.

the part of speech ("gbz" being verb, "sbz" noun, "pbz" adjective and "rbz" adverb) while the number denotes the case, and "sbz4" stands for a noun in the accusative case. Finally, one can observe that in the grammatical relation the preposition is lexicalized, which is taken over from the Sketch grammar formalism.

Which part of the grammatical relation is the actual headword the MWE candidate occurs under is labeled by uppercasing that grammatical relation element, so under the verb *tražiti#Vm*, the Croatian MWE candidate *tražiti rupu u zakonu* will appear under the grammatical relation "GBZ sbz4 u sbz6".

2.3 Candidate extraction

The candidate extraction procedure is the following: over each parsed sentence from the corpus, each pattern tree makes an exhaustive search for sentence subtrees that satisfy its constraints. All subtrees corresponding to a pattern tree of a specific grammatical relation are written to standard output as (subtree, grammatical relation) pairs.

2.4 Candidate scoring

Once all (subtree, grammatical relation) pairs are extracted from the corpus in a given language, co-occurrence weighting is performed and MWE candidates are organized by their headwords and their grammatical relations. For now only the log-Dice measure [Rychlỳ, 2008], the association measure used in the Sketch Engine, is implemented in the tool. A selection of the resulting output for the Croatian headword *tražiti#Vm* is given in Table 1.

3 Resource description

3.1 The corpora

The Croatian and Serbian lexicons were extracted from the web corpora of the corresponding languages, namely the 1.9 billion token Croatian Web corpus hrWaC and the parsed half of the 894 million token Serbian Web corpus srWaC [Ljubešić and Klubička, 2014]. These corpora were annotated with morphosyntactic, lemmatization and dependency parsing models built on the SETimes.HR corpus [Agić and Ljubešić, 2014] of 4.000 sentences.

On the other hand, the 100 million token balanced corpus of Slovene KRES [Erjavec and Logar, 2012] was used for building the Slovene lexicon. Our assumption is that this corpus is better suited for the task of extracting lexical information than the web corpora used for Croatian and Serbian for which there are no other freely available corpora. The KRES corpus was annotated with models trained on the SSJ500k corpus[2] consisting of 11.000 sentences.

3.2 The grammars

The grammars of the three languages used in the DepMWEx tool were based on the Slovene sketch grammar used in the SSJ project.[3] Once the morphosyntax-level grammar was transformed to the corresponding dependency syntax level for Slovene, the grammar was adapted for Croatian and Serbian. At this point the Slovene grammar consists of 75 grammatical relations defined through the same number of pattern trees while the Croatian and Serbian grammars consist of 63 grammatical relations with Slovene-specific relations removed.

3.3 The resulting lexicons

The size of the resulting lexicons is given in Table 2. The size of the Croatian lexicon in the number of headwords is very similar to the size of the Slovene lexicon, although the Croatian corpus from which the lexicon is extracted is almost 20 times the size. The reason for this lies in the fact that in the extraction of the Croatian and Serbian lexicons stricter frequency thresholds were applied due to the expected higher level of noise in web corpora in comparison to the manually built and balanced Slovene corpus. The (subtree, grammatical relation) pair frequency threshold applied on Croatian and Serbian data was 5 while for Slovene the threshold was 2.

There was a second threshold, identical for all three languages, applied on the lexicons, namely that each headword had to contain at least 5 MWE candidates (i.e. above mentioned pairs) satisfying the first frequency threshold to be included in the lexicon.

Finally, the Croatian list of headwords and dependents was filtered through two available morphological lexicons

[2] http://eng.slovenscina.eu/tehnologije/ucni-korpus

[3] http://eng.slovenscina.eu

tražiti#Vm	logDice	freq
GBZ sbz4		
pomoć#Nc	8.358	9410
odšteta#Nc	7.958	1949
odgovor#Nc	7.851	4339
povrat#Nc	7.775	1952
ostavka#Nc	7.763	1900
zvijezda#Nc	7.503	2490
smjena#Nc	7.354	1385
rješenje#Nc	7.116	3127
posao#Nc	7.071	6353
naknada#Nc	7.031	1713
sbz1 GBZ sbz4		
prodavač#Nc način#Nc	8.457	330
tužiteljstvo#Nc kazna#Nc	7.295	147
čovjek#Nc mudrost#Nc	6.932	114
čovjek#Nc pomoć#Nc	6.840	108
sindikat#Nc povećanje#Nc	6.801	104
tužitelj#Nc kazna#Nc	6.575	89
prosvjednik#Nc ostavka#Nc	6.057	62
čovjek#Nc odgovor#Nc	6.001	60
žena#Nc muškarac#Nc	5.893	58
radnica#Nc pomoć#Nc	5.832	53
rbz GBZ		
uporno#Rg	7.589	715
stalno#Rg	7.579	1434
GBZ sbz4 za sbz4		
ponuda#Nc podizanje#Nc	10.831	587
rješenje#Nc problem#Nc	7.465	60
sredstvo#Nc ideja#Nc	6.995	39
stan#Nc najam#Nc	6.871	36
naknada#Nc šteta#Nc	6.869	36
obračun#Nc život#Nc	6.756	33
GBZ po sbz5		
vrlet#Nc	6.118	7
internet#Nc	5.612	227
džep#Nc	5.487	36
kontejner#Nc	5.334	29
oglasnik#Nc	4.718	10
kvart#Nc	4.714	21
inercija#Nc	4.623	5
forum#Nc	4.263	115
knjižara#Nc	4.181	8

Table 1: An excerpt of the output of the DepMWEx tool for the Croatian headword *tražiti#Vm*

	lexemes	MWE candidates
hrMWELex	46,293	12,750,029
slMWELex	47,579	6,383,963
srMWELex	23,594	3,279,864

Table 2: The size of the automatically generated lexicons

of Croatian, the Croatian Morphological Lexicon[4] and the Apertium lexicon for Croatian[5]. There was no such lexicon available for Serbian. There was no need for such a filtering process for Slovene since the lemmatization of the corpus is relying on a large morphological lexicon and thereby of very high quality.

The resources, being currently in version 0.5, are encoded in XML and published[6][7][8] under the CC-BY-SA 3.0 license.

4 Resource evaluation

We performed an evaluation of the Croatian and Slovene lexicon by inspecting up to 20 top-ranked MWE candidates for each grammatical relation of 12 selected lexemes for each language. The analyzed Croatian and Slovene lexemes were sampled as follows: 3 lexemes were taken for each part of speech, one in the upper, one in the medium and one in the lower frequency range. One human annotator per language decided whether a MWE candidate was a genuine MWE or not.

Score 1 was assigned to each candidate that represented the appropriate syntactic relationship between the headword and its collocate, regardless of its semantic (un)transparency or syntactic (in)completeness. In other words, if the two-word collocation candidate in question was a syntactically valid lexical realisation of the given grammatical pattern, it was assigned score 1, despite the fact that it was a completely transparent collocation (e.g. *green leaf*) or an idiom (e.g. *green card*). Similarly, the candidate was assigned score 1 also if it formed a semantically complete unit by itself or was only part of a larger multi-word unit (e.g. *zaspati z vestjo*, "to_fall_asleep with conscience", as part of *zaspati z isto/slabo/mirno vestjo*, "to_fall_asleep with clear/guilty conscience"). Although semantically transparent or structurally incomplete two-word units might be of a lesser interest to the community, their recall is more a matter of adjusting the statistical score and/or extending the grammatical patterns to combinations of three or more words rather than a feature of the tool itself.

Score 2, on the other hand, was assigned to each candidate that did not form a valid two-word collocation for the given grammatical pattern due to incorrect pre-processing. This either means that it was assigned an incorrect MSD tag or lemma, which is frequently the case in ambiguous word forms (e.g. noun instead of verb for *stoja* - "stand/stand" or *leglo* -"lie/litter", or adverb instead of neuter adjectives *sanitarno* – "sanitary(ly)", *preventivno* – "preventive(ly)")

[4] http://hml.ffzg.hr

[5] http://sourceforge.net/p/apertium/svn/HEAD/tree/languages/apertium-hbs/

[6] http://nlp.ffzg.hr/resources/lexicons/hrmwelex/

[7] http://nlp.ffzg.hr/resources/lexicons/slmwelex/

[8] http://nlp.ffzg.hr/resources/lexicons/srmwelex/

Croatian				Slovene		
lexeme	# evaluated	precision	diff	lexeme	# evaluated	precision
burza#Nc	559	0.735		ureditev#Nc	563	0.863
lampa#Nc	154	0.422		krč#Nc	200	0.905
lavež#Nc	34	0.324		varovalo#Nc	49	0.755
N	747	0.652	-0.215	N	812	0.867
gurati#Vm	311	0.296		razmišljati#Vm	293	0.816
razumjeti_se#Vm	161	0.484		zaspati#Vm	197	0.843
tužiti_se#Vm	77	0.26		žagati#Vm	23	0.696
V	549	0.346	-0.475	V	513	0.821
dužan#Ag	279	0.29		odgovoren#Ag	171	0.871
legendaran#Ag	64	0.609		zdravstven#Ag	62	0.645
svrhovit#Ag	20	0.4		medgeneracijski#Ag	21	1.000
A	363	0.353	-0.474	A	254	0.827
naprosto#Rg	85	0.859		nenehno#Rg	101	0.871
trostruko#Rg	78	0.615		dosledno#Rg	69	0.986
jednoglasno#Rg	62	0.806		šepetaje#Rg	23	1.000
R	225	0.76	-0.167	R	193	0.927
all	1884	0.518	-0.336	all	1772	0.854

Table 3: MWE candidate precision and difference between languages on each of the 12 evaluated lexemes

or an incorrect dependency relation or label (e.g. relating an adverbs as an attribute of an adjective instead of as an adverbial of a noun).

The precision obtained on each of the 12 lexemes, along with summaries for each part of speech and all lexemes for both evaluated languages, is given in Table 3. We can observe that the overall precision of the MWE candidates is just above 50% for Croatian but is as high as 85.4% for Slovene. The big difference in precision can be explained in most part by two factors:

1. Slovene has a more mature text pre-processing chain which was trained on more than double the amount of training data

2. the Slovene corpus is manually built (and balanced), while the Croatian corpus (similarly to the Serbian one) is automatically built from the web.

Regardless of the absolute difference in precision, same precision trends can be observed in both languages between different parts-of-speech. Adverbs are the most precise PoS, followed by nouns. Verbs and adjectives have an almost identical and the lowest precision in both languages. As one would expect, the drop in accuracy correlates with the task complexity on a specific part-of-speech (measured through precision, i.e. false positive error), showing a larger precision drop between languages on nouns (21.5%) than on adverbs (16.7%), while on verbs and adjectives the drop is the highest and almost identical (47.4% and 47.5%).

Inside each part of speech the MWE candidate accuracies vary significantly and there is no correlation between the frequency range of a lexeme and its precision (the lexemes are ordered by falling frequency).

Next, we analyzed the precision of each specific grammatical relation. The precision for each grammatical relation occurring 10 or more times in the 12 lexemes is given in Table 4. The worst performing set of grammatical relations in Croatian are the *in/ali* ("and/or") relations which search for the same-POS constituents combined with the "and" or "or" conjunction. Another frequent and poorly performing relation is the one of a noun subject and its main verb predicate when the verb is the head (sbz1 GBZ) while significantly better results (0.64 vs. 0.167) are obtained with the subject as the head of a relation (SBZ1 gbz). A similar phenomenon can be observed with the grammatical relation consisting of a main verb and its direct object which performs very poorly when the verb is considered the head of the relation (GBZ sbz4), but with noun as head (gbz SBZ4), the obtained precision is much higher (0.214 vs. 0.714). This result stresses the fact that some relations are actually not symmetric and that the relations as they are defined now have to be reconsidered in the future. In Slovene, on the other hand, the worst performing grammatical relation is the gbz SBZ2, which matches verb+noun_genitive combinations (e.g. *veseliti se poletja* – "look forward to summer") with as little as 0.158 accuracy. There are several top-performing grammatical relations with all candidates extracted correctly in the Slovene evaluation sample, including the most frequent pbz0 SBZ0 pattern that matches adjective+noun_nominative (e.g. *zdravstveno zavarovanje* – "health insurance").

Croatian			Slovene		
relation	frequency	precision	relation	frequency	precision
pbz0 SBZ0	94	0.809	pbz0 SBZ0	109	1.000
RBZ gbz	73	0.822	rbz GBZ	107	0.953
RBZ pbz0	65	0.923	SBZ1 gbz	86	0.791
rbz GBZ	60	0.5	sbz0 SBZ2	85	0.906
sbz1 GBZ	60	0.167	rbz Inf-GBZ	78	0.974
RBZ RBZ	52	0.558	gbz SBZ4	76	0.750
SBZ1 gbz	50	0.64	rbz PBZ0	69	0.696
GBZ u sbz5	49	0.204	GBZ v sbz5	66	0.879
GBZ0 in/ali GBZ0	47	0.213	GBZ z sbz6	53	0.962
PBZ0 in/ali PBZ0	47	0.277	zveze s predlogi	42	1.000
GBZ na sbz4	46	0.283	sbz1 Vez-gbz PBZ1	42	0.976
SBZ0 in/ali SBZ0	45	0.0	PBZ0 in/ali PBZ0	41	1.000
gbz SBZ4	42	0.714	SBZ0 in/ali SBZ0	41	0.707
GBZ sbz4	42	0.214	SBZ0 v sbz5	40	0.975
rbz PBZ0	42	0.357	gbz PBZ1	38	0.447
sbz0 SBZ2	42	0.667	gbz SBZ2	38	0.158
GBZ u sbz4	41	0.829	SBZ0 za sbz4	37	0.784
SBZ0 sbz2	32	0.656	GBZ na sbz5	36	0.972
RBZ Vez-gbz pbz1	27	0.704	GBZ o sbz5	34	0.971
gbz Inf-GBZ	25	0.64	gbz za SBZ4	34	0.941

Table 4: Precision scores for 20 most frequent grammatical relations in each evaluated language

5 Lexicon refinement

At this point we produced a recall-high resource with satisfactory precision, just over 50% for Croatian and 85% for Slovene, and the next obvious step is additional filtering of the resource with the goal of getting the precision rate up without hurting recall. Besides filtering, classifying the MWE candidates into types of MWEs should be looked into as well.

5.1 Semantic transparency

One of the properties of MWEs we are especially interested in is semantic transparency. In this section we report on the initial experiments on Croatian in identifying that type of idiosyncrasy by using the distributional approach.

We built context vectors for all MWE candidates that fall under the following grammatical relations: "pbz0 SBZ0", "SBZ0 sbz2" and "VBZ sbz4". Besides building context vectors for MWE candidates, we also built vectors for their heads.

We built context vectors from three content words to the left and right, stopping at sentence boundaries. We took into consideration only MWE candidates occurring 50 times or more, which we consider minimum context information for any prediction. We used TF-IDF for weighting the vector features and Dice similarity for comparing vectors. We obtained the IDF statistic from head context vectors. The full procedure applied in calculating semantic transparency is the following:

1. build the frequency context vector for each MWE and its head;
2. subtract the MWE vector frequencies from the head-word vector (thereby remove contextual information of that MWE);
3. transform both vectors to TF-IDF vectors;
4. calculate the Dice similarity score between each MWE and its head.

By inspecting MWE candidates, organized under their heads and ordered by the computed similarity to the head, we observed quite promising results. We give a few examples for the simplest "pbz0 SBZ0" relation:

- for the head *voda* ("water"), the most distant MWE candidate is *amaterska voda* (*amaterske vode* refers to a person who moves from professional to amateur)
- for the head *selo* ("village"), the most distant MWE candidate is *španjsko selo* ("Spanish village", refers to something absolutely unknown to someone, like *it's all Greek to me*)
- for the head *stan* ("flat") the most distant MWE is *tkalački stan* ("sewing machine")
- for the head *ured* ("office"), the most distant MWE is *ovalni ured* (the Oval office)
- for the head *zlato* ("gold"), among the most distant MWEs is *crno zlato* ("black gold", referring to oil)

On the other hand, once we sorted all the results, regardless of their head, the results seem much less usable. Besides non-transparent MWEs, we obtain probable parsing errors, low-frequency entries, entries with very static context etc. Nevertheless, the obtained results can be very useful for a lexicographer inspecting a specific headword and will therefore be added to the new version of the lexicon.

5.2 Multilinguality

Since the grammatical relations have the same names in grammars of all the languages used in the experiment, we can use (grammatical relation, dependents) pairs as features for our context vectors, thereby obtaining a more detailed and selective formalization of the context of a lexeme than in the standard distributional approach as implemented in the previous subsection. This leads to more potent distributional memories [Baroni and Lenci, 2010] for tasks of inducing multilingual lexicons of closely related languages by using lexical overlap or similarity, as was done in [Ljubešić and Fišer, 2011]. It would be interesting to inspect how such a memory compares to the already existing distributional memory of Croatian DM.HR [Šnajder et al., 2013] which takes into account only binary relations.

We give here one example for the Croatian–Serbian language pair. The Serbian noun *vaspitanje* is not present in Croatian, but by observing its strongest MWE candidates, which are for the relation "sbz0 SBZ2" *nastava*, *profesor*, *nastavnik* and for the relation "pbz0 SBZ0" *fizički*, *predškolski*, *građanski*, for a human it becomes obvious that the two Croatian counterparts are *odgoj* and *obrazovanje*, which have very similar entries under the same grammatical relations, such as *uvođenje*, *nastava* and *nastavnik* for the "sbz0 SBZ2" relation and *predškolski*, *zdravstven* and *građanski* for the "pbz0 SBZ0" relation. If a model was constructed by using (grammatical relation, dependent) pairs as features and log-Dice as their weights, the models of those two lexemes on the Croatian side would have an overwhelming similarity with the Serbian lexeme in comparison to other lexeme combinations with that Serbian lexeme.

6 Conclusion

In this paper we presented the process of building a recall-oriented MWE lexicon of Croatian, Serbian and Slovene with the newly developed DepMWELex tool which uses syntactic patterns for MWE candidate extraction. Although MWEs are an important part of a lexicon of a certain language, and often key for proficient knowledge and use of a language, they are still not sufficiently represented in dictionaries, lexicons and other resources. This is especially the case with the languages used in this experiment as well as many other under-resourced languages. Thus the intention of building this MWE lexicon was to build a MWE resource that has a wide range of use, including HLT applications, professionals and the general public. Such an extensive resource offers a vast array of possibilities of researching Croatian, Serbian and Slovene and its MWEs. Foreign language learners, as well as professional translators translating into Croatian, Serbian or Slovene as their non-mother tongue, are still lacking such a resource.

Since the recall-high approach was taken in producing the resource, the overall precision of the candidates lies slightly above 50% for Croatian, whereas it is 85% for Slovene. Nevertheless, there are big differences in accuracies of specific grammatical relation, so a lexicon with precision of $\sim 80\%$ for Croatian and $\sim 95\%$ for Slovene can be produced easily by just filtering out the noisy grammatical relations. The possibility of calculating semantic transparency of MWE candidates with the distributional approach was inspected as well with very promising results on the lexeme level. Using the produced output for modeling the context of a lexeme and using it for cross-language linking was shown as well.

This work presents only the first step towards a rich MWE resource of not just Croatian, but its neighboring languages as well. Future work on the resource will start by increasing the size of the underlying corpora for the lexicons of Slovene and Serbian and publishing a three-lingual resource. For that resource to be of maximum value, the possibilities of cross-language linking on both the headword and MWE candidate levels with the distributional approach will be looked into. Finally, focused research on identifying non-transparent MWEs will be undertaken as well.

Acknowledgement

The research leading to these results has received funding from the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement PIAP-GA-2012-324414 (Abu-MaTran) and the Slovenian-Croatian bilateral project "Bilingual Lexicon Construction for Closely Related Languages from Existing Language Resources" (BI-HR/14-15-047).

References

[Agić and Ljubešić, 2014] Agić, Ž. and Ljubešić, N. (2014). The SETimes.HR linguistically annotated corpus of Croatian. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. European Language Resources Association (ELRA).

[Arhar Holdt, 2011] Arhar Holdt, Š. (2011). *Luščenje besednih zvez iz besedilnega korpusa z uporabo dvodelnih in tridelnih oblikoskladenjskih vzorcev*. Trojina, zavod za uporabno slovenistiko.

[Baldwin and Kim, 2010] Baldwin, T. and Kim, S. N. (2010). Multiword expressions. In Indurkhya, N. and

Damerau, F. J., editors, *Handbook of Natural Language Processing, Second Edition*. CRC Press, Taylor and Francis Group, Boca Raton, FL.

[Baroni and Lenci, 2010] Baroni, M. and Lenci, A. (2010). Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

[Bejček et al., 2013] Bejček, E., Stranak, P., and Pecina, P. (2013). Syntactic identification of occurrences of multiword expressions in text using a lexicon with dependency structures. In *Proceedings of the 9th Workshop on Multiword Expressions*, pages 106–115, Atlanta, Georgia, USA. Association for Computational Linguistics.

[Church et al., 1991] Church, K., Gale, W., Hanks, P., and Hindle, D. (1991). Using statistics in lexical analysis. In *Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon*, pages 115–164. Erlbaum.

[Clear, 1993] Clear, J. (1993). *Text and Technology: In honour of John Sinclair*, chapter From Firth Principles - Computational Tools for the Study of Collocation. John Benjamins Publishing Company.

[Delač et al., 2009] Delač, D., Krleža, Z., Šnajder, J., Bašić, B. D., and Šarić, F. (2009). Termex: A tool for collocation extraction. In Gelbukh, A. F., editor, *CICLing*, volume 5449 of *Lecture Notes in Computer Science*, pages 149–157. Springer.

[Erjavec and Logar, 2012] Erjavec, T. and Logar, N. (2012). Referenčni korpusi slovenskega jezika (cc)Gigafida in (cc)KRES. In *Zbornik Osme konference Jezikovne tehnologije*.

[Gantar and Peterlin, 2006] Gantar, P. and Peterlin, A. P. (2006). Korpusni pristop v frazeologiji in slovarske aplikacije. *Slavistična revija*.

[Kilgarriff et al., 2004] Kilgarriff, A., Rychly, P., Smrz, P., and Tugwell, D. (2004). The Sketch Engine. *Information Technology*, 105:116.

[Kosem et al., 2013a] Kosem, I., Gantar, P., and Krek, S. (2013a). Avtomatizacija leksikografskih postopkov. *Slovenščina 2.0*.

[Kosem et al., 2013b] Kosem, I., Krek, S., and Gantar, P. (2013b). Automatic extraction of data: Slovenian case revisited. In *SKEW-4: 4th International Sketch Engine Workshop*, Talinn, Estonia.

[Krek and Dobrovoljc, 2014] Krek, S. and Dobrovoljc, K. (2014). Sketch grammar or parser – a comparison of two extraction methods. Poster.

[Ljubešić and Fišer, 2011] Ljubešić, N. and Fišer, D. (2011). Bootstrapping bilingual lexicons from comparable corpora for closely related languages. In *Text, Speech and Dialogue - 14th International Conference, TSD 2011, Pilsen, Czech Republic, September 1-5, 2011. Proceedings*, volume 6836 of *Lecture Notes in Computer Science*, pages 91–98. Springer.

[Ljubešić and Klubička, 2014] Ljubešić, N. and Klubička, F. (2014). {bs,hr,sr}WaC – web corpora of Bosnian, Croatian and Serbian. In *Proceedings of the 9th Web as Corpus Workshop (WaC-9)*, pages 29–35, Gothenburg, Sweden. Association for Computational Linguistics.

[Martens and Vandeghinste, 2010] Martens, S. and Vandeghinste, V. (2010). An efficient, generic approach to extracting multi-word expressions from dependency trees. In *Proceedings of the 2010 Workshop on Multiword Expressions: from Theory to Applications*, pages 85–88, Beijing, China. Coling 2010 Organizing Committee.

[Pecina and Schlesinger, 2006] Pecina, P. and Schlesinger, P. (2006). Combining association measures for collocation extraction. In *Proceedings of the COLING/ACL on Main Conference Poster Sessions*, COLING-ACL '06, pages 651–658. Association for Computational Linguistics.

[Pinnis et al., 2012] Pinnis, M., Ljubešić, N., Ştefănescu, D., Skadiņa, I., Tadić, M., and Gornostay, T. (2012). Term extraction, tagging, and mapping tools for underresourced languages. In *Proceedings of the Terminology and Knowledge Engineering (TKE2012) Conference*, Madrid, Spain.

[Rychlỳ, 2008] Rychlỳ, P. (2008). A lexicographer-friendly association score. *Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN*, pages 6–9.

[Sag et al., 2001] Sag, I. A., Baldwin, T., Bond, F., Copestake, A., and Flickinger, D. (2001). Multiword expressions: A pain in the neck for nlp. In *In Proc. of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2002*, pages 1–15.

[Seretan et al., 2003] Seretan, V., Nerima, L., and Wehrli, E. (2003). Extraction of multi-word collocations using syntactic bigram composition. In *In Proceedings of the International Conference RANLP'03*, pages 424–431.

[Šnajder et al., 2013] Šnajder, J., Padó, S., and Agić, Ž. (2013). Building and evaluating a distributional memory for croatian. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789. Association for Computational Linguistics.

[Tadić and Šojat, 2003] Tadić, M. and Šojat, K. (2003). Finding multiword term candidates in croatian. In *Proceedings of Information Extraction for Slavic Languages 2003 Workshop*, pages 102–107.

Modeling Semantic Compositionality of Croatian Multiword Expressions

Jan Šnajder and Petra Almić
University of Zagreb
Faculty of Electrical Engineering and Computing
Text Analysis and Knowledge Engineering Lab
Unska 3, 10000 Zagreb, Croatia
E-mail: jan.snajder@fer.hr, petra.almic@gmail.com

Keywords: multiword expressions, semantic composition, distributional semantics, Croatian language

Received: March 30, 2015

A distinguishing feature of many multiword expressions (MWEs) is their semantic non-compositionality. Determining the semantic compositionality of MWEs is important for many natural language processing tasks. We address the task of modeling semantic compositionality of Croatian MWEs. We adopt a composition-based approach within the distributional semantics framework. We build and evaluate models based on Latent Semantic Analysis and the recently proposed neural network-based Skip-gram model, and experiment with different composition functions. We show that the compositionality scores predicted by the Skip-gram additive models correlate well with human judgments (ρ=0.50). When framed as a classification task, the model achieves an accuracy of 0.64.

Povzetek: Razvita je metoda za dekompozicijo hrvaškega jezika.

1 Introduction

The peculiarity of multiword expressions (MWEs) has long been acknowledged in natural language processing (NLP). According to [29], MWEs can be defined as idiosyncratic interpretations that cross word boundaries (or spaces). Because of their unpredictable and idiosyncratic behavior, such expressions need to be listed in a lexicon and treated as a single unit ("word with spaces") [10, 5]. One dimension along which the MWEs can be analyzed is their semantic compositionality, sometimes referred to as semantic idiomaticity or semantic transparency. Semantic compositionality is the degree to which the features of the parts of an MWE combine to predict the features of the whole [4]. The meaning of a non-compositional MWE cannot be deduced from the meaning of its parts. In reality, MWEs span a continuum between completely compositional expressions (e.g., *world war*) to non-compositional ones [6]. A prime example of non-compositional MWEs are idioms, such as *kick the bucket (to die)* or *red tape (excessive rules and regulations)*.

Being able to model the semantic compositionality of MWEs – and in turn determine whether a given MWE is semantically transparent or opaque – has been shown to be important for many NLP tasks, ranging from machine translation [8] and information retrieval [1] to word sense disambiguation [11]. It is thus not surprising that the task of automatically determining semantic compositionality has gained a lot of attention [15, 4, 7, 28, 18].

In this paper we address the task of modeling semantic compositionality of Croatian MWEs comprised of two words. We follow up on the work of [15] and [7] and adopt a compositionality-based approach: the basic idea is to compare the meaning of an MWE against the meaning of the composition of its parts. To model the meaning of the MWEs and its parts, we use distributional semantics, which represents the word's meaning based on the distribution on its contexts in a corpus, assuming that similar words tend to appear in similar contexts [13]. To determine the compositionality of an MWE, we compare its context distribution in the corpus to the context distribution approximated by the composition of its parts.

The contribution of our work is twofold. First, we build a dataset of Croatian MWE annotated with semantic compositionality scores. Secondly, we build and evaluate a set of semantic compositionality models based on Latent Semantic Analysis (LSA) [20] and the recently proposed neural network-based Skip-gram model [24]. Our results show that the compositionality scores predicted by additive compositional models correlate well with human-annotated scores, thereby confirming similar results for the English language. To the best of our knowledge, this is the first work to consider the modeling of semantic compositionality for the Croatian language.

The remainder of the paper is structured as follows. In the next section we give an overview of related work. We describe the creation of the dataset in Section 3 and the compositionality models in Section 4. In Section 5 we present evaluation results. Section 6 concludes the paper.

2 Related work

The approaches for modeling semantic compositionality can be broadly divided into two groups: knowledge-based approaches and corpus-based approaches. The former rely on linguistic resources (e.g., WordNet) to measure the semantic similarity between an MWE and its parts [16]. An obvious downside of knowledge-based approaches is that for most languages the linguistic resources are not available, while the construction of such resources is labor-intensive and expensive. In contrast, corpus-based approaches rely on statistical properties of MWEs and the constituting words, which can be readily extracted from corpora. E.g., [23] rely on the hypothesis that non-compositional MWEs tend to be syntactically more fixed than compositional MWEs, while [27] assumes that lexical association correlates with non-compositionality.

Related to the work presented in this paper are specifically the corpus-based approaches that make use of distributional semantic modeling of MWEs and their constituents. The pioneering work in this direction is that of [21], who used a statistical association measure to discriminate between compositional and non-compositional MWEs. Lin compared the mutual information of an MWE and of an expression obtained as a slight modification of the original MWE (e.g., *red tape* vs. *orange tape*). Although this method has not shown to work well, the idea that non-compositional expressions have a "different distributional characteristic" than similar compositional expressions paved a way for other distributional semantics based approaches. [5] used LSA to compare the similarity between an MWE and its head, and showed that there exists a correlation between the measured semantic similarity and compositionality. Along the same lines, [15] used LSA to compare the semantic vector of an MWE against the semantic vector of the composition of its constituents, obtained simply as the sum of the corresponding vectors.

To consolidate the research efforts, [7] organized a shared task on Distributional Semantics and Compositionality (DISCo), and provided datasets in English and German with human compositionality judgments. The task was shown to be hard and no clear winner emerged. However, the approaches based on distributional semantics seemed to outperform those based on statistical association measures. Following up on DISCo, [18] performed a systematic evaluation of various distributional semantic approaches to compositionality detection, and showed that LSA-based models perform quite well.

In this paper we adopt the methodology of [15] to compare the distribution of an MWE to the composition of its parts, but we experiment with different composition functions, proposed by [26]. To build the dataset, we adopt the methodology of [7].

3 Annotated dataset

The starting point of our work is a dataset of representative Croatian MWEs annotated with human compositionality judgments. In building this dataset, we adopted the approach of [7], but depart from it in some key aspects that we discuss below. As a source of data, we used the 1.2 billion words corpus fHrWaC[1] [30], a filtered version of the Croatian web corpus *hrWaC* [22]. The corpus has been tokenized, lemmatized, POS tagged, and dependency parsed using the HunPos tagger and CST lemmatizer for Croatian [3], and the MSTParser for Croatian [2], respectively. We next describe the construction of the dataset.[2]

3.1 MWE extraction

Following the work of [7], we restricted ourselves to the following three MWE types:

- **AN**: an adjective modifying a noun, e.g., *žuti karton* (*yellow card*);
- **SV**: a verb with a noun in the subject position, e.g., *podatak govori* (*data says*);
- **VO**: a verb with a noun in the object position, e.g., *popiti kavu* (*drink coffee*).

We extracted all dependency bigrams (i.e., possibly non-contiguous bigrams) from the corpus that match one of these three types and sorted them by frequency in descending order.[3] Going from the top of list, we (the two authors) manually annotated the MWEs (i.e., for each bigram we annotated whether it constitutes an MWE) and additionally pre-annotated each MWE as either compositional (C) or non-compositional (NC). We next selected the bigrams on which both annotators agreed, and then balanced the set so that it contains an equal number of compositional and non-compositional MWEs. The so-obtained dataset does not reflect the true distribution of MWEs, as the compositional MWEs are much more frequent in the corpus. However, balancing the dataset is justified because our focus is on discriminating between the compositional and non-compositional MWEs. The final dataset contains 100 compositional and 100 non-compositional MWEs (125 AN, 10 SV, and 65 VO expressions). Note that the C/NC annotation is preliminary; each of the 200 MWEs has subsequently been annotated with compositionality scores by multiple human annotators other than the authors (cf. Section 3.3).

3.2 Levels of compositionality

During MWE pre-annotation, we identified various flavors of compositionality. For example, a *yellow card* really is a

[1] http://takelab.fer.hr/data/fhrwac/

[2] The dataset is available under the Creative Commons BY-SA license from http://takelab.fer.hr/cromwesc

[3] By considering only the most frequent MWEs, we limit ourselves to MWEs with most reliable distributional representations.

yellow card, but its predominant sense is a figurative one (a warning indication). In contrast, *gray economy* is indeed a type of economy, but *gray* does not have a literate meaning. Further along these lines, *chain* in a *chain store* is not a chain in its predominant sense. One can argue that all these expressions are non-compositional to a certain extent. In an attempt to give an operational account of the different levels of non-compositionality, we propose the following typology:

NC3: Expressions that are completely non-compositional, i.e., the meaning of constituents cannot be combined to give the meaning of the expression. E.g., *žuti karton (yellow card)* and *preliti čašu* (literal meaning: *spill over the cup*; figurative meaning: *the last straw*), *trljati ruke (to rub ones hands)*;

NC2: Partially compositional expressions, i.e., the meaning of one but not both constituents is opaque, e.g., *siva ekonomija (gray economy)*, *bilježiti rast (to record a growth)*, *morski pas* (literal meaning: *sea dog*; compositional meaning: *a shark*);

NC1: The expressions that are non-compositional if we consider only the predominant senses of one or both of its constituents. For example, if we consider the predominant sense of *chain* to be a series of metal rings, then a *mountain chain* is a non-compositional expression.[4]

Note that our typology is motivated by practical rather than theoretical concerns. When concerned with automatic compositionality detection, we expect type NC3 to be more easily determinable than type NC1. However, from a theoretical perspective, the proposed typology is oversimplified and we make no attempt here to relate it to the different types of figures of speech studied in linguistics (e.g., metaphors, metonyms, synegdochs, etc.).

3.3 Annotation

[7] used the crowdsourcing service Amazon Turk to annotate their dataset. For every expression, they provided five different context sentences. For each in-context MWE, they asked the turkers to annotate how literal the MWE is, on a scale from 1 (non-compositional) to 10 (compositional). Because the set of annotators differs across MWEs, they were not able to estimate the inter-annotator agreement. However, they argued that the judgments for the expressions should be reliable because they were averaged over several sentences and several annotators. As the final compositionality scores, they computed the mean score for each MWE.

We departed from the above-described setup for two reasons. Methodologically, we argue that annotating MWEs across contexts is inappropriate for the task of type-based semantic compositionality detection, which is what we are addressing here. The reason is that this setup ignores the fact that MWEs may have different meanings (compositional and non-compositional ones) depending on the context, thus averaging across the contexts will lump together the various senses.[5]On a practical side, in-context annotation is more expensive and would require more resources (we feel that annotating five sentences per MWE would not suffice to reliably capture the sense variability of MWEs). For these reasons, we chose not to annotate MWEs across different contexts.

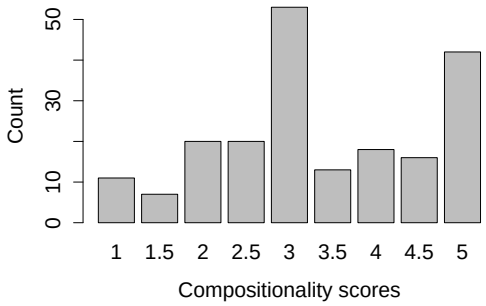


Figure 1: Histogram of MWE compositionality scores.

Our annotation setup was as follows. A total of 24 volunteers (mostly students) participated in the annotation. To reduce the workload, we divided the 200 MWEs into four groups (A, B, C, D) and randomly assigned one group to each annotator. Thus, each MWE was annotated by six annotators. To be able to computer the inter-annotator agreement, we ensured that there is a 10% overlap among all four groups (20 expressions that were annotated by all 24 annotators). We asked our annotators to judge how literal each MWE is on the scale from 1 (non-compositional) to 5 (compositional). For each MWE, we provided one context sentence that instantiates its non-compositional meaning (for non-compositional MWEs) or typical compositional meaning (for compositional MWEs). We did this to ensure that annotators consider the same sense of an MWE, so that the judgments would not diverge because of sense mismatches.

We computed the final compositionality score for each MWE as the median of its compositionality scores. Fig. 1 shows the scores histogram, while Table 1 shows some examples from the annotated dataset.

3.4 Annotation analysis

Table 2 shows the inter-annotator agreement in terms of the Krippendorff's alpha coefficient [17] for each of the groups as well as the overlapping part of the dataset. We

[4]We are aware that the notion of a predominant sense is a problematic one. Many of the NC1 MWEs in our dataset are in fact borderline cases between NC1 and C classes.

[5]In-context MWE compositionality annotation would be adequate for the task of token-based semantic compositionality detection (detection of semantic compositionality of a MWE instance in context). Curiously enough, [7] were also addressing the type-based ask, but used in-context annotation.

MWE	Type	Score
maslinovo ulje (olive oil)	AN	5
krvni tlak (blood pressure)	AN	5
telefonska linije (telephone line)	AN	4
pružiti pomoć (to offer help)	VO	4
kućni ljubimac (a pet)	AN	3.5
crno tržište (black market)	AN	3
voditi brigu (to worry)	VO	3
ostaviti dojam (to leave an impression)	VO	2.5
zeleno svjetlo (green light)	AN	1
hladni rat (cold war)	AN	1

Table 1: Examples from the annotated dataset.

Sample	AN+SV+VO	AN	SV+VO
Group A	0.587	0.620	0.535
Group B	0.506	0.510	0.478
Group C	0.490	0.544	0.337
Group D	0.586	0.505	0.648
Overlap (10%)	0.456	0.452	0.439

Table 2: Inter-annotator agreement (Krippendorff's α).

consider the agreement to be moderate and indicative of the high subjectivity of the task. The agreement on the verb expressions is somewhat lower in comparison to adjective-noun expressions. In Table 3 we present some example MWEs from the dataset where the annotators achieved a high level of agreement (zero standard deviation) and a low level of agreement (st. dev. larger than 1.3).

As an indication of the ceiling performance, we computed the correlation between every annotator's scores and the median scores. The average Spearman's correlation coefficient over 24 annotators is 0.77.

High agreement ($\sigma = 0$)	
igrati nogomet (play soccer)	5.0
služiti kaznu (serve sentence)	3.0
financijska pomoć (financial aid)	5.0
pjevati pjesmu (sing song)	5.0
nemati sumnje (have no doubt)	5.0
Low agreement ($\sigma > 1.3$)	
zabilježiti rast (record growth)	4.5
žuti karton (yellow card)	3.0
prvi korak (first step)	3.0
telefonska linija (phone line)	4.0
crveni karton (red card)	4.5

Table 3: Examples of MWEs and median compositionality scores with high and low inter-annotator agreement.

3.5 Test set

To optimize and experiment with the various parameters, we randomly split our dataset into the train and test set, each consisting of 100 MWEs. The breakdown of MWEs by type in the test set is as follows: 65 AN, 30 VO, 5 VS. Furthermore, we (the two authors) annotated the level of compositionality (cf. Section 3.2) for each MWE from the test set and resolved the disagreements by consensus. Our primarily motivation for this was to be able to investigate how the level of non-compositionality influences the performance of the model. The breakdown of compositionality levels in the test set is as follows: 48 C, 31 NC1, 7 NC2, 14 NC3.

4 Compositionality models

To build our models, we use the fHrWaC corpus, the same corpus we used to build the dataset. To determine the semantic compositionality of a MWE, we carry out the following three steps: (1) model the meaning of the constituent words, (2) model the composition of the meaning, and (3) compare these meanings.

4.1 Modeling word meaning

To model the meaning of constituent words, we use two distributional semantics models. First is the well-known Latent Semantic Analysis [20] model. LSA has shown to perform quite good in the task of semantic compositionality detection [15, 18], and performed very good in the task of identifying synonyms in the Croatian language [14]. We defined the context as a ± 5 word window around the word, or, in the case of the MWEs, a ± 5 word window around both constituents. For the constituent words, we only consider the contexts in which they appear alone, i.e., not as a part of any MWE from our dataset. Motivation behind this is to emphasize the independent contribution of the constituents in an expression, as proposed by [15]. As context elements (the columns of the LSA matrix), we use the 10k most frequent lemmas from the corpus (excluding stop words). As target elements (the rows of the matrix), we use the MWEs and their constituting words, as well as the 5k most frequent lemmas from the corpus. For weighing the word-context associations, we experimented with two functions: log-entropy [19] and Local Mutual Information (LMI) [9]. Since log-entropy gave consistently better results, we use only log-entropy in subsequent experiments. We use singular value decomposition to reduce the dimensionality of the matrix from 10000 to 100 dimensions per target.

The second model we experiment with is the neural network-based Skip-gram model recently proposed by [24]. Skip-gram model produces low-dimensional, real-valued vector representations of words (also known as word embeddings) by learning to predict the context of each input word. Skip-gram model has been shown to

be very effective for predicting the semantic similarity of words and has excelled in the synonymy detection and relation modeling task for Croatian [31], outperforming LSA by a large margin. Furthermore, it has been shown that with Skip-gram model the semantic composition of short phrases can be modeled quite effectively via simple vector addition [25], which makes it well-suited for our task. We build 100-dimensional vector representations of MWEs and their constituting words using the `word2vec` tool,[6] with default parameters (number of negative examples set to 5, no hierarchical softmax, maximum skip length of 5), but without a frequency threshold.

4.2 Modeling composed meaning

The second step was to model the composition of the word meanings. [26] introduced a number of composition models (additive, weighted additive, multiplicative, tensor product, and dilation), which they evaluated on a phrase similarity task (e.g. *vast amount* vs. *large quantity*). In this work, we experiment with additive ($\vec{z} = \vec{x} + \vec{y}$), weighted additive ($\vec{z} = \alpha\vec{x} + \beta\vec{y}$), and the multiplicative model ($\vec{z} = \vec{x} \odot \vec{y}$), where z stands for the composed vector and $\vec{x}$ and $\vec{y}$ stand for vectors of its constituent words.

We experiment with two weighted additive models. In the first one (model Opt), similarly to [26], we optimize the weights on the train set to maximize the correlation with human-assigned scores. The weights are optimized globally and they are identical for every MWE. In the second model (model Dyn), we calculate the weights dynamically, separately for each MWE, as proposed by [28]. The two weights, α and β, are defined as

$$\alpha = \frac{\cos(\overrightarrow{xy}, \vec{x})}{\cos(\overrightarrow{xy}, \vec{x}) + \cos(\overrightarrow{xy}, \vec{y})}, \quad \beta = 1 - \alpha \qquad (1)$$

where $\overrightarrow{xy}$ is the MWE vector. The intuition behind this method is that more importance should be given to the constituent that is semantically more similar to the whole MWE, i.e., the constituent whose vector is closer, in terms of the cosine similarity, to the vector of the MWE. For example, in the expression *gray economy*, more importance should given to the word *economy* than the word *gray*.

4.3 Compositionality prediction

Finally, in the third step, we use the cosine similarity measure to compare the vector of the MWE and the vector of its composition-derived meaning. We expected that for the compositional MWEs these two meaning vectors will be similar, i.e., cosine similarity will be closer to 1, while for non-compositional it will be closer to 0. Thus, the model simply predicts the semantic compositionality score as the cosine between the MWE vector and the composed MWE vector.

[6] `http://code.google.com/p/word2vec/`

Additionally, we consider the linear combination model proposed by [28]. Instead of relying on a single compositionality prediction, this model uses the collective evidences from several models. More precisely, the model predicts the semantic compositionality score as a linear combination of the prediction of the additive model, the multiplicative model, and the two individual constituents model:

$$\begin{aligned}\lambda = a_0 &+ a_1 \cdot \cos(\overrightarrow{xy}, \overrightarrow{x+y}) + a_2 \cdot \cos(\overrightarrow{xy}, \overrightarrow{x \odot y}) \\ &+ a_3 \cdot \cos(\overrightarrow{xy}, \overrightarrow{x}) + a_4 \cdot \cos(\overrightarrow{xy}, \overrightarrow{y})\end{aligned} \qquad (2)$$

We optimized the parameters a_0–a_4 using least squares regression on the train set.

5 Evaluation

The task of determining semantic compositionality can be framed as a regression problem (prediction of compositionality scores) or a classification problem (compositionality vs. non-compositionality). We consider both settings.

5.1 Predicting compositionality scores

In Table 4 we show the correlation (Spearman's ρ) between model-predicted and human-assigned compositionality scores on the test set. Generally, additive models outperform the other considered composition models. This is in contrast to the conclusions of [26], but in accordance with the results of [12] and [18]. For both LSA and Skip-gram, correlation for verbal MWEs is much worse then for adjective-noun MWEs. This is expected, as it has been observed that the semantics of verbs is not fully covered by distributional spaces (cf. [31, 14]) Skip-gram model outperforms LSA, confirming the findings from other tasks [31]. The difference in performance is most prominent for verbal (SV+VE) MWEs. Overall, the best performing models are the Skip-gram additive and linear combination models ($\rho = 0.50$). Specifically, if one considers the AN and SV+VO subsets, Skip-gram linear combination model emerges as the clear winner, suggesting that combining the evidence from multiple models is beneficial. We consider the obtained correlation of 0.50 to be satisfactory, considering that the average correlation of human annotators is 0.77. Our results seem to be comparable to those of [7, 18] obtained for English.

5.2 Compositionality classification

For the compositionality classification task, we converted the compositionality scores to binary labels. To this end, we analyzed the distribution of the scores in the dataset (Fig. 1). The distribution is bimodal, so we chose to set the cut-off after the first mode: MWEs with a score in the $[1, 3]$ range are labeled as non-compositional (NC), while

	LSA			Skip-gram		
Composition model	AN+SV+VO	AN	SV+VO	AN+SV+VO	AN	SV+VO
Multiplicative	−0.19	−0.20	−0.18	0.01	−0.14	0.38
Simple additive	0.45	0.54	**0.35**	**0.50**	0.55	0.40
Weighted additive (Opt)	0.46	0.56	0.28	**0.50**	0.55	0.40
Weighted additive (Dyn)	0.46	**0.57**	0.26	**0.50**	0.55	0.40
First constituent	0.41	0.50	0.19	0.37	0.43	0.21
Second constituent	0.28	0.31	0.31	0.41	0.49	0.36
Linear combination (λ)	**0.48**	0.56	0.34	**0.50**	**0.58**	**0.47**

Table 4: Spearman's correlation coefficient on the test set for LSA and Skip-gram model and different composition functions.

	AN+SV+VO	AN	SV+VO
Precision	0.56	0.63	0.44
Recall	0.82	0.84	0.92
F1-score	0.67	0.72	0.60
Accuracy	0.64	0.69	0.54

Table 5: Classification results for the Skip-gram linear combination model.

those with a score in the $\langle 3, 5]$ range are labeled as compositional (C). This gave us 44 compositional (C) and 56 non-compositional MWEs in the test set. We consider only the best-performing model from the previous experiment (the Skip-gram linear combination model). The model predicts C (positive class) if the prediction of the linear combination model defined by (2) is above a certain threshold, otherwise it predicts NC (negative class). We set the threshold to $t = 3.11$, obtained by optimizing the F1-score on the train set. The results are shown in Table 5. The overall classification accuracy is 0.64. The accuracy is higher for adjective-noun MWEs (0.72) than for verbal MWEs (0.54), which is in line with the results from the previous experiment. Precision is substantially lower than recall (0.56 vs. 0.82), indicating that the model more often predicts compositionality for a non-compositional MWE than the other way around, i.e., the predictions for non-compositional MWEs are often higher than they ought to be. Our model outperforms the accuracy of a majority class (NC) baseline, which is 0.56, but not the F1-score, which is 0.72.

The classification task is similar to the one considered by [15]. In their experiment, they achieved an F1-score of 0.48, but they only considered the additive model.

5.3 Result analysis

In this section we give some insights about the model performance. Results show moderate level of correlation, so we are interested in investigating on what MWEs the model fails. We are also interested in relating the model performance to the levels of compositionality introduced in Section 3.2 and the inter-annotator agreement levels.

In Table 6 we list the MWEs on which the Skip-gram linear combination model performs the worst. We define the error as an absolute difference between z-scored model-predicted and human-annotated compositionality scores. The results suggest that most errors occur on compositional expressions (C).

To explore this hypothesis a bit further, we divide our test set into the subsets based on the compositionality levels and analyze compositionality scores and correlation on these subsets. Fig. 2a shows z-scored human-assigned compositionality scores and z-scored model predictions across different compositionality levels. Both human-assigned and predicted scores increase with the level of compositionality, however the model tends to assign lower scores to compositional MWEs (C) and higher scores to completely non-compositional MWEs (NC3); the latter is in line with the classification results (low precision).

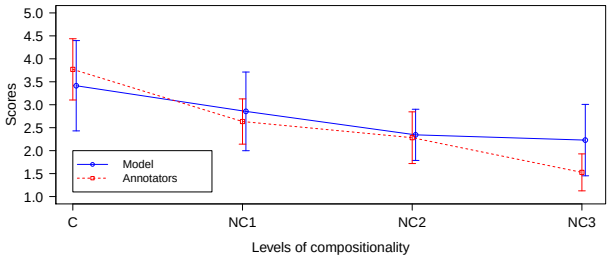
Fig. 2b shows the correlation between human-assigned and model-predicted scores across different compositionality levels. The plot shows that the model performs best on non-compositional MWEs of type NC1 (non-compositional in the predominant reading) and much worse on other non-compositional MWEs as well as compositional MWEs.[7] While this is in line with the previous analysis (model underestimates C scores and overestimates NC3 scores), it remains unclear why the model performs better on NC1. The results seems counterintuitive, as one would expect NC3 (completely non-compositional) MWEs to be more easily detectable than NC1 (non-compositional in the predominant reading).

A more systematic analysis, which is out of the scope of this paper, would be required to determine the underlying causes. One of the possible reasons could be the low quality of vector representations for some (rare) words. The low quality of the individual words propagates to the low quality of compositional representations, which in turn makes the composed vector too dissimilar to the MWE vector. A further problem might stem from the polysemy, another weakness of distributional semantic models.

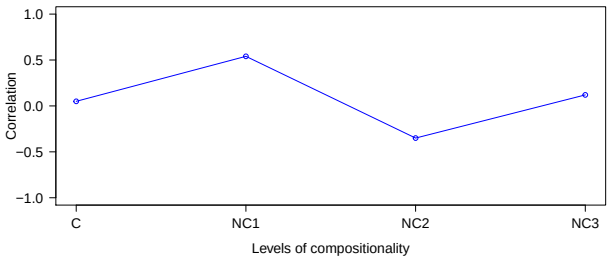
[7] Note that NC2 and NC3 have few data instances, hence their correlation results should be taken with caution.

MWE	Type	Level	Score	Prediction	Error
oglasna ploča (announcement board)	AN	C	4.5	1.69	3.33
organizacijski odbor (organizing committee)	AN	C	5	2.76	2.26
motorno vozilo (motor vehicle)	AN	C	5	2.79	2.22
nemati sumnje (have no doubt)	VO	C	5	2.82	2.18
optužnica tereti (charged with)	SV	C	2.5	4.35	1.96
životno djelo (lifework)	AN	C	3	1.81	1.94
novi val (new wave)	AN	NC3	1	3.33	1.78

Table 6: MWEs on which the Skip-gram linear model performs the worst (human-assigned scores and model predictions are not scaled, while the error is between z-scored values).



(a)



(b)

Figure 2: Analysis across four compositionality levels: (a) z-scored human-assigned scores and model predictions, (b) correlations.

6 Conclusion

In this paper we modeled of semantic compositionality of Croatian multiword expressions (MWEs) using composition-based distributional semantics. We built a small dataset of Croatian MWEs, manually annotated with semantic compositionality scores. To represent the meaning of the MWEs and their constituents, we build two kinds of models (LSA and Skip-gram), and experimented with additive and multiplicative composition functions. The best-performing model combines the predictions of the additive and the multiplicative models, and achieves a correlation of 0.50 and a classification accuracy of 0.64. The model tends to underestimate scores for compositional MWEs and overestimate scores for non-compositional MWEs. Surprisingly, the model works best on MWEs that that are non-compositional if one considers the predominant reading of MWE constituents.

Future work might address a more systematic analysis. This implies annotating a larger dataset (possibly one that is unbalanced and hence more realistic) and accounting for confounding factors such as MWE frequency and ambiguity.

References

[1] Otavio Costa Acosta, Aline Villavicencio, and Viviane P. Moreira. Identification and treatment of multiword expressions applied to information retrieval. In *Proc. of the Workshop on Multiword Expressions: From Parsing and Generation to the Real World*, pages 101–109. ACL, 2011.

[2] Željko Agić and Danijela Merkler. Three syntactic formalisms for data-driven dependency parsing of Croatian. In *Text, Speech, and Dialogue*, pages 560–567. Springer, 2013.

[3] Željko Agić, Nikola Ljubešić, and Danijela Merkler. Lemmatization and morphosyntactic tagging of Croatian and Serbian. In *Proc. of ACL*, 2013.

[4] Timothy Baldwin. Compositionality and multiword expressions: Six of one, half a dozen of the other. In

Invited talk given at the COLING/ACL'06 Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties, 2006.

[5] Timothy Baldwin, Colin Bannard, Takaaki Tanaka, and Dominic Widdows. An empirical model of multiword expression decomposability. In *Proc. of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment-Volume 18*, pages 89–96. ACL, 2003.

[6] Colin Bannard, Timothy Baldwin, and Alex Lascarides. A statistical approach to the semantics of verb-particles. In *Proc. of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment - Volume 18*, MWE '03, pages 65–72. ACL, 2003. doi: 10.3115/1119282.1119291.

[7] Chris Biemann and Eugenie Giesbrecht. Distributional semantics and compositionality 2011: Shared task description and results. In *Proc. of the Workshop on Distributional Semantics and Compositionality*, pages 21–28. ACL, 2011.

[8] Marine Carpuat and Mona Diab. Task-based evaluation of multiword expressions: A pilot study in statistical machine translation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 242–245. ACL, 2010.

[9] Stefan Evert. *The statistics of word cooccurrences*. PhD thesis, Dissertation, Stuttgart University, 2005.

[10] Stefan Evert. Corpora and collocations. *Corpus Linguistics. An International Handbook*, 2:223–233, 2008.

[11] Mark Alan Finlayson and Nidhi Kulkarni. Detecting multi-word expressions improves word sense disambiguation. In *Proc. of the Workshop on Multiword Expressions: From Parsing and Generation to the Real World*, pages 20–24, 2011.

[12] Emiliano Guevara. Computing semantic compositionality in distributional semantics. In *Proc. of the Ninth International Conference on Computational Semantics*, pages 135–144. ACL, 2011.

[13] Zellig Harris. Distributional structure. *Word*, 10(23): 146–162, 1954.

[14] Mladen Karan, Jan Šnajder, and Bojana Dalbelo Bašić. Distributional semantics approach to detecting synonyms in Croatian language. *Information Society*, pages 111–116, 2012.

[15] Graham Katz and Eugenie Giesbrecht. Automatic identification of non-compositional multi-word expressions using latent semantic analysis. In *Proc. of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*, pages 12–19. ACL, 2006.

[16] Su Nam Kim and Timothy Baldwin. Automatic identification of English verb particle constructions using linguistic features. In *Proc. of the Third ACL-SIGSEM Workshop on Prepositions*, pages 65–72. ACL, 2006.

[17] Klaus Krippendorff. Reliability in content analysis. *Human Communication Research*, 30(3):411–433, 2004.

[18] Lubomír Krčmář, Karel Ježek, and Pavel Pecina. Determining compositionality of word expresssions using various word space models and methods. In *Proc. of the Workshop on Continuous Vector Space Models and their Compositionality*, pages 64–73. ACL, 2013.

[19] Landauer. *Handbook of Latent Semantic Analysis*. Lawrence Erlbaum Associates, 2007. ISBN 0805854185.

[20] Thomas K. Landauer and Susan T. Dumais. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, pages 211–240, 1997.

[21] Dekang Lin. Automatic identification of non-compositional phrases. In *Proc. of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 317–324. ACL, 1999.

[22] Nikola Ljubešić and Tomaž Erjavec. hrWaC and slWaC: Compiling web corpora for Croatian and Slovene. In *Text, Speech and Dialogue*, pages 395–402. Springer, 2011.

[23] Diana McCarthy, Sriram Venkatapathy, and Aravind K Joshi. Detecting compositionality of verb-object combinations using selectional preferences. In *EMNLP-CoNLL*, pages 369–379, 2007.

[24] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *Proc. of ICLR*, Scottsdale, AZ, USA, 2013a.

[25] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013b.

[26] Jeff Mitchell and Mirella Lapata. Composition in distributional models of semantics. *Cognitive science*, 34(8):1388–1429, 2010.

[27] Ted Pedersen. Identifying collocations to measure compositionality: shared task system description. In *Proc. of the Workshop on Distributional Semantics and Compositionality*, pages 33–37. ACL, 2011.

[28] Siva Reddy, Diana McCarthy, Suresh Manandhar, and Spandana Gella. Exemplar-based word-space model for compositionality detection: Shared task system description. In *Proc. of the Workshop on Distributional Semantics and Compositionality*, pages 54–60. ACL, 2011.

[29] Ivan A Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. Multiword expressions: A pain in the neck for NLP. In *Computational Linguistics and Intelligent Text Processing*, pages 1–15. Springer, 2002.

[30] Jan Šnajder, Sebastian Padó, and Željko Agić. Building and evaluating a distributional memory for Croatian. In *In Proc. of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 784–789. ACL, 2013.

[31] Leo Zuanovic, Mladen Karan, and Jan Šnajder. Experiments with neural word embeddings for croatian. In *Proc. of the Ninth Language Technologies Conference, Information Society (IS-JT 2014)*, pages 69–72, 2014.

Denoising Human-Motion Trajectories Captured with Ultra-Wideband Real-time Location System

Rok Piltaver, Božidara Cvetković, and Boštjan Kaluža
Department of Intelligent Systems, Jožef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia
E-mail: {rok.piltaver, boza.cvetkovic, bostjan.kaluza}@ijs.si

Keywords: motion capture data, human motion analysis, filtering, real-time locating system

Received: September 21, 2015

Real-time locating system (RTLS) based on UWB radio technology can be used to track people performing every-day activities. However, the quality of obtained data is relatively low and, therefore it is difficult to perform a reliable advanced analysis of human motion based on it. The paper analyses the noise of RTLS measurements and suggests filtering methods that reduce the impact of the noise on the accuracy of activity recognition. The methods are based on the statistical properties of the noise and human anatomy and motion limitations. First, a rule based method for inserting missing measurement values is suggested and compared with simple insertion of the last known value. Second, an adaptive low-pass filter that reduces impulsive noise is suggested and compared with median filter. Third, a filter that ensures human motion constraints are meet is suggested. In addition, an implementation of Kalman filter that can be used to estimate the missing values, estimate the velocity of movement from recorded locations, and for smooth the signal is described. The advantages and limitations of the suggested filtering approach are demonstrated on synthetic and real data. Finally, influence of each phase of the suggested filtering chain on the accuracy of activity recognition is analysed.

Povzetek: Z UWB redijskim sistemom za lociranje v realnem času je mogoče spremljati gibanje človeka pri vsakdanjih opravilih. Kvaliteta tako zajetih podatkov je relativno slaba, kar otežuje natančne analize gibanja. Prispevek analizira šum tako zajetih podatkov in predlaga postopek za zmanjšanje vpliva šuma na točnost prepoznavanje aktivnosti. Metode so osnovane na statističnih lastnostih šuma in omejitvah, ki izhajajo iz anatomije ter fiziologije človeškega telesa. Za vstavljanje manjkajočih vrednosti predlagamo postopek na osnovi pravil in ga primerjamo z vstavljenjem zadnje znane vrednosti. Za odpravi impulznega šum predlagamo prilagodljiv nizkopasovni filter in ga primerjamo z mediana filtrom. Zadnji v zaporedju je filter, ki zagotovi, da filtrirani podatki ustrezajo omejitvam gibanja človeka. Opisan je tudi Kalmanov filter, ki vstavi manjkajoče vrednosti, oceni hitrost gibanja in odpravi splošen šum. Sistem je ovrednoten na podlagi vpliva vsakega predlaganega filtra na točnost prepoznavanja aktivnosti, prednosti in omejitve filtrov pa so prikazane na sintetičnih in realnih podatkih.

1 Introduction

There is a significant amount of research in human activity recognition, since it is important in many domains such as ambient assisted living, security, sports, and recognition of health problems. The goal of human activity-recognition algorithms [7, 13] is to build a model that maps a sequence of sensor readings (and some additionally computed features) to an activity label, such as walking, sitting, cycling, etc. Such algorithms require that there are no missing values and that the level of noise is low. An important sensor technology that provides sensor measurements, which are useful for human activity recognition, relies on real-time location systems (RTLS) that output 3-dimensional coordinates of tags attached to the human body. High-fidelity optical RTLS such as Vicon [24] and SMART [2] provide accurate measurements (± 2 mm) but often suffer from tag mislabelling due to problems with tracking when tag occlusion happens. Furthermore, they require a line-of-sight between the tag(s) attached to the human body and several cameras. They are appropriate for use in controlled environment (laboratory or animation studio), but fail in real-world applications as they are expensive, difficult to install, and have limitations such as line-of-sight requirement and confined operational area (e.g. 2×3 m). More affordable system relay on radio technology, which makes them less privacy-invasive and cheaper, but less accurate. Systems based on ultra-wideband technology (UWB) such as Ubisense [22] achieve ± 15 cm accuracy in ideal setting, which makes human activity recognition challenging. The main problem addressed in this paper is how to denoise human-motion trajectories captured with UWB RTLS in order to improve human activity recognition. Methods described in the paper can also be applied in other applica-

tions based on UWB RTLS as well.

Denoising human-motion data captured with UWB RTLS raises several challenges. [13, 16]. First, motion-capture data may contain a certain percentage of missing values due to packet loss or temporal sensor malfunction. Second, sensor noise and environment disturbances cause a percentage of motion-capture data to have a high error –so called outliers – and unstable measurements, which corrupt the reconstruction of human-body posture. The noisy data that violates physical body constraints as well as spatial-temporal motion constraints, which in turn causes additional problems for robust human activity recognition. Finally, some essential features used in activity-recognition process that are computed from noisy measurements, such as velocity and acceleration, have an integral error term, which accumulates error over time.

This paper proposes an efficient approach for denoising human-motion trajectories that filters corrupted motion data and enforces spatial-temporal constrains of human body, which enables a more accurate computation of features used by activity-recognition model. The key idea is to apply a series of filters that address the above-mentioned challenges: (i) inserting missing values, (ii) filtering values with high error, (iii) enforcing spatial-temporal constrains of human body, (iv) smoothing the noise, and (v) estimating derived features such as velocity.

The next section presents related work and compares it with the methods suggested in this paper. Section 3 introduces the real-time locating system used in the experiments, describes how it was used to track human motion and gives detailed analysis of the sensor noise. Section 4 gives an overview of the suggested sequence of filters and explains each of them: filters for dealing with missing values are given in Section 4.1, outlier filters are discussed in Section 4.2, filter that enforces spatial-temporal body constraints is proposed in Section 4.3, and filter for smoothing and estimating velocities is given in Section 4.4. Section 5 evaluates the proposed filters. Evaluation on synthetic data is given in Section 5.1, on real data in Section 5.2 and on human activity recognition in Section 5.3. Two applications using the proposed filters are shortly described in Section 6. The paper conclusion is in Section 7.

2 Related Work

Related work from the field of signal processing provides numerous signal denoising methods. This section provides a quick overview of methods for filtering extreme values, enforcing constraints and estimating values from noisy signals.

Qui et al. [19] reviewed and evaluated various impulsive noise filtering techniques for aircraft engine sensor data. Kernel smoothing and local regression method performed best on slowly changing signals such as ramp signal with white noise and outliers. Cascaded recursive median filter performed best on the step change signals with standard deviation of the Gaussian noise lower than half of the change in the signal value.

Verma et al. [23] reached similar conclusions and confirmed that median filter successfully removes outliers while preserving signal features in jet engine gas path measurements.

Sul et al. [21] presented a Kalman filter framework that handles the following problems related to motion capture sensor noise: satisfaction of physical constraints inherent to human body, user-specified kinematic constraints, and noise reduction. The constraints are added to the Kalman filter as an error function that needs to be minimised. The filter also guaranties seamless motion transition between concatenated motion segments and can be used for motion generation.

Musić et al. [14] presented an Extended Kalman filter for filtering sit-to-stand-motion using low cost inertial sensors. They define dynamic human body model and fused it with sensor measurement into an Extended Kalman filter. This approach successfully estimates and filters angles between body segments, angular velocities, angular accelerations, and joint moments.

This paper focuses on measurements captured with UWB RTLS, which are known to have low accuracy. As analysed in Section 3.3, the measurements contain different types of noise, which requires combination of multiple filters. The main contribution of the paper is the complete analysis of a comprehensive set of filters that enable effective sensor readings denoising. This work is based on thfindings presented in our previous papers [4, 5, 6, 15].

3 Location Trajectories of UWB System

This section first introduces the ultra-wide band (UWB) real-time location system (RTLS) used in the experiments. Second, placement of RTLS tags on human body is described according to the needs of activity-recognition – the domain used for evaluation of the proposed denoising approach. Finally, a detailed analysis of UWB RTLS measurement noise is given as it defines the denoising algorithm requirements and points out the essential evaluation tests.

3.1 Ubisense Location System

A commercially available localization system Ubisense [22] was selected as the sensing component. It allows locating by tracking a set of small tags (40 $\times$ 40 $\times$ 16 mm, 25 g), which are attached to a person's body, within an area of up to 30 $\times$ 30 m. A sampling frequency of around 9 Hz can be achieved with no more than four tags simultaneously. Each tag communicates using ultra-wideband radio signal [25] with four to six stationary sensors, for example, mounted on the wall. To calculate the x, y and z position of a tag, both the time

difference of arrival and the angle of arrival of radio signal are used. Location accuracy of about ± 15 cm in each of the three axes can be achieved across approximately 95% of the readings in a typical open environment [22]. However, in a real-life scenarios the absolute measurement error is higher than 100 cm in 1% of measurements, which represents a significant challenge for preprocessing and filtering.

3.2 Tag placement

The effect of body tag placement on classification accuracy was studied in [12], where it turned out that in general more tags enable more accurate classification. However, given large enough noise, even an increased number of tags does not necessarily improve the results. For example, the accuracy of the activity-recognition is comparable when using eight or four tags. Nevertheless, using only one or two tags significantly impacts the classification accuracy. Based on these results and the fact that the Ubisense sampling rate for four tags is limited to 9 Hz, the tags were positioned at the following locations on the body: chest, waist, left and right ankles.

3.3 Noise Analysis

In order to successfully denoise the RTLS measurements, the analysis of noise was conducted first. The Ubisense RTLS was installed and calibrated in a 7×4 m room used for measuring the noise of the RTLS readings. The tags were placed on the following positions (black circles in Figure 1): left ankle, right ankle, waist, and chest. To analyse the noise, static measurements were collected on a grid with 0.5 m density while the person wearing the tags was faced in one of the four directions (north, east, south, west). Over 150 measurements (lasting at least 15 s) were taken at each grid location. As a result, approximately 100,000 measurements were collected at the known locations and orientations. The data was analysed with statistical methods as described bellow.

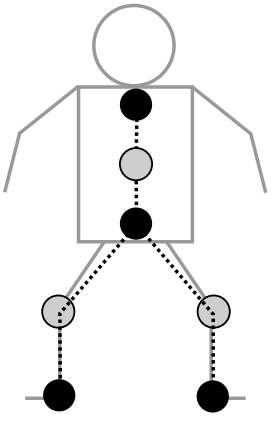


Figure 1: Positions of the RTLS tags on the body.

Figure 2 shows the observed measurement error. The black dots represent the distance from the true position in 3D while the grey crosses represent the 2D projections. Figure 2 illustrates that the noise level makes applications such as activity recognition challenging.

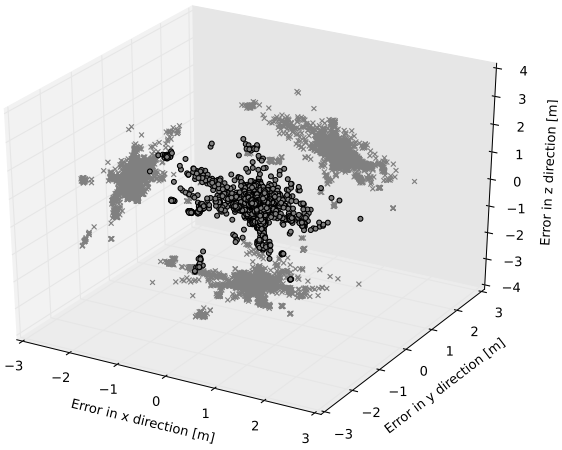


Figure 2: RTLS measurement error in 3D (black circles) with 2D projections (grey crosses).

The noise was further analysed by RTLS error histograms shown in Figure 3 for each of the three directions as well as the combined absolute error. Figure 3 show that the error is the highest in z direction (up-down), and the smallest in x direction. The Shapiro-Wilk test for normality [20] was performed for all three directions and all four tags confirming that the measurement noise is not normally distributed. This was also confirmed by Q-Q plots analysis.

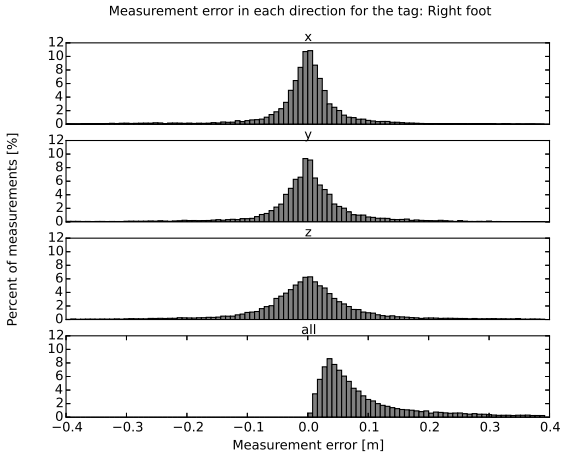


Figure 3: Histograms of RTLS measurement error in all three directions (x, y, z) and the total absolute error.

The standard deviation of measurement error in each direction is between 10.6 and 17.6 cm. Measurement error in single directions is below 10.4 to 29.9 cm (depending on the tag position and direction of error) for 95% of the measurements. The average absolute error is between 9.8 and 14.4 cm depending on the tag placement. The median of the absolute error is between 3.2 and 6.9 cm. The abso-

lute measurement error is below 22.3 to 53.7 cm (depending on tag position) for 95% of the measurements. Only one percent of measurements has absolute error higher than 1.389 m. The tags that are placed higher, for instance, chest, have lower noise level as illustrated by Figure 4.

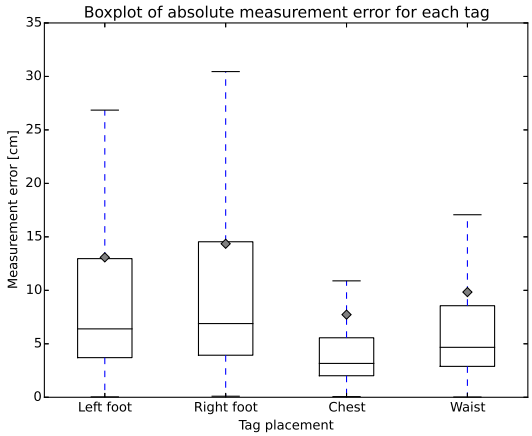


Figure 4: Boxplots (without outliers) of RTLS absolute measurement error for each tag placement. Diamonds represent mean values.

The maximal Spearman rank-order correlation coefficient between error in two directions for any of the four tags is 0.135. This shows that the error in various directions is not correlated. The maximal observed error in single direction is 3.97 m which is more than half of the longest side of the room in which the measurement noise was analysed. Additional noise analysis using auto-correlation confirmed that the noise is random, that is, there is no external process that influences the measurement error.

Up to 0.84% of the RTLS measurements are missing. However, in most cases (0.57%) only one consecutive value is missing, while more than 3 consecutive values (corresponding to 1/3 s or more) between two consecutive measurements, are missing in 0.07% of measurements.

4 Denoisinig Human Motion

This section propose an efficient approach for denoising human-motion trajectories that filters corrupted motion data and enforces the human-body spatio-temporal constraints thus enabling more accurate feature computation.

The key idea is to apply a series of filters as shown in Figure 5 that mitigate the identified measurement errors. First, the missing values are inserted by either inserting the last known value or by using rule-based insertion. Second, the raw RTLS signal of each tag is filtered to remove the impulse noise, using either a median filter or an adaptive low-pass filter. Third, measurements are corrected by a constraint propagation procedure in order to satisfy the following constraints: human anatomic constraints enforcing expected distances between joints and human motion constraints enforcing acceleration and velocity limits. Finally, a Kalman filter is applied in order to smooth the signal and obtain an estimation of velocity, which is an important feature for activity recognition.

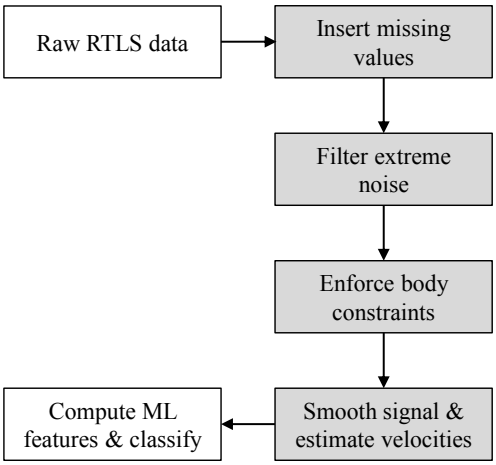


Figure 5: A series of filters for denoising the raw UWB RTLS measurement – preprocessing step for activity recognition.

4.1 Dealing with Missing Values

In most applications, the algorithms for RTLS signal analysis can be simplified if no values are missing and a constant sampling rate is used. However, data from UWB RTLS contains missing values due to packet loss, delay during transmission, sensor failure, or corrupted packets. Therefore, the first step in RTLS denoising is to insert the missing values. This paper compares two approaches described bellow. Other methods, such as Kalman filter (see Section 4.4) or retrospective interpolation, could be used as well.

4.1.1 Insert the Previous Value

A simple method for dealing with the missing values replaces the missing value x_t at time t with the last known value x_{t-1} according to Equation 1.

$$x'_t = x_{t-1} \tag{1}$$

This approach is simple to implement, has a constant time complexity, and can be executed online. The error of the locations inserted using this approach is reasonable if only one or a few consecutive measurements are missing and sampling frequency is high. Nevertheless, if the tag with missing value is moving, applying this method results in a signal with discontinuous derivation – velocity suddenly changes to zero, which is not desirable.

4.1.2 Rule-based Insertion

Rule-based insertion uses height of the person ($height$) and values of the non-missing tag measurements to estimate the values of the missing tag measurements. The locations of

the tags are denoted as follows: c for chest, w for waist, aR for right ankle, and aL for left ankle tag.

This approach uses a simple rule to first identify the activity of the user and then estimate the positions of the missing tags based on the activity, which is treated as the context.

The identification of the activity is done according to the height of the chest tag. If the tag is below 0.65 m the assumed activity is lying, otherwise the activity is upright. If the identified activity is lying, the values of the missing tags are estimated using one of the rules presented in Algorithm 2.

Algorithm 1: Rule-based insertion if the identified activity is lying.

if w *and* aR *and* aL *are missing* **then**
  $w(x,y,z) = (c_x - height/3, c_y, c_z)$ $aR(x,y,z) = (c_x - height, c_y - 0.2, c_z)$ $aL(x,y,z) = (c_x - height, c_y + 0.2, c_z)$
else if w *and* aR *are missing* **then**
  $w(x,y,z) = (1/2(c_x + aL_x), c_y, c_z)$ $aR(x,y,z) = (aL_x, 2 * c_y - aL_y, aL_z)$
else if w *and* aL *are missing* **then**
  $w(x,y,z) = (c_x, 1/2(c_y + aR_y), c_z)$ $aL(x,y,z) = (aR_x, 2 * c_y - aR_y, aR_z)$
else if w *is missing* **then**
  $w(x,y,z) = (c_x, c_y, 2/3c_z)$
else if aR *is missing* **then**
  $aR(x,y,z) = (aL_x, 2 * w_y - aL_y, aL_z)$
else if aL *is missing* **then**
  $aL(x,y,z) = (aR_x, 2 * w_y - aR_y, aR_z)$

If the identified activity is upright, the values of the missing tags are estimated using one of the rules presented in Algorithm 3.

Algorithm 2: Rule-based insertion if the identified activity is upright.

if w *and* aR *and* aL *are missing* **then**
  $w(x,y,z) = (c_x, c_y, 2/3c_z)$
  $aR(x,y,z) = (c_x - height, c_y - 0.2, 0)$
  $aL(x,y,z) = (c_x - height, c_y + 0.2, 0)$
else if w *and* aR *are missing* **then**
  $w(x,y,z) = (c_x, c_y, 2/3(c_z + aL_z))$ $aR(x,y,z) = (2 * w_x - aL_x, 2 * w_y - aL_y, aL_z)$
else if w *and* aL *are missing* **then**
  $w(x,y,z) = (c_x, c_y, 2/3(c_z + aR_z))$ $aL(x,y,z) = (2 * w_x - aR_x, 2 * w_y - aR_y, aR_z)$
else if w *is missing* **then**
  $w(x,y,z) = (c_x, c_y, 2/3c_z)$
else if aR *is missing* **then**
  $aR(x,y,z) = (2 * w_x - aL_x, 2 * w_y - aL_y, aL_z)$
else if aL *is missing* **then**
  $aL(x,y,z) = (2 * w_x - aR_x, 2 * w_y - aR_y, aR_z)$

This approach is constrained by the mandatory availability of the chest tag location upon which the activity is identified.

4.2 Dealing with Outliers

The second filter used in the suggested denoising approach deals with the impulse noise (outliers). As explained in Section 3.3, a few percent of the RTLS measurements are outliers, which should be filtered before other data processing is executed. This paper compares two approaches for outlier filtering explained in the following subsections: the median filter and an adaptive low-pass filter.

4.2.1 Median Filter

Median filter is a non-linear filter that can suppress impulsive, isolated noise without blurring sharp changes in the signal [26]. The filter uses a window of sequential samples with odd length $w = 2n + 1$. At each time step t the filter returns the median of the elements in the window:

$$x'_t = median(x_{t-n}, ..., x_t, ..., x_{t+n}) \quad (2)$$

The only parameter of the median filter is the window length w, which introduces a delay of length $\lfloor w/2 \rfloor$. Too long window may smooth the signal too much, while too short window does not remove the high density noise. A common approach is to choose a window length that preserves the desired signal features and attenuates the impulse noise well.

The majority of computational time for the median filter is spent on calculating the median value of each window, hence an efficient median calculation is crucial for the filter run-time. While a naive approach sorts samples in each window, a histogram-based algorithms implemented with binary search trees are more efficient.

In the case of RTLS data, the median filter is applied at each tag, separately for each dimension. The filter is able to remove isolated spikes in the signal, while parts of the signal with high oscillation remain unsuppressed. However as demonstrated in Section 3.3, the Ubisense RTLS data does not contain long periods with many outliers, which makes the median filter suitable for dealing with outliers.

4.2.2 Adaptive Low-Pass Filter

Another method that filters outliers is the low-pass filter, also termed high-cut filter. It passes signals with a frequency lower than a certain cut-off frequency and attenuates signals with frequencies higher than the cut-off frequency. It provides a smoother form of a signal by removing the short-term fluctuations (outliers) and preserving the longer-term trend. The output x'_t of a discrete low-pass filter is a weighted sum of the input x_t and the preceding output x'_{t-1} for a given constant smoothing factor $0 \leq \alpha \leq 1$ that defines the cut-off frequency:

$$x'_t = \alpha x_t + (1 - \alpha)x'_{t-1} \quad (3)$$

The main idea of the adaptive low-pass filter is to set the smoothing factor α dynamically. If the tag is stationary the cut-off frequency should be lower compared to the cut-off frequency used when the tag is moving. The key challenge is how to detect whether or not the tag is moving.

This is done using movement detection algorithm described in [15]. The algorithm computes a set of attributes from time windows of RTLS data and uses them as the input to the movement detection classifier trained using a machine-learning algorithm. The attributes of the classifier are: average velocity, standard deviation of velocity, average difference between two consecutive velocities, approximate length of travelled path, standard deviation of velocity direction, and average change of direction within the time window. The accuracy of the classifier is above 96%. The miss-classification happens mainly in time windows that include a transition from a stationary state to motion and vice versa. The classifier achieves even higher accuracy during long periods without transitions: when the tag is stationary or when it moves.

The advantage of the adaptive low-pass filter is that the smoothing is dynamically adjusted. Therefore, sequences of stationary locations are smoothed more while the features of the signal during motion are preserved.

4.3 Spatial-Temporal Body Constraints

After the missing values are inserted and the outliers are filtered, more advanced filtering methods are applied. So far each tag was considered as a separated measurement. In reality, however, the tags are attached to a human body, which implies a set of constraints regarding relative tag positions and motion dynamics. In activity-recognition process as well as in other applications using RTLS data, it is usually expected that a set of measurements resembles human body proportions as well as spatial-temporal patterns in natural human motion. Therefore, we developed a filter based on iterative constraint relaxation that: (i) projects measured values in a valid domain; (ii) applies human body proportion constraints to the measured positions; and (iii) constraints spatial-temporal motion patterns.

4.3.1 Mapping Measurements to a Valid Domain

In the first step, an assumption about a valid domain of measurements is made. For example, it is expected that all the measurements are within a cuboid bounded with two extreme points $\mathbf{p}_A$ and $\mathbf{p}_B$ (assuming the coordinate system is aligned with the room) as shown in Figure 6. To keep the measurement $\mathbf{p}_t$ within the expected bounds, it has to be translated to an edge (in case it is not already within the cuboid) as shown in Figure 6. The update step is:

$$\mathbf{p}'_t = min(max(\mathbf{p}_t, \mathbf{p}_A), \mathbf{p}_B). \quad (4)$$

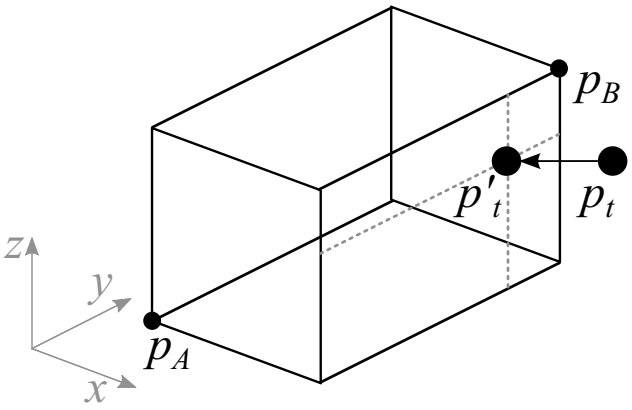


Figure 6: All the measurements are bounded with a cuboid.

4.3.2 Body Constraints

The human body is modelled using rigid-body components, which assume that there is no deformation. Rigid-body components are connected to each other with joints and form an articulated rigid body that approximates the human body as shown with dotted lines in Figure 1. The distance between any two connected joints is constant regardless of external forces exerted on it. Note that at this point on joint constraints are posed.

In our case, the four RTLS tags provide the positions of the joints (ankles, waist and chest), but do not allow the reconstruction of the skeleton displayed in Figure 1 since locations of knees and abdomen are missing. They are reconstructed as follows. Suppose there are two points A and C with known position and a joint B, which interconnects A and C, with an unknown position. Since the distances $r_A = d(A, B)$ (between A and B) and $r_c = d(C, B)$ are known, the point B then lies at the intersection of the two spheres centred at A with radius r_A and at C with radius r_C.

In general, there are three cases to consider: (i) $r_A + r_B = d$, that is, the intersection is a single point; (ii) $r_A + r_B > d$, that is, there is no solution; and (iii) $r_A + r_B < d$, that is, the intersection lies on a circle. In the second case, the point B is positioned on the line between points A and C so that the distances between points is in the same proportion as the lengths of r_A and r_B. In the third case, a new coordinate system is used.to calculate the position of the point B. In the new coordinate system the first sphere is centred at the origin and the second sphere is centred at a point on the positive x-axis, at distance d from the origin, as shown in Figure 7. Subtracting the sphere equations gives a set of points representing a circular intersection of the spheres:

$$x = \frac{d^2 - {r_C}^2 + {r_A}^2}{2d} \quad (5)$$

$$y^2 + z^2 = {r_A}^2 - (\frac{d^2 - {r_C}^2 + {r_A}^2}{2d})^2 \quad (6)$$

The exact position of the point B is not important, hence

an arbitrary point on the circle is picked and transformed to the original coordinate system. As explained below, the distance between joints is enforced with Equations (8) and (9).

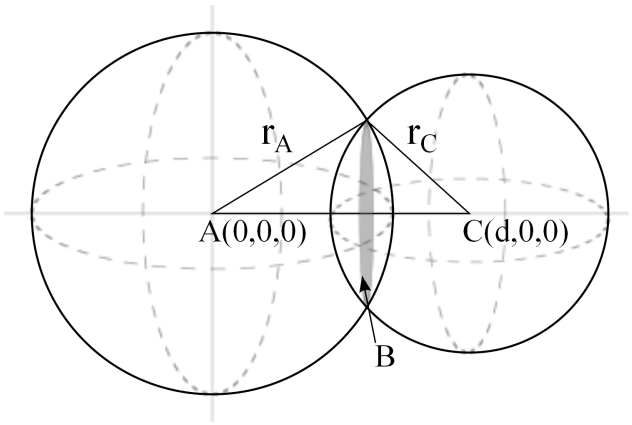


Figure 7: The result of sphere-sphere intersection is a circle.

After all the joint positions are known, constraints between the connected pairs of points can be introduced. For example, suppose the true distance between joints A and B is r_A, that is

$$\|\mathbf{p}_A - \mathbf{p}_B\| = r_A. \tag{7}$$

If measurements p_A and p_B violate the constraint given by the Equation (7), the position of both points is adjusted. Each point is translated along the line connecting the points for half of the error defined as the difference between the measured and the true distance as shown in Figure 8. The update is:

$$\mathbf{p}'_A = \mathbf{p}_A + \frac{\|\mathbf{p}_B - \mathbf{p}_A\| - r_A}{2\|\mathbf{p}_B - \mathbf{p}_A\|}(\mathbf{p}_B - \mathbf{p}_A) \tag{8}$$

$$\mathbf{p}'_B = \mathbf{p}_B - \frac{\|\mathbf{p}_B - \mathbf{p}_A\| - r_A}{2\|\mathbf{p}_B - \mathbf{p}_A\|}(\mathbf{p}_B - \mathbf{p}_A) \tag{9}$$

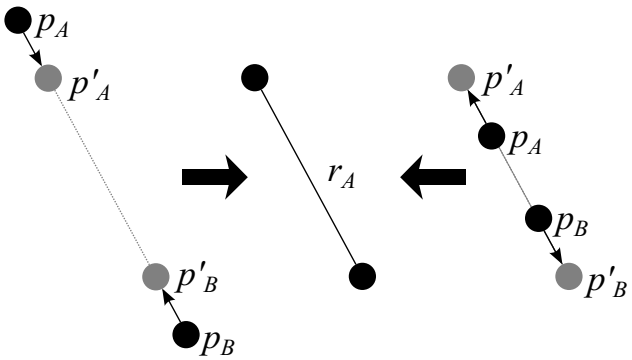


Figure 8: Move the points $\mathbf{p}_A$ and $\mathbf{p}_B$ to match the constraint given by Equation (7).

4.3.3 Spatial-Temporal Motion Patterns

In addition to the constraints introduced by human body proportions, physical motion constraints of limbs (such as velocity and acceleration) are also considered. Suppose that a $[m/s^2]$ is the greatest possible acceleration of a body joint. This implies that it can travel at most $l_{max} = (v_{t-1} + a\Delta t/2)\Delta t$ $[m]$ in time interval Δt, where $1/\Delta t$ is the sampling frequency. Hence the next position of the joint $\mathbf{p}_t$ is limited with a sphere with radius l_{max}, that is

$$\|\mathbf{p}_t - \mathbf{p}_{t-1}\| \leq l_{max} = (v_{t-1} + a\Delta t/2)\Delta t \tag{10}$$

In the case the new position $\mathbf{p}_t$ is outside the sphere, the position is translated onto the edge of the sphere in the direction of the previous position as shown in Figure 9. The update step is:

$$\mathbf{p}'_t = \mathbf{p}_t + \frac{l_{max}(\mathbf{p}_t - \mathbf{p}_{t-1})}{\|\mathbf{p}_t - \mathbf{p}_{t-1}\|} \tag{11}$$

In order to speed up the computations, the sphere can be approximated with a cube with edge length l_{max}. In this case, the new position is limited using Equation 4 where:

$$\mathbf{p}_A = p_{t-1} - (l_{max}, l_{max}, l_{max}) \tag{12}$$

$$\mathbf{p}_B = p_{t-1} + (l_{max}, l_{max}, l_{max}) \tag{13}$$

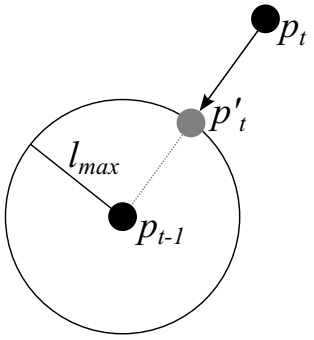


Figure 9: Constrain the maximal distance according to Equation (11).

4.3.4 Iterative Constraint Relaxation

Finally, the three types of constraints are put together. Consider $\mathbf{C} = \{\Phi_i\}$ as a set of constraints, where $\Phi(\mathbf{p})$ applies the update step on point $\mathbf{p}$ first using Equation (11) and then using Equation (4); that is, $\mathbf{p}' \leftarrow \Phi(\mathbf{p})$, while $\Phi(\mathbf{p}_A, \mathbf{p}_B)$ applies the update step on both points A and B using Equations (8) and (9); that is, $\mathbf{p}'_A, \mathbf{p}'_B \leftarrow \Phi(\mathbf{p}_A, \mathbf{p}_B)$. If a constrained between points A and B is not present, then $\Phi(\mathbf{p}_A, \mathbf{p}_B)$ does not alter the corresponding points. The algorithm 4 takes the set of constraints $\mathbf{C}$ and set of points points $\mathbf{P}$ as an input and iteratively updates the values until the convergence threshold τ_c or maximal number of iterations k is reached [1].

Algorithm 3: Iterative constraint relaxation.

Data: set of constraints $\mathbf{C}$, set of points $\mathbf{P}$, maximal number of iterations k, convergence threshold τ_c
Result: filtered set of points $\mathbf{P}$

```
while k > 0 and Δ > τ_c do
    Δ = 0;
    for p ∈ P do
        for q ∈ P do
            p', q' ← Φ(p, q);
            Δ ← Δ + |q − q'|;
            q ← q';
        p' ← Φ(p);
        Δ ← Δ + |p − p'|;
        p ← p';
    k ← k − 1;
```

4.4 Smoothing and Velocity Estimation

The final step of RTLS denoising smooths the signal and estimates additional quantities such as velocity, which are used as attributes in activity-recognition process. For this task, Kalman filter [3] is used. It is a recursive linear filter for determining the best estimation of the system's state. The main assumption of the Kalman filter is that the underlying system is a linear dynamical system and that all measurement errors have a multivariate Gaussian distribution. In our case the system is a single RTLS tag moving in 3D space.

The Kalman filter performs the following three tasks: smooths the UWB measurements, estimates the velocities of tags, and predicts the missing measurements. In our case, the Kalman filter state is a six dimensional vector $\mathbf{x}_t$ that includes positions and velocities in each of the three dimensions at time t, $\mathbf{x}_t = [p_{x,t}, p_{y,t}, p_{z,t}, v_{x,t}, v_{y,t}, v_{z,t}]^T$. The next state is estimated from the previous state using the following equation:

$$\mathbf{x}_{t+1} = \mathbf{F}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t + \mathbf{w}_t, \tag{14}$$

where matrix $\mathbf{F}$ encodes the linear dynamical system, $\mathbf{B}$ is a control matrix and $\mathbf{w}_t$ is noise covariance matrix. In our case, the Kalman update is simplified to Equation 15. The next state is calculated as a sum of the previous position and a product between the previous velocity and the sampling interval Δt for each direction separately. The velocities remain constant. The measurement noise covariance matrix is set based on UWB RTLS specification, while the process noise covariance matrix is fine-tuned experimentally.

$$\begin{bmatrix} p_{x,t+1} \\ p_{y,t+1} \\ p_{z,t+1} \\ v_{x,t+1} \\ v_{y,t+1} \\ v_{z,t+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \Delta_t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta_t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta_t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_{x,t} \\ p_{y,t} \\ p_{z,t} \\ v_{x,t} \\ v_{y,t} \\ v_{z,t} \end{bmatrix} + \mathbf{w}_t \tag{15}$$

5 Results

The proposed denoising method was tested on synthetic data, real data, and in an activity-recognition application. The evaluation is based on the insights obtained by the noise analysis and shows that the proposed method successfully deals with missing values, outliers and general noise while preserving the original signal features.

5.1 Evaluation on Synthetic Data

The proposed denoising method was first evaluated on synthetic data. The locations of the four tags (ankles, waist and chest) were simulated as follows. The time-series starts with a period of normal positions of the four tags during standing – simulating a period with no noise. In the second phase, the signal is corrupted by moving the chest tag into outlier position – simulating a long series of outliers. Finally the position of the chest tag returns to normal.

Figure 10 shows x, y, and z coordinates of two tags. The blue line represents the synthetic signal, the green line represents the output of the median filter, the red line represents the output of the Spatio-temporal body constraints filter, and the pink line represents the output of the Kalman filter, that is, the final denoising result.

Figure 10 shows that the output of median filter follows the noisy signal due to a long period of outliers. Short periods of outlier values are successfully filtered by the median filter as explained in the next section. On the other hand, the filter that enforces the human-body proportion and motion constraints follows the noisy signal conservatively. This significantly reduces the error of the chest tag, however it introduces a relatively small error of the waist tag while the ankle tags are not affected. The Kalman filter smooths the transition. This example illustrates the benefits of the Spatio-temporal body constraints filter on the noisy data that can not be filtered appropriately using median filter alone.

5.2 Evaluation on Real Data

An example of filter response is shown in Figure 11, which shows x (top) and y (bottom) coordinates for a tag attached to the waist for $T = 600$ time steps (approximately 67 s). The vertical axis corresponds to the position of the tag in meters. The blue line represents the original location measurements, the green line represents the median filter

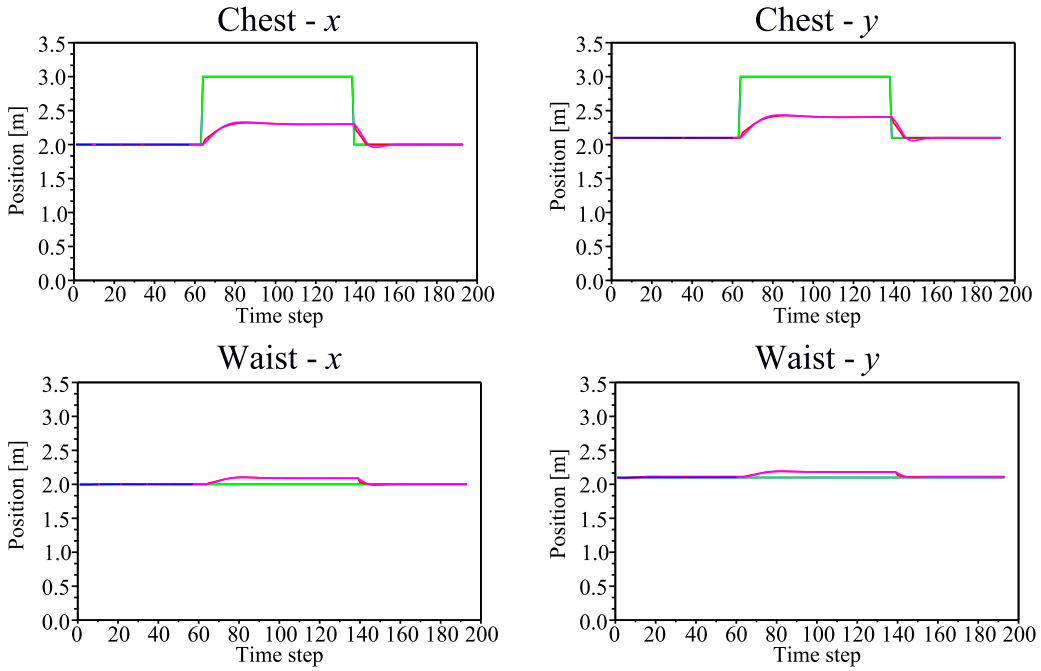


Figure 10: Response of filters on synthetic data with a long period of outliers: measurements of chest tag have a high error for extended period of time.

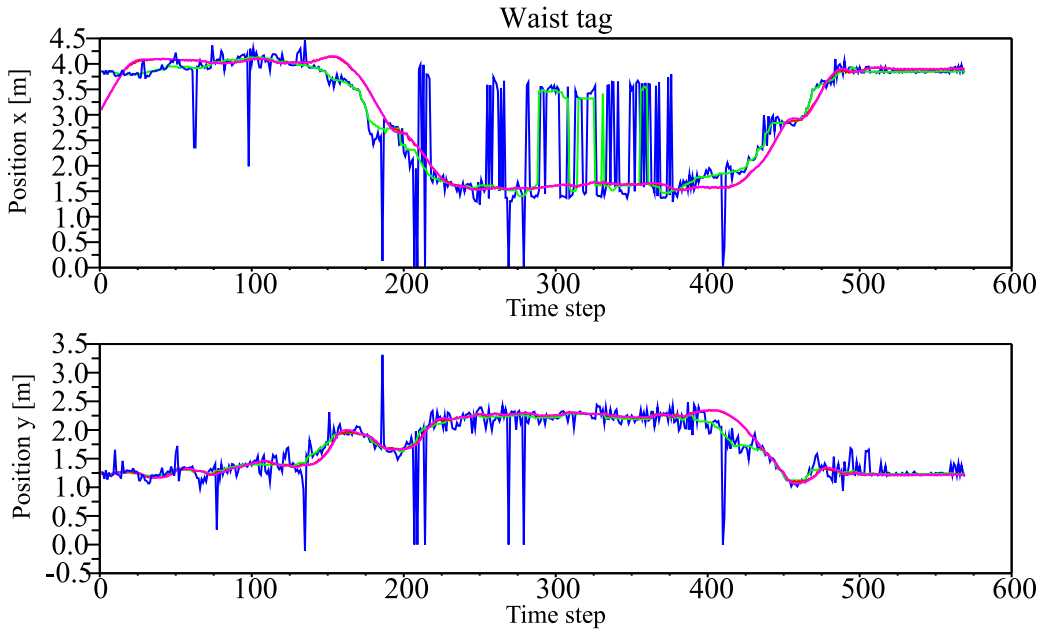


Figure 11: Filtered coordinates x (top) and y (bottom) of a tag attached to the waist.

output, and the red line represents the output of Spatio-temporal body constraints filter followed by the Kalman filter.

Figure 11 illustrates that the median filter successfully filters sparse outliers while it fails when it encounters a long sequence of outliers. In addition it shows that Spatio-temporal body constraints filter successfully correct such errors using the information about positions of the other tags.

5.3 Impact on Human Activity Recognition

In order to quantitatively evaluate the proposed denoising method, combinations of suggested filters were used as the preprocessing step that filtered the raw RTLS data for activity-recognition [6, 9, 10]. The effect on the activity-recognition classification accuracy is analysed in order to evaluate the effect of each suggested filter. The test dataset includes 32,000 to 55,000 instances for each of the ten persons, amounting to over 400,000 instances in total. Leave-one-person-out method was performed to estimate the classification accuracy using each subset of filters (used for raw-data preprocessing). The results are shown in Table 1. One-tailed paired t-test was performed to calculate the statistical significance of differences in classification accuracy (shown in Figure 12) between using various subsets of denoising filters in pre-processing step.

ID	p	r	r+m	r+l	r+m +b	r+m +b+K
A	62.5	65.6	65.4	62.8	76.1	76.8
B	61.2	61.5	60.6	58.9	72.6	73.0
C	55.2	58.0	62.6	59.7	73.7	74.4
D	65.3	67.8	68.7	65.6	74.8	76.6
E	60.4	64.4	68.4	63.4	67.7	68.1
F	64.2	64.1	62.0	61.1	74.2	74.5
G	59.0	59.0	59.2	58.2	71.9	72.2
H	56.5	59.8	65.8	61.5	69.2	72.2
I	61.9	62.1	64.6	63.3	76.3	76.5
J	63.5	64.0	64.0	63.0	75.4	75.4
$\bar{x}$	**61.0**	**62.6**	**64.1**	**61.7**	**73.2**	**74.0**
σ	3.1	3.0	3.0	2.2	2.7	2.6

Table 1: Accuracy of activity recognition (in %) using subsets of filters for preprocessing.

First, the two methods for inserting the missing values are compared. Classification accuracy using rule-based insertion ($\bar{x} = 62.6\%, \sigma = 3.0$) is significantly higher ($p = 0.005$) compared to inserting the last known value ($\bar{x} = 61.0\%, \sigma = 3.1$).

Second, the two methods for filtering impulse noise are compared. They are applied after the rule-based insertion of missing values. Classification accuracy using median filter ($\bar{x} = 64.1\%, \sigma = 3.0$) is higher compared to using the adaptive low-pass filter ($\bar{x} = 61.7\%, \sigma = 2.2$). However fine-tuning the parameters of the adaptive low-pass filter could improve its influence on the classification

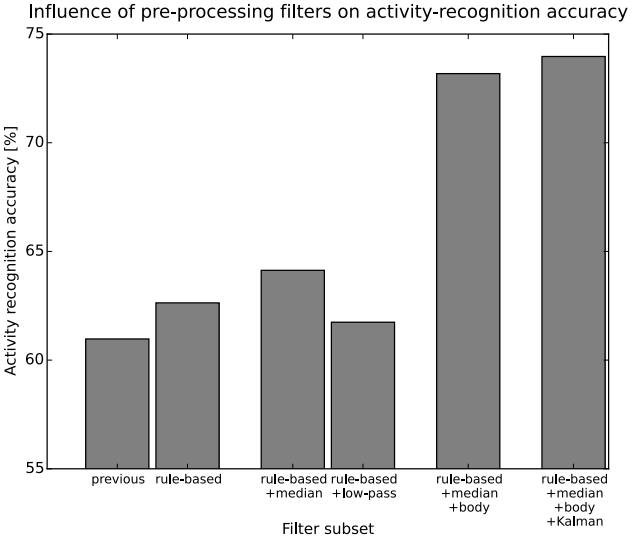


Figure 12: Comparison of activity-recognition accuracy using various subsets of filters for raw-data pre-processing.

accuracy. Furthermore, the observed difference in classification accuracy means that the adaptive low-pass filter is worse than the median filter in this application, however it does not mean that this is so in general. We argue that the adaptive low-pass filter is preferred to median filter for stationary RTLS tag positions since the accuracy of detecting stationary sequences is high (>96%). Adding median filter after rule based insertion of missing values significantly improves classification accuracy ($p = 0.054$) compared to only inserting the missing values.

Third, adding the body filter after the rule-based insertion and median filter significantly ($p \approx 0$) increases classification accuracy to $\bar{x} = 73.2\%$ ($\sigma = 2.7$). Finally, adding Kalman filter at the end of the filter chain significantly ($p = 0.013$) increases classification accuracy to $\bar{x} = 74.0\%$ ($\sigma = 2.6$).

6 Applications

The proposed set of filters was successfully used in two applications. The first is an intelligent system for surveillance of personnel and equipment movement that triggers an alarm when unusual or forbidden activities are detected. The second one is a care system that detects abnormal events (such as falls) or unexpected behaviour that may be related to a health problem in elderly people. Each of the two application and the importance of denoising for robust operation of the application is briefly described below.

The first application is Commander's Right Hand [16, 17, 18]. It is an intelligent system for surveillance of movement of personnel and equipment in high security indoor environment. It uses a RTLS system to monitor movement of personnel and important equipment. It learns a model of the usual behaviour and compares it with the currently observed movement in order to detect abnormalities.

The goal is to alarm the commander about unusual and forbidden activities and enable centralized overview of monitored environment and analysis of past events. Comparison of motion detected by the RTLS and the motion detected by intelligent video surveillance detects motion that is not monitored by the RTLS, which triggers an alarm. Furthermore, the expert system module enables a simple way of specifying prohibited behavior in terms of forbidden motion patterns.

The system uses the filters for inserting the missing values, filtering the outliers, detecting motion, identifying basic activities (lying, sitting, and standing), smoothing the motion trajectories and estimating movement velocities. Only one RTLS tag per person is used is the system, therefore the Spatio-temporal body constraints filter can not be applied.

The second application is the Confidence system [6, 8, 10, 11]. It collects RTLS information about movement of an elderly that lives home alone and wears four tags positioned as described in this paper. The short term motion analysis detects unusual events such as a fall or an accident. The system triggers a rapid actuation of the health services, which decreases the negative consequences of the accident (worsening of injuries, psychological impact of being alone and injured, etc.). The long term motion analysis detects deviations in motion patterns and elderly habits which often correspond to changes in persons health. For instance, when the person's health status is worsening, there is usually less activity. It reflects in longer periods of lying and sitting, less walking and slower speed of walking and general movement. In addition, frequency of visits to the toiled often increases. When such deviation from normal behaviour is detected, the system notifies the caretaker in order to check on the elderly.

The Confidence system [6] uses the denoising method described in this paper as a preprocessing step for motion analysis described above. The denoising provided by the proposed filters significantly improves performance of activity recognition and modelling of the usual behaviour as well as simplifies motion analysis software.

7 Conclusion

The paper analyses the noise of commercially available real-time location system (RTLS) based on ultra-wide band radio technology. The results of the analysis are used to design a series of efficient denoising filters integrated into a denoising method consisting of four steps. The effect of suggested filters on the accuracy of activity recognition (based on RTLS data) is analysed to evaluate the filters. The first step inserts the missing values. A rule-based insertion method is suggested and shown to enable significantly higher accuracy compared to insertion of the last known value. The second step filters the measurements with high error – so called outliers. A low-pass filter with dynamically adapted parameters based on the motion detection is suggested and compared with median filter. It is empirically shown that the suggested motion detection algorithm achieves accuracy of 96%, which enables adaptive filtering. The third step enforces spatial-temporal constraints of human-body proportions and motion, which additionally reduces the noise by exploiting information about location of the other tags attached to the same person. The filter significantly improves the accuracy of activity-recognition. The fourth and final step smooths the signal and estimates motion velocities, which are used as attributes for activity recognition. If the filter parameters are set correctly, the sequence of filters successfully attenuates RTLS noise while preserving the features of the observed motion. This simplifies the software for intelligent motion analysis and improves its accuracy. Furthermore, the advantages and limitations of suggested filters and their interaction are illustrated on synthetic and real data.

Acknowledgments

The experiments were made within the EU FP7 project Confidence and national project Commander's Right Hand. The authors are very thankful to both project teams, in particular to Mitja Luštrek, Matjaž Gams, Domen Marinčič, Erik Dovgan, Violeta Mirchevska, Blaž Strle, and other colleges from the Department of Intelligent Systems at the Jožef Stefan Institute, as well as the anonymous volunteers that made experimental recordings possible and Robert Jakomin who made the RTLS noise recordings.

References

[1] R. Barrett, M. Berry, T. F. Chan, et al. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd edition*. Philadelphia, 1994.

[2] BTS Bioengineering Corp. *BTS SMART-DX*. `http://www.btsbioengineering.com/products/kinematics/bts-smart-dx/`, 12-03-2015.

[3] R. E. Kalman. *A New Approach to Linear Filtering and Prediction Problems*. Transaction of the ASME – Journal of Basic Engineering 35, pp. 35–45, 1960.

[4] B. Kaluža and E. Dovgan. *Glajenje trajektorij gibanja človeškega telesa zajetih z radijsko tehnologijo*. Proceedings of the 13th International Multiconference Information Society – IS 2009, vol. A, pp. 97–100, 2009.

[5] B. Kaluža. *A Unified Framework for Detection of Suspicious and Anomalous Beahvior from Spatio-Temporal Traces*. Informatica (Slovenia) 38(2), 2014.

[6] B. Kaluža, B. Cvetković, E. Dovgan, H. Gjoreski, M. Gams, M. Luštrek. *Multiagent Care System to Support Independent Living*. International Journal on Artificial Intelligence Tools 23(1), 2013.

[7] B. Kaluža, M. Gams. *Analysis of Daily-Living Dynamics*. Journal of Ambient Intelligence and Smart Environments 4(5), pp. 403–413, 2012.

[8] B. Kaluža, V. Mirchevska, E. Dovgan, M. Luštrek, M. Gams. *An Agent-based Approach to Care in Independent Living*. Lecture Notes in Computer Science 6439, pp. 177–186, AmI 2010, 2010.

[9] B. Kaluža, M. Gams. *An Approach to Analysis of Daily Living Dynamics*. Proceedings of the WCECS 2010, vol. 1, pp. 485–490, 2010.

[10] B. Kaluža, M. Luštrek, E. Dovgan, M. Gams. *Context-Aware MAS to Support Elderly People*. AAMAS 2012, Valencia, Spain, June 2012.

[11] M. Luštrek, B. Kaluža, B. Cvetković, E. Dovgan, H. Gjoreski, V. Mirchevska, M. Gams. *Confidence: Ubiquitous Care System to Support Independent Living*. ECAI 2012.

[12] M. Luštrek, B. Kaluža, E. Dovgan, B. Pogorelc, and M. Gams. *Behavior Analysis Based on Coordinates of Body Tags*. Lecture Notes in Computer Science 5859, pp. 14–23, AmI 2009, 2009.

[13] M. Luštrek, B. Kaluža. *Fall detection and activity recognition with machine learning*. Informatica 33(2) pp. 205–212, 2009.

[14] J. Musić, R. Kamnik, M. Munih. *Model based inertial sensing of human body motion kinematics in sit-to-stand movement*. Simulation Modelling Practice and Theory 16, pp. 933–944, 2008.

[15] R. Piltaver. *Strojno učenje pri načrtovanju algoritmov za razpoznavanje tipov gibanja*. Proceedings of 11th International multiconference Information Society – IS 2008, pp. 37–40, 2008.

[16] R. Piltaver, E. Dovgan, M. Gams. *An intelligent indoor surveillance system*. Informatica 35(3) pp. 383–390, 2011.

[17] R. Piltaver, B. Pogorelc, M. Gams. *Ambient intelligence for indoor surveillance* Proceedings of International Joit Conference on Ambient Intelligence – AMI 2011, pp. 5–8, 2011.

[18] R. Piltaver, M. Gams. *Expert system as a part of intelligent surveillance system* Proceedings of 18th International Electrotechnical and Computer Science Conference -– ERK 2009, vol. B, pp. 191–194, 2009.

[19] H. Qiu, N. Eklund, N. Iyer, X. Hu. *Evaluation of Filtering Techniques for Aircraft Engine Condition Monitoring and Diagnostics*. Proceedings of the International Conference on Prognostics and Health Management, pp. 1–8, 2008.

[20] S. S. Shapiro, M. B. Wilk. *An analysis of variance test for normality (complete samples)*. Biometrika 52 (3–4), pp. 591-–611, 1965.

[21] C. W. Sul, S. K. Jung, K. Wohn. *Synthesis of Human Motion using Kalman Filter*. Proceedings of the International Workshop on Modelling and Motion Capture Techniques For Virtual Environments 1537, pp. 100–112, 1998.

[22] Ubisense GmbH. *In-building Location Systems*. `http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4449097`, 12-03-2015.

[23] R. Verma, R. Ganguli. *Denoising Jet Engine Gas Path Measurements Using Nonlinear Filters*. IEEE/ASME Transactions on Mechatronics 10(4), pp. 461–464, 2005.

[24] Vicon Motion Systems Ltd. *Vicon Bonita*. `http://www.vicon.com/System/Bonita`, 12-03-2015.

[25] Wikipedia. *Ultra-wideband*. `http://en.wikipedia.org/wiki/Ultra-wideband`, 12-03-2015.

[26] L. Yin, R. Yang, M. Gabbouj, Y. Neuvo. *Weighted median filters: a tutorial*. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing 43(3), pp. 157–192, 1996.

PGO-DLLA: Parallel Grid Optimization by the Daddy Long-Legs Algorithm for Preventing Black Hole Attacks in MANETs

Khalil I. Ghathwan
School of Computing, Universiti Utara Malaysia, Kedah, Malaysia;
Department of Computer Sciences, University of Technology, Baghdad, Iraq
Email: k.i.ghathwan@gmail.com, s93453@student.uum.edu.my

Abdul Razak Yaakub
School of Computing, Universiti Utara Malaysia, Kedah, Malaysia
Email: ary321@uum.edu.my

Keywords: mobile ad hoc networks (MANETs), black hole attack, swarm intelligence, malicious node, secure routing, optimization

Received: December 10, 2014

Mobile ad hoc networks (MANETs) are wireless networks that are considered a good alternative to the other types of networks during the hardest times such as wars or natural environment disasters. MANETs have the capability of working without any need for base stations or infrastructures. However, MANETs are subject to severe attacks, such as the black hole attack. Many researchers in the field of secure routing and network security have been working on acceptable solutions to prevent black hole attacks in MANETs. Unfortunately, most of the proposals are not attainable or have performance difficulties. One of the most ambitious goals in the research is to find a way to prevent black hole attacks without decreasing network throughput or increasing routing overhead. Swarm intelligence is a research area for information models that studies the collective behavior of insects or animal swarms. Some algorithms have been proposed to address black hole attacks through new protocols and improving routing security with swarm intelligence. In this paper, we propose a parallel grid algorithm for MANETs that optimizes both routing discovery and security in an Ad Hoc On-Demand Distance Vector (AODV). The new technique, called Parallel Grid Optimization by the Daddy Long-Legs Algorithm (PGO-DLLA), simulates the behavior of the biological spiders known as daddy long-legs spiders. Experiments were conducted on an NS2 simulator to demonstrate the efficiency and robustness of the proposed algorithm. The results indicate better performance than the AntNet algorithm with respect to all metrics except throughput, for which AntNet is the better algorithm. In addition, the results show that PGO-DLLA outperforms the standard AODV algorithm in simulations of both a peaceful environment and a hostile environment represented by a black hole.

Povzetek: Razvit je algoritem za obrambo pred napadi na omrežja MANET.

1 Introduction

Worldwide, there are more than 30,000 kinds of spiders, which are characterized by a unique way of hunting prey. Most types of spiders respond to vibrations that come from their web. Spiders have special methods for quick access to prey and capture them as soon as possible. Some vibrations coming from the web may signal a source of danger, and changing strategies is essential for avoiding the threat [1]. In this paper, we propose a new algorithm for parallel grid optimization that simulates the behavior of daddy long-legs spiders (PGO-DLLA). This type of spider responds to the first vibration that comes from the web and chooses the shortest path to catch the prey without giving it a chance to escape from the trap [2]. Spiders have a huge number of strategies to capture prey, such as trapping the prey in a sticky web [1], [2]. In the case of daddy long-legs spiders, all paths in the web are available for access to a destination, because daddy long-legs spiders do not use the glue that other spiders use. The absence of glue on the yarns of daddy long-legs spiders provides them with unique features, such as the ability to change their location in the web to avoid any dangers coming from outside the web. In addition, when there is more than one source of vibrations, the daddy long-legs spider chooses the smallest vibration frequency value to avoid the risk. This spider is sometimes called a hopper spider because it generates inverse artificial vibrations [2], which can be useful to tighten restrictions on its prey or to discard the prey when it is an unwanted kind. Daddy long-legs spiders are slightly different from other spiders because they have high sensing precision

using their eight legs, which act like sensors or agents to receive signals or to discover their prey's position.

A MANET contains many varieties of dynamic nodes. The network can be active in an actual environment without any infrastructure [3]. MANETs have numerous implementations in several fields, including emergency operations, military operations, civilian environments, and personal area networking [4]. However, they suffer from several limitations, such as short battery lives, limited capacities, and vulnerability to malicious behaviors. A black hole is one type of attack that occurs in MANETs. Black hole nodes attack routing protocols such as the AODV protocol [5], causing network packets to be dropped. The main goal of the AODV protocol is to find a path from a source to its destination node and then to forward the packets. The routing mechanism in AODV uses route requests (RREQs; for discovering routes) and route replies (RREPs; for receiving paths). However, this mechanism is vulnerable to attacks by malicious black hole nodes that can easily adjust the values of routing table fields such as *hop count* and *DSN* in order to deceive the source node after sending a RREQ, a source node will respond to the first RREP it receives. This RREP may be from a black hole node, and the source will not reply to other intermediate nodes. This could cause the end of cooperative work in MANET [3], [6], [7], [8], [9], [10].

Intensive computations are required to make AODV secure against black hole attacks [11]. Most of the proposed solutions with limited computations such as trusting neighboring nodes, using cross-layer cooperation, or allowing route redundancy are fail to detect cooperative black hole attacks [6], [8], [9]. However, use of intensive computations as a solution to cooperative black hole attacks may lead to depletion of the limited energy of batteries. In this paper, we develop methods to find the shortest secure path and reduce overhead using the information that is available in the routing tables [12], [13]. However, we use this information as input to propose a more complex algorithm using swarm intelligence. Mathematical formulas such as Hooke's law [14] and, Newton's second and third laws [15] are utilized to evaluate the route reply and choose the best path. For example, the vibration between two nodes, depending on Hooke's law.

The remainder of this paper is organized as follows Section 2 discusses some related work, Section 3 presents the proposed approach and methodology, and Section 4 presents the solution scenarios and parameters. Finally, Section 5 concludes the paper.

2 Related work

In [16], the authors proposed a new taxonomy to classify approaches to detecting black hole attacks in MANETs. This taxonomy classifies processes according to their computation: whether they are computationally limited or computationally intensive. In this taxonomy, the computationally limited approaches are simple processes that use network parameters, while the computationally intensive approaches use artificial intelligence techniques such as mobile agents, genetic algorithms, clustering, and fuzzy logic to implement the detection. Some approaches to detecting and defending against black hole attacks in MANETs are proposed in [6], [7], [8], [9], [10].

In [6], an anti-cooperative solution to black hole attacks that modified the standard AODV protocol was proposed. In the modified protocol, a source node does not respond directly when it receives the first RREP, but rather waits for a specific period of time. The source node has a cache list to save all RREPs and all details about the next hop that it gathers from other nodes. It then chooses the correct path from a list of response paths after checking for a repeated next hop node, and if there are none, it chooses a random path. The new contribution of this study was the use of a "fidelity table" and assigning fidelity levels to the participating nodes. The important point in their study is that it proposes a solution for collective black hole attacks. However, this method suffers from an increase in the control overhead, because of the exchanges of fidelity packets to achieve security.

In [7], a dynamic anomaly training method, which is one of the learning methods in data mining, was used. The authors create a database that contains the features that result from attacks to compare these with a regular network status. They use statistical theory to produce an anomaly threshold by measuring a projection distance. This method can detect black holes in AODV with low overhead, but false positives are a major drawback.

In [8], the authors suggested a method based on the fact that attackers rely on changing the destination sequence number to the maximum number and will therefore acquire the routing and drop the packets.

In [9], the authors suggested a computationally limited method to detect black hole attacks during the routing discovery in the AODV protocol. This method combines the technique of trusting neighboring nodes with a route redundancy message parameter. When intermediate nodes receive a RREQ from a source node, they send back a new RREP contain a sequence number (SN) to the source node. At the same time, the intermediate node will generate a newly defined SN request (SREQ) and send it to the destination node through the route. The destination node receives the SREQ message and sends a SN reply (SREP) message containing its SN. The source node saves each new SN that it obtains from the destination node in a special field of a SN table (SNT), for comparison with the current sequence source number. The exchanges of the RREQs, RREPs, SREQs, and SREPs in the entire network between the source and destination nodes increase the control overhead. This method is probably not effective in a large topology that has high mobility.

In [10], an algorithm was proposed that provides a security mechanism in AODV by trusting neighboring nodes based on feedback from other nodes and their reputations in the network. This is a distributed collaborative approach for ad hoc wireless networks. In this algorithm, each node does intrusion detection system (IDS) work locally and independently, but nearby nodes work together to monitor a larger area. Each node is

responsible for overseeing the activities involving local data. If an anomaly is detected in the local data, or if the evidence is not sufficient and requires a more comprehensive search, neighboring IDS agents cooperate on the global intrusion detection. However, this algorithm has a high routing control overhead.

3 Proposed approach and methodology

In this paper, we propose a new mechanism that works as an intelligent swarm algorithm based on the VDLLA algorithm, which is integrated into the AODV routing protocol. The new technique, which is intended to enhance security in the AODV protocol, is called Parallel Grid Optimization by the Daddy Long-Legs Algorithm (PGO-DLLA). It tries to reduce financial and technical constraints by reducing the number of hops in the route discovery for finding the destination. This algorithm is proposed in order to reduce the severity of black hole attacks and eliminate them.

3.1 Virtual Daddy Long-Legs Algorithm (VDLLA)

The VDLLA is a swarm of spiders. We assume that each spider has nine positions represented as a 3×3 matrix in a grid space, where eight of the positions are for the spider's eight legs and the center position is for the spider's body. Each spider evaluates the nine positions based on the objective function and determines the best location from the nine positions. The best position for each spider is then evaluated to choose a global position. The computational procedure of the VDLLA is as follows.

Step 1: Generate Initial population of spider members, considering N as the total number of members.

Step 2: Generate Initial location for each body of spider members randomly, and then calculate the legs position based on body position:
Assume the body position = (X, Y), the legs position is eight direction where: from up = (X,Y+0.1), from down =(X,Y-0.1), from left =(X-0.1,Y), from right = (X+0.1,Y), from up left = (X-0.1, Y+0.1), up right =(X+0.1,Y+0.1), from down left =(X-0.1,Y-0.1) and downright =(X+0.1, Y-0.1) as shown in Table 1 below.

Table 1: The positions of each agent (spider).

Leg5 = (X-0.1, Y+0.1)	Leg1 = (X,Y+0.1)	Leg6 = (X+0.1,Y+0.1)
Leg3 = (X-0.1,Y)	body = (X,Y)	Leg4 = (X+0.1,Y)
Leg7 = (X-0.1,Y-0.1)	Leg2 = (X,Y-0.1)	Leg8 = (X+0.1, Y-0.1)

Step 3: Evaluate the fitness for each agent (spider) where the evaluation includes all position of agent (body + legs).

Step 4: Select the best fitness for each agent (spider) and save the position as best position.

Step 5: Select the global fitness from all best fitness and save the position as global position.

Step 6: Do while global fitness greater than tolerance value (tolerance value is based on objective function).

Step 7: Find new position for each agent where the body move to best position and legs position change based on body.

Step 8: Find new best fitness and new global fitness.

Step 9: If new global fitness less than global fitness.

Step 10: Global fitness = new global fitness.

Step 11: Else if new global equal global fitness

Step 12: Change the global position using (1) below:

$$Gpos_{new} = Gpos_{old} + 0.01(RND(1,d)) \quad (1)$$

Where, d is the dimension of objective function.

Step 13: iteration=iteration +1

Step 14: End while.

3.2 Problem formulation and solution representation

Aggregative conduct or swarm behavior in animals or insects is intelligent behavior of their biological group. The study of swarm intelligence is aimed at understanding the behavior of a group in nature. Biological scientists have found that many models can mimic the living systems of animals or insects.

Most spiders do not live in communities, so swarm intelligence does not reflect the collective behavior directly: rather, in this research we consider the sensitive behavior of spider legs to represent the collective performance. This approach is a relatively new orientation in the area of swarm intelligence. It is very important to develop new frameworks, which may be very useful in highly dynamic routing networks, in this area. We apply the new algorithm to MANETs, to address the problem of black hole attacks in the AODV routing protocol. The new proposal is based on the daddy long-legs spider's behavior in nature, as described in the next section.

3.3 The proposed PGO-DLLA algorithm

In AODV routing protocol, each node has a routing table which includes the information such as; hop count, destination sequence number (DSN), life time, source IP address. PGO-DLLA have three routing tables; the first table (L1) contains a source sequence number (SSN), destination sequence number (DSN) and lifetime of the leg (LTL1). The second table (L2) contains SSN, DSN, and the force (F). The third table is the routing table that contains all a routes discovery (RD), current route discovery (CRD), life time (LT), and the best route (BR) to destination node. Figure 1 illustrates the PGO-DLLA routing tables.

L1	L2	Routing Table
SSN DSN LTL1	SSN DSN F	RD CRD LT BR

Figure 1: PGO-DLLA routing tables.

The route discovery in PGO-DLLA is shown in Figure 2. The spider sends an agent (L1), to neighboring nodes to discover the route to the destination (prey).

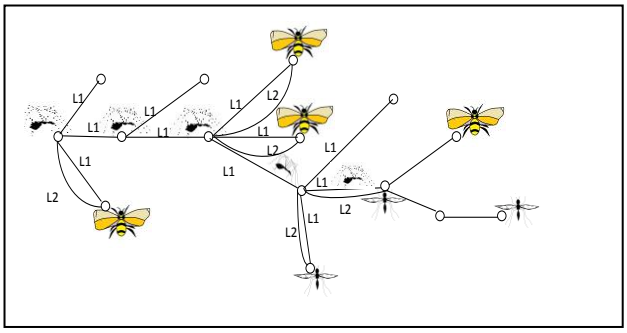


Figure 2: The route discovery in PGO-DLLA.

After broadcasting legs to all neighbor nodes, the spider (source) waits for a lifetime (LT) for receive (L2), if source receives L2 that means this node has a route to the destination or it is a destination. Then, the source node evaluates all route reply that comes from neighboring nodes using (5), to find the best move and select the next path. The Newton second law is computed the force. According to [18] Newton second law is stated as "The vector sum of the forces on an object is equal to the total mass of that object multiplied by the acceleration of the object". (2) shows the original Newton second law.

$$F_{net} = ma \tag{2}$$

Where, *m* is the mass, *a* is the acceleration where can be calculated also by Newton second law (3).

$$a = \left(F_{net} / m\right) \tag{3}$$

Depending on Hook's law [14] that is stated "The force exerted by the spring which is proportional to the length of stretch or compression of the spring and opposite in direction to the direction of the stretch or the compression". (4) shows the original Hook's law.

$$F = -kx \tag{4}$$

Where : K is constant, X is displacement. By replace the (3) by (4) we get the acceleration equal to (5).

$$a = -\left(\frac{k}{m}\right)x \tag{5}$$

We suppose that m equals to DSN, and K is constant number which sets 0.1.

3.4 Solution representation

The PGO-DLLA algorithm has one main goal (shortest secure path). The main goal can be achieved by using objective function that includes two sub goal; shortest path and secure path. The Shortest Secure path in PGO-DLLA from source to destination can be calculated by the following process Figure 3:

Step 1: Distribute one agent to every node that is a central station to its neighbors, and this is done by checking the table of each node separately.

Step 2: For each agent simultaneously (applied at same time).

Step 3: Create two tables for each agent

a) The distances table which represented the distance between agent and neighbor nodes.

b) The acceleration table which represented the evaluation function for agent to choose best path.

Step 4: Find the result of evaluation function for agent using (6).

$$\alpha = \frac{kx}{m} \tag{6}$$

Step 5: Create an ascending table for the (α) values (*ListMin*).

Step 6: Calculated the value of threshold as (7).

$$Th_{Dynamic} = \frac{kx}{DSN(6\%)} \tag{7}$$

Step 7: For ListMin (node)
If ListMin (node) <= $Th_{Dynamic}$
select Path
Exit For
else
delete Path from routing table

Step 8: Next For (new node)

Step 9: Stop

Figure 3: Pseudo code of PGO-DLLA.

An example of the route discovery in PGO-DLLA is shown in Figure 4, in this figure the source A send requests to each neighbor node (B, C and D) to discover a route to the destination. The neighbor node (B and D) are replying to source A. The source A will evaluate the node B and D separately using (5). Source Node A decided to choose the main value that less than the threshold which in this example, source A choose node B as Best route and secure at the same time.

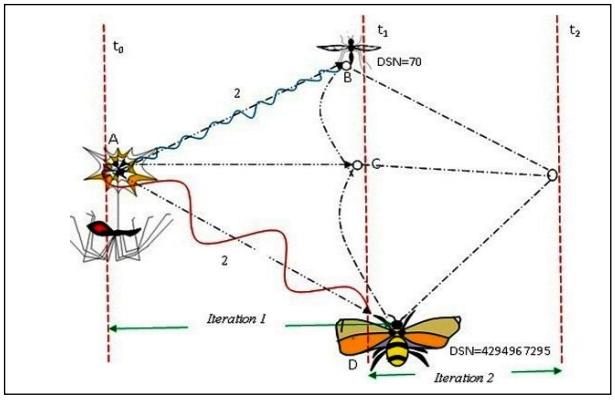


Figure 4: Example of the route discovery in PGO-DLLA.

4 Solution scenarios and parameters

We used NS2 simulator, version 2.33 [19], to conduct simulation scenarios in order to determine the efficacy and accuracy of our AODV routing protocol. We use the traffic rate and mobility models similar to parameters setting in simulation model that reported in [17]. The traffic sources have a continuous bit rate (CBR). The mobility model is the random waypoint model. The map area uses a square 800×800 field with 50 nodes. The pause time varies (between 10 and 100 sec.). The simulation were run 40 times for each scenario (1–4).

4.1 Experimental results

Simulation 1 tests the original AODV, simulation 2 tests the black hole AODV, simulation 3 tests the AntNet algorithm [20], [21], and simulation 4 tests the proposed PGO-DLLA for discovering the shortest secure path. The parameters for simulations 1–4 are shown in Table 2.

4.2 Performance metrics

Four performance indicators are used to measure the performance of the proposed PGO-DLLA, the standard AODV, the black hole AODV (BAODV) and AntNet. The details of these performance metrics are as follows:

- The packet delivery ratio (*PDR*) is the percentage of data packets sent by the source that are received by the destination. A larger packet delivery ratio indicates better protocol performance. (8) shows how the packet delivery ratio is computed:

$$PDR = \frac{\text{Number of Data Packets Received}}{\text{Number of Data Packets Sent}} \quad (8)$$

- Packet loss (*PL*) is the percentage of packets that are lost during the simulation. A lower packet loss rate indicates better protocol performance. (9) shows how packet loss is computed:

$$PL = \text{Packets Sent - Packets Received} \quad (9)$$

- The end-to-end delay (*EtoE*) is the average time taken for data packets to reach the destination. Only the data packets that are successfully addressed and delivered are counted. A lower end- to-end delay indicates better performance. (10) shows how the end-to-end delay is computed:

$$EtoE = \frac{\sum \text{Arrival Time - Transsmission Time}}{\sum \text{Connections}} \quad (10)$$

- Throughput (*TH*) is the number of packets received per unit of simulation time. A higher throughput value indicates better protocol performance. (11) shows how throughput is computed:

$$TH = \frac{\sum \text{Packets Received}}{\text{Simulation Time}} \quad (11)$$

4.3 Results of the comparison of PGO-DLLA with AntNet and discussion

For these scenarios, the pause time was varied from 0 to 100 sec., as shown in the parameters for scenarios 3 and 4 in Table 2. In Figure 5-a, the PDR for PGO-DLLA was better than the PDR for the AntNet algorithm for most sets of pause times. This is because of the new routing characteristics of the proposed algorithm, which finds the shortest route to the destination node with the smallest number of hops.

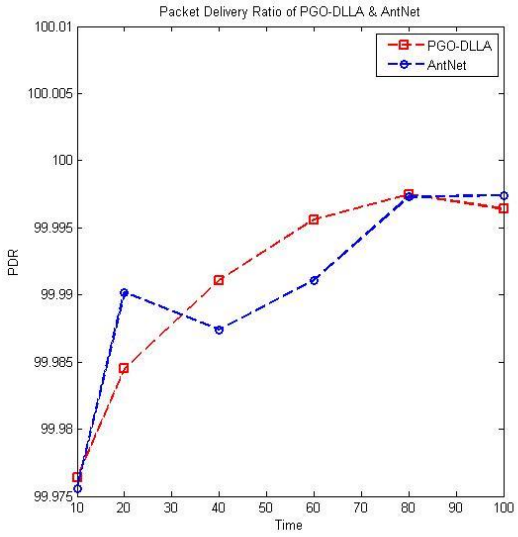
Generally, when the algorithm selects a route based on a smaller hop count, it chooses the shortest path to the destination node and thus avoids some potential link failures. For this reason, the average end-to-end delay may decrease [22]. In Figure 5-b, the value of the EtoE for PGO-DLLA was slightly higher than for the AntNet algorithm. One reason for this is the calculation that is required to find a new route in order to avoid attacks.

Throughput measures the number of packets from the source that are received by the destination node. If any delay occurs as the result of complex routing or updating the route, the throughput will be decreased [23].

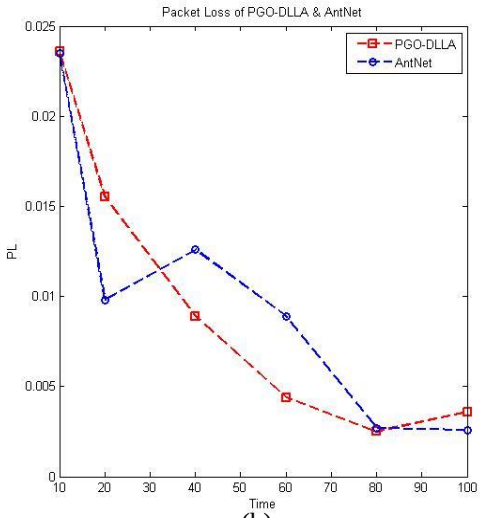
As shown in Figure 5-c, PGO-DLLA has better throughput than the AntNet algorithm during the first four time periods (t = 10, 20, 30,40), and, the throughput of the AntNet algorithm then becomes better. Nevertheless, the throughput of PGO-DLLA and the AntNet algorithm both increase across time.

Figure 5 (a, b, and c) shows a distinctive peak in AntNet graphs, the reason behind that is the using of two different routing discovery in AntNet and PGO-DLLA algorithms. Specifically, AntNet algorithm has a multipath routing that helps it to avoid the link fails, while PGO-DLLA has multi agent's strategy that takes some period (from 10 to 20 Sec.) to configure the routing tables. Hence, some distinctive peaks have been appeared in the beginning of executing AntNet compared against PGO-DLLA.

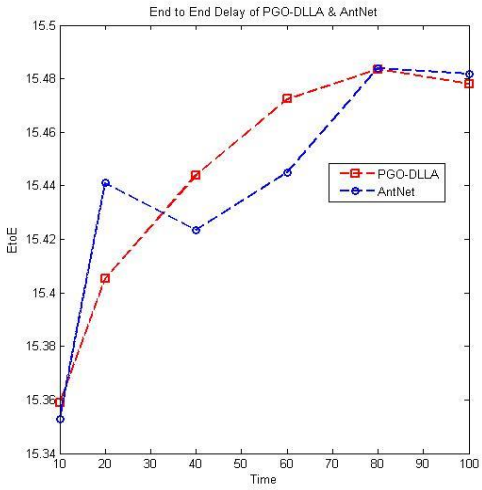
In some cases where intermediate nodes make routing decisions, such as in self-adaptive algorithms, the nodes update the routing after each iteration. In such algorithms, the routing discovery is not done by the source node, and most of these algorithms are designed to work in dynamic environments. As a result, the packet dropping rate will increase, which results in an increased packet loss rate. However, PGO-DLLA has a more stable packet loss rate than the AntNet algorithm because of its special way of routing to the destination node, as shown in Figure 5-d.



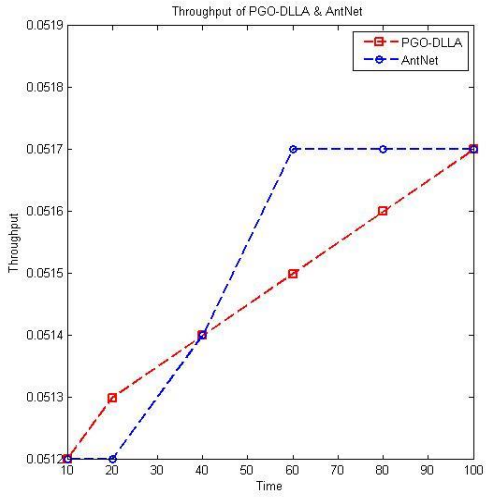
(a)



(b)



(c)



(d)

Figure 5: The results of compression of PGO-DLLA and AntNet; (a) PDR, (b) PL, (c) EtoE, (d) TH.

4.4 Results of the comparison of PGO-DLLA with AODV and BAODV and discussion

Even though the rate of throughput is small, because the pause time equals zero (continuous motion), the PDR may be not affected [24]. In such a situation, the new proposed algorithm has more than one strategy to ensure that all packets are received by the destination nodes. In Figure 6-a, we can see some decrease in the PDR for BADOV and standard AODV, as the effect of black hole attacks from a malicious node. In contrast, the PDR rate for PGO-DLLA increases, because of its strategy to avoid black hole attacks while retaining the shortest path to the destination node.

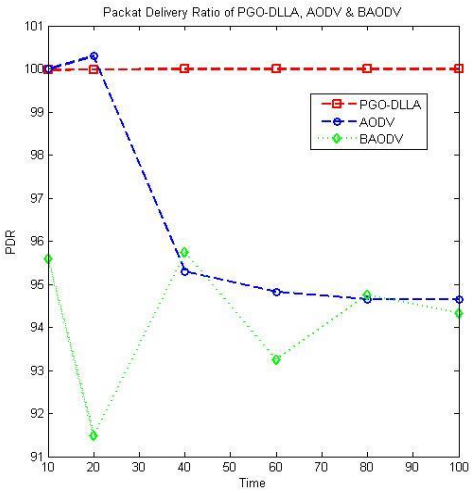
Figure 6-b shows a comparison of PGO-DLLA, BADOV, and standard AODV with respect to the average end-to-end delay. In this figure we can see that PGO-DLLA has a lower rate of delay, which is because

(d)

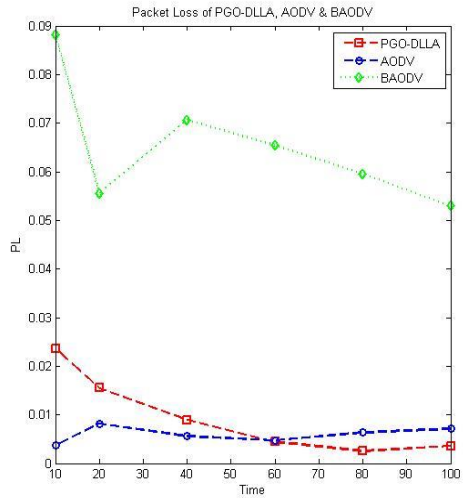
of its strategy to change the route when it is broken as a result of misbehaving nodes. In contrast, the average end-to-end delay for BADOV increases, as the effect of black hole attacks.

Figure 6-c shows a comparison of PGO-DLLA, BADOV, and standard AODV with respect to throughput. In this figure, we can see that PGO-DLLA has a higher throughput, because it can avoid dropping packets as a result of black hole attacks and change the route to the destination if it finds any disconnection. In contrast, the throughput for BADOV is very low, which is the effect of having black hole attacks without any strategy to avoid the attacks.

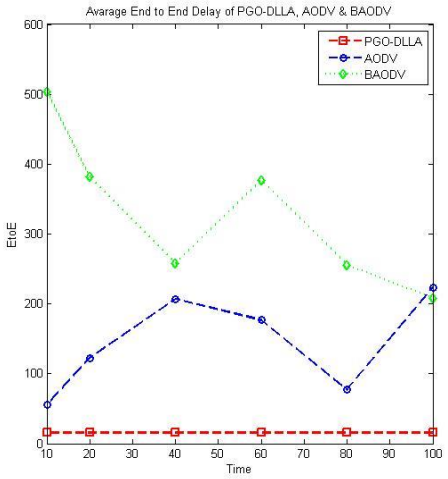
Figure 6-d shows a comparison of PGO-DLLA, BADOV, and standard AODV with respect to the rate of packet loss. In this figure, we can see that BADOV has a higher loss rate, as the result of black hole attacks. In contrast, PGO-DLLA has a very low loss rate, which is very close to that of the standard AODV.



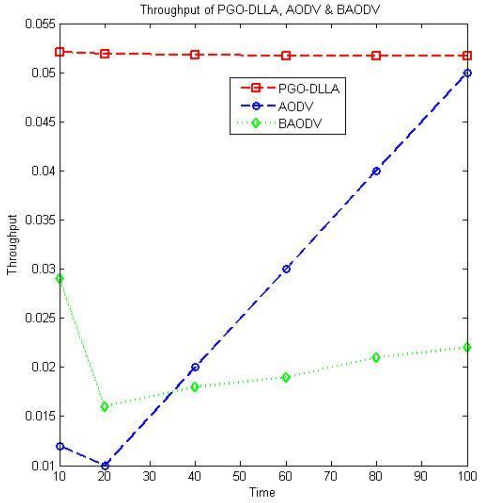
(a)



(b)



(c)



(d)

Figure 6: The results of compression of PGO-DLLA with standard AODV and Black Hole AODV; (a) PDR, (b) PL, (c) EtoE, (d) TH.

5 Conclusions

This paper proposes a defense mechanism against a cooperative black hole attack in a MANET that relies on the AODV routing protocol. The new method is called the PGO-DLLA protocol, modifies the standard AODV and optimizes the routing process. The idea inspired by a spider called daddy long-legs is a new technique for finding suspicious nodes and avoiding black hole attacks. As a swarm algorithm, the PGO-DLLA can consolidate the routing mechanism. Some changes are made in the routing tables to store the shortest and secure path from source to destination node. The main objective in this method is to avoid black hole attacks without causing delays in the routing protocol. The experimental results show that PGO-DLLA is able to improve the performance of the network with respect to most of the performance metrics examined. For future work, we plan to examine the enforcement of additional complex

attacks and the latest routing. The PGO-DLLA algorithm could not work on real maps directly, some adjustments would be needed (for instance, we need to adjust the distances between the nodes to the real distances among cities in the real maps, and we need to calculate a risk level value rather than the destination sequence number DSN).

References

[1] E. Bechinski, D. Schotzko, and C. Baird, "Homeowner guide to spiders around the home and yard," 2010.

[2] A. E. Wignall and M. E. Herberstein, "Male courtship vibrations delay predatory behaviour in female spiders.," *Sci. Rep.*, vol. 3, p. 3557, Jan. 2013.

[3] F.-H. Tseng, L.-D. Chou, and H.-C. Chao, "A survey of black hole attacks in wireless mobile ad hoc networks," *Human-centric Comput. Inf. Sci.*, vol. 1, no. 1, p. 4, 2011.

[4] J. Burbank, P. Chimento, B. Haberman, and W. Kasch, "Key Challenges of Military Tactical Networking and the Elusive Promise of MANET Technology," *IEEE Commun. Mag.*, vol. 44, no. 11, pp. 39–45, Nov. 2006.

[5] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings WMCSA'99. Second IEEE Workshop on Mobile Computing Systems and Applications*, 1999, pp. 90–100.

[6] L. Tamilselvan and V. Sankaranarayanan, "Prevention of Co-operative Black Hole Attack in MANET," *J. Networks*, vol. 3, no. 5, pp. 13–20, May 2008.

[7] H. Nakayama and S. Kurosawa, "A dynamic anomaly detection scheme for AODV-based mobile ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 5, pp. 2471–2481, 2009.

[8] S. Kurosawa, H. Nakayama, N. Kato, A. Jamalipour, and Y. Nemoto, "Detecting blackhole attack on AODV-based mobile Ad Hoc networks by dynamic learning method," *Int. J. Netw. Secur.*, vol. 5, no. 3, pp. 338–346, 2007.

[9] X. Zhang, Y. Sekiya, and Y. Wakahara, "Proposal of a method to detect black hole attack in MANET," in *Proceedings - 2009 International Symposium on Autonomous Decentralized Systems, ISADS 2009*, 2009, pp. 149–154.

[10] Y. Zhang and W. Lee, "Intrusion detection in wireless ad-hoc networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking - MobiCom '00*, 2000, pp. 275–283.

[11] R. Mitchell and I.-R. Chen, "A survey of intrusion detection in wireless network applications," *Comput. Commun.*, vol. 42, pp. 1–23, Apr. 2014.

[12] K. I. Ghathwan, A. R. Yaakub, and R. Budiarto, "EAODV: A*-Based enhancement ad-hoc on demand vector protocol prevent black hole attacks," *J. Ilmu Komput. dan Inf.*, vol. 6, no. 2, pp. 45–51, 2013.

[13] K. I. Ghathwan and A. R. B. Yaakub, "An Artificial Intelligence Technique for Prevent Black Hole Attacks in MANET," in *Recent Advances on Soft Computing and Data Mining*, Springer International Publishing, 2014, pp. 121–131.

[14] S. Horibe, "Robert Hooke, Hooke's Law & the Watch Spring," 2011.

[15] C. Benjamin, *Laser-Tissue Interactions*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, p. 218.

[16] A. Sherif, M. Elsabrouty, and A. Shoukry, "A Novel Taxonomy of Black-Hole Attack Detection Techniques in Mobile Ad-hoc Network (MANET)," in *2013 IEEE 16th International Conference on Computational Science and Engineering*, 2013, pp. 346–352.

[17] C. Perkins, E. Royer, S. R. Das, and M. K. Marina, "Performance comparison of two on-demand routing protocols for ad hoc networks," *IEEE Pers. Commun.*, vol. 8, no. 1, pp. 16–28, 2001.

[18] G. L. Lucas, F. W. Cooke, and E. A. Friis, *A Primer of Biomechanics*. New York, NY: Springer New York, 1999.

[19] A. Bright, J. R. Waas, C. M. King, and P. D. Cuming, "Bill colour and correlates of male quality in blackbirds: An analysis using canonical ordination," *Behav. Processes*, vol. 65, pp. 123–132, 2004.

[20] G. Di Caro and M. Dorigo, "AntNet: Distributed Stigmergetic Control for Communications Networks," *J. Artif. Intell. Res.*, vol. 9, pp. 317–365, May 1998.

[21] H. Huang, H.-B. Xie, J.-Y. Guo, and H.-J. Chen, "Ant colony optimization-based feature selection method for surface electromyography signals classification.," *Comput. Biol. Med.*, vol. 42, no. 1, pp. 30–8, Jan. 2012.

[22] C. Sarr, I. Guérin-Lassous, and others, "Estimating average end-to-end delays in IEEE 802.11 multihop wireless networks," 2007.

[23] P. Li, Y. Fang, and J. Li, "Throughput, delay, and mobility in wireless ad hoc networks," in *INFOCOM, 2010 Proceedings IEEE*, 2010, pp. 1–9.

[24] S. K. Tiong and H. S. Jassim, "EMNet: Electromagnetic-like Mechanism based routing protocol for Mobile ad hoc network," *Trendas Appl. Sci. Res.*, vol. 7, no. 11, pp. 881–900, Nov. 2012.

The Information Fragmentation Problem Through Dimensions of Software, Time and Personal Projects

Matjaž Kljun
Faculty of Mathematics, Natural Sciences and Information Technologies
University of Primorska
E-mail: matjaz.kljun@upr.si

Thesis Summary

Keywords: personal information management, information fragmentation, personal project management

Received: March 25, 2015

Abstract: This paper is an extended abstract of the doctoral thesis [1]. It presents an overview of the research into the field of Personal Information Management (PIM) and a study of information fragmentation on the three axes: software, time and personal projects. These axes were investigated through three studies: (i) a preliminary investigative study by interviews and observation of project information management, (ii) the observed usage of the purpose-built project information management research prototype, and (iii) logging of the usage of the same prototype in the wild. The findings show (i) the extensive information fragmentation in each individual PIM tool besides cross-tool fragmentation, (ii) the information overload preventing focusing on the subset of fragmented project related information and changing focus over time, and (iii) the importance of support information (information scraps) and its integration into project flow.

Povzetek: Pričujoče delo je razširjen povzetek doktorske disertacije [1]. Predstavlja pregled raziskav na področju upravljanja zasebnih podatkov in študijo razdrobljenost podatkov na treh oseh: programski opremi, času in vodenju zasebnih projektov. Te osi smo raziskovali skozi tri študije: (i) uvodno študijo z intervijuji in opazovanjem upravljanja podatkov povezanih s posameznim projektom, (ii) opazovanjem uporabe namensko razvitega prototipa, in (iii) dnevnike istega prototipa, ki so ga uporabniki uporabljali v vsakdanjem življenju. Glavne značilnosti upravljanja projektnih podatkov so: (i) obsežna razdrobljenost podatkov ne samo med programsko opremo ampak tudi v posameznem PIM orodju, (ii) preobremenjenost s podatki, kar preprečuje osredotočenost na podmnožico trenutno pomembnih podatkov in spreminjanje fokusa skozi čas, in (iii) pomen podpornih podatkov in vključenost le-teh v tok projekta.

1 Introduction and problem statement

Personal projects are an undertaking made up of numerous tasks and sub-projects that may last for days, weeks, months or years. These projects my include official granted projects, projects ordered to be undertaken by someone else as well as undertakings initiated by an individual. They are personal in a sense that it is (most commonly) up to knowledge workers to decide on how to manage them, which marks the management with a personal touch.

The core resource to be managed is personal information – information that an individual manages to satisfy their needs, requirements and fulfil their roles. This process is studied in the field of Personal Information Management (PIM). The three most common personal information types are files, emails and web bookmarks. Studies have shown that information fragmentation is a severe hindrance to project flow. However, despite a considerate body of research and a handful of prototypes providing unification it is still not clear what are the characteristics of such day-to-day, semi-formal and loosely planned projects, and how should unification be implemented. This thesis focuses on (tacit) knowledge behind personal project information management that is not captured by current PIM applications (for example the level of fragmentation, information importance, project stages, context recreation, etc.), and that could offer a solution to project information management.

2 Methodology

The research approach in this thesis follows the classical Human Computer Interaction (HCI) practice: (i) empirical analysis of user needs, (ii) design a solution to meet these needs, and (iii) evaluate the solution. The first step consisted of repeated semi-structured interviews over the course 4 months during which participants described their projects and related information management. This step formed the empirical conceptualisation of tacit knowledge of the project management process and the basis of the development of the Task Information Collection (TIC) proto-

type. The third step consisted of two studies. In addition to evaluate the prototype and observe its usage in the real life settings (repeated weekly observations of usage and interviews), the in-the-wild usage data (TIC is offered as an open source software to general public) was logged to confirm and further the findings of the exploratory study as well as evaluation study.

3 Results

Based on the studies and observations the thesis provides a definition of a personal project as "a self defined or given undertaking lasting from days to months that is (ii) directed towards and defined by a specific goal in the form of information or/and a path to achieve it, (ii) managed (information, time, people, equipment, budget) by an individual on a day-to-day, semi-formal manner based on this individual's ingenuity, past experiences, and knowledge (of technology and information), and (iii) made up of loosely planned tasks and sub-projects affected by planning fallacy and completed when remembered, when time permits or when approaching formal due-dates."

Quantitative data provided additional insight into the problem and made several classifications, comparisons and listings possible: classification of factors behind information importance such as time spent and (mental and physical) effort invested, comparison of fragmentation patterns in the file hierarchy alone (between 2 or more folders in a file hierarchy), and comparison on how project information spaces evolve which shows how information focus changes when sub-projects are completed. The data also showed how projects overlap through information and how information is reused or recycled for different projects constituting to even greater fragmentation in the file hierarchy. In particular, the data revealed the importance of support information (web pages, information scraps) to the project flow, which has never been observed in the personal project context (e.g. its relation to other information types) and its unification (in TIC) with other information types proved very helpful at the beginning as well as through projects' lifetime.

4 Discussion and further work

The main findings revealed (i) the preference over selective unification focusing on the subset of cross-tool project related information, (ii) the evolution of such unification over time, (iii) re-use of information in various projects, (iv) the extensive information fragmentation in each PIM tool due to different organisational needs and ease of information access, (v) the factors behind information value (time and effort spent), and (vi) the importance of support information in relation to project goals. Nevertheless, the studies presented form an initial in-the-wild study of project information unifications and further studies are needed to contribute towards even greater understanding of how to support information unification in the project management context.

References

[1] M. Kljun, "The information fragmentation problem through dimensions of software, time and personal projects," Ph.D. dissertation, Lancaster University, 2013.

Designing Effective Mobile Augmented Reality Interactions

Klen Čopič Pucihar
Faculty of Mathematics, Natural Sciences and Information Technologies, University of Primorska
E-mail: klen.copic@famnit.upr.si

Thesis Summary

Keywords: augmented reality, dual-view problem, user perspective, device perspective

Received: Junij 20, 2015

Abstract: This paper is an extended abstract of the doctoral thesis [1] which attempts to fill the knowledge gap between user understanding and available Augmented Reality (AR) technology, a result of the general lack of user studies in AR and high-pace technology-driven AR development. The thesis pursues this goal by: (i) reviewing perceptual issues that relate to handheld AR in order to identify usability issues; (ii) reviewing handheld AR system utility in order to propose utility improvements; (iii) conducting empirical user studies to explore identified usability issues; (iv) designing, building and evaluating solutions that will enhance handheld AR utility and usability.

Povzetek: Pričujoče delo je razširjen povzetek doktorske disertacije [1], ki strmi k zmanjšanju razkoraka med uporabnikom in tehnologijo dopolnjene resničnosti (DR) namenjene mobilnim napravam. Odtujenost je posledica pomanjkanja uporabniških študij DR in tehnološkega napredka, ki ne temelji na potrebah uporabnika. Predstavljeno dela poizkuša razkorak zmanjšati s: (i) pregledom percepcijskih problem in identifikacijo problemov uporabnosti; (ii) pregledom primernosti obstoječih sistemov; (iii) empiričnim raziskovanjem problemov; (iv) načrtovanjem in izdelavo izboljšav sistemov DR.

1 Introduction

The environment we live in is a dynamic heterogeneous space, filled with countless objects and people. This vibrant space is possibly the richest source of visual, audio and tactile stimuli that fabricates our everyday life experiences. Yet, we spend more time than ever interacting within digital media confined to screens, which often serve to disconnect us from the physical space we live in.

How can we escape the glow of the screen, and bring digital and physical worlds closer together, and how can we make the world itself a user interface for digital interactions?

Researchers raising these questions identified a possible solution in Augmented Reality (AR), an interaction concept based on superimposing computer-generated content on top of the real world. However, irrespective of the mass market reach through AR on mobile devices (e.g. handheld AR), the first generation of AR interfaces failed as online user survey conducted by Olsson and Salo revealed *"generally positive evaluations are overshadowed by mentions of applications' pragmatic uselessness in everyday life and technical unreliability"* [2]. This is the result of the general lack of user studies, coupled with high-pace technology-driven AR development, which increased the knowledge gap between the available technology and user understanding. This disconnect between technology and the user resulted in poorly designed AR implementations that did not take full advantage of AR paradigms.

The primary goal of this thesis is to reduce this gap between the user and the technology and improve the usefulness of future handheld AR interfaces where usefulness comprises of system utility and usability. Utility is concerned with systems' ability to do what it was intended for, and usability focuses on how well the user can use the designed system. This thesis aims to improve both by pursuing a four step procedure outlined in abstract.

2 Research approach

The research approach in this thesis follows the classical Human Computer Interaction (HCI) practice. The thesis starts by framing research questions and hypothesise through exploratory studies and synthesis of existing knowledge. This step is followed by the design process in which solutions or test systems are being built. This enables experimentation that generates data of various types allowing researchers to confirm or reject initial hypothesis. The methodology follows a mixed research methods approach where quantitative and qualitative data types are being captured and analysed utilizing action-based and empirical research methods.

3 Results

Through the review of perceptual problems that relate to the handheld AR, within the class of table-top sized AR workspace, three usability issues were identified and confirmed through empirical user studies. In particular, results highlighting the prominence of the dual-view perceptual problem–the result of the difference between common implementation of the magic-lens, known as device-perspective rendering, and what the user would expect to see when looking though a magic-lens, which acted as a transparent glass pane [3]. Results showed that users find particular difficulty in dealing with the effect caused by the camera-screen offset e.g. the camera is not positioned in the centre of the device screen [4]. The thesis takes a design approach to the problem in which a hybrid magic-lens is proposed as one possible solution to the problem [5, 6].

Finally, by reviewing basic system utility and through design and implementation of prototypes, the research identified and confirmed three utility improvements, namely: (i) reintroduction of scale into online markerless AR systems utilizing auto-focusing feature of a camera phone [7]; (ii) improving system initialization by optimizing map initialization through sensor fusion of phone camera and other sensing capabilities, commonly available on handheld devices; and (iii) improving system utility of scene readability by enhancing rendering quality, or through an interaction paradigm that replicates a magnifying glass.

4 Discussion and future work

The main finding of this thesis is the identified prominence of the dual-view problem on usability of handheld AR interfaces. Hence, amongst others, future research should focus on different methods that will minimize the dual-view problem. One such method is a hybrid magic-lens approach which was designed and implemented within the thesis, but remains to be throughly evaluated. Additionally, future research shoulde explore how dual-view problem affects multiple user situations, particularly important as the magic-lens interaction paradigm presents information in a contextually meaningful way. Because the context is the real-world, such visualization enables intuitive context sharing amongst multiple users, making it a compelling choice for collocated multiple user collaboration.

References

[1] K. Čopič Pucihar, "Designing Effective Mobile Augmented Reality Interactions," Ph.D. dissertation, Lancaster University, 2014.

[2] T. Olsson and M. Salo, "Online user survey on current mobile augmented reality applications," in *ISMAR*, 2011.

[3] K. Čopič Pucihar, P. Coulton, and J. Alexander, "The use of surrounding visual context in handheld AR: device vs. user perspective rendering," in *CHI*, 2014.

[4] K. Čopič Pucihar, P. Coulton, and J. Alexander, "Evaluating dual-view perceptual issues in handheld augmented reality: device vs . user perspective rendering," in *ICMI*, 2013.

[5] K. Čopič Pucihar and P. Coulton, "Contact-view: a magic-lens paradigm designed to solve the dual-view problem," in *ISMAR*, 2014.

[6] K. Čopič Pucihar and P. Coulton, "Utilizing contact-view as an augmented reality authoring method for printed document annotation," in *ISMAR*, 2014.

[7] K. Čopič Pucihar and P. Coulton, "Estimating scale using depth from focus for mobile augmented reality," in *EICS*, 2011.

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 submit a manuscript to: http://www.informatica.si/Editors/PaperUpload.asp. 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.

QUESTIONNAIRE

☐ Send Informatica free of charge

☐ Yes, we subscribe

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 twentyone 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 is free of charge for major scientific, educational and governmental institutions. Others should subscribe (see the last page of Informatica).

ORDER FORM – INFORMATICA

Name: .....

Title and Profession (optional): .....

.....

Home Address and Telephone (optional): .....

.....

Office Address and Telephone (optional): .....

.....

E-mail Address (optional): .....

Signature and Date: .....

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. Čosić, 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 2015 (Volume 39) 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.
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):
Robotics Society of Slovenia (Jadran Lenarčič)
Slovene Society for Pattern Recognition (Janez Perš)
Slovenian Artificial Intelligence Society (Dunja Mladenić)
Cognitive Science Society (Urban Kordeš)
Slovenian Society of Mathematicians, Physicists and Astronomers (Andrej Likar)
Automatic Control Society of Slovenia (Sašo Blažič)
Slovenian Association of Technical and Natural Sciences / Engineering Academy of Slovenia (Vojteh Leskovšek)
ACM Slovenia (Andrej Brodnik)

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 39 Number 3 September 2015 **ISSN 0350-5596**

Informatica

An International Journal of Computing and Informatics

Informatica **39** (2015) Number 3, pp. 219–335