Volume 17 Number 1 March 1993 ISSN 0350-5596

Informatica

An International Journal of Computing and Informatics

Profiles: Terry Winograd
Japan: Knowledge Archives
Stanford: Language and Information
ALPHA Project

The Slovene Society Informatika, Ljubljana, Slovenia

Informatica Volume 17 Number 1 March 1993 pp. 1-108

1977

Informatica

An International Journal of Computing and Informatics

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, Vožarski pot 12, 61000 Ljubljana, Slovenia.

The subscription rate for 1993 (Volume 17) is
- DEM 50 for institutions,
- DEM 25 for individuals, and
- DEM 10 for students
plus the mail charge.

Claims for missing issues will be honored free of charge within six months after the publication date of the issue.

LaTeX Technical Support: Borut Žnidar, DALCOM d.o.o.
Stegne 27, 61000 Ljubljana, Slovenia

Printed by Biro M, d.o.o., Žibertova 1, 61000 Ljubljana, Slovenia.

Orders for subscription may be placed by telephone or fax using any major credit card. Please call Mr. R. Murn, Department for Computer Science, Jožef Stefan Institute: Tel (+38) 61 214 499, Fax (+38) 61 158 058, or use the bank account number LB 50101-678-51841.

According to the opinion of the Ministry for Informing (number 23/216-92 of March 27, 1992), the scientific journal Informatica is a product of informative matter (point 13 of the tariff number 3), for which the tax of traffic amounts to 5%.

The issuing of the Informatica journal is financially supported by the Ministry for Science and Technology, Slovenska 50, 61000 Ljubljana, Slovenia.

TOWARDS AN INFORMATIONAL ORIENTATION

The international issue of *Informatica* is only a beginning of the new informational orientation, which is already on hand in research and technology communities concerning the realm of the informational. This realm extends and will more and more extend into the cultural, knowledge, and technology, irrespective of the national, regional, continental, or global. The new perspective from the computer scientific to the informational began with the criticism at the very beginning of artificial intelligence, in 1965, when *Hubert L. Dreyfus* launched his report to RAND, in which he argued that *"Early success in programming digital computers to exhibit simple forms of intelligent behavior, coupled with the belief that intelligent activities differ only in their degree of complexity, have led to the conviction that the processing underlying any cognitive performance can be formulated in a program and thus simulated on a digital computer."* (Alchemy and Artificial Intelligence, The RAND Corporation Paper P-3244, December 1965.)

The next important milestone of understanding computers and cognition was the book published by *Terry Winograd* and *Fernando Flores,* in 1986, by Ablex Publishing Corporation (see Profiles on the next page). By this work, philosophical thought concerning hermeneutics, phenomenology, Being and autopoiesis (philosophers Husserl, Austin, Gadamer, Heidegger, Habermas and biologist Maturana) was introduced, that is, meaningly legalized within the global artificial intelligence community.

The last push came from Japan when, in 1993, the *Knowledge Archives Project* started its work (look at *Mission and Research Reports*). The goal of this project is to become a project for projects and to connect and manage internationally, interdisciplinary, and interindustrially the fields of artificial intelligence, knowledge engineering, humanities (cultures over the globe), social sciences, computer sciences, multimedia technologies, etc. with the goal to implement the global knowledge system by means of both humans and new technological tools which will emerge during the run of the project.

Among others, *Informatica* will follow the problems of research and technology, which fall into the domain (niches) of the informational, through an international cooperation, joining disciplines and researchers of different professional orientations. It will enable the publishing of submitted matters, which fall into the field of informational conceptualism, theory, design, and technology. In this respect, cybernetics, humanities, social sciences, philosophy, etc. concerning informationally oriented problems will be treated, searching new paths of informational research and technology.

At the beginning of the international issuing of *Informatica* there are still unsaid editorial and organizational problems, which have to be solved in the coming years. For instance, we are establishing the final LaTeX style for several kinds of submitted matters (articles, reports, news, special columns, etc.). The E-mail net of editors of *Informatica* over the globe is beginning to function effectively. *Informatica* is becoming "our journal" for all participating parties. We are still in acquiring of new editors and authors at the instantaneous journal circulation of a thousand copies. And this beginning is promising.

—*Anton P. Železnikar,* Editor–in–chief

PROFILES

On this page we present editors of *Informatica* in a spontaneous and free way, in the form of the reply for which the Editor-in-chief has asked editors to deliver their biographies. These biographies may be of particular interest for readers of *Informatica* to learn about the circumstances in which editors live and work. In the first attempt of *Informatica*'s editors profiles, we have the opportunity to give a short story of professor Terry Winograd who together with Fernando Flores surprised the artificial intelligence community by the work *Understanding Computers and Cognition*. This work gave a substantial impulse not only to a new orientation but also to the critical look to the phenomenon of intelligence and possibilities of machine intelligence in particular.

Terry Winograd

Terry Winograd is Professor of Computer Science at Stanford University. He received his B.S. in mathematics from The Colorado College in 1966 and a Ph.D. in Applied Mathematics from MIT in 1970. He taught at MIT from 1970 to 1973 and has been on the faculty of the Computer Science Department of Stanford University since 1973. He also has appointments in the Department of Linguistics and the Program in Values, Technology, Science, and Society and is on the advisory board of the Stanford Humanities Center.

Winograd's research on natural language understanding by computers is often cited as a major milestone in artificial intelligence. It was the basis for his book *Understanding Natural Language* (Academic Press, 1972), and his textbook *Language as a Cognitive Process* (Addison-Wesley, 1983) as well as numerous articles in both scholarly journals and popular magazines. His most recent book, co-authored with *Fernando Flores*, takes a critical look at work in artificial intelligence and presents an alternative theory of language and thought, which suggests new directions for the design of intelligent human/computer systems. The book, entitled *Understanding Computers and Cognition: A New Foundation for Design* (Addison-Wesley, 1987), was named as the best information science book of 1987 by the American Society for Information Science. He recently co-edited a book with *Paul Adler* entitled *Usability: Turning Technologies into Tools* (Oxford University Press, 1992).

Winograd's current research on the foundations of the design of computer systems builds on the theories developed in his book with *Flores*. He is developing a "language-action perspective" in which current and potential software and hardware devices are analyzed and designed in the context of their embedding in work and communicative structures. The language/action perspective grew out of his earlier work in artificial intelligence, but it shifts the focus of attention away from the mental and the individual, to the social activity by which we generate the space of cooperative actions in which we work and to the technology that is the medium for those actions. He also has developed several new courses at Stanford, including one on *Computers, Ethics and Social Responsibility*, and a series on the *Design of Human-Computer Interaction* (sponsored by the National Science Foundation). He directs a project at Stanford called the *Project on People, Computers and Design*. During the 1992-93 academic year he is on leave from Stanford, working with *Interval Research*, a new research laboratory in Palo Alto.

Winograd was the keynote speaker for the *1988 Conference on Office Information Systems*, the *1990 Conference on Computer-Human Interaction* (CHI'90), and the first *National Conference on Computing and Values* (1991). He edited a special issue of the *ACM Transactions on Office Information Systems* (Spring 1988) on the "Language/action perspective." He is on the editorial board of a number of journals, including *AI Expert*, *AI & Society*, *Journal of Computing and Society*, *Human-computer Interaction*, and *Computer-Supported Cooperative Work*.

Winograd is a board member and consultant to *Action Technologies*, a developer of workgroup productivity software. He was a founding member of *Computer Professionals for Social Responsibility*. He has been on the national board since the organization was founded, and served as national President from 1987-1990.

—*A.P. Železnikar*

METHODOLOGIES FROM MACHINE LEARNING IN DATA ANALYSIS AND SOFTWARE

D. Michie
The Turing Institute, George House, 36 North Hanover Street,
Glasgow G1 2AD

Keywords: artificial intelligence, machine learning, software

Edited by: Matjaž Gams

Received: February 25, 1993 **Revised:** March 8, 1993 **Accepted:** March 15, 1993

In the last few decades the rise of computing and telecommunications has flooded the world of government, business, medicine and engineering with unprecedented volumes of stored data. These databases provide the raw material for information supply but have been largely impenetrable as potential sources of export knowledge. Computer-oriented techniques can now be used, however, in integration with established methods from classical statistics to generate rule-structured classifiers which not only make a better job of classifying new data sampled from the same source but also possess the quality of clear explanatory structure. New developments in the computer induction of decision rules have contributed to two areas, multivariate data analysis and computer assisted software engineering. Practical connections between the two are thereby coming to light. This paper reviews some of the more significant of these developments.

1 Introduction

Computer induction of decision rules from sample multivariate data was already known a quarter of a century ago. But Hunt, Marin and Stone's CLS (Concept Learning System) initially aroused interest only among cognitive psychologists [11]. In recent years, however, new developments have contributed to two applied disciplines, namely (1) multivariate data analysis and (2) computer-assisted software engineering. Practical connections between (1) and (2) are also thereby coming to light. This paper reviews some of the more significant of these developments. [1]

In 1977 Friedman independently incorporated the essentials of CLS in an algorithm suitable for inducing decision trees from statistical-type (i.e. 'noisy') data. For this he introduced an important innovation, automatically pruning the tree's more distal nodes under the control of a user-supplied parameter. At about the same time Quinlan devised and demonstrated an algorithm, ID3 (Iterative Dichotomizer 3) [25], capable of extracting complete logical descriptions from large files of multivariate noise-free data, adverse to analysis for reasons of logical rather than statistical complexity. The data-analytic era of rule induction was consolidated with the pioneering work *Classification and Regression Trees* by Breiman, et al. [5].

The opening passage of the above treatise contains the statement: 'An important criterion for a good classification procedure is that it not only produces accurate classifiers but that it also provides insight and understanding into the predictive structure of the data.' In the last few decades the rise of computing and telecommunications has flooded the worlds of business, government, medicine and engineering with unprecedented volumes of stored data. These databases provide the raw material for information supply, but have been largely impenetrable as potential sources of expert knowledge. However, computer-oriented techniques can now be used in integration with established methods from classical statistics to

[1] Similar paper with the same title was published in the Computer Journal. Permission for reprint obtained by the British Computer Society.

generate rule-structured classifiers which not only make a better job of classifying new data sampled from the same source, but also possess the quality of clear explanatory structure emphasised in the above-quoted passage. Applicability has been shown both to data derived from databases (real case histories) and from simulators (model case histories), as in the following knowledge-intensive application areas.

Synthesis from simulation data
- aerospace
- instrumentation
- manufacturing
- pharmaceuticals
- electronics trouble-shooting
- interpretation of biomedical monitoring
- generating software from specifications

Synthesis from captured data
- nuclear engineering
- gas and oil processing
- circuit fault diagnosis
- steel and chemical process industries
- seismic measurement interpretation
- clinical diagnosis
- credit control
- stockmarket assessment

In what follows the essentials of rule-based techniques drawn from machine learning are summarised in the context of earlier approaches.

2 Nature of the Problem

Given. A 'training set', or estimation sample, of case-descriptions, each in the form of a list of attribute-values (e.g. age, duration of pregnancy, number of previous births, number of previous pregnancies, marital status, etc.), together with a classification of each case into, say, YES, NO, or more generally $class_1, class_2, \ldots, class_i$, (e.g. Did patients with these characteristics elect to have an amniocentesis test? Did loan applicants giving these questionnaire answers turn out to be creditworthy? Was this pattern of meteorological measurements followed by thunderstorms? Which category of fault was identified from the observed engine test results?)

Required. A classifier (i.e. some formula or rule defined over the attributes) which can classify new cases sampled from the same population.

2.1 Two Approaches

We demand of a classifier: (1) that it should predict with high accuracy; (2) that it should be simple and easy to understand.

Decision formulae derived from standard multivariate statistics, such as discriminant analysis or 'naive Bayes' methods, have the form of sets of positive and negative weights, scoring attribute-values for their individual contributions (assumed independent) to an ACCEPT-versus-REJECT preference for each decision class. Such lists of numbers mean less to the user than they do to the machine.

Logical decision formulae ('rules') get away from the arithmetic. Instead of the operators addition, multiplication etc., decision-tree rules for example use the logical operator 'if...then...else' for combining relevant subsets of attributes into classifying expressions. The above-referenced treatise by Leo Breiman and his colleagues opens with the following illustration:

> At the University of California, San Diego Medical Center, when a heart attack patient id admitted, 19 variables are measured during the first 24 hours. These include blood pressure, age and 17 other ordered and binary variables summarising the medical symptoms considered as important indicators of the patient's condition.
> The goal of a recent medical study was the development of a method to identify high risk patients (those who will not survive at least 30 days) on the basis of the initial 24 hours.

The tree-structured classification rule which these authors obtained from their data is below.

if the minimum systolic blood over
    the initial 24-hour period ≤ 91
  then risk is HIGH
else if age ≤ 62.5
  then risk is NOT-HIGH
else if sinus tachycardia is present
  then risk is HIGH
  else risk is NOT-HIGH

The reason for using the term 'tree-structured' is clear from the graphical representation of this same rule, as shown in Fig. 1.

Breiman and his colleagues comment on this rule that 'its simplicity raises the suspicion that standard statistical classification methods may give classification rules that are more accurate. When these were tried, the rules produced were considerably more intricate, but less accurate.' During the six years which have elapsed since then, decision-tree induction from data has been subjected to field trials in various countries by both academic and industrial groups. The results have been in conformity with the above-cited observation. The chief advantages have been that:

1. the amount of calculation is much smaller;
2. the classifiers produced are easier to understand;
3. filtering out irrelevant attributes is done automatically;
4. decision-tree classifiers induced from data in the style of Breiman and his colleagues have been found in practice to be usually more accurate than classifiers formed by adding up discriminant scores.

2.2 Reasons for Improved Accuracy

When applied to the kinds of data which make difficulties for standard statistical analysis decision-tree methods gain improved accuracy in two ways.

1. They can handle both numerical and non-numerical attributes with equal ease.
2. They do not suffer any loss of discriminant power when some of the attributes violate the simplistic assumption of mutual independence. Consider the four items of decision data (cases) in Table 1 which, if read as statements from an 'oracle' instead of passively as data, collectively define the exclusive-or relation between two binary attributes.

Case	a1	a2	Class
1	+1	+1	False
2	+1	−1	True
3	−1	+1	True
4	−1	−1	False

Table 1: A simple example of a data-set which gives trouble to linear numerical estimation methods. For rule-induction procedures (including standard Boolean simplification procedures) the problem is trivial

Linear scoring systems such as discriminant analysis are powerless to find scoring coefficients for a1 and a2 such that they can be multiplied by a1's and a2's values, added up, and compared with a threshold in order reliably to distinguish true and false cases. If the reader cares to try it he will find that it cannot be done. But in common with other logic-based induction algorithms, such as those routinely employed by electrical engineers for circuit simplification, decision-tree induction trivialises the problem:

if a1 = +1
  then if a2 = +1
    then false
    else true
  else if a2 = +1
    then true
    else false

To construct from such data a classifying expression using numerical multivariate analysis would require an excursion into non-linear regression equations, or (equivalently, see Angus) [1] into multilayer neural networks. Moreover, decision-rule methods are not limited, as are Boolean simplification techniques, to problems where attributes are guaranteed to be of logical type. Modern rule-induction techniques are equally at home with inputs which include numerical attribute-values. For this, the standard approach converts input values to logical type by automatically splitting numerical ranges into intervals according to an entropy-minimisation principle. Among other sources, the book by Breiman and his colleagues can be consulted for details.

3 Machine Learning As Data Analysis

Multivariate statistical methods were developed from mathematical and scientific foundations by

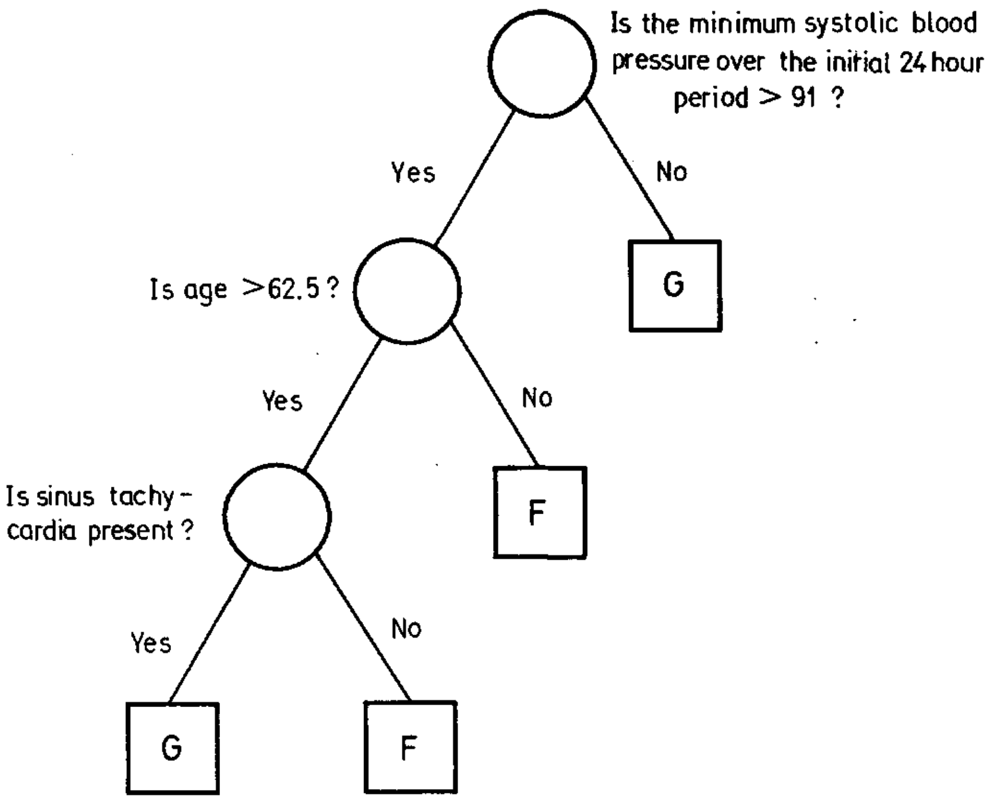

Figure 1: Decision tree induced from heart-attack data corresponding to if-then-else rule (see text). 'G' stands for high risk, 'F' for lower risk.

pioneers such as Francis Galton, Karl Pearson and Ronald Fisher. These men made it possible to give precise answers to questions phrased within the statistical paradigm. 'Within what limits of error can this or that classifier be expected to perform?' or 'How much will the error be narrowed by a given increase in the size of estimation sample?' Studies in the computational theory of learning are directed towards building scientific foundations for the machine-learning approach as an extension of classical probability and statistics (in the case of decision-tree induction see, for example Refs 7, 16, 17 and 18). The connection between inductive learning and statistical data analysis can be explained as follows.

Approximately two decades of exposure to data turns a baby into a mentally capable adult. Evidently the developing brain extracts something of continuing and incremental value. We call the process of extraction 'learning'. For the something which is extracted, stored, refined, built upon and exploited, we have variations on notions of 'knowledge'. We speak about behaviour-patterns, habits, skills, hypotheses, beliefs, models, descriptions, concepts, theories. These structures have one thing in common: they all act as classifiers. Empirical scientists here recognise something familiar. They too are concerned with extracting theories from data, alternating with the theory-guided sampling of new data. In this cycle of extraction and testing the scientist commonly calls on the aid of a special breed of numerical craftsman, the data analyst.

According to one rather undemanding definition, the statistical data analyst's fitting of models to data would qualify as a form of machine learning. This definition says:

a learning system uses sample data (the training set)
to generate an up-dated basis
for improved classification
of subsequent data from the same source.

Notice that the definition, although phrased strictly in terms of classification, logically extends to acquisition of improved performance on tasks which do not look at all like classification. Iterative situation-action tasks come to mind such as riding a bicycle, solving an equation, or parsing a sentence. The extension becomes obvious when for the decision classes we choose names which refer to partitions of the space of situations as 'suitable for action A', 'suitable for action B', etc.

4 Machine Learning As Problem Solving

To illustrate the way in which iterative situation-action problems can be coded as classification, consider the task of solving simple equations in schoolroom algebra (see Ref. 20). Thus $3x+1=2$ is transformed by an appropriate action ('collect like terms on same side of the equation') into $3x = 2 - 1$, which is in turn transformed into $3x = 1$ (by 'combine like terms') and thence into the final 'situation', $x = \frac{1}{3}$(by 'divide by the coefficient of the unknown'). This is the solution or 'goal'.

Using the attributes/classes format, the problem-description is given in Table 2. Line 1 gives the number of attributes, lines 2-7 describe the attributes and the last two give the number and names of the classes.

The key to the six attribute names is:

- a1 Does the equation have a common factor? (comfact)
- a2 Are there like terms on opposite sides? (likeopp)
- a3 Are there any bracketed terms? (bracket)
- a4 Does either side have like terms? (likesam)
- a5 Is exactly the same present on both sides? (sametrm)
- a6 Is there only one unknown term and is its coefficient equal to one? (xcoeffi)

The seven class names given earlier have the following interpretations:

1. Divide the equation by its common factor (divef).

6	
comfact	logical yes no
likeopp	logical yes no
bracket	logical yes no
likesam	logical yes no
sametrm	logical yes no
xcoeffi	logical yes no
7	
divef collect multbr combine cancel divex stop	

Table 2: Problem description for inductive derivation of an equation-solving rule

2. Collect like terms on the same side of the equation (collect).
3. Multiply out bracketed terms (multbr).
4. Combine like terms (combine).
5. Cancel out a term that appears on both sides (cancel).
6. Divide by the coefficient of the unknown (divex).
7. Stop because the equation is solved (stop).

The four lines of examples given at the outset, namely

$$3x + 1 = 2, 3x = 2 - 1, 3x = 1, x = \frac{1}{3}$$

would appear in an induction file as:

ID no	a1	a2	a3	a4	a5	a6	Class
1	no	yes	no	no	no	no	collect
2	no	no	no	yes	no	no	combine
3	no	no	no	no	no	no	divex
4	no	no	no	no	no	yes	stop

From the output of a skilled human equation-solver on a number of example cases of the above kind, an induction system acquires expertise. From sequences of illustrative solution steps, a solver is built. Application of the solver to actual problems is iterative. In each cycle the output action is applied to the current state of the equation to yield a modified state. This is then fed back to the classifier as a new input, and so on until the output 'stop' appears. Although seemingly remote from the task of synthesizing a controller from a recorded behavioural trace of a skilled human operator (see Ref. 19 for a worked example), the logic is the same. Successive 'snapshots' of the state of the dynamical system correspond to successive lines of symbols in equation-solving. The STOP action is of course normally dropped to allow for instabilities and perturbations from the goal state, normally absent from symbol-manipulation tasks.

5 A More Demanding Definition Of Learning

A more demanding definition of learning is now coming from applied artificial intelligence:

a learning system uses sample data
to generate an up-dated basis
for improved classification
of subsequent data from the same source
and expresses the new basis in intelligible symbolic form.

According to this more demanding definition, not only improved performance results from the learning process, but also an explicit set of rules. Decision-tree induction in the algebra domain meets not only the less demanding but also the more demanding criterion. An intelligible symbolic form obtained by the ACLS algorithm [23] from a training set of equation solutions is shown in Fig. 2. Such a form constitutes an operational theory, a prescription which can be followed by any agent able to interpret it, whether human or machine.

Michie *et al.* [20] found that use of a version of ACLS by high-school children could be an effective mode of self-instruction. Children were asked to teach the machine equation-solving from an elementary algebra text-book by selecting and supplying example solvings. The children's subsequent grasp was tested against that of a group of classmates who had been exposed instead to a conventional CAI algebra package.

if exactly the same term occurs on both sides
then cancel the same term from each side
else if the equation has a common factor
then divide by the common factor
else if there is a bracketed term
then multiply out the brackets
else if there are like terms on opposite sides
then collect like terms on the same side
else if one side has more than one like term
then combine like terms
else if the unknown term has a coefficient not equal to one
then divide by the coefficient of the unknown
else stop; the equation is solved.

Figure 2: Induced rule as an operational theory discovered form supplied data

6 Machine Learning As Theory-Construction

Scientists of the past have been content if the automated procedures of data analysis satisfied the undemanding definition only, taking on themselves the responsibility of abstracting from the analysis the desired explanatory or predictive theories. AI-based machine learning, combining as it does both logical and statistical idioms, follows the more demanding definition. It thus enters directly into the theory-building process, as was first shown in 1976 by groups working in chemistry at Stanford, USA, [6] and in plant pathology at the University of Illinois [8]. Subsequent studies in USA, Australia, Slovenia and Britain established the tree-structured paradigm of computer induction as the dominant form for rule-based data analysis. The following were the chief advances.

(1) In 1977 J.H. Friedman introduced ways of deriving tree structures from data in the style of Earl Hunt's 1966 scheme but constrained by criteria for pruning unprofitable branches [9]. Decision-tree induction was thus generalised to noisy data, yielding trees with confidence measures associated with their leaves (outcome nodes).

(2) In 1979 J.R. Quinlan published the first of a series of papers describing the ID3 series of decision-tree algorithms, the efficiency and versatility of which led to their widespread adop-

tion as the basis of commercial inductive software engineering [25]. For large problems (the 1979 paper describes a rule-based solution of a problem which in unreduced form constituted some $2\frac{1}{2}$ million records) extracted decision trees were efficient at run-time but unstructured and hence obscure – the automated equivalent of hand-crafted 'spaghetti code'.

(3) A.D. Shapiro and T.Niblett [28] elaborated the Quinlan paradigm with a method known as 'structured induction', subsequently studied in depth by Shapiro [27]. To supplement the initially given primitive attributes they added separately induced procedural attributes. Structured induction confers on inductive data analysis the benefits of top-down problem decomposition, as in the discipline of structured programming. In its original form, it can only be applied where noise is absent and where the problem can be fully specified in terms of the primitives. Today the structuring steps can in suitable cases themselves be automated (see (7) below).

(4) In 1987 J.R. Quinlan adapted his C4 algorithm for inducing trees from noisy data so as to generate solutions in the form of compact sets of logic rules [26]. After pooling the branches harvested from the trees separately induced from the same data set, the program winnows out redundancy and delivers a compact and intelligible local theory of the data.

(5) Quinlan's current version, C 4.5, has been used to recover from the recorded behaviour of simulator-trained human subjects sets of 'production rules' of the type postulated by cognitive psychologists [19]. Such productions, in some neurally encoded form, are believed to underlie learned decision skills. As compared with the original expert behaviour, induced rules exhibit a 'clean-up effect', having shed some of the inconsistencies and noise which even the most highly trained nervous system introduces into the recognise-act cycle.

(6) D. Michie derived, [17] and with A.Al-Attar, partly tested, a formulation of decision-tree induction from the axioms of Bayesian probability [18]. Main gains obtained from this approach are: (i) reunion of rule induction with statistical decision theory; (ii) a practical way of extending Shapiro-Niblett structured induction into the analysis of noisy data.

(7) Full automation of structured induction requires algorithms for generating new procedural attributes, rather than depending for this on the knowledge-based insight of human domain specialists. Advances by Muggleton and Buntine and by Bain and Muggleton at the Turing Institute, Glasgow [2,22] have yielded needed algorithms within the framework of first-order predicate logic (see also Ref. 21).

7 Inductive Program Generation

The work by A.D. Shapiro referred to above extended data-oriented induction into the realm of computer-assisted software engineering (CASE), and recent extensions to first-order level have established points of contact with the formal methods school. On the practical side, a path has finally been found to circumvent what has sometimes been termed the 'bottleneck problem' facing knowledge acquisition from experts.

A few years ago it was hoped that experts-doctors, engineers, etc. - would be able to teach their skills directly to computers, which would then be able to carry out much of their routine diagnostic work. Faults or symptoms would be fed into a computer, which would then give a diagnosis of the problem. Unfortunately, it was found that if explicit how-to-do-it rules are required from them, experts cannot effectively feed their own decision-making processes into computers. The soya bean specialist, the analytical chemist or the cardiologist largely reacts 'intuitively' to data, in ways he or she cannot fully explain. In the realm of evaluating credit-worthiness in the finance industry. L. Sterling and E. Shapiro give a telling account of the phenomenon [30].

> The major difficulty was formulating the relevant expert knowledge. Our expert was less forthcoming with general rules for overall evaluation than for rating the financial record, for example. He happily discussed the profiles of particular clients, and the outcome of their credit requests and loans, but was reluctant to generalise.

The observation that specialists transmit their inarticulate skills to trainees by example, rather than by using explicit rules, led in the mid-1980s to the development of commercial programs to exploit 'teaching by showing' in the style illustrated with schoolroom algebra. The machine learns 'how to do it' from experts who supply examples. A number of large corporations, notably British Petroleum [29], have begun using this approach as a cost-effective way of building large applications. The resultant programmer productivities exceed current industry standards by an order of magnitude. The world's largest expert system, BMT, for configuring fire-detection equipment, was built by the German company Brainware using the RuleMaster and 1st Class inductive shells [10]. BMT consists of 150,000 lines of inductively generated C code and is in routine use by the client organisation. The total figure of 9 man-years expended on the project includes management, support staff and domain specialists as well as programmer time. Reductions of software maintenance overheads have also been impressive, as can be seen from the summary results set out in Table 3.

The reader can appreciate the connection with CASE methods by thinking of expert supplied examples as statements in a requirements-specification language: 'in situations of this kind we require the system to do x; in situations of that kind we require it to do y, etc.' In this context an induction routine can be thought of as a kind of compiler, translating from a specification consisting of example cases into a program consisting of if-then-else expressions. A mass of partly redundant and partly incomplete specifications becomes a structured set of efficient executable rules. Permissiveness towards redundancy and incompleteness in the user's tabulation of cases marks the sole but significant departure of rule-induction programming from mainstream decision-table methods [12, 13]. Rule induction in a CASE context is indeed little more than the rebirth of structured decision tables, with this difference: that specification tables, as we may here term them, are initially partial, and are developed incrementally in successive cycles of generate-and-test. Herein lies the key to the extraordinary productivities, illustrated in Table 3, where numbers of the order of 100 lines of installed code per programmer-day are the norm. The even more remarkable gains in software maintain-ability can be similarly explained. Modern inductive shells automatically flag not only incomplete parts of the rule-base but also those which have been derived from generalisation steps in the induction process. The user can then interactively validate flagged passages, editing specification tables as required, re-inducing at each stage from the edited 'spec' [15, 16]. General implications for software manufacture were discussed a few years ago in the author's Royal Society Technology Lecture [14], and have more recently been elaborated in the specific context of interactive validation and of inductive logic programming [16, 21].

8 Automated Synthesis Of Operational Knowledge

The possibility had long been foreseen of using inductive inference to derive operational models (a formal equivalent of 'skills') from deep models (a formal equivalent of 'understanding'). Thus from a numerical model of an aero-engine a qualitative model can in principle be prepared, from which all possible combinations of engine faults may automatically be listed, together with the corresponding sensor readings and test responses. From such a giant tabulation, efficient rules for use in future fault diagnosis may be machine-induced. The wild card in the foregoing is the phrase 'principle'. But Slovenian work using a computer model of the human heart resulted in the discovery by machine of a corpus of new clinical rules for interpreting electrocardiogram patterns [3, 4]. Work at Glasgow's Turing Institute on diagnosing electronic faults in a space satellite confirmed the methodology as fully practicable [24]. These synthetic rule-bases comprise new diagnostic knowhow beyond the achievements of human specialists. Their means of construction also automatically guarantees completeness and correctness with respect to the formal models used to generate the exhaustive datasets. Since the latter are logically equivalent to the descriptive specifications from which they were derived, they too may be viewed as formal specifications, in the form of very long sentences written in a ground-level data-description language. This insight, however, logically irrefutable, seems strange to some who approach via formal methods, where conciseness in a specification is of the essence.

Application	No.of rules	Development man-yrs	Maintenance man-years/year	Inductive tools	
MYCIN	Medical diagnosis	400	100	N/A	N/A
XCON	VAX computer configuration	8,000	180	30	N/A
GASOIL	Hydrocarbon separation system configuration	2,800	1	0.1	ExpertEase and Extran 7
BMT	Configuration of fire-protection equipment in buildings	>30,000	9	2.0	1 st Class and RuleMaster

Table 3: Tabulation from Ref. 29, with 1990 data on BMT added

The Bratko and Pearce syntheses do indeed start from concise (intensional) specifications, but find the path to automated synthesis via an intermediate product, namely a complete extensional form which is its logical equivalent, with the added property of inductive transformability into a set, again concise, of operational rules.

References

[1] J.E. Angus, *On the connection between neural network learning and multivariate nonlinear least squares estimation.* International Journal of Neural Networks 1, 42-46 (1989).

[2] M. Bain and S. Muggleton, *Non-monotonic learning.* In Machine Intelligence 12, edited J.E.Hayes,D.Michie and E.Tyugu), Oxford University Press, Oxford (1991).

[3] I. Bratko, I. Mozetic and N. Lavrac, *Automatic synthesis and compression of cardiological knowledge.* In Machine Intelligence II, edited J.E.Hayes, D.Michie and J.Richards, p.p. 435-454. Oxford University Press, Oxford (1988).

[4] I. Bratko, I. Mozetic and N. Lavrac, *Kardio: a Study in Deep and Qualitative Knowledge for Expert Systems.* MIT Press, Cambridge, MA (1989).

[5] L. Breiman, J.H. Friedman, R.A. Olshen and C.J. Stone, *Classification and Regression Trees.* Wadsworth and Brooks, Monterey, CA (1984).

[6] B.G. Buchanan, D.H. Smith, W.C. White, R.J. Gritter, E.A. Feigenbaum, J. Lederberg and C. Djerassi, Applications of Artificial Intelligence for chemical inference. XXII. Automatic rule formation in mass spectrometry by means of the meta-DENDRAL program. *Journal of the American Chemical Society* 98, 6068-6078 (1976).

[7] L. Cestnik and I. Bratko, *On estimating probabilities in tree pruning.* In Proceedings of the Sixth European Working Session on Learning (EWSL-91), Porto, Portugal (1991).

[8] R. Chilausky, B. Jacobsen and R.S. Michalski, *An application of variable-valued logic to inductive learning of plant disease diagnostic rules.* Proceedings of the Sixth Annual International Symposium on Multi-variable Logic, Utah (1976)

[9] J.H. Friedman, *A recursive partitioning decision rule for nonparametric classification.* IEEE Transactions on Computers, C-26, 404-408 (1977).

[10] J.E. Hayes-Michie (ed.) *Pragmatica* (International Bulletin of the Inductive Programming Special Interest Group), No. 1. Turing Institute Press, Glasgow (1990).

[11] E.B. Hunt, J. Marin and P.Stone, *Experiments in Induction.* Academic Press, New York (1966).

[12] L.S. Levy and H.T. Stump, *Inverted decision tables and their application: automating the translation of specifications to programs.* AT&T Technical Journal 64, 533-558 (1985).

[13] A. Lew, *Proof of correctness of decision table programs.* The Computer Journal 27, 230-232 (1984).

[14] D. Michie, *The superarticulacy phenomenon in the context of software manufacture.* Proceedings of the Royal Society of London A 405, 185-212. Also reproduced in The Foundations of Artificial Intelligence, edited D. Partridge and Y. Wilks), pp. 411-439. Cambridge University Press.

[15] D. Michie, *Problems of automatic concept formation.* In Applications of Expert Systems, vol. 2, edited J.R. Quinlan), pp. 310-333. Addison-Wesley, London (1989).

[16] D. Michie, *Human and machine learning of descriptive concepts.* In ICOT Journal (ed. Hiroshi Hara), March issue, pp. 2-20. Institute for New Generation Technology, Tokyo (1990).

[17] D. Michie, *Personal models of rationality.* In Journal of Statistical Planning and Inference (special issue on foundations of statistics and probability), 21, 381-399.

[18] D. Michie and A. Al-Attar, *Use of sequential Bayes with class probability trees.* In machine Intelligence 12, edited J.E. Hayes, D. Michie and E. Tyugu. Oxford University Press, Oxford (1991).

[19] D. Michie, M. Bain and J.E. Hayes-Michie, *Cognitive models from subcognitive skills.* In Knowledge-based Systems in Industrial Control, edited M. Grimble, S. McGhee and P. Mowforth. Peter Peregrinus, Stevenage (1990).

[20] D. Michie, A. Paterson and J.E. Hayes-Michie, *Learning by teaching.* Second Scandinavian Conference on Artificial Intelligence (SCAI 89).

[21] S. Muggleton, *Inductive logic programming.* In Algorithmic Learning Theory, Springer, Heidelberg (1990).

[22] S. Muggleton and W. Buntine, *Machine invention of first-order predicates by inverting resolution.* Proceedings of the Fifth International Conference on Machine Learning, pp. 339-352.

[23] A. Paterson and T. Niblett, *ACES User Manual.* Intelligent Terminals Ltd, Glasgow (1982).

[24] D. Pearce, *The induction of fault diagnosis system from qualitative models.* In Proceedings of the Seventh National Conference on Artificial Intelligence (AAAI-88) St Paul, Minnesota, pp. 353-357 (1989).

[25] J.R. Quinlan, *Discovering rules by induction from large collections of examples.* In Expert System in the Microelectronic Age, edited D. Michie, pp. 168-201. Edinburgh University Press, Edinburgh.

[26] J.R. Quinlan, *Generating production rules from decision trees.* International Joint Conference on Artificial intelligence 1987 (IJCAI-87), pp. 304-307. Kaufmann, Los Altos, CA (1987).

[27] A.D. Shapiro, *Structured Induction in Expert Systems.* Turing Institute Press and Adisson Wesley, London (1987).

[28] A.D. Shapiro and T. Niblett, *Automatic induction of classification rules for a chess endgame.* In Advances in Computer Chess, vol. 3 edited M.R.B. Clark, pp. 73-92 (1982).

[29] S. Slocombe, K. Moore and M. Zelouf, *Engineering expert system applications.* Presented at the BCS Annual Conference (December 1986).

[30] I. Sterling and E. Shapiro, *The Art of Prolog.* The MIT Press, Cambridge, MA (1986).

ANALYTICAL FORM SOLUTION OF THE DIRECT KINEMATICS OF A 4-4 FULLY IN-PARALLEL ACTUATED SIX DEGREE-OF-FREEDOM MECHANISM

C. Innocenti
DIEM - Facoltà di Ingegneria - Università di Bologna
Viale Risorgimento, 2 - 40136 Bologna -ITALY
Tel 051-6443450 - Fax 051-6443446
e-mail:Meccapp8@ingbo1.cineca.it
AND
V. Parenti-Castelli
Istituto di Ingegneria
Facoltà di Ingegneria - Università di Ferrara
Via Scandiana, 21 - 41100 Ferrara - ITALY
Tel +39-532-65150 - Fax +39-532-740983

Keywords: robots, kinematics, parallel mechanisms, closure equations

Edited by: Jadran Lenarčič

Received: February 10, 1993 **Revised:** March 4, 1993 **Accepted:** March 15, 1993

The paper presents the direct position analysis in analytical form of a six-degree-of-freedom 4-4 fully-parallel mechanism. For a given set of actuator displacements the mechanism becomes a structure and the analysis finds all the possible closures of the structure. The analysis is performed in two steps. First, the two closures of the tetrahedron-like subchain of the structure are found. Then, for each tetrahedron closure, two transcendental equations are determined that represent the closure of the remaining part of the 4-4 structure. The two equations can be reduced to algebraic equations and, after eliminating the unwanted unknowns, a final 8th order equation in only one unknown is obtained. Hence, the maximum number of possible real closures of the 4-4 structure is sixteen. Numerical examples are reported which illustrate and confirm the new theoretical result.

1 Introduction

The requirement of improving the performances of robotic systems seems to find a promising answer in adopting parallel structures which, indeed, experience high payload to link weight ratios, high stiffness and a better position accuracy with respect to the widely used serial manipulators. A wide bibliography for the research on parallel mechanisms and manipulators can be found in [1].

Parallel mechanisms are closed chains with one or more loops where only some pairs are actively controlled. The basic arrangement of a fully in-parallel actuated mechanism consists of two bodies (base and platform) connected to each other by six adjustable-length legs. The leg ends are connected to the base and the platform by spherical pairs. Several connection patterns are possible since two or three spherical pairs can coalesce to form multiple spherical pairs, consequently different mechanisms can be devised. The actuated legs provide the platform with six degrees of freedom relative to the base.

The direct position analysis (DPA) of parallel mechanisms asks for position and orientation (location) of the output link (platform) when a set of actuator displacements is given. It represents a challenging problem for the non-linear equations involved. Indeed, many solutions are possible and numerical methods, currently adopted to this purpose, prove difficulty to find all solutions. On the contrary the analytical form solution, if feasible, would provide all the possible solutions and add insight into the kinematics of the mechanism.

The analytical form solution is represented, in general, by a system of equations in echelon form, that is, if the equations are solved in the appropriate order, each equation can be regarded as one equation in only one unknown. By referring to algebraic equations, the closed form solution is feasible when the order of the equations is less or equal than four; for higher orders the solutions must be determined numerically and the DPA is said to be solvable in analytical form. In general, the first equation to be solved is of high order while the remaining are linear (in the unknown to be solved for). Thus, in this case, the order of the first equation represents the maximum number of possible real solutions.

Only few parallel mechanisms have been solved in analytical form, as reviewed in [2], and some of them have been solved after mechanism geometrical simplification such as base and platform planar [3-7] and symmetrically shaped [8] have been introduced. Mechanisms with general geometry, although more difficult to be solved in analytical form, would give, however, more freedom to the mechanism design, also allowing simpler hardware constructions.

This paper presents the DPA in analytical form of the fully-parallel mechanism schematically shown in Fig. 1. The six legs meet both the base and the platform at four points, centers of spherical pairs. Namely, two legs meet singly the base and the platform, while the remaining four meet in pair both the base and the platform, forming with these a tetrahedron- like pattern. Consequently, either single and double spherical pairs occur at the connection points. The freedom each leg has to rotate about its axis does not affect the gross motion of the mechanism, however it can be eliminated by a suitable hardware design (For instance, by substituting a universal joint for one spherical pair). The four connection points does not necessarily belong to the same plane. It is worth noting that the base and the platform have a symmetric topological role. The mechanism is denoted as 4-4 parallel mechanism. When the leg lengths are frozen the mechanism becomes a statically determined 4-4 structure.

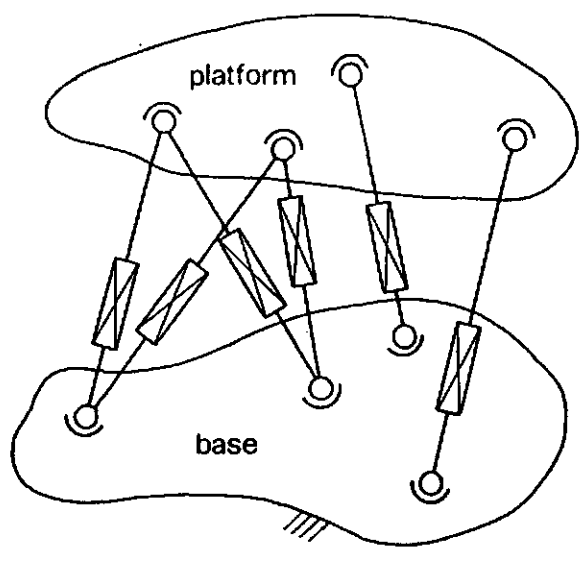

Figure 1: The 4-4 fully-parallel mechanism.

Still maintaining the 4-4 pattern, different mechanisms can be obtained by devising various leg connection arrangements to the base and platform. Several of these have been studied in [4] where the DPA analytical form solution of three cases (and one subcase of these) has been presented. In their study both base and platform have been considered as planar.

The 4-4 mechanism presented in this paper is a new case that has not been treated in [4]. It completes the 4-4 parallel mechanism class, disregarding arrangements with three leg ends that coalesce.

The analytical form solution of the 4-4 mechanism is obtained in two steps. The first step solves for the two possible closures of the tetrahedron-part of the 4-4 structure. For each of them, the second step provides two transcendental equations that represent the closure of the remaining part of the 4-4 structure. The two equations can be reduced to algebraic form and, after eliminating one unknown, a final 8th degree polynomial equation in only one unknown is obtained. Thus, in the complex field, the number of possible solutions for the 4-4 mechanism is sixteen. Finally a numerical example is reported that supports the new theoretical result.

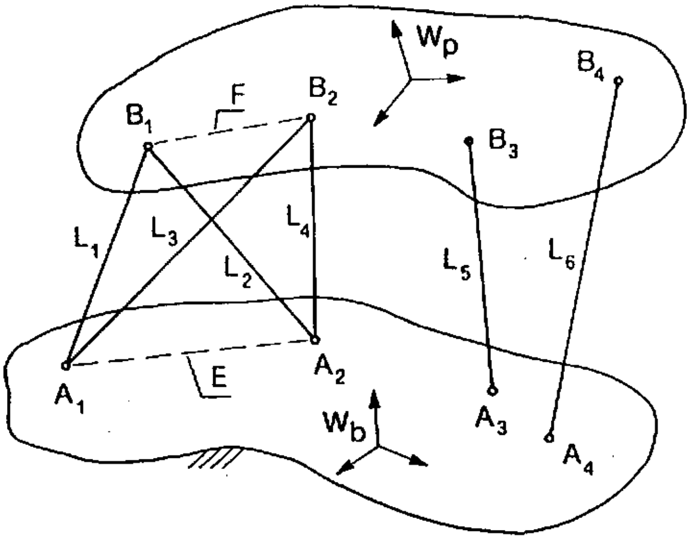

Figure 2: The 4-4 structure.

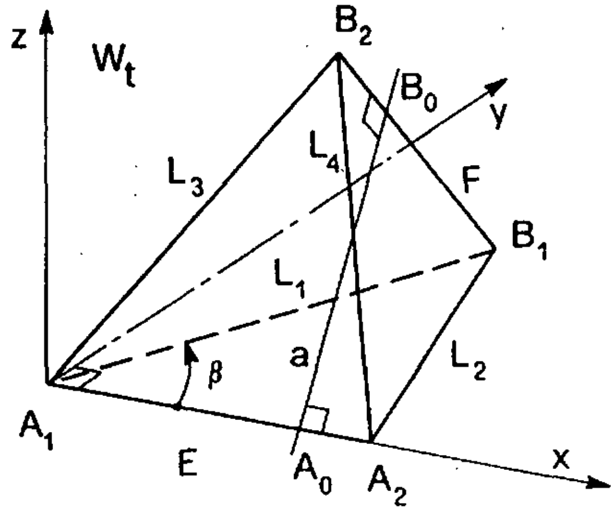

Figure 3: The tetrahedron structure.

2 Direct Kinematics

In this section, the kinematic model of the 4-4 structure, schematically shown in Fig. 2, is developed.

The base points A_j, $j = 1,4$, and the platform points B_j, $j = 1,4$, the centers of spherical pairs, and their positions are known in reference systems W_b and W_p, respectively. System W_b is fixed to the base while system W_p is fixed to the platform. It can be recognized that points A_1, A_2, B_1 and B_2 are the vertices of a tetrahedron, which can be regarded as a rigid body when the leg lengths L_j, $j = 1,4$, are given (see Fig. 2). It is also recognized that the tetrahedron itself can be assembled in different ways.

The closure of the structure is thus performed in two steps. First the closures of the tetrahedron regarded as a separate rigid body are determined. Then, for each of them, the closures of the 4-4 structure are found. Henceforth, the position vector of a point P in reference system W_k is denoted by $(P)_k$, and the components of a vector $\underline{u}$ in reference system W_k are denoted by $(\underline{u})_k$.

2.1 Tetrahedron's assembling

With reference to Fig. 3, system W_t fixed to the tetrahedron is chosen with origin in A_1, positive direction of x axis from A_1 to A_2, y axis such that point B_1 lies in the plane (x,y) with positive y component, and z axis according to the right-hand rule.

Position vector $(A_2)_t$ is:

$$(A_2)_t = (E, 0, 0)^T \tag{1}$$

where

$$E = [(A_2 - A_1)b^2]^{\frac{1}{2}} \tag{2}$$

The coordinates of point B_1 in W_t are determined as follows. The angle $\beta \in [0, \pi]$ of vector $(B_1 - A_1)_t$ forms with x-axis of W_t is given by:

$$\cos\beta = (L_1^2 - L_2^2 + E^2)/(2 \cdot L_1 \cdot E)(3) \tag{3}$$

If $|\cos\beta| > 1$ the tetrahedron cannot be assembled in the real field and the whole 4-4 structure could not be assembled either. Supposing $|\cos\beta|s_1$, it stems:

$$(B_1)_t = L_1 \cdot (\cos\beta, \sin\beta, 0)^T \tag{4}$$

where

$$\sin\beta = +[1 - \cos^2\beta]^{\frac{1}{2}} \tag{5}$$

The position of point B_2 can be determined by the following conditions:

$$(B_2 - A_1)_t^2 = L_3^2 \qquad (6)$$
$$(B_2 - A_2)_t^2 = L_4^2$$
$$(B_2 - B_1)_t^2 = F^2$$

where:

$$F = [(B_2 - B_1)_p^2]^{\frac{1}{2}} \qquad (7)$$

It is worth considering the following position:

$$\begin{aligned}(B_2 - A_1)_t &= \delta \cdot (A_2 - A_1)_t + \\ &\mu \cdot (B_1 - A_1)_t + \\ &\sigma \cdot (A_2 - A_1)_t \cdot (B_1 - A_1)_t\end{aligned} \qquad (8)$$

where δ, μ and σ are quantities to be determined.

The system of equation (6) can then be rewritten in the form:

$$(B_2 - A_1)_t^2 = L_3^2 \qquad (9)$$
$$[(B_2 - A_1)_t - (A_2 - A_1)_t]^2 = L_4^2$$
$$[(B_2 - A_1)_t - (B_1 - A_1)_t]^2 = F^2$$

which gives:

$$\begin{aligned}&(B_2 - A_1)_t^2 = L_3^2 \\ &L_3^2 + (A_2 - A_1)_t^2 - \\ &\quad 2(A_2 - A_1)_t^T \cdot (B_2 - A_1)_t = L_4^2 \\ &L_3^2 + (B_1 - A_1)_t^2 - \\ &\quad 2(B_1 - A_1)_t^T \cdot (B_2 - A_1)_t = F^2\end{aligned} \qquad (10)$$

where, for obtaining the second and the third equation, the first has been taken into account.

By substituting in (10) the position (8) for $(B_2 - B_1)_t$, the following system is obtained:

$$\begin{aligned}&\sigma^2 \cdot [(A_2 - A_1)_t^2 \cdot (B_1 - A_1)_t^2 - \\ &\quad [(A_2 - A_1)_t^T \cdot (B_1 - A_1)_t]^2] = \\ &\quad L_3^2 - \delta^2 \cdot (A_2 - A_1)_t^2 - \mu^2 \cdot (B_1 - A_1)_t^2 - \\ &\quad 2 \cdot \delta \cdot \mu \cdot (A_2 - A_1)_t^T \cdot (B_1 - A_1)t \\ &\delta \cdot (A_2 - A_1)_t^2 + \mu \cdot (A_2 - A_1)_t^T \cdot (B_1 - A_1)_t = \\ &\quad [L_3^2 - L_4^2 + (A_2 - A_1)_t^2]/2 \\ &\delta \cdot (A_2 - A_1)^T \cdot (B_1 - A_1)_t + \mu \cdot (B_1 - A_1)_t^2 = \\ &\quad [L_3^2 - L_4^2 + (A_2 - A_1)_t^2]/2\end{aligned} \qquad (11)$$

Equations (11) represent a linear system in δ and μ that admits a unique solution if the condition

$$\begin{aligned}&(A_2 - A_1)_t^2 \cdot (B_1 - A_1)_t^2 - \\ &\quad [(A_2 - A_1)_t^T \cdot (B_1 - A_1)_t]^2 \neq 0\end{aligned} \qquad (12)$$

is satisfied, i.e., if points A_1, A_2 and B_2 are not aligned.

Let condition (12) be verified. Thus δ and μ can be determined and, from equation (11), the value of σ^2 can be determined. The tetrahedron can be assembled only if:

$$\sigma^2 > 0 \qquad (13)$$

In this case, indeed, the two real solutions for e can be obtained:

$$\sigma_1 = +[\sigma^2]^{\frac{1}{2}},\ \sigma_2 = -[\sigma^2]^{\frac{1}{2}} \qquad (14)$$

and the tetrahedron can be assembled in two distinct ways.

Once the tetrahedron is assembled, it can be considered as a binary rigid body connected to base and platform by means of two revolute pairs whose axes are defined by points A_1, A_2 and points B_1, B_2 respectively (henceforth, axes A_1A_2 and B_1B_2 for brevity).

In order to perform the second step of the DPA, the description of the tetrahedron's shape in terms of Denavit-Hartenberg parameters is of interest. The shortest distance between axes A_1A_2 and B_1B_2 is represented by line A_0B_0 (see Fig. 3). The position of points A_0 and B_0 can be determined as follows.

With the positions:

$$\begin{aligned}(A_0 - A_1) &= l \cdot (A_2 - A_1) \\ (B_0 - B_1) &= m \cdot (B_2 - B_1)\end{aligned} \qquad (15)$$

where l and m are scalar quantities to be determined, the geometrical conditions:

$$(A_2 - A_1)^T \cdot (B_0 - A_0) = 0 \qquad (16)$$
$$(B_2 - B_1)^T \cdot (B_0 - A_0) = 0$$

that represent the orthogonality conditions of the line A_0B_0 with the axes A_1A_2 and B_1B_2 can be written as:

$$\begin{bmatrix} (A_2-A_1)^2, -(A_2-A_1)^T(B_2-B_1) \\ (A_2-A_1)^T(B_2-B_1), -(B_2-B_1)^2 \end{bmatrix} \cdot \begin{bmatrix} l \\ m \end{bmatrix} = \begin{bmatrix} (A_2-A_1)^T(B_1-A_1) \\ (B_2-B_1)^T(B_1-A_1) \end{bmatrix} \quad (17)$$

With condition (12) still verified, linear system (17) can be solved for l and m, after all vectors are measured in reference system W_t.

By equation (15), position of points A_0 in W_b and B_0 in W_p can be determined respectively.

The D-H parameter a, distance between axes A_1A_2 and B_1B_2, is given by:

$$a = \frac{|[(A_2-A_1)_t \times (B_2-B_1)_t] \cdot (B_1-A_1)_t|}{|(A_2-A_1)_t \times (B_2-B_1)_t|} \quad (18)$$

The D-H parameter α, skew angle of axes A_1A_2 and B_1B_2, can be obtained by:

$$\begin{aligned} \cos\alpha &= \frac{(A_2-A_1)_t^T \cdot (B_2-B_1)_t}{E \cdot F} \quad (19) \\ \sin\alpha &= \frac{[(A_2-A_1)_t \times (B_2-B_1)_t] \cdot (B_1-A_1)_t}{E \cdot F \cdot a} \end{aligned}$$

The two possible closures of the tetrahedron are two mirrored closures, thus the parameters l, m, a, and $\cos\alpha$ have the same values for both of them, while parameter $\sin\alpha$ has opposite values.

2.2 4-4 structure closure

Let the tetrahedron be assembled in one of the two possible closures and, recalling the previously reported consideration, consider the tetrahedron as a binary link T connected to base and platform by two revolute pairs. Hence, the 4-4 structure (see Fig. 2) can be represented by the kinematically equivalent structure shown in Fig. 4.

A reference system W_A fixed to the base is chosen with origin in A_0 and z axis directed from A_1 to A_2. Let R_A be the 3×3 rotation matrix for the (coordinate) transformation from W_A to W_b. Similarly a reference system W_B fixed to the platform is chosen with origin in B_0 and z axis directed from B_1 to B_2. Let R_B be the 3×3 rotation matrix for the (coordinate) transformation from W_B to W_p.

A careful inspection of Fig. 4 shows that the location of the platform (with respect to the base) can be uniquely parametrized by the angles ϕ_1 and ϕ_2, which represent respectively the angular position of link T with respect to base and platform. Conversely, for a given location of the platform with respect to the base, angles ϕ_1 and ϕ_2 result uniquely determined.

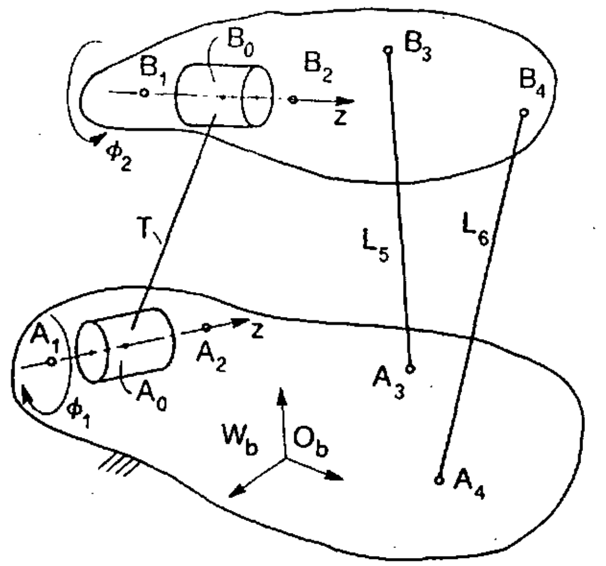

Figure 4: The auxiliary structure.

Let R be the 3×3 rotation matrix for the coordinate transformation from W_B to W_A. Matrix R is a function of angles ϕ_1 and ϕ_2 for a given geometry of the 4-4 structure.

By imposing the constraints due to the legs A_3B_3 and A_4B_4, whose lengths are respectively L_5 and L_6, the closure equations of the equivalent 4-4 structure (of Fig. 4) can be written as:

$$\begin{aligned} &[R \cdot (B_3-B_0)_B + (B_0-A_0)_A - \\ &\quad (A_3-A_0)_A]^2 = L_5^2 \\ &[R \cdot (B_4-B_0)_B + (B_0-A_0)_A - \\ &\quad (A_4-A_0)_A]^2 = L_6^2 \end{aligned} \quad (20)$$

where

$$R = \begin{bmatrix} c_1c_2 - us_1s_2 & -c_1s_2 - us_1s_2 & vs_1 \\ s_1c_2 - uc_1s_2 & -s_1s_2 - uc_1c_2 & -vc_1 \\ v_1s_2 & vc_2 & u \end{bmatrix} \quad (21)$$

$$(B_0 - A_0)_A = \begin{bmatrix} a \cdot c_1 \\ a \cdot s_1 \\ 0 \end{bmatrix} \tag{22}$$

and $u = \cos\alpha$, $v = \sin\alpha$, $c_j = \cos\phi_j$, $s_j = \sin\phi_j$, with $j = 1,2$.

By considering that $(B_0 - A_0)^2 = a^2$, system (20) can be rewritten as follows:

$$\begin{aligned}
&(A_3 - A_0)^T_A R(B_3 - B_0)_B + \\
&\quad (A_3 - A_0)^T_A (B_0 - A_0)_A - \\
&\quad (B_3 - B_0)^T_B R^T (B_0 - A_0)_A = \\
&[(A_3 - A_0)A_2 + (B_3 - B_0)B_2 + a^2 - L_5^2]/2 \\
&(A_4 - A_0)^T_A R(B_4 - B_0)_B + \\
&\quad (A_4 - A_0)^T_A (B_0 - A_0)_A - \\
&\quad (B_4 - B_0)^T_B R^T (B_0 - A_0)_A = \\
&[(A_4 - A_0)A_2 + (B_4 - B_0)B_2 + a^2 - L_6^2]/2
\end{aligned} \tag{23}$$

where

$$R^T \cdot (B_0 - A_0)_A = \begin{bmatrix} a \cdot c_2 \\ -a \cdot s_2 \\ 0 \end{bmatrix} \tag{24}$$

Relation (24) puts into evidence that equations (23) are linear in the cosine and sine of angles ϕ_1 and ϕ_2.

Equations (23), after the well known expressions:

$$c_j = \frac{1 - t_j^2}{1 + t_j^2}; \quad s_j = \frac{2 \cdot t_j}{1 + t_j^2} \tag{25}$$

where $t_j = \tan(mj/2)$, are substituted for sine and cosine, can be rearranged in the following form:

$$\begin{aligned}
&\sum_{i,j=0,2} e_{ij} \cdot t_1^i \cdot t_2^j = 0 \\
&\sum_{i,j=0,2} f_{ij} \cdot t_1^i \cdot t_2^j = 0
\end{aligned} \tag{26}$$

where coefficients e_{ij} and f_{ij} $(i,j = 0,2)$ are functions of both the geometry of the 4-4 structure and the assembly configuration of the tetrahedron.

Equations (23) are two algebraic equations in the unknowns t_1 and t_2. One unknown, t_2 for instance, can be eliminated from (23), and one final equation in unknown t_1 only is thus obtained.

The elimination of t_2 leads to the following condition:

$$\begin{vmatrix} G_0 & G_1 & G_2 & 0 \\ 0 & G_0 & G_2 & G_2 \\ H_0 & H_1 & H_2 & 0 \\ 0 & H_0 & H_1 & H_2 \end{vmatrix} = 0 \tag{27}$$

that, after developing the 4×4 determinant, leads to:

$$\begin{aligned}
&(G_0 H_2 - G_2 H_0)^2 + \\
&\quad (G_0 H_1 - G_1 H_0)(G_2 H_1 - G_1 H_2) = 0
\end{aligned} \tag{28}$$

where the following positions have been considered:

$$\begin{aligned}
G_j &= \sum_{i=0,2} e_{ij} \cdot t_1^i = 0 \\
H_j &= \sum_{i=0,2} f_{ij} \cdot t_1^i = 0
\end{aligned} \tag{29}$$

Indeed, the determinant in (27) is the eliminant of system (26) and its vanishing represents the necessary and sufficient condition for equations (26) to have the same solutions for t_2.

Equation (28) is an 8th order algebraic equation in the unknown t_1, which has eight solutions in the complex field.

Determination of t_2. For each solution $t_1 = t_{1k}$ $(k = 1,8)$ of equation (28), left-hand sides of equations (26) are polynomials in the variable t_2. They generally admit a first-order greatest common divisor whose vanishing provides the common root $t_2 = t_{2k}$.

Thus, the position of the platform with respect to the base can be determined by means of relations (25), (22) and (21). In particular, the position of points B_j of the platform in reference system W_b is given by:

$$\begin{aligned}
(B_j)_b = \; & R_A[R(B_j - B_0)_B + (B_0 - A_0)_A] \\
& + (A_0 - O_b)_b \quad (j = 1,4)
\end{aligned} \tag{30}$$

where $(A_0 - O_b)_b$ is the position vector of point A_0 in W_b, and both matrix R and vector $(B_0 - A_0)_A$ are computed for $t_1 = t_{1k}$ and $t_2 = t_{2k}$.

In conclusion, one closure of the 4-4 structure can be obtained for every solution of equation

	B_1	B_2	B_3	B_4		B_1	B_2	B_3	B_4
	3.91900000	4.90000000	1.03031378	10.14352599		3.91900000	4.90000000	0.10270299	9.31884867
1	4.75758710	3.95490931	7.97888183	6.83380904	9	0.74090168	4.92208620	1.18131426	-3.24071401
	3.31765037	-3.05756315	-0.59115408	5.03362784		-5.75260843	-0.87353730	-1.36604783	-4.75844002
	3.91900000	4.90000000	-0.19569624	8.77551303		3.91900000	4.90000000	0.62022597	2.15722486
2	3.06371316	4.84015608	5.91293201	6.80550135	10	5.02835369	3.67370583	7.46465198	-1.17611044
	4.92494677	-1.25015562	1.93308669	7.82551607		-2.89086460	3.39026334	1.25431822	-4.98866659
	3.91900000	4.90000000	7.41316951	-2.73221333		3.91900000	4.90000000	0.07820448	3.40584298
3	4.88683211	3.82908511	0.58749265	3.57714394	11	5.26323603	3.36111971	6.55728872	-0.85848049
	3.12414963	-3.21373726	1.30578233	3.33880768		-2.43716751	3.70038840	1.75493429	-5.31140770
	3.91900000	4.90000000	8.58081974	-1.86682394		3.91900000	4.90000000	1.11843459	1.25479252
4	5.41905454	-0.30361999	2.15808595	3.92495221	12	4.63028153	4.06640942	8.07611175	-1.50044306
	-2.06767670	-4.98977103	-0.78958414	1.14042236		-3.49312638	2.90763039	0.28617083	-4.64039027
	3.91900000	4.90000000	5.87188051	-2.78349578		3.91900000	4.90000000	1.89922819	10.38741880
5	-5.79059278	-1.21931066	-0.49562467	-6.51139250	13	-4.36007315	2.02193037	-0.53643883	-6.29873348
	0.33237673	4.84801831	-1.13376461	1.07869976		-3.82507531	-4.57184838	0.08648990	-3.19170784
	3.91900000	4.90000000	1.67644660	0.41347608		3.91900000	4.90000000	6.27569737	8.09930050
6	-5.79709458	-1.64849734	-0.41612749	-10.52353198	14	-5.60975474	-2.65814506	-0.28244892	-10.87964686
	-0.18743919	4.71937035	-0.31451165	3.18481399		1.47380147	-4.23370581	1.21730553	0.60585867
	3.91900000	4.90000000	3.45743097	-1.51222561		3.91900000	4.90000000	0.51056606	9.78766240
7	0.84555973	4.93718069	-0.96582475	3.85844994	15	4.74653541	-1.51797840	0.58605307	5.98020127
	5.73815892	0.78373898	0.21494968	8.46291228		3.33344275	4.76295513	1.08111846	0.16534076
	3.91900000	4.90000000	-0.05781944	3.79421623		3.91900000	4.90000000	8.25733044	5.10272935
8	0.96474020	4.95241549	1.48536843	5.60883534	16	4.23881056	-2.16335729	1.61164238	5.98679108
	5.71932823	0.68086767	1.48685219	10.66066503		3.95903069	4.50664900	1.08207460	10.40444371

Table 1: Coordinates of points B_j, $j = 1,4$, in reference system W_b for all 16 closure s of the 4-4 structures.

(28). Moreover, taking into account of the two possible assembly configurations of the tetrahedron, the 4-4 structure admits sixteen closures in the complex field.

3 Case Study

The direct position analysis of a 4-4 mechanisms is reported. For a given set of actuator displacements, which is characterized by a given set of leg lengths L_j, $j = 1,6$, the mechanism becomes the 4-4 structure shown in Fig. 2.

Positions of points A_j, $j = 1,4$, and B_j, $j = 1,4$, are known respectively in W_b and W_p. Arbitrary length unit are considered. The coordinates of A_j in W_b are: $A_1 = (0,0,0)$, $A_2 = (5,0,0)$, $A_3 = (4,4,0)$, $A_4 = (5.5,-2,4.4)$, and the coordinates of B_j in W_p are $B_1 = (0,0,0)$, $B_2 = (6.5,0,0)$, $B_3 = (3,5,0)$, $B_4 = (-1,-3,6)$. The leg lengths are: $L_1 = 7$, $L_2 = 5.9$, $L_3 = 7$, $L_4 = 5$, $L_5 = 5$, $L_6 = 10$.

According to the procedure presented in the paper, the two possible assembly configurations of the tetrahedron have been found each providing eight real solutions. All solutions have been verified to perform the same set of leg lengths.

The sixteen solutions are reported in Table 1, in terms of the coordinates in W_b of points B_j, $(j = 1,4)$, of the platform.

4 Conclusions

The paper presented the direct position analysis of one type of 4-4 Stewart platform mechanism in analytical form. That is, for a given set of actuator displacements, all the possible ways of assembling the 4-4 structure can be determined. The geometry of the mechanism is quite general, that is, neither the base nor the platform are necessarily planar.

The analysis can be solved in two steps. First a quadratic algebraic equation provides the two possible ways of assembling a tetrahedron-like substructure of the 4-4 structure, then an 8th order algebraic equation provides the closures of the remaining 4-4 structure. Thus, in the complex field, the closures of the 4-4 structure resulted to be sixteen.

The new theoretical result has been confirmed by a numerical example that has been reported in the paper.

Acknowledgements

The financial support of the Ministero della Ricerca Scientifica e Tecnologica under MPI 40% and 60% funds is gratefully acknowledged.

References

[1] Merlet J.P., "Les robots les", Hermes, Paris, 1990.

[2] Innocenti C., Parenti-Castelli V., "Basic Ideas and Recent Techniques for the Analytical Form Solution of the Direct Position Analysis of Fully-Parallel Mechanisms", Int. J. of Laboratory and Automation, Vol.4, 1992, pp. 107-113.

[3] Griffis M., Duffy J., "A Forward Displacement Analysis of a Class of Stewart Platform", J. of Robotic Systems, Vol. 6, No. 6, 1989, pp. 703- 720.

[4] Lin W., Duffy J., Griffis M., "Forward Displacement Analysis of the 4-4 Stewart Platforms", 21st ASME Mechanisms Conference, DE-Vol. 25, Sept. 16-19, 1990, Chicago, pp. 263-269.

[5] Nanua P., Waldron K.J., Murthy V., "Direct Kinematic Solution of a Stewart Platform", IEEE Trans. on Robotics and Automation, Vol. 6. No. 4, August 1990, pp. 438-444.

[6] Zhang C., Song S., "Kinematics of Parallel Manipulators with a Positional Subchain", 1990 21th ASME Mechanisms Conference, 16-19 Sept., 1990 Chicago, IL, DE-Vol. 25, pp. 271-278.

[7] Zhang C., Song S., "Forward Kinematics of a Class of Parallel (Stewart) Platforms With Closed-Form Solutions", 1991 IEEE Int. Conference on Robotics and Automation, Sacramento, California, April, 1991, pp. 2676- 2681.

[8] Merlet J.P., "An Algorithm for the Forward Kinematics of General Parallel Manipulators", 5th Int. Conference on Advanced Robotics, ICAR'91, June 19-22, 1991, Pisa, Italy, pp. 1136-1140.

TOWARDS A MATHEMATICAL THEORY OF NATURAL-LANGUAGE COMMUNICATION

Vladimir A. Fomichov
Department "Math. Provision of Information-Processing and Control Systems",
Moscow Institute of Electronics and Mathematics,
Bolshoy Vuzovsky pereulok, 3/12,
109028 Moscow, Russia.
E-mail: onims@miem.msk.su (to Fomichov)
Fax: 007/095/227 28 07 (to Fomichov)

Keywords: communication, formalization, integral formal semantics, knowledge, linguocybernetics, mathematical linguistics, natural language processing

Edited by: A.P. Železnikar

Received: February 26, 1993 **Revised:** March 15, 1993 **Accepted:** March 26, 1993

The need of effective mathematical means for solving a number of tasks associated with the design of natural-language—processing systems (NLPSs) is argued. The basic ideas of cognitive science concerning natural language (NL) understanding are stated. It is shown that the main popular approaches to the formalization of NL–semantics do not satisfy the requirements of cognitive science and computer science as concerns many important aspects of NL–communication.

The interdisciplinary problem of developing a mathematical theory of NL–communication (as a collection of models based on common mathematical means of describing knowledge and texts' structured meanings and being useful for the design of NLPSs) is posed.

Several principles of a new approach to the mathematical study of NL–communication called Integral Formal Semantics are set forth. This approach provides, in particular, the definition of a new class of formal languages called standard K–languages. Some new opportunities afforded by standard K–languages for modeling NL–communication are characterized.

The conclusion is drawn that the premises have been created already for developing in larger scope than before the researches aimed at working out a mathematical theory of NL–communication. It is noted that such a theory will be, in essence, cognitive mathematical linguistics and may be called also mathematical linguocybernetics taking into account the peculiarities of its methods and models.

1 Introduction

The work on constructing the natural–language–processing systems (NLPSs) has been carried out for forty years already, beginning with the first systems of machine translation. A considerable progress has been achieved during this time, and NLPSs are being built now for use in a large spectrum of applications, from communication with relational databases to computer–aided design of complex technical systems with the aid of their natural–language specifications. The great inventory of diverse applications of NLPSs can be found, for instance, in Hahn (1989).

The attained progress permitted to start a number of projects on creating highly complicated NLPSs, first of all, full–text databases (DBs), telephones–interpreters, and computer systems enriching and updating knowledge bases (KBs) of artificial intelligence systems (AISs) by means of extracting information from scientific papers, text–books, patents, etc.

The first attempts in the fifties to build systems of machine translation led to the discovery that the designers of such systems knew very little about natural language (NL) and were able to formalize only a few of its numerous regularities. This situation together with the need of high–level algorithmic languages and translators from such languages caused the emergence and quick development of the theory of formal grammars, languages, and translators.

It seems that an analogical situation takes place nowadays. The complexity of some tasks like the creation of full–text DBs is so great that many researchers acutely feel the necessity of developing effective formal tools for constructing such systems. In particular, Hajičová (1989) notes that the projects like automatic compilation of KBs cannot be realized with a brute force method, but require the development of theories formalizing intricate regularities of NL–comprehension and the use of NL in communication. Sgall (1989) expresses a similar opinion, pointing out the importance of elaborating formal and implementable descriptions of NL–semantics for solving tasks like automatic enriching and updating KBs.

Many aspects of creating effective mathematical methods for the design of NLPSs, especially of their semantic components, are analyzed in Fomichov (1992).

At the very beginning of the nineties the main popular approaches to the formalization of NL–semantics were Montague Grammar and its extensions, Situation Semantics, Discourse Representation Theory, Theory of Generalized Quantifiers (the references can be found, in particular, in Fomichov (1993)), and Dynamic Predicate Logic (Groenendijk & Stokhof, 1989). All these approaches stem from mathematical logic and are underlain by model–theoretic semantics.

Unfortunately, there exists a great distance between the possibilities of mentioned approaches and the demands of cognitive science and computer science. This is shown in sections 2 and 8 of Fomichov (1993) and in sections 2 and 3 of this paper.

Most important is that the expressive power of each of these approaches is insufficient in order (a) to describe structured meanings of real discourses—abstracts, scientific papers, patents, etc.; (b) to represent knowledge about the reality, in particular, to build formal descriptions of notions; (c) to represent goals of intelligent systems; (d) to reflect in models the activity of intelligent systems in the course of natural–language communication: such systems can pose questions, carry out various operations, etc.

Nevertheless, the stock of mathematical means useful for the design of NLPSs is now rather rich. The reason is that new, practically effective approaches to the formal study of NL–semantics and NL–pragmatics were elaborated beyond the frameworks (partially or completely) of enumerated most popular approaches.

One of such new approaches was developed under the LILOG-project funded by the IBM Germany (Herzog & Rollinger, 1991). The other approach called Integral Formal Semantics (IFS) has been created in Russia from the end of the seventies on. IFS provides powerful formal means to describe structured meanings of sentences and discourses, represent knowledge and goals, build models of NLPSs and of NLPSs' subsystems (see Fomitchov (1984), Fomichov (1992, 1993)).

It seems that these new approaches and the works of some other researchers have created a "critical mass" of scientific results permitting to raise the interdisciplinary problem of developing a mathematical theory of NL–communication.

This problem is formulated in section 4 of the present paper. In section 5 several principles of IFS are set forth, and some important new opportunities afforded by IFS (more exactly, by standard K–languages) for modeling NL–communication are described.

The posed problem is discussed in section 6.

2 The Context of Cognitive Psychology and Cognitive Linguistics for the Formal Study of Natural Language

The problems and achievements in the field of constructing NLPSs, on the one hand, and great difficulties on the way of formalizing regularities of NL–comprehension, on the other hand, have evoked an increasing interest of many psychologists and linguists to investigating such regularities.

In linguistics a new branch is formed called cognitive linguistics and being a part of cognitive science. Cognitive linguists consider language "as an instrument for organizing, processing, and conveying information .... The formal structures of languages are studied not as if they were autonomous, but as reflections of general conceptual organization, categorization principles, processing mechanisms, and experiential and environmental influences" (Geeraerts, 1990, p. 1).

The obtained results permitted to formulate the following now widely accepted principles of NL–comprehension.

1. The meaning of a natural–language text (NL–text) is represented by means of a special mental language, or a language of thought (Mel'cuk & Zolkovskij, 1970; Chafe, 1971; Schank, 1972; Apresyan, 1974; Fodor, 1975, 1988; Lakoff & Johnson, 1980; Wierzbicka, 1980; Johnson–Laird, 1983; Fauconnier, 1985; Lakoff, 1987; Langacker, 1987, 1990; Caron, 1989).

2. People build two different (though interrelated) mental representations of a NL–text. The first one is called by Johnson–Laird (1983) the *propositional representation* (PR). This representation reflects the semantic microstructure of a text and is close to the text's surface structure.

   The second representation being a *mental model* (MM) is facultative. The MM of a text reflects the situation described in the text. Mental models of texts are built on the basis of both texts' PRs and diverse knowledge—about the reality, language, discussed situation, and communication participants (Johnson–Laird, 1983).

3. A highly important role in building the PRs and MMs of NL–texts is played by diverse cognitive models accumulated by people during the life—semantic frames, explanations of notions' meanings, prototypical scenarios, social stereotypes, representations of general regularities and area–specific regularities, and other models determining, in particular, the use of metaphors and metonymy (Minsky, 1975; Lakoff & Johnson, 1980; Johnson–Laird, 1983; Fauconnier, 1985; Fillmore, 1985; Seuren, 1985; Johnson, 1987; Lakoff, 1987).

4. The opinion that there exists syntax as an autonomous subsystem of language system has became out of date. Syntax should depend on descriptions of cognitive structures, on semantics of NL.

   Natural language understanding by people doesn't include the phase of constructing the pure syntactic representations of texts. The transition from a NL–text to its mental representation is carried out on the basis of various knowledge and is of integral character (Thibadeau et al, 1982; Johnson–Laird, 1983; Seuren, 1985; Lakoff, 1987; Langacker, 1987, 1990; Caron, 1989; Fisher et al, 1991).

5. Semantics and pragmatics of NL are inseparably linked and should be studied and described by the same means (Schank et al, 1985; Sgall et al, 1986).

A significant role in formulating the enumerated principles was played by the researches on developing computer programs capable to carry out the conceptual processing of NL–texts. This applies especially to the works which can be attributed to the semantics–oriented (or semantically driven) approaches to natural language parsing (for more details see Hahn (1989)).

It appears that the set of principles stated above may serve as an important reference–point for the development and comparison of approaches to mathematical modeling NL–understanding.

3 The Restrictions of Main Popular Approaches to the Formalization of Natural–Language Semantics

3.1 The standpoint of philosophy, cognitive psychology, and cognitive linguistics

The shortcomings of main known approaches to the formal study of NL-semantics were felt in the eighties by many philosophers, psychologists, and linguists.

Basic philosophical ideas of model–theoretic semantics were criticized, in particular, by Putnam (1981), Johnson–Laird (1983), Fillmore (1985), Seuren (1985, 1986), and Lakoff (1987).

The main approaches to the formalization of NL–semantics popular in the eighties—Montague Grammar and its extensions, Situation Semantics, Discourse Representation Theory, and Theory of Generalized Quantifiers are strongly connected with traditions of mathematical logic, of model–theoretic semantics and do not provide formal means permitting to model the processes of NL–comprehension in correspondence with enumerated above principles of modern cognitive science.

In particular, these approaches do not afford effective formal tools to build (a) semantic representations of arbitrary discourses (e.g., of discourses with references to the meaning of fragments being sentences or larger parts of texts), (b) diverse cognitive models, for instance, explanations of notions' meanings, representations of semantic frames, (c) descriptions of sets, relations and operations on sets.

Besides, these approaches are oriented towards regarding assertions. However, it is important to study also goals, promises, advises, commands, questions.

The dominant paradigm of describing surface structure of sentences separately from describing semantic structure (stemming from the pioneer works of Montague (1970, 1974a, 1974b)) contradicts to one of key principles of cognitive linguistics—the principle assuming the dependency of syntax on semantics.

Highly emotionally the feeling of dissatisfaction with the possibilities of the main popular approaches to the formalization of NL–semantics was put into words by Seuren (1985, 1986). In particular, P. Seuren expressed the opinion that the majority of studies on the formalization of NL–semantics was carried out by researchers interested, first of all, in demonstrating the use of formal tools possessed by them, but not in developing formal means permitting to model the mechanisms of NL–comprehension.

As it is known, ecology studies the living beings in their natural environment. In Seuren (1986) the need of new, adequate, *ecological approaches* to studying the regularities of NL–comprehension is advocated. Many reasonings and observations useful for working out ecological approaches to the formalization of discourses' semantics can be found in Seuren (1985). In this monograph a peculiar attention is given to the questions of expressing and discerning the presuppositions of discourses, and the so called Presuppositional Propositional Calculus is suggested.

In the second half of the eighties a number of new results concerning the formalization of NL–semantics was obtained. Let us mention here the approach of Saint–Dizier (1986) motivated by the tasks of logic programming, the results of Cresswell (1985) and Chierchia (1989) on describing sentences' structured meanings, the theory of situation schemata (Fenstad et al, 1987), Dynamic Semantics in the forms of Dynamic Predicate Logic (Groenendijk & Stokhof, 1989) and Dynamic Montague Grammar (Groenendijk & Stokhof, 1990; Dekker, 1990).

Unfortunately, the restrictions pointed above in this section apply also to these new approaches. It should be added that Chierchia (1989) describes structured meanings of some sentences with infinitives. But the expressive power of semantic formulae corresponding to such sentences is very little in comparison with the complexity of real discourses from scientific papers, text–books, encyclopedic dictionaries, legal sources, etc.

Thus, the approaches mentioned in this section do not provide effective and widely–applicable formal tools for modeling NL–understanding in accordance with stated principles of cognitive psychology and cognitive linguistics.

The lack of such means for modeling NL–understanding can be seen also from the most complete text–book on mathematical methods in linguistics (Partee, ter Meulen, & Wall, 1990).

3.2 The standpoint of computer science

One can't say that all approaches to the formalization of NL–semantics mentioned in the precedent subsection are not connected with the practice of designing NLPSs. There are publications, for example, on using for the design of NLPSs Montague Grammar (MG) in modified forms (Clifford, 1988; Hirst, 1988; Sembok & van Rijsbergen, 1990), Situation Semantics (Yasukawa et al, 1988), Discourse Representation Theory (Herzog & Rollinger, 1991).

The language of intensional logic provided by MG is used also in Generalized Phrase Structure Grammars (Gazdar, Klein, Pullum, & Sag, 1985) for describing semantic interpretations of sentences. Such grammars have found a number of applications to natural language processing.

Nevertheless, these and other approaches mentioned in this section possess a number of important shortcomings as concerns applying formal methods to the design of NLPSs and to developing the theory of NLPSs.

The demands of diverse application domains to the means of formal describing natural language may differ. That is why let distinguish for further analysis the following groups of application domains:

1. Natural–language interfaces to databases, knowledge bases, autonomous robots.

2. Full–text databases; computer systems automatically forming and updating knowledge bases of artificial intelligence systems by means of extracting information from scientific papers, text–books, etc., in particular, automatic abstracting systems.

3. Such subsystems of automatized programming systems which are destined for transforming the NL–specifications of tasks into the formal specifications for the further synthesis of programs; such similar subsystems of CAD–systems which are destined for transforming the NL–specifications of designed technical objects into the formal specifications.

Obviously, the enumerated application domains represent only a part of all possible domains, where the development and use of NLPSs are actual. Much larger list of such domains can be found, in particular, in Hahn (1989).

However, for our purpose it is sufficient to consider only mentioned important domains of applying NLPSs. The analysis of formal means for the study of NL needed for these domains will allow us to get a rather complete list of demands to the formal theories of NL which should be satisfied by useful for practice and widely–applicable mathematical tools of studying NL–semantics and NL–pragmatics.

Let us regard for each distinguished group of applications the most essential restrictions of enumerated approaches in this section to the formal study of NL–semantics.

3.2.1 Group 1

Semantics–oriented, or semantically driven NL–interfaces work in the following way (for more details see Hahn (1989)). They transform a NL–input (or at first its fragment) into a formal structure reflecting the meaning of this input (or the meaning of some input's fragment) and called *semantic representation* (SR) of the input or input's fragment. Then the SR is used (possibly, after transforming into a problem–oriented representation) for working a plan of the reaction to the input with respect to a knowledge base, and after this some reaction is produced. The reactions may be highly diverse: AISs can pose questions, fulfill calculations, search required information, transport things, etc.

For constructing NL–interfaces in accordance with these principles, the following shortcomings of MG and its extensions, including Dynamic Montague Grammar, of Situation Semantics, Discourse Representation Theory, Theory of Generalized Quantifiers, Dynamic Predicate Logic, and of other approaches mentioned in this section are important.

1. The effective formal means of describing knowledge fragments, the structure of KBs are not provided.

2. There are no sufficiently powerful and flexible formal means to describe surface and seman-

tic structures of questions and commands expressed by complicated NL–utterances.

3. There are no sufficiently powerful and flexible formal means to represent surface and semantic structures of intelligent systems' goals formed by complicated NL–utterances.

   The possibilities of intelligent systems to understand the goals of communication participants and to use the information about these goals for planning the reaction to a NL–input are not modeled.

4. The enumerated approaches do not give the flexible and powerful means of formal describing structured meanings of NL–discourses (including real discourses from scientific papers, legal sources, patents, etc.). The means of describing structured meanings of discourses are extremely restricted and unsatisfactory from the viewpoint of practice. In particular, discourses with references to the meaning of sentences and larger fragments of texts are not considered.

5. The existence of sentences of many types widely used in real life is ignored. For instance, the structure of the following sorts of sentences is not studied; (a) containing expressions built out of descriptions of objects, sets, notions, events, etc. by means of logical connectives ("Yves has bought a monograph on mathematics, a text–book on chemistry, and a French–Russian dictionary"), (b) describing the operations on sets ("It will be useful to include Professor A. into the Editorial Board of the journal B."), (c) with the words 'notion' or 'term' (the latter in the sense "a notion"), (d) with the words 'respectively' or 'correspondingly' ("Ljubljana, Oslo, and Bratislava are capitals of Slovenia, Norway, and Slovakia, respectively").

6. The models of the correspondences between texts, knowledge about the reality, and texts' semantic representations are not built, and the adequate means for developing models of the kind are not provided.

7. The inputs of NLPSs may be incomplete phrases, even separate words (e.g., the answers to questions in the course of a dialogue). The interpretation of such inputs is to be found in the context of precedent phrases and with respect to the knowledge about the reality and about the concrete discussed situation.

   However, such a capability of NL–interfaces isn't studied and isn't formally modeled.

8. The structure of metaphors and incorrect, but understandable expressions from input texts, the correspondences between metaphors and their meanings are not investigated by formal means.

9. The same situation takes place relatively to the formal study of metonymy—the phenomenon which often manifests in input texts of applied intelligent systems. As Lakoff (1987, p. 77) notes, "metonymy is one of the basic characteristics of cognition. It is extremely common for people to take one well–understood or easy–to–perceive aspect of something and use it to stand either for the thing as a whole or for some other aspect or part of it".

10. Wilks (1990, p. 348) writes that many NLPSs (in particular, systems of machine translation) do not work so as it is explained by the "official" theories in publications about these systems and function "in such a way that it cannot be appropriately described by the upper–level theory at all, but requires some quite different form of description".

The approaches mentioned in this section do not afford the opportunities to describe adequately the main ways of processing information by semantic components of NLPSs.

3.2.2 Group 2

Obviously, the restrictions 1, 3–6, and 8–10 are important also from the viewpoint of solving the tasks like the development of full–text DBs.

The restriction 7 should be replaced by a similar restriction, since fragments of discourses pertaining to business, technology, science, etc. may be incomplete, elliptical phrases.

The following restriction is to be pointed out additionally: the semantic structure of discourses with promises (protocols, contracts often include such discourses), interrelations between surface and semantic structures are not studied and modeled.

Let us make some remarks. The possibility to build semantic representations (SRs) of complicated goals is necessary, in particular, for the development of algorithms permitting to find referents of such expressions as "this success", "this failure", etc.

Yet twenty years ago Wilks (1973, p. 116) noted that "any adequate logic must contain a dictionary or its equivalent if it is to handle anything more than terms with naive denotations such as 'chair' ".

However, all approaches to the formalization of NL–semantics enumerated in this section do not take into account the existence and roles of various semantic dictionaries. Because of this, in particular, reason there is no opportunity to model the correspondence between texts, knowledge about the reality, and SRs of texts.

At first sight, the demands to the means of describing structured meanings of discourses and to the models of the correspondences between texts, knowledge, and SRs of texts are much higher for the second group of applications than for the first one.

Nevertheless, it is not excluded that the joint future work of philosophers, linguists, specialists on computer science, and mathematicians will show that such demands are in fact very similar or the same for these two groups of NLPSs' applications.

3.2.3 Group 3

Additionally to the shortcomings important for the groups 1 and 2, the following restrictions should be mentioned.

1. There are no effective formal means to represent structured meanings of NL–discourses describing algorithms, methods of solving diverse tasks. In particular, there are no adequate formal means to describe on a semantic level the operations with sets.

2. The opportunity to build semantic representations of complicated notions' descriptions (from encyclopedic dictionaries, etc.) is not afforded.

It appears that the collection of restrictions stated above provides a useful reference–point for further enriching the stock of means and models for the mathematical study of NL–communication.

4 The Interdisciplinary Problem of Developing a Mathematical Theory of Natural–Language Communication

In situation when known formal methods of studying NL–semantics proved to be ineffective for solving many actual tasks of designing NLPSs, a number of researchers in diverse countries pointed out at the necessity to search new mathematical ways of modeling NL–communication.

According to the opinion of Peregrin (1990), additional efforts are to be undertaken in order to develop powerful formal means for describing the regularities of NL–understanding. For this a full–fledged linguistic analysis of NL–phenomena should be carried out.

Habel (1988) notes the actuality of creating adequate mathematical foundations of computational linguistics and underlines the necessity to model the processes of NL–communication on the basis of formal methods and theories of cognitive science.

Fenstad & Lonning (1990, p. 70) posed the task of working out adequate formal methods for Computational Semantics—"a field of study which lies at the intersection of three disciplines: linguistics, logic, and computer science". Such methods should permit, in particular, to establish interrelations between pictorial data and semantic content of a document.

A. P. Ershov, the prominent Russian theoretician of programming, raised the problem of developing a formal model of Russian language (Ershov, 1986).

It is very interesting how close the ideas of Seuren (1986) about the need of ecological approaches to the formal study of NL are to the following words of A. P. Ershov published in the same 1986: "We want as deeply as it is possible to get to know the nature of language and, in particular, of Russian language. A model of Russian language should become one of manifestations of this knowledge. It is to be a formal system which should be adequate and equal–voluminous to the living organism of language, but in the same time it should be anatomically prepared, decomposed, accessible for the observation, study, and modification" (Ershov, 1986, p. 12).

The need of good formal models for good engineering in the field of designing NLPSs is argued in (Joshi, 1989).

Thus, the idea of developing new formal methods destined for the study of NL–semantics and adequate to the complexity of NL has been finding gradually supporters in various countries.

A number of such formal methods useful for the design of NLPSs was developed during the eighties. Let us mention now only three examples.

Appelt & Kronfeld (1987) elaborated a formal theory of referring within the framework of a general theory of speech acts and rationality.

A large volume of studies aimed at creating a formal theory of natural–language question–answering systems was carried out in Germany under the known project LILOG funded by the IBM Germany. One of the goals of this project was to determine formal languages permitting to reflect on a semantic level a wide range of NL phenomena and serving as target languages in order to represent (in a logical form) the information extracted from texts in German (Herzog & Rollinger, 1991).

A new, powerful approach to the mathematical study of NL–semantics and NL–pragmatics called Integral Formal Semantics (IFS) has been developed in Moscow from the end of the seventies on. This approach provides, in particular, the highly effective mathematical means to describe structured meanings of sentences and discourses, to represent knowledge and goals of intelligent systems (see the next section).

It seems that the demands of practice, on the one hand, and the results obtained in mathematical linguistics, computer science, and cognitive science, on the other hand, permit and force us to go to a new level of mathematical studying natural language and to raise the problem of developing a *mathematical theory of NL–communication.*

It is the problem of building a collection of mathematical models based on common mathematical means and useful for the design of NLPSs. The collection of such models should permit to reflect and to study all the regularities of NL–communication which were mentioned in the precedent section.

Besides, the new theory should help to solve at least the tasks pertaining to three groups of application domains distinguished above.

Naturally, such a new mathematical theory is to be developed jointly by philosophers, psychologists, linguists, specialists on computer science, and mathematicians. Hence the posed problem is interdisciplinary.

5 Shortly About Integral Formal Semantics

The analysis carried out in the sections 2 and 3 may create the impression that a long way should be gone yet in order to get mathematical tools permitting to overcome stated restrictions and to develop complicated models useful for the design of NLPSs and for representing hypotheses in cognitive psychology and cognitive linguistics.

Fortunately, the real situation is not so pessimistic, and in fact powerful mathematical means for modeling NL–communication were elaborated yet in the eighties.

Such means are provided by an original approach to the mathematical study of Nl–communication called Integral Formal Semantics (IFS). This term is introduced, in particular, in Fomichov (1993).

The first version of IFS was developed in 1978 – 1983. By 1984 a great volume of theoretical results was obtained. A part of these results is reflected in Fomichov (1978, 1980, 1981), Fomitchov (1983, 1984). One can find additional references in Fomichov (1992).

It is interesting that approximately in the same time, in 1981–1983, the first publications on the Theory of Generalized Quantifiers (Barwise & Cooper, 1981), Discourse Representation Theory (Kamp, 1981), Lexical–Functional Grammars (Kaplan & Bresnan, 1982), and Situation Semantics (Barwise & Perry, 1983) appeared.

Strong impulses to the development of IFS were given by the well known works of T. Winograd on the program understanding NL, W. A. Woods on the system LUNAR, R. Schank on the Conceptual Dependency theory, Y. Wilks on machine translation published in the first half of the seventies, by the paper of D. Bobrow and T. Winograd on the knowledge representation language KRL, and by the main ideas of the known theory of linguistic models "Meaning–Text" developed in Moscow (Mel'cuk & Zolkovskij, 1970; Apresyan, 1974).

The set of basic philosophical ideas of IFS includes, in particular, the following principles.

1. The chief task of researches on the formalization of NL–semantics is to be the elaboration of formal models of NLPSs and of such subsystems of NLPSs which belong to the so called semantic components of NLPSs.

   The structure of such models of a number of kinds is suggested in Fomichov (1992); the additional references can be found in Fomichov (1993).

2. The studies are to be oriented towards considering not only assertions, but also commands and questions which may be inputs of NLPSs.

3. The basis of researches is to be a formal model reflecting many peculiarities of semantic structures of sentences and discourses of arbitrary great length and providing the description of some class of formal languages convenient for building semantic representations (SRs) of NL–texts in a large spectrum of applications and on different levels of representation.

Two main variants of such a model are developed. The first variant is provided by the theory of free S–models, S–calculuses and S–languages, T–calculuses and T–languages (the STCL–theory). This theory afforded already in 1981 – 1983 a really ecological approach to the formalization of NL–semantics providing powerful and convenient for linguists mathematical means for representing both structured meanings of NL–texts and knowledge about the reality (Fomichov, 1981; Fomitchov, 1983, 1984).

The expressive power of S–languages and T–languages is very great (see the section 5 of Fomichov (1992)) and essentially exceeds, for example, the expressive power of Discourse Representation Theory.

The basic ideas of the STCL–theory were reported at the First symposium of the International Federation of Automatic Control (IFAC) on Artificial Intelligence which was held in 1983 and were published in the proceedings of this symposium (it should be noted that the paper Fomitchov (1984) is a considerably abridged version of Fomitchov (1983)).

The STCL–theory became the starting point for developing the theory of K–calculuses, algebraic systems of conceptual syntax, and K–languages (the KCL–theory) being nowadays the central component of IFS. The foundations of the KCL–theory are stated in Fomichov (1988a) (it is simultaneously a monograph and a text–book) and also, in particular, in Fomichov (1988b, 1992, 1993).

The KCL–theory provides the definition of a class of formulae permitting (a) to describe structured meanings of complicated sentences and discourses and (b) to build the representations of diverse cognitive structures.

For instance, this theory allows us to describe in a mathematical way structured meanings of:

a. the expressions "a group of seven students", "a man in the age from 21 to 27 years being a chemist or a biologist" (Fomichov, 1992);

b. sentences "Namur and Lyon are the cities of Belgium and France, correspondingly", "Namur, Leuven, Gent belong to the cities of Belgium, and London does not belong to the cities of Belgium, or France, or Sweden", "There exists a country in Europe with the number of cities greater than 15", "Belongs Gent to the cities of Belgium?" "To whom did P. Carpenter phoned at 4:30 p.m.?" (Fomichov, 1992), "The terms 'cytosine', 'ischemia', and 'insulin' are used in genetics, cardiology, and endocrinology, respectively" (Fomichov, 1993);

c. discourses

"The chemical action of a current consists in the following: for some solutions of acids (salts, alkalis), by passing an electrical current across such a solution one can observe isolation of the substances contained in the solution and laying aside these substances on electrodes plunged into this solution.

For example, by passing a current across a solution of the blue vitriol ($CuSO_4$) a pure copper will be isolated on the negatively charged electrode.

One use this to obtain pure metals" (Fomichov, 1992),

"An adenine base on one DNA strand links only with a thymine base of the opposing DNA strand. Similarly, a cytosine base links only with a guanine base of the opposite DNA strand" (Fomichov, 1993).

The referenced papers contain semantic representations of these NL-expressions built with the help of the so called standard K-languages.

Besides, standard K-languages afford large opportunities to represent structured meanings of notions' definitions. In particular, semantic representations of the following notions' definitions are built in Fomichov (1993):

"Thrombin is an enzyme which helps to convert fibrinogen to fibrin during coagulation",

"Sphygmomanometer is instrument destined to measure blood pressure",

"A genotype is a collection of all genes located in chromosomes of an organism",

"Type A blood group are persons who possess type A isoantigen on red blood cells and anti-B agglutinin in plasma",

"Messenger RNA is molecule which is formed from DNA and transfers the genetic code to the cytoplasm where protein synthesis occurs",

"Tympanic membrane is a membrane between the outer and middle ear. Surgical evacuation of the purulent material through the tympanic membrane prevents hearing loss and mastoiditis. This procedure is called a myringotomy".

It should be mentioned that SRs of all texts adduced here (except the texts with the word 'correspondingly' or with the expression 'the terms') can be constructed also by means of S-languages of type 5 provided by the STCL-theory and characterized in (Fomitchov, 1983, 1984).

Hence the expressive power both of standard K-languages and of S-languages of type 5 considerably exceeds the expressive possibilities of other approaches to the formalization of Nl-semantics discussed above.

The *integral character* of IFS manifests in affording opportunities to describe both structured meanings of texts and knowledge about the reality.

Some important possibilities of S-languages to represent knowledge are described in Fomitchov (1983, 1984). K-languages also provide similar possibilities (see the section 9 of Fomichov (1992)).

In the subsection 9.4 of Fomichov (1992) some opportunities of recording NL-communication by means of standard K-languages are explained. I.e., it is shown how it is possible to represent in a formal manner the actions carried out by intelligent systems in the course of communication.

The subsection 9.5. of Fomichov (1992) is devoted to describing with the aid of standard K-languages semantic-syntactic information associated with words and fixed word combinations.

It may be added that IFS gives also two variants of a complicated and useful for practice mathematical model of the correspondence "Text (phrase or discourse) + Knowledge about reality ⟶ Semantic representation of a text". The references and brief information about this model can be found in Fomichov (1992). This model satisfies the demand of cognitive linguistics about the dependency of syntax on semantics.

6 Discussion

It should be noted that the problem of creating a mathematical theory of NL-communication was raised in Fomichov (1978, 1980, 1981) and Fomitchov (1983, 1984). These publications appear to be the first ones (or belong to the first ones), where this problem is formulated.

In Fomichov (1988b, 1992) the conclusion is drawn that the KCL-theory provides the premises for starting in more large scope than before the development of a mathematical theory of NL-communication and gives good chances to work out such a theory.

Taking into account the experience obtained in the framework of Integral Formal Semantics, it appears to be worth-while to base a mathematical theory of NL-communication on the powerful, widely-applicable or universal formal means of describing knowledge and discourses' structured meanings.

Such means, on the one hand, can be used for building models of knowledge bases and bases of goals. On the other hand, formal representations of texts' structured meanings can be used for describing, firstly, surface structures of texts and, secondly, the correspondences "Text + Knowledge ⟶ Semantic representation of a text".

The idea of describing surface structures consists in getting these structures by means of indeterministic transformations of semantic structures (transformations are to be based on various semantic–syntactic dictionaries).

It is the idea absolutely contradictory to the approach of Montague Grammar, but corresponding to the principle of cognitive linguistics about the dependency of syntax on semantics. It seems that this idea points out at the only effective way of describing surface structures of scientific papers, patents, etc. and is in a good agreement with the principles of designing the semantics–oriented NLPSs stated by Hahn (1989).

A mathematical theory of NL–communication should be based on the ideas of cognitive linguistics. Hence such a theory will be, in essence, and may be called *cognitive mathematical linguistics.*

Since this hypothetical new theory will describe the activity of intelligent systems having knowledge bases and goals, the new theory may be called also *mathematical linguocybernetics.*

Železnikar (1988a, 1988b, 1989) expresses the opinion that computer systems will be gradually replaced in the future by information machines. Natural languages are the most important means of storing and conveying information.

That is why it appears that the creation of a domain–independent and realization–independent mathematical theory of NL–communication will contribute essentially to developing a theory of information machines.

7 Conclusions

It may be hoped that this paper will help to join the efforts of philosophers, psychologists, linguists, specialists on computer science, and, naturally, mathematicians and to work out a formal theory of natural–language communication (or cognitive mathematical linguistics, or mathematical linguocybernetics).

Quite good premises for speeding–up the studies in this direction creates Integral Formal Semantics—a new approach to the mathematical investigation of NL–semantics and NL–pragmatics developed in Russia. The task of demonstrating all these premises goes far beyond the scope of the present paper and will be the subject of the future research work.

It seems that the creation of such a new theory will be of high significance for the development of Informatics.

References

[1] Appelt, D. & Kronfeld, A. (1987). A computational model of referring. In Proc. of the Tenth Int. Joint Conf. on Art. Intelligence. Milan, 640-647.

[2] Apresyan, Yu. D. (1974). Lexical Semantics. Synonymic Means of Language. Moscow: Nauka Publ. (in Russian).

[3] Barwise, J. & Cooper, R. (1981). Generalized quantifiers and Natural Language. Linguistics & Philosophy, 4 (2), 159-219.

[4] Barwise, J. & Perry, J. (1983). Situations and Attitudes. Cambridge, MA: The MIT Press.

[5] Caron, J. (1989). Precis de Psycholinguistique. Paris: Presses Universitaires de France.

[6] Chafe, W. L. (1971). Meaning and the Structure of Language. Chicago and London: The University of Chicago Press.

[7] Chierchia, G. (1989). Structured meanings, thematic roles and control. In Chierchia, G., Partee, B. H., & Turner, R. (Eds.), Properties, Types and Meaning, 2. Semantic Issues. Dordrecht: Kluwer Acad. Publ., 131-166.

[8] Clifford, J. (1988). Natural language querying of historical data bases. Computational Linguistics, 14 (4), 10-34.

[9] Cresswell, M. J. (1985). Structured Meanings: the Semantics of Propositional Attitudes. Cambridge, MA, London: The MIT Press.

[10] Dekker, P. (1990). The scope of negation in discourse: towards a flexible dynamic Montague grammar. In Quantification and Anaphora 1. DYANA. Deliverable R2.2.A. July 1990. Edinburgh, 79-134.

[11] Ershov, A. P. (1986). Machine fund of Russian language: An external task statement. In Karaulov, Yu. N. (Ed.), Machine Fund

of Russian Language: Ideas and Opinions. Moscow: Nauka Publ., 7-12 (in Russian).

[12] Fenstad, J. E., Halvorsen, P.-K., Langholm, T., van Benthem, J. (1987). Situations, Language and Logic. Dordrecht: D. Reidel.

[13] Fenstad, J. E. & Lonning, J. T. (1990). Computational semantics: steps towards "Intelligent" Text Processing. In Studer, R. (Ed.), Natural Language and Logic. Berlin, etc.: Springer-Verlag, 70-93.

[14] Fillmore, C. (1985). Frames and the semantics of understanding. Quaderni di Semantica, 6, 222-253.

[15] Fisher, C., Gleitman, H., & Gleitman, L. R. (1991). On the semantic content of subcategorization frames. Cognitive Psychology, 23, 331-392.

[16] Fodor, J. A. (1975). The Language of Thought. New York: Crowell.

[17] Fodor, J. A. (1988). Psychosemantics: The Problem of Meaning in the Philosophy of Mind. 2nd print. Cambridge, MA, London: The MIT Press.

[18] Fomichov, V. A. (1978). A mathematical model of natural-language texts' semantic analysis. In Theses of reports of the Second sc.-techn. conf. of young specialists of VNTITSentr. Moscow: All-Union Scientific-Technical Information Centre (VNTITSentr), 58-60 (in Russian).

[19] Fomichov, V. A. (1980). Some principles of mathematical describing natural language's subsets. In A. K. Ailamazyan (Ed.), Reports of 1st and 2nd sc.-techn. conf. of young specialists. Informational Processes and Their Automatization. Moscow, VNTITSentr, 125-136 (in Russian).

[20] Fomichov, V. A. (1981). To the theory of logic-algebraic modelling the speech-forming mechanisms of conceptual level. 1. Task statement and the idea of an approach to its resolving. Moscow Institute of Electronic Engineering, 85 pp. - Paper deposited in AUISTI of Ac. Sc. USSR (All-Union Institute of Scientific and Technical Information; in Russian - Vsesoyuzny Institut Nauchnoy i Teknicheskoy Informatsii, or VINITI) 27/10/81, No. 4939-81 Dep. - The abstract 2B1386DEP in the Abstracts Journal "Mathematika" of AUISTI, 1982, No. 2 (in Russian).

[21] Fomitchov, V. A. (1983). Formal systems for natural language man-machine interaction modelling. In International Symposium on Artificial Intelligence IFAC, Vol. 1. Leningrad: Ac. Sc. USSR, 223-243 (in two variants - in Russian and in English). Paper presented at the First Symposium of the Int. Federation of Automatic Control on Artificial Intelligence.

[22] Fomitchov, V. A. (1984). Formal systems for natural language man-machine interaction modelling. In V. M. Ponomaryov (Ed.), Artificial Intelligence. Proceedings of the IFAC Symposium. Leningrad, USSR, 4 - 6 October 1983 (IFAC Proc. Series, 1984, No. 9). Oxford, New York, etc.: Pergamon Press, 203-207.

[23] Fomichov, V. A. (1988a). Representing Information by Means of K-calculuses: Textbook. Moscow: The Moscow Institute of Electronic Engineering (MIEE) Press (in Russian).

[24] Fomichov, V. A. (1988b). About the means of constructing natural-language-communication mathematical theory. In V. N. Markov (Ed.), Mathematical provision of calculating, informational, and control systems. Moscow: The MIEE Press, 21-25 (in Russian).

[25] Fomichov, V. (1992). Mathematical models of natural-language-processing systems as cybernetic models of a new kind. Cybernetica (Belgium), XXXV (1), 63-91.

[26] Fomichov, V. A. (1993). K-calculuses and K-languages as powerful formal means to design intelligent systems processing medical texts. Cybernetica, XXXVI (1).

[27] Gazdar, G., Klein, E., Pullum, G., & Sag, I. (1985). Generalized Phrase Structure Grammar. Oxford: Blackwell.

[28] Geeraerts, D. (1990). Editorial Statement. Cognitive Linguistics, 1 (1), 1-3.

[29] Groenendijk, J. & Stokhof, M. (1989). Dynamic Predicate Logic, towards a compositional, non-representational semantics of discourse. The University of Amsterdam. The ITLI Prepublication Series, LP-89-02.

[30] Groenendijk, J. & Stokhof, M. (1990). Dynamic Montague Grammar. In Quantification and Anaphora I. DYANA. Dynamic Interpretation of Natural Language. ESPRIT Basic Research Action BR3175. Deliverable R2.2.A. July 1990, 1-35.

[31] Habel (1988). Kognitionswissenschaft als Grundlage der Computerlinguistik. In Computerlinguistik und ihre theoretischen Grundlagen. Berlin etc.: Springer-Verlag, 204-209.

[32] Hahn, U. (1989). Making understanders out of parsers: semantically driven parsing as a key concept for realistic text understanding applications. International Journal of Intelligent Systems, 4 (3), 345-385.

[33] Hajičová, E. (1989). The introduction to the special issue of the journal. Computers and Artificial Intelligence, 8 (5).

[34] Herzog, O. & Rollinger, C.-R. (1991) (Eds.). Text Understanding in LILOG. Integrating Computational Linguistics and Artificial Intelligence. Final Report on the IBM Germany LILOG-Project. Berlin etc.: Springer-Verlag.

[35] Hirst, G. (1988). Semantic interpretation and ambiguity. Artificial Intelligence, 34, 131-177.

[36] Johnson, M. (1987). The Body in the Mind: The Bodily Basis of Meaning, Imagination, and Reason. Chicago: The University of Chicago Press.

[37] Johnson-Laird, P. (1983). Mental Models. Cambridge, MA: Harvard University Press.

[38] Joshi, A. K. (1989). Formal and computational models. The Prague Bulletin of Mathematical Linguistics, 51, 5-9.

[39] Kamp, H. (1981). A theory of truth and semantic representation. In J. Groenendijk, T. Janssen, & M. Stokhof (Eds.), Formal Methods in the Study of Natural Language. Part 1. Amsterdam: Mathematical Centre, 227-322.

[40] Kaplan, R. M. & Bresnan, J. (1982). Lexical-Functional Grammar: a formal system for grammatical representation. In J. Bresnan (Ed.), The Mental Representation of Grammatical Relations. Cambridge, MA: The MIT Press, 173-281.

[41] Lakoff, G. & Johnson, M. (1980). Metaphors We Live By. Chicago: The University of Chicago Press.

[42] Lakoff, G. (1987). Women, Fire and Dangerous Things: What Categories Reveal about the Mind. Chicago: The University of Chicago Press.

[43] Langacker, R. W. (1987). Foundations of Cognitive Grammar. Vol. 1. Theoretical Prerequisites. Stanford, CA: Stanford University Press.

[44] Langacker, R. W. (1990). Concept, Image, and Symbol. The Cognitive Basis of Grammar. Berlin, New York: Mouton de Gruyter.

[45] Mel'cuk, I. & Zolkovskij, A. (1970). Towards a functioning meaning-text model of language. Linguistics, 57, 10-47.

[46] Minsky, M. (1975). Framework for representing knowledge. In The Psychology of Computer Vision, Winston, P. H. (Ed.). N. Y.: Mc Graw-Hill book comp.

[47] Montague, R. (1970). Universal Grammar. Theoria, 36, 373-398.

[48] Montague, R. (1974a). English as a formal language. In Formal Philosophy. Selected papers of Richard Montague, Thomason, H. (Ed.), New Haven and London: Yale University Press, 188-221.

[49] Montague, R. (1974b). The proper treatment of quantification in ordinary English. Ibid., 247-270.

[50] Partee, B. H., ter Meulen, A., & Wall, R. E. (1990). Mathematical Methods in Linguistics. Dordrecht etc.: Kluwer.

[51] Peregrin, J. (1990). On a logical formalization of natural language. Kybernetika, 26 (4), 327-341.

[52] Putnam, H. (1981). Reason, Truth, and History. Cambridge: Cambridge University Press.

[53] Saint-Dizier, P. (1986). An approach to natural-language semantics in Logic Programming. Journal of Logic Programming, 3 (4), 329-356.

[54] Sembok, T. M. T. & van Rijsbergen, C. J. (1990). A simple logical-linguistic document retrieval system. Information Processing & Management, 26 (1), 111-134.

[55] Seuren, P. A. M. (1985). Discourse Semantics. Oxford: Basil Blackwell.

[56] Seuren, P. A. M. (1986). Formal theory and the ecology of language. Theoretical Linguistics, 13, 1-18.

[57] Schank, R. (1972). Conceptual Dependency: A theory of natural language understanding. Cognitive Psychology, 3 (4).

[58] Schank, R., Birnbaum, L., & Mey, J. (1985). Integrating semantics and pragmatics. Quaderni di Semantica, VI, No. 2.

[59] Sgall, P., Hajicova, E., & Panevova, J. (1986). The Meaning of the Sentence in Its Semantic and Pragmatic Aspects. Prague and Dordrecht: Academia and D. Reidel.

[60] Sgall, P. (1989). The tasks of semantics and the perspectives of computers. Computers and Artificial Intelligence, 8 (5), 403-421.

[61] Thibadeau, R., Just, M. A., & Carpenter, P. A. (1982). A model of the time course and content of reading. Cognitive Science, 6, 157-203.

[62] Wierzbicka, A. (1980). Lingua Mentalis. The Semantics of Natural Language. Sydney, etc.: Academic Press.

[63] Wilks, Y. (1973). An artificial intelligence approach to machine translation. In Schank, R. & Colby, K. (Eds.), Computer Models of Thought and Language. San Francisco: Freeman, 114-151.

[64] Wilks, Y. (1990). Form and content in semantics. Synthese, 82 (3), 329-351.

[65] Yasukawa, H., Suzuki, H., & Noguchi, N. (1988). Knowledge representation language based on Situation Theory. In K. Fuchi and L. Kott (Eds.), Programming of Future Generation Computers II: Proc. of the 2nd Franco-Japanese Symp. Amsterdam etc.: North-Holland, 431-460.

[66] Železnikar, A. P. (1988a). Principles of Information. Cybernetica, XXXI (2), 99-122.

[67] Železnikar, A. P. (1988b). Information Determinations I. Cybernetica, XXXI (3), 181-213.

[68] Železnikar, A. P. (1989). Information Determinations II. Cybernetica, XXXII (1), 5-44.

ALPHA AXP OVERVIEW

Jurij Šilc[†], Borut Robič[†] and Jože Buh[‡]

[†] Jožef Stefan Institute
Laboratory for Computer Architectures, Jamova 39, Ljubljana, Slovenia
Phone: +38 61 159 199, Fax: +38 61 161 029
E-mail: jurij.silc@ijs.si or borut.robic@ijs.si

[‡] EuroComputer Systems
Digital Authorized Representative, Vojkova 50, Ljubljana, Slovenia
Phone: +38 61 182 500, Fax: +38 61 181 005

Keywords: Alpha AXP architecture, RISC architecture, multiple instruction issue, shared-memory multiprocessing, overview

Edited by: Marco Botta

Received: February 22, 1993 **Revised:** March 15, 1993 **Accepted:** March 22, 1993

The paper describes the Alpha AXP architecture and some existing implementations. It is a true 64-bit RISC architecture supporting multiple instruction issue, shared-memory multiprocessing, and several today's leading operating system environments. The first Alpha AXP microprocessor DECchip 21064 and several hardware products using it are also briefly described.

1 Introduction

As the 20th Century draws to a clone, more and more computer power is being needed to drive our extremely complex applications. The application of computing took us from mainframes and 16-bit minicomputers of the early 1970s to the 32-bit microprocessors of the mid-1980s, in the age of desktop computing with windowing user interfaces. What will stimulate the next technological leap that brings us applications of the future? One of the answers might be an advanced 64-bit RISC architecture.

In late 1970s DEC introduced the VAX architecture with one hardware product (the VAX 11/780), one operating system (VAX VMS), one network (DECnet) and one high-level language (Fortran).

In early 1990s it introduced Alpha AXP architecture with several hardware products, three operating systems, multiple networking protocols, multiple languages, and so forth.

In this paper, we shall present a short overview of the Alpha project. We will start with description of main project goals and proceed to basic architectural features such as multiple instruction issuing and the possibility of multiprocessing. The first implementation of the Alpha AXP architecture is DECchip 21064. The chip and several hardware products using it are briefly described. Finally, we give a short overview of the operating systems supported by Alpha AXP architecture.

2 Project overview

Alpha was the largest engineering project in Digital's history, spanning more than 30 engineering groups in ten countries [8]. It started with a task force chartered to define a high-performance RISC architecture for the 1990s and beyond. Even before the architecture definition was complete, work began on implementing a high-performance microprocessor. The work was done in the summer 1991 when a product-level chip DECchip 21064 was fabricated [4]. However, a prototype chip was fabricated in late 1990 and was used in an experimental multiprocessor system called ADU (Alpha demonstration unit) [9]. This system was of great benefit to software developers since it allowed them to boot the first Alpha AXP operating systems early in 1991.

3 Project goals

The Alpha AXP architecture[1] project started with a small list of goals:

• **High performance and longevity.** In current architectures, a primary limitation is the 32-bit memory address. Therefore, the project adopted a full 64-bit architecture (with a minimal number of 32-bit operations for backward compatibility). It was estimated that it would be reasonable for raw clock rates to improve only by a factor of 10 over the coming 25 years. If the clock cannot be made faster, more work should be done per clock tick to obtain increase in performance. Alpha AXP architecture was therefore designed to encourage *multiple instruction issue* implementations eventually sustaining about 10 new instructions starting every clock cycle. Additional performance improvements are to be expected from multiple processors. Hence, Alpha AXP architecture project early focused on multiple processors, and designed a multiprocessor memory model and matching instructions from the very beginning.

• **Run several operating systems.** Underpinnings were placed for interrupt delivery and return, exceptions, context switching, memory management, and error handling, all in a set of privileged software subroutines called PALcodes. By having different sets of PALcode for different operating systems, neither the hardware nor the operating system is burdened with a bad interface match, and the architecture is not biased toward a particular computing style.

• **Easy migration from other architecture customer bases.** To run an existing (old architecture, such as VAX and MIPS) binary version of a complex application, the idea of *binary translation* was adopted [7]. It allows a user to get applications up and running immediately, with minimal porting effort.

[1] *Computer architecture* is defined as the attributes and behavior of a computer as seen by a machine language programmer, while *implementation* is defined as the actual hardware structure, logic design, and data-path organization of a particular embodiment of the architecture. The architecture therefore carefully describes the behavior that a machine language programmer sees, but does not describe the means by which a particular implementation achieves that behavior.

4 Alpha AXP architecture

4.1 Its approach to RISC architecture

Alpha is a 64-bit load/store RISC architecture designed with particular emphasis on clock speed, multiple instruction issue, and multiple processors [1]. Its architects examined and analyzed current and theoretical RISC architecture design elements and developed high-performance alternatives for the Alpha architecture.

4.2 True 64-bit architecture

All registers are 64 bit in length and all operations are performed between 64-bit registers. Hence, it is not a 32-bit architecture that was later expanded to 64 bits. There are 32 integer registers R0..R31 and 32 floating-point registers F0..F31.

The basic unit of data is 64-bit *quadword.* There are three fundamental datatypes: integer (32-bit longword, 64-bit quadword), IEEE floating-point (32-bit S-floating point, 64-bit T-floating point), and VAX floating-point (32-bit F-floating point, 64-bit G-floating point).

Table 1: MIN and MAX Values for the Floating-point Data Formats

Data Format	MIN	MAX
F-floating	0.294e-38	1.70e38
G-floating	0.56e-308	0.899e308
S-floating	1.175e-38	3.40e38
T-floating	2.225e-308	1.798e308

Each of 168 Alpha instructions is 32 bits in length. There are four major instruction formats (PALcode, Branch, Memory, Operate) and all have 6-bit opcode.

• **PALcode instructions.** These instructions specify one of few dozen complex operations from Privileged Architecture Library[2] to be performed.

[2] A Privileged Architecture Library is a set of subroutines that is specific to a particular Alpha operating-system implementation.

• **Branch instructions.** Conditional branch instructions can test a register for positive/negative or for zero/nonzero. They can also test integer registers for even/odd. Unconditional branch instructions can write a return address into a register. There is also a calculated jump instruction that branches to an arbitrary 64-bit address in a register.

• **Memory instructions.** Memory instructions are used for loads, stores, and a few miscellaneous operations.

• **Operate instructions.** There are five groups of register-to-register operate instructions: integer arithmetic, logical, byte-manipulation, floating-point, and miscellaneous.

4.3 Multiple instruction issue

Alpha implementation will issue multiple instructions in a single cycle. To improve the odds of multiple-issue, compilers should choose pairs of instructions to put in aligned quadwords. Pick one from type A and one from type B but only a total of one Load/Store and Branch per pair:

Type A	Type B
Integer Operate	Floating Operate
Floating Load/Store	Integer Load/Store
Floating Branch	Integer Branch
Integer Operate	Floating Operate
	Unconditional Branch
	Branch to Subroutine
	Jump to Subroutine

To avoid any mechanism that would hinder such implementations, all special or hidden processor resources were avoided [6]. Therefore, there are:

• **No branch delay slots.** Branch delay slots require exactly one following instruction to be executed after a conditional branch. This, however, does not scale well to a multiple-way issue chip with a multiple-cycle instruction cache where several instructions will be needed in the delay slot.

• **No suppressed instructions or skips.** When execution of one instruction conditionally suppresses or skips a following one (found in some other RISC architectures) the suppression bits represent a nonreplicated hidden state. Hence, it is difficult to multi-use more than one potential suppressor.

• **No precise arithmetic exceptions.** Reporting an arithmetic exception (such as overflow and underflow) means that instructions subsequent to the one causing the exception must not be executed. This, however, becomes difficult in a pipelined multiple issue implementation. Alpha architecture uses the Trap Barrier instruction which stalls instruction issuing until all prior instructions are guaranteed to complete without incurring arithmetic traps. A code-generation design was documented by Alpha project which needs one trap barrier per branch to give precise reporting.

• **No single-byte writes to memory.** The byte load/store instructions found in some other RISC architectures can be a performance bottleneck because they require an extra byte shifter in the speed-critical load and store paths, and they force a hard choice in fast cache design. Therefore, in the Alpha AXP architecture, a byte load is done as an explicit load/shift sequence; a byte store as an explicit load/modify/store sequence. Instructions in these sequences can be multi-issued with other computation.

Moreover, there are no condition codes, no global exception enables, and no multiplier-quotient or string registers.

4.4 Shared-memory multiprocessing

An Alpha system consists of a collection of processors and shared coherent memories that are accessible by all processors. (There may also be unshared memories.) There are several types of accesses[3] that a processor may generate to shared memory locations (I-stream access, D-stream accesses, and barriers).

Writes to shared data must be synchronized by the programmer.

The basic multiprocessor interlocking primitive

[3]Instruction fetch by processor i to location x, returning value a.
Data read by processor i to location x, returning value a.
Data write by processor i to location x, returning value a.
Memory barrier instruction issued by processor i.
I-stream memory barrier instruction issued by processor i.

Table 2: Competitive Position

Vendor	Digital	MIPS	Sun/TI	IBM	HP	Intel	Motorola
Device	21064	R4000	Viking	RIOS	PA-4	i860XP	88110
Max freq. (internal)	200MHz	100MHz	50MHz	50MHz	66MHz	50MHz	50MHz
No. chips required	1	1	1	7-9	2	1	1
Peak MIPS	400	100	150	200	132	150	150
Peak MFLOPS	200	50	50	100	132	100	100
Base arch. design	64-bit	64-bit	32-bit	32-bit	32-bit	32-bit	32-bit

Table 3: CMOS Technology Roadmap

CMOS	1	2	3	4
Mfg Year	1985	1987	1989	1991
Min features size	2.0μm	1.5μm	1.0μm	0.75μm
Chip power supply	5.0V	5.0V	3.3V	3.3V
Max μP chip size	0.9cm^2	1.2cm^2	1.5cm^2	2.2cm^2
Gate oxide	30nm	22nm	15nm	10nm
Effective L	1.3μm	1.0μm	0.7μm	0.5μm
# of wiring levels	2	2	3	3
# of transistors	200,000	400,000	800,000	1,680,000

is a RISC-style load-locked, in-register modify, store-conditional sequence of instructions.

If the sequence runs without interrupt, exception, an interfacing write from another processor, or a PALcode instruction, then the conditional store succeeds – an atomic update was in fact performed. Otherwise, the store fails and the program eventually must branch back and retry the sequence until it succeeds.

This style of interlocking scales well with very fast caches, and makes Alpha a suitable architecture for building multiprocessor systems. There is no strict multiprocessor read/write ordering, whereby the sequence of reads and writes issued by one processor is delivered to all other processors in exactly the order issued. The strict ordering can be specified when needed by insertion of Memory Barrier instruction. This instruction guarantees that all subsequent loads or stores will not access memory until after all previous loads and stores have accessed memory, as observed by other processors.

5 An implementation: DECchip 21064

DECchip 21064 microprocessor represents the first implementation of the Alpha AXP architecture [4]. It is a super-scalar super-pipelined processor, using dual instruction issue, that has sampled up to 200MHz cycle time. Super-pipelined means that an instruction is issued to the functional unit at every clock tick and the results are pipelined. The integer pipeline is seven stages deep, where each stage is a 5 ns clock cycle. The first four stages are associated with instruction fetching, decoding, and scoreboard checking of operands. Pipeline stages 0 through 3 can be stalled. Beyond 3, however, all pipeline stages advance every cycle. Most ALU operations complete in cycle 4 allowing single-cycle latency, with the shifter being the exception. Primary cache accesses complete in cycle 6, so cache latency is three cycles. The floating-point pipeline is identical and mostly shared with the integer pipeline in stages 0 through 3, however, the execution phase is three cycles longer.

Table 2 shows competitive position of DECchip 21064 microprocessor.

Table 4: Alpha AXP System Comparison Chart

System	3000/400	3000/500	4000	7000	10000
No. of processors	1	1	1 or 2	up to 6	up to 6
CPU	DECchip 21064	DECchip 21064	DECchip 21064	DECchip 21064	DECchip 21064
Clock speed	133 MHz	150 MHz	160 MHz	182 MHz	200 MHz
Max memory capacity	512 MB	1 GB	2 GB	14 GB	14 GB
Max disk capacity	9.5 GB	11.6 GB	56 GB	over 10 TB	over 10 TB
Max I/O throughput	90 MB/s	100 MB/s	160 MB/s	400 MB/s	400 MB/s

The CMOS process is a .75 micron and 1.4- by 1.7-cm chip incorporating 1.68 million transistors. DECchip 21064 include 8KB instruction cache, 8KB data cache and two associated translation buffers, a four-entry 32B/entry write buffer, a pipelined 64B integer execution unit with 32-entry register file, and a pipelined floating-point unit with an additional 32 registers. The bus interface unit handles all communication between the chip and environment. The CMOS technology used to manufacture the DECchip 21064 evolved from three previous generation used to produce very high-performance microprocessors (Table 3).

6 Hardware products

At the time of writing, there are several hardware products available spanning desktop through data center: DEC 3000/400 AXP (Desktop Workstation or System), DEC 3000/500 AXP (Deskside Workstation or System), DEC 4000 AXP (Distributed/Departmental System), DEC 7000 AXP (Data Center System), and DEC 10000 AXP (Mainframe-Class System). Three more products are due in the next few months [3]. See Table 4.

7 Operating systems

Alpha AXP supports today's leading operating system environments: OpenVMS, UNIX, and Windows NT (to be announced soon).

- **OpenVMS AXP.**

VMS, now known as OpenVMS, supports open industry standard interfaces[4], system purchasing flexibility, and licensing. This combination is so new to the industry that the VMS operating system has been renamed to OpenVMS. It runs on both VAX and Alpha AXP hardware platforms. OpenVMS Alpha AXP provides the same features as OpenVMS VAX, enabling users and applications to easily move from one system to another. The benefits of VAXclusters will be made available on OpenVMS Alpha AXP, allowing OpenVMS VAX and OpenVMS Alpha AXP systems to co-exist in the same cluster. The main purpose of moving the OpenVMS system to the Alpha AXP architecture was to deliver the performance advantages of RISC to OpenVMS applications [5].

- **DEC OSF/1 AXP.**

DEC OSF/1 AXP is a true native UNIX. It implements the common definition agreed upon by UNIX Systems Labs (System V) and the Open Software Foundation (OSF/1):
- OSF Application Environment Specification;
- Systems V Interface Definition;
- OSF/Motif User interface;
- Distributed Computing Environment support; and
- Distributed Management Environment support commited.

DEC OSF/1 AXP supports several key standards in the area of the operating and window systems.[5] It adds a range of enhanced program-

[4]The OpenVMS operating environment complies with IEEE POSIX and OSF/Motif standards and is X/Open XPG3 BASE Branded. Future plans also call for OpenVMS compliance with OSF Distributed Computing Environment, and support for XPG4.

[5]IEEE POSIX 1003.1 (1990), 1003.2 (partial), 1003.4a (threads), and 1004.3 D11 (threads); FIPS 160 (ANSI C); X/Open Portability Guide 3; System V Interface Definition 2; System V Release compatibility (all SVID3 Base and Kernel Extensions with the exceptions of streams, signals, and counters); 4.3 BSD; Applications Environment specification (AES); MITs X Window System, X11 Release 5; Motif version 1.1.3; and ISO 9660 (CDROM f.s.).

ming tools available with DEC OSF/1 Developer Extension package. These tools provide a complete software development environment for programmers and application developers. It also provides ULTRIX compatibility, through standards conformance, development tools networking, user interfaces, data interoperability and compilation systems, allowing ULTRIX customers to move to DEC OSF operating system on the Alpha AXP architecture in one step.

To support realtime, DEC OSF/1 offers:
– *A pre-emptive kernel*, to ensure that external realtime events get immediate attention
– *Fixed priority scheduling*, to ensure that realtime applications aren't delayed by background activity;
– *Clocks and timers*, to provide the increased functionality and granularity needed for realtime applications;
– *Process memory locking*, to prevent system paging and swapping that could cause the system to respond unpredictably;
– *Asynchronous I/O*, that enables application between realtime processes;
– *Semaphores*, for fast, reliable communication between realtime processes;
– *Shared memory*, for fast data sharing between processes or applications.

- **Windows NT AXP.**

Alpha AXP systems will run Microsoft's Windows NT. Through Windows NT, users who use DOS and Window-based applications today can continue to use them tomorrow on Alpha AXP-based systems that run many times faster than today's fastest PCs.

For these operating systems, the November 1992 edition of the Alpha AXP Software Application Listing [2] provides a compendium of information on over 1500 software products.

References

[1] Digital Equipment Corporation (1992) *Alpha Architecture Handbook*. EC-H1689-10.

[2] Digital Equipment Corporation (1992) *Alpha AXP Software Application Listing*. First Edition, EC-J2088-10, November 1992.

[3] Digital Equipment Corporation (1993) *Alpha AXP Systems Summary*. EC-F2183-46.

[4] Dobberpuhl D. W. et al. (1992) A 200 MHz 64B Dual Issue CMOS Microprocessor. *IEEE Trans. Solid State Circuits*, **27**, 11, p. 1555-1567.

[5] Kronenberg N. et al. (1993) Porting OpenVMS from VAX to Alpha AXP. *Comm. ACM*, **36**, 2, p. 45-53.

[6] Sites R. L. (1993) Alpha AXP Architecture. *Comm. ACM*, **36**, 2, p. 33-44.

[7] Sites R. L. et al. (1993) Binary Translation. *Comm. ACM*, **36**, 2, p. 69-81.

[8] Supnik R. M. (1993) Digital's Alpha Project. *Comm. ACM*, **36**, 2, p. 30-32.

[9] Thacker C. P., Conroy D. G. & Stewart L. C. (1993) The Alpha Demonstration Unit: A High-Performance Multiprocessor. *Comm. ACM*, **36**, 2, p. 55-67.

The following trademarks are used in this paper: AXP, Alpha AXP, DEC, DECchip, DECnet, Digital, OpenVMS, ULTRIX, VAX, VAXcluster, VMS are trademarks of Digital Equipment Corporation, OSF/1 and OSF/Motif are registered trademarks of the Open Software Foundation, Windows NT is a trademark of Microsoft Corporation, and UNIX is a registered trademark of UNIX System Laboratories, Inc.

OPEN SECURE MODEL AND ITS FUNCTIONALITY IN NETWORKS WITH VALUE-ADDED SERVICES

Borka Jerman-Blažič
Jožef Stefan Institute, Jamova 39, Ljubljana, Slovenia
Phone: +38 61 159 199, Fax: +38 61 158 058

Keywords: security mechanisms, kryptography, networks, value-added services

Edited by: Rudi Murn

Received: December 12, 1992 **Revised:** February 9, 1993 **Accepted:** March 1, 1993

The contribution gives an overview of the security functions employed in the Value Added Networks technology. The overview represents briefly the user requirements for secure communication, the basic threats to which communication systems are exposed and the framework of Open Secure Model defining the security functions and services to be used in an open network. The security mechanisms used for provision of the security services and functions are briefly described. Several applications used in Value Added Networks with inbuilt security functions are briefly introduced at the end.

1 Introduction

Information gains value once it is exchanged or consumed. To be exchanged or consumed it must be transferred or delivered. The need for effective and safety means of carrying this process is today growing faster then the growth of information and electronic data processing. Nowadays, information interchange and data communication is an integral part of any modern information system. Information interchange is a process taking part in the services offered through networks known as Value Added Networks or VANs. These networks usually interconnect communicating users with various information services.

Every VAN is different in its structure and the services it offers. Some of the services are very generalized and some are extremely specialised. A comprehensive VAN is likely to have the following general components [1]:

basic network,

generic services,

transaction relay,

application enabling,

information databases,

network management and help desk.

Generic services are general purpose services needed by a wide range of customers rather being application or industry specific. The main examples are electronic mail, bulk data transfer, Electronic Data Interchange (EDI) and information services support for managers or professionals.

VANs do not have to own a proper wide area network, but they generally do. At its simplest they might just provide point-to-point packet switching. At a more advanced level they may also offer protocol conversion to support a wide range of different equipment. Overall, their aim is to provide full connectivity between all the equipment and systems that their customers need. Connectivity and security are inherently contradictory requirements. However, with specially added security services it is possible to built open, fully connected network with any required level of security.

This contribution gives an overview of the security functions taken as a part of globally interconnected network. The overview deals with the user requirements for secure communication, the basic threats to which communication systems are exposed and the framework of the model which defines the security functions and services in an open network. The security mechanisms employed for the provision of the security services are briefly described. Several applications used in

Value Added Networks with inbuilt security functions are briefly introduced at the end.

2 Open networks and threats

Today, separate networks are integrated into a global connected network known as Global Internet [2] consisting of a number of interconnected networks. Individual computers and work stations are connected to fast Local Area Networks (LAN) spanning office building or parts of them. These LANs are usually connected into fast area backbone networks interconnecting one building, building complex or campus area at a speed comparable to that of LANs (100 mb/s). Metropolitan Area Networks are emerging. They will span entire cities with speeds above 100 Mb/s. Unlike LANs, MANs will be an integral part of the modern public network infrastructure being owned and operated by teleoperators and sharing the addressing and network management schemes of other public network. Wide Area Networks are used for long-haul services. Modern WANs are based on fibber optics transmission systems in the Gb/s speed range. All these networks today are not anymore separate physical networks but rather virtual networks, i.e collections of Network Service Access points (NSAPs) forming one world-wide logical network. One NSAP can simultaneously belong to any number of such logical networks. The protocol suites used in these interconnected logical network are mainly the Internet protocol i.e TCP/IP suite defined on the Request for Comment standards [3] and the OSI protocols developed within OSI Reference Model [4]. Upper layer ISO protocols can be run on the top of TCP/IP and vice versa, enabling the connected networks with different technology to provide global connectivity. Recently, the Connectionless Network Service and Protocol developed within ISO was adopted as an RFC standard and that among the other developed interworking techniques can be considered as step forward towards better coexistence of these technologies and provision of global connectivity.

Connectivity and security are inherently contradictory requirements. However, openness as is understood today does not mean lack of security but it means interconnectivity and the ability to interoperate between systems in different organizations and from different manufacturers. When an open distributed system is built up it becomes essential to define the user requirements regarding the security of communication. The users, depending on which service of the communicating system are they using may require different level of security. Usually, users are concerned with the following:

- the identity of the other communicating party,
- that nobody else can listen to the session,
- that nobody can undetect delete from, change or add to the information they are interchanging with other party,
- that commitments made during the session can beyond reasonable doubt, afterward be provided to an impartial judge.

The user apprehensions come out from the fact that the communication systems and resources connected are usually targets of different threats.

The threats can be oriented towards the communication network itself or towards unauthorized access to local system where the communication network is used only as a medium of access. So, three categories of assets within a globally interconnected network can be identified, the manipulation of which is a serious threat:

- the resources in the network,
- the informations conveyed
- and partner relations.

Local systems are resources accessed through the communication system and they must be protected. The communication system itself is a resource and must be protected too. The users of communication systems expect the communication system components to be present and to function and in that sense, the availability of services and stability of services are also the assets of the communication system and need protection.

Informations are the actual content of communication. Unauthorized access to informations, both by eavesdropping and by damaging, can destroy the value of information. Informations held locally and accessible through communication media also belong to that category.

The relation between communicating partners is another basic asset of communication. Without trust in the authenticity of a communicating partner all communication with is worth nothing. Trusted partner relations are characterised by: the trust in the identity of the partner and the trust in the actions of communications.

All the assets of the communication system are exposed to two fundamentally different kinds of threats i.e intentional or not intentional. In the classical security technology only one type of threats is considered i.e the intentional threats represented by the act of espionage and sabotage. Espionage comprises all passive intentional threats such as to get unauthorized knowledge of confidential or classified information. Sabotage comprises all active intentional threats i.e all kind of unauthorized manipulation of data, access to the resources to the communication system etc.

The other potential accidents which are also regarded as security relevant in the communication networks are accidental threats such as bad maintenance leading to an interruption of the network services. From the point of view of the users it does not make any difference if this is caused by a malicious or by an unable administrator.

The various threats and attacks in an open environment are classified within the framework document of ISO (International Standard Organization) [5]. This document (ISO -7498 part 2) identify five different attacks to the open communicating system i.e :

- masquerade,
- repudiation of action or service or,
- denial of service.
- data interception
- data manipulation

a. Masquerade can happen during the mutual validation of the message transfer agent (MTA is an entity which transfer/exchange messages in the electronic mail service) is by the exchange of the MTA names in plain text. An unknown MTA (for example in testing procedure) may be interconnected with some operational MTA by sending one of the known MTA names. This is a typical masquerade of identity with the intention to steal working resources or information. The masquerade of user identity is possible also by tricky handling of routing oriented addresses [6].

b. Repudiation of action or service: repudiation of origin, submission, or delivery of information is extremely painful if contracts or other business documents are considered. How to trust to an invoice received by an EDI service if no evidence of the sender identity can be provided?

c. Denial of services: denial of services can happen due to accidental interruption caused by local system failures or by nonconformant components in cooperating systems, such as erroneous entries in address routing or name mapping tables. Intentional interruptions are normal for maintenance purposes.

d. Data interception: the breach of confidentiality is the most common attack in the existing networks. It is impossible to guess the number of intentional espionage by system administrator or other unauthorised persons able to read data on their own or on other systems. Data may be intercepted also non intentionally in case of misrouted messages etc.

e. Data manipulation: is any kind of unauthorized modification of data and thus violates their integrity. The managing of electronic mail addresses is also in some sense a violation of integrity, accidentally caused by bad maintenance. This is obviously a case in gatewaying, electronic message get loss or cut of their bodies. This type of vulnerability of the communication system includes also manipulation of a message contents in the originator's local store after non-repudiation of submission and/or manipulation of message contents in the recipient's store after non- repudiation of delivery of the message.

The situation that communication services not being provable and that different security failures can happen in a globally interconnected network was acknowledged on many forums. Some of them spent a lot efforts to develop security functions and to provide security models. ISO has addressed the security issue in several documents defining the Security Services or more properly the Security Functions in an Open Environment. General overview of the Open Secure Architecture is given in the Security Frameworks document [7].

3 Security functions and services

The Security Framework is intended to address the application of security services in an Open System Environment, where the term "Open Systems" is taken to include areas such as Data Bases, Distributed Applications, Office Document Processing and Communication Networks. This framework defines the means of providing protection for systems and objects within the systems and with interactions between systems. The framework address both information and sequence of operations which are used to obtain specific security services. These security services may apply to the communication systems as well as to the information exchanged between systems and to the local resources or data managed systems. The term security in the ISO framework is defined as "a mean of minimizing the vulnerabilities of assets and resources". Security is therefore understood as a system preventing the attacks and protecting the assets from the threats. Threats are therefore, encountered by security services implemented at different layer of communicating networks or within the user interfaces. Security services are implemented by employing security mechanisms. Some mechanisms prevent attacks, other detect attacks, some of the latter provide recovery of an unmanipulated state. They are:

Authentication: Many open systems applications have security requirements which depend upon correctly identifying the principles involved. Such requirements may include the protection of assets and resources against unauthorized access, for which an identity based access control mechanism might be used, and/or for accounting and charging purposes. The process of corroborating an identity is called authentication.

Access control: Many open systems applications have security requirements which demand that resources be only used in a manner consistent with the prevailing security policy. The process of determining whether the use of resources within an open system environment is permitted and subsequently preventing such use is called access control.

Non-repudiation: the non-repudiation services ensures the proper collection and maintenance of information consisting of the origin or delivery of data in order to protect an originator against the false denial of a recipient that the data has been received or to protect a recipient against the false denial by an originator that the data has been sent.

Data Integrity: the maintenance of data value is actually its integrity. Many open system applications have security requirements which depend upon the integrity of information. Such requirements may include the protection of information used in the provision of other security services such as authentication, access control, confidentiality, audit and non-repudiation, that, if an attacker could modify them could reduce or nullify the effectivness of those services.

Data Confidentiality: Many applications have requirements which depend upon the secrecy of information. Such requirements may include the protection of information used in the provision of other security services such as authentication, access controls or integrity, that if known by an attacker, could reduce or nullify the effectiveness of those services. The maintenance of the secrecy of data is called confidentiality.

Audit: A security audit is an independent review and examination of system records and activities. The purpose of a security audit is an independent review and examination of system records and activities. The security audits: tests the adequacy of system controls, confirm compliance with established security policy, recommend any indicated changes in controls, policy and procedures, assists in the analysis of the attacks, and hence recommend damage control procedures. A security audits requires the collection and recording of security related events in a security audit trail. A security audit itself involves the analysis and reporting of the information collected by the security audit trail.

Key management: In communication and information systems there is an ever increasing need for data to be protected against unauthorized disclosure or manipulation using cryptographic mechanisms. The security and reliability of such mechanisms is directly depended on the protection afforded to a security parameter, called the key. The purpose of the key management is to provide procedures for handling cryptographic keying material to be used in symmetric or asymmetric cryptographic mechanisms. Key management includes key generation, key distribution, key installation, key storage and key deletion. A fundamental problem in a key management is to establish keying material whose origin, integrity, and in the case of secret keys, confidentiality can be guaranteed.

The placement of particular security function in the Open Architecture is not exactly defined. Security services or functions may be provided by different layers and by different protocols depending of the user and application requirements. Some applications are more and some are less vulnerable. The protection of particular application depends also of the adopted security policy and on the technology used. It can be said, that no universal model exists and that the placement of particular function is chosen after the features and the requirements of particular application regarding security are identified and by pragmatic considerations.

For example, in connection oriented end to end services the transport connection is dedicated to serve one end to end instance of communication and for that reason any security function can be placed at the transport layer or between the transport and the network layer. There are several protocols that implement several security functions on that level i.e NLSP, SDNS, EESP and SP4 [7]. The placement of the function depends also on particular user requirement such as for example traffic flow confidentiality. This security function can only be reliable implemented at layers 1 through 3.

In the case of connection-less protocols such as IP the label techniques known as IPSO (IP Security Option) is used. Such labels (sensitive, unclassified, top-secret etc) are usually accompanied with encrypted data. If the data are sent to a trusty communication system (the delivery of data is guaranteed to be to a authorized local system) then the label could be satisfactory protection but in the case of untested network i.e a public data network then the packets of data are encrypted.

The placement of Authentication, Integrity and Confidentiality functions in the higher layers or directly in the Application Processes such is electronic mail is straight forward solution which is pragmatic but not optical. Placement of the security functions and mechanisms for each application (i.e for Virtual terminal, for File Transfer, for Directory services etc) separately requires excessive development and duplication of functionality. This approach also contradicts the principle that security should be an integral part of the whole communication system and services provided. However, the practice has shown that this approach is much more used today due to the complexity of the interconnected networks and different requirements for security in different applications.

The security functions and services in the networks are provided by employment of security mechanisms. The security mechanisms are also defined in the Open Framework [5]. They are briefly described in the chapter that follows.

4 Security mechanisms

Mechanisms and algorithms providing different security functions and services are all called security mechanisms. In fact these mechanism form a hierarchy:

Higher level mechanisms, such as security protocols and semantic message contents,

Lower level mechanism, such as cryptosystems, forming parts of the above mentioned higher level mechanisms,

Physical mechanisms, such as encryption chips and pieces of program code, implementing the above mentioned mechanisms.

The application of these mechanisms depend mainly on the security functions required and on the complexity of the system to be protected. The security of a local system can, to a great extent, can be ensured by physical security arrangement. However, in a global open network it is impossible to guarantee security of communications by means of physical safeness and there cryptographic techniques are applied.

4.1 Cryptography

Cryptography is a long-established way to keep information secret but nowadays the cryptographic mechanisms are specially developed and used to protect transfer of data and information. There are many cryptographic mechanisms but as basic ones used in globally connected networks the following are considered: encryption and techniques for providing integrity and authentication of the messages. For more details see [8,9]

An encryption mechanism is used to convert a cleartext message into a cryptogram. An encryption mechanism is based on a public algorithm and at least one key whose value is randomly chosen from larger set. A symmetric cryptosystem has two functions i.e encrypt and decrypt. A message encrypted with key K can be only decrypted with the same key [10].

In asymmetric encryption mechanisms the key is divided into two parts, the encryption key and the decryption key, in such a way that the encryption key specifies the encryption transformation and the decryption key determines its left inverse mapping decryption. The receiver of data holds a secret key with which he can decifer but a different key is used by the sender to encipher and this can be made public without in any way compromising the system. This system provides secure communication in only one direction. It is known as asymmetric or public key encryption mechanism.

If it is unfeasible to derive the encryption key from the decryption key then the system is called public key signature mechanism. In that case additional information is required to check the digital signature. An asymmetric encryption mechanism provides complete confidentiality (only the legitimate recipient in possession of the secret key can decrypt the message) but no authentication of the sender (anybody with access to the recipient's public key could generate the message) but if the technique of digital signature is applied then authentication can be provided too. Unlike a normal signature on a document, the value of the message i.e the whole plaintext of the message is transformed. In order to check the signature, the receiver applies the encipherment function using the public key of the sender. If the result is a plaintext message, the signature is considered to be valid. The argument for its validity is that only by possessing the secret key could anyone produce the transformed message which enciphers with the public key to generate a valid plaintext. There are other ways of forming digital signature in which the signature is not transformation of the message itself but an additional and separate value that goes along with the plaintext. In that case also origin and integrity of the message may be claimed.

Data Integrity mechanisms provides means for the sequence of messages to stay intact. This means that no message have, undetected been omitted or duplicated and that the original ordering of the messages is preserved. Data integrity provides detection of the changes in the data being transferred, optionally recovering from the changes, when possible and reporting the cases where recovery is not possible. Usually the technique of a Checksum [11] or, preferable Cyclic Redundancy Check [12] is used to detect changes in the data stream. Both represent a special field in the message which content guarantee that the data were not changed.

Authentication is today used by use of passwords which are concerned as very vulnerable and that sort of Authentication is known as weak Authentication. Strong Authentication can be based on symmetric or public key cryptography. The procedure of Strong Authentication and key exchange is described in the CCITT Recommendation X.509 [13] or ISO 9594 [14]. With symmetric cryptosystems a mutually agreed pairwise key belonging to the appropriate security context is used for Strong Authentication between any pair of parties A and B. Public key signature mechanisms have several advantages over symmetric cryptosystems when used for authentication. Key management is greatly simplified by the fact that only public keys of the pairs need to be shared and only one key pair is needed for each party.

A rapidly growing area in that field is that of zero-knowledge techniques [15]. In these techniques, the secret authentication information of each party plays very much the same role as the secret key in the public key cryptographic system but it can not be used for data encryption, only for data authentication and possible digital signature. Some existing zero-knowledge techniques make key management very simple by completely abolishing the need for user dependent public key. The draw back of these techniques is that they require the secret keys to be generated by a trusted third party and they can not be used for confidentiality.

In the case of cryptography, the physical mechanism at the bottom of the hierarchy that are needed to actually perform the cryptographic functions employed can be pieces of software running on a piece of hardware or hardware. With the transmission speeds offered by current data networks, the efficiency of the physical mechanisms used is becoming a major issue of system design. The choice between various physical mechanism is trade-off between economy, flexibility and performance. For details see [16].

4.2 Known cryptosystems

The most commonly used today cryptosystems are the Data Encryption Standard (DES) which development was initiated by US National Bureau of Standards but later resulted in many commercial applications [17]. The Rivest-Shamir-Adleman (RSA) algorithm is the most commonly used and probably most usable Public Key Cryptosystem today [18]. The Diffie-Hellman scheme first proposed as the first published "public key algorithm" is still concerned as one of the best methods for secretly sharing pairwise symmetric keys [19]. The algorithm is based on public "half-keys" and secret values associated with them. From their public half-keys the communicating parties can determine a pairwise session, which remains secret from other parties. This key can then be used for mutual authentication and or exchanging secret information.

5 Security policy

An integral part of the Open Security Framework is the Security Policy. A security policy is a set of rules which constrain one or more sets of security relevant activities of one or more sets of elements. Secure policy need not apply to all activities and elements in a communication system. This means that its specification must include a specification of the activities and the elements to which the policy applies. The rules for each security service are derived from the security policy.

Security policies are conventionally divided into Identity-Based and Rule-Based policies; Identity-Based security policies are based on privileges or capabilities given to users and/or Access Control Lists associated with data items and other resources. In a Rule-Based security policy, Security Classes are normally used for determining what is authorized behaviour. In identity-based systems, the users traditionally identifies himself by presenting to the system something he knows (e.g a password). This is often called "need to know" policy.

It is only after an explicit security policy has been stated that security becomes an engineering problem and every organization seriously interested in security should have one. The enforcement of the adopted security policy and monitory of security related events lies in the domain of engineering.

A body responsible for the implementation of a security policy is called Security Authority. Security Authority may be a composite entity but such entity must be always identifiable. A security domain is a set of elements under a given policy administered by a single authority for some specific security relevant activities. An activity involves one or more elements such as: connections between different layers in the protocol suite, operation relating to a specific management function, non-repudiation operations involving a notary etc. The enforcement of the adopted security policy usually goes through generation of security control information. One of them is a Security Label. A security label is a set of security attributes that is bound to an element, communication channel or data item. A security label also indicates, either explicitly or implicitly, the authority responsible for creating the binding and the security policy applicable to the use of the label. A security label can be used to support a combination of security services. Examples of security labels are: indication of sensitivity i.e unclassified, confidential etc, to indicate protection, disposal and other handling requirements.

Another very important Security Control Information (SCI) is the Certificate. A certificate contains SCI relating to one or more security services. Certificates are issued by Certification Authority. It is used to convey SCI from an authority to entities which require this information to perform a security function. In general a certificate may contain SCI for all security functions.

The security mechanism described in the chapter above involve the exchange of SCI either between two communicating parties or between the security authority and the interacting parties. There are two common forms of protected security information that are used by the described mechanisms. One is called security token, used to protect security information that is passed between interacting parties. The other is called a security certificate, used to protect security information obtained from an authority for use by one or more of the interacting parties.

The Security Framework [5] does not define the methods and the procedures for implementing the Security Policy and related SCI. This is left to be developed by particular organization and system. The security models and techniques developed for VANs and globally interconnected networks is relatively new and the number of publicly known implementations is relatively small. The following chapter gives a brief overview of known applications in the field.

6 Applications providing secure functions in VANs

6.1 Kerberos

The most prominent strong authentication service in wide use today is the Kerberos Authentication Server created in the Athena project at MIT [20]. Kerberos is in everyday use in several major U.S universities and obviously has solved a number of security problems in them. In Kerberos, authentication is based on symmetric encryption which precludes the stronger service of non-repudiation and leads to the problems of key management. However, non-repudiation is not considered as a serious threat in university environment. Kerberos works in limited environments and therefore the number of shortcomings such as the possession of the all master keys by only one party i.e Kerberos itself can become unfeasible to be managed one day when the number of users and service grow.

6.2 Private Enhanced Mail

The other forthcoming application within the Internet is PEM (Private Enhanced Mail) [21]. PEM provides security services for e-mail users and is a result of the development efforts by BBN in Cambridge, U.S based on the RFC 1113-1115 which have been developed by IETF (Internet Engineering Task Force) Privacy and Security Research Group [22]. The services provided are the following: confidentiality, data origin authentication and connectionless integrity as defined by ISO [5]. These services are bundled into two groups:

1. default services meaning that all messages processed via PEM incorporate the authenticity, integrity and non-repudiation support facilities and

2. optional services such as confidentiality.

For compatibility purposes PEM is designed to be transparent for X.400 message transfer agent systems. In the recipient's workstation PEM messages may be retrieved also by Post Office Protocol [22] or by IPMS protocol P7 as defined in X.400 environment [23].

PEM message processing involves three steps: SMTP (Small Mail Transfer Protocol) canonicalization needed for compatibility with the MTAs, computation of the message integrity code (MIC) and computation of the optional message encryption code. The second step begins with the calculation of MIC (similarly to DES message authentication code) then encryption follows if required by the originator. Message key, used exclusively to encrypt the particular message, is generated specially for that message. The encryption algorithm employed is specified in the Key-info field in the PEM header along with any parameters required by the algorithm. The message text is then encrypted using this per-message key. The third and final processing step renders encrypted or MIC-only message into a printable form suitable for transmission via SMTP or other messaging systems.

To provide data origin authentication and message integrity, and to support non- repudiation with proof of origin, the MIC computed in step 2 is padded and then encrypted using private component of the originator's public public key pair. This effects a digital signature on the message, which can be verified by any PEM user. If the message is encrypted, this signature value is encrypted using the secret, per message key, which was employed to encrypt the message itself. The resulted value is 6-bit encoded and included in the MIC-Info field along with the identifiers of the MIC algorithm and digital signature algorithm. The MD2 hash function is employed as the MIC algorithm and the RSA algorithm is employed as the digital signature algorithm [22]. A hash function is a well defined function of a message which appears to generate a random number.

The PEM specification recommend use of public key cryptography for message integrity and authentication and for distribution of message encryption keys. PEM uses the public key certificates as defined in the CCITT X.509 Recommendation [13]. On the basis of X.509 definition of certificate handling an Internet Certification Hierarchical (ICA) scheme is envisaged in which different Certification Policy's are employed. ICA is expected to be developed in near future. For details see [21].

6.3 SecuDe System

SecuDE is software package which consists of Security Application Programmers Interfaces providing support for the application of the Authentication Scheme and Certificate Handling, The Privacy Enhanced Mail Support, X.400 Message handling and Key Management. The system provides various cryptographic mechanisms such as DES, RSA, hash functions, key generation and generation and verification of digital signature [24]. The signature algorithm employed is a composition of a hash function followed by an RSA function. The signer's public key which is used for the verification of the signature is certified by Certification Authority. For encryption, the DES algorithm is used and for the transfer itself the secret DES key is RSA encrypted. For every user, the pair of RSA keys used for encryption and decryption is different from the pair of RSA keys used for signature and verification. A special module is provided for support of the functions for the generation and distribution of keys, certificates and certification paths enabling the functionality of the Certification Authority as envisaged in X.509. Additional module is also developed for support of PEM and secure X.400 mail [25].

6.4 EDI and Inter-Bank payment system

Other applications of the Security Framework Services are in the EDI environment and in the Inter-Bank payment systems [26]. Some of them (i.e the system ETEBAC 5 developed in FRANCE [27] use the authentication mechanism as defined in X.509 and the C (Message Authentication Code) computed on plaintext data. MAC is defined in the document ANSI Financial Institution Message Authentication and is a sort of authenticator). The MAC key is exchanged (encrypted under RSA) for each session. The confidentiality is configured by another key drawn by the sender. The non repudiation is based on the RSA algorithm. The digital signature comprises the MAC calculated on the Message Identifier and MAC calculated on the whole message. The secret key of the sender issued for computation of the signature.

The Holland AMRO-ABN bank implementation comprises two modules: a one way hash function which compresses the bulk payment to a code of a fixed length. This module is based on the DES algorithm. The output code of the module 1 (hash function) is encrypted with an RSA digital signature to provide currently authenticity and non-repudiation . Three main functions are essential: user identification, generation of digital signature and verification of digital signature. The necessary key management is based on the generation of the keys by every user and the certification of the keys by a Certification Authority after checking both the integrity and authenticity of the keys. Key generation is planned to be based on CCITT X.509.The response messages are used to provide non-repudiation of receipt.

Teletrust (Trustworthy Telematic Transactions) implements the public key mechanism and reliable hashing functions. The basic Teletrust device is a token.It is a credit card size chip that can not be tampered. The token is protected by a PIN (Personal Identification Number, a sequence of digits used for identification of the holder of a bank card) and is used as payment device. The token is activated by the user and the mutual authentication with the service provider is performed by exchange of the public keys with RSA. Once completed, a payment transaction may take place and the token is used to "sign" the payment (digital signature).The authentication requires a Certification Authority (in this case the central bank). The token stores therefore the public keys and the signatures of the central bank used for the authentication of the users's bank, which is used for authentication of the user and the user, which is used for the authentication of the transaction.The digital signature is used for authentication, integrity and non-repudiation service.

7 Conclusion

Security is a central consideration in the evolution of Value Added Networks. Security services and functions are needed to protect the infrastructure of the communication system, the local resources as well as to provide enough assurance to the prospective users by guaranteeing safe transport of sensitive and high value information. Fortunately, today the fast progress of technical developments is rapidly improving the security of the networks providing in the same time openness, connectivity and safety.

References

[1] R.Reardon (ed.) *Future Networks*, Blenheim Online, London 1989.

[2] *Internet: Getting started*, SRI International, Menlo Park, CA, 1992.

[3] J.B. Postel, (ed.) *IAB official protocol standards*, 1992 J.B. Postel, Internet Protocol, RFC 791, 1981.

[4] ISO, Information Processing Systems, *Open System Interconnection Reference Model, Part:1 Basic Reference Model*, ISO 7498-1, Geneva 1984.

[5] ISO, Information Processing Systems, *Open System Interconnection Reference Model, Part:1 Security Architecture*, ISO 7498-2, Geneva 1988.

[6] R.Grimm, *Security on Networks: Do WE Really Need it?*, Comp.Networks and ISDN Systems, Vol.17, No 4&5, October 1989, p.315–321.

[7] A.T.Karila, *Open System Security - an Architectural Framework*, Espoo 1991, Helsinki.

[8] D.W.Davies and W.L.Price, *Security for Computer Networks*, Sc.ed.,J.Willey and Sons, Chichester, 1989.

[9] S.Muftic, (ed.) *Security Mechanisms for Computer Networks*, Ellis Horwood Ltd, Chichester, 1989.

[10] C.Shannon, *Communication Theory of Secrecy Systems*, Bell System Technical Journal, Vol.28, 1949, p.656–715.

[11] ISO, Information Technology, *Security Techniques, A Data Integrity Mechanism*, ISO DP 9797, Geneva 1990

[12] S.Walker, *Network security: The parts of the Sum*, Proceedings of the 1989 IEEE Computer Society Symposium on Security and Privacy, Oakland 1989, p.2–9.

[13] *The Directory - Overview of Concepts, Models and Services*, CCITT Recommendation X.500, Melbourne 1988, and The Directory, Part 8: Authentication Framework, CCITT Recommendation X.509 (Melbourne 1989,).

[14] ISO 9594, Information Processing Systems, *OSI - The Directory*, 9594 through parts 1 - 8, Geneva 1989.

[15] ISO 10181, Information Technology, *OSI Security Model*, Part 1 Security Framework, Part 2, Authentication framework A.Shamir, Identity-Based Cryptosystem and Signature Scheme, Advances in Cryptology: Proceedings of Crypto 84, Springer, Berlin, 1985, pp.47–53.

[16] J.Smith, *Cryptographic Support in a Gigabit Network*, INET 92, Proceedings, Internet Society Reston VA, Kobe, 1992, p.229–239.

[17] R.M.Davis, *The Data Encryption Standard in perspective*, Computer Security and the Data Encryption Standard, National Bureau

of Standards Special Publication, 500-527, Washington, 1978.

[18] R.Rivest, A.Shamir, L.Adleman, *A Method for Obtaining Digital Signatures and Public Key Cryptosystems*, Communication of the ACM, Vol.21, No.2, 1978, p.120–126.

[19] W.Diffie,M.Hellman, *New Directions in Cryptography*, IEEE, Transaction on Information Theory, Vol IT-22, No.6, 1976, p.644–654.

[20] D.Otway, O.Rees, *Efficient and Timely Mutual Authentication*, ACM Operating System Review, Vol 21, No.1, 1987, p.8–10.

[21] S.Kent, *An Overview of Internet Privacy Enhanced Mail*, INET 92, Proceedings, Internet Society Reston VA, Kobe, 1992, p.217–227.

[22] J.Linn, *Privacy Enhancement for Internet Electronic Mail*, Part III, Algorithms, SRI NIC International, RFC 1115, Auvgust 1989.

[23] M.Rose, *Post Office Protocol: Version 3*, SRI NIC International, RFC 1225, May 1991.

[24] R.Grimm, R.Nauster, W.Scheider, *SecuDE, Principles of Security Operations*, Technical Report, Version 2.0, GMD, Darmstadt, Germany, 1991.

[25] *Message Handling Systems*, Recommendation X.400, CCITT, Blue Book Vol VII, Fascicle VII.7, Melbourne 1988.

[26] B.Jerman-Blažič, *Security in Value Added Networks – security requirements for EDI*, Comp.Stands, North Holland, Vol.12, 1991, p.23–33.

[27] *The TEDIS Programme 1988-1989*, Activity Report, and TEDIS-EDI Security Workshop, Security in a Multi-Owner System, Brussels, June 20-21, 1989, Digital Signature in EDIFACT a TEDIS programme report prepared by CRYPTOMATHIC A/S, final version, Nov.29, 1990.

REAL AI

Matjaž Gams
Jožef Stefan Institute, Jamova 39, Ljubljana, Slovenia
Phone: +38 61 159 199, Fax: +38 61 158 058
E-mail: matjaz.gams@ijs.si

Keywords: artificial intelligence, viewpoint

Edited by: Branko Souček

Received: December 12, 1992 **Revised:** February 9, 1993 **Accepted:** March 1, 1993

In the first part, four viewpoints on AI are presented. It is proposed that a program exhibiting AI is one that can change as a result of interactions with the environment. While no program can be proclaimed as intelligent at the moment, intelligence may be just an emerging property of successful information machines or beings. In the second part, ideas, problems and misconceptions about AI are analysed through grouping into three categories: (1) facts - opinions that are supported by facts; (2) legends - opinions, based on facts but largely exaggerated; (3) myths - opinions not based on facts.

1 Introduction

Discussions about AI have attracted most of the researchers inside the field as well as many outside (Fox 1990, Hayes-Roth & Fikes 1991, Hayes et. al. 1992, Mettrey 1992, Schank 1991, Searle 1982, Wilkes 1992). In particular, there have been important shifts and modifications in world-wide opinion observed in recent relevant publications. These debates have motivated us to make an attempt to summarise them in a coherent way. [1]

First, let us analyse the notion of artificial intelligence.

2 Viewpoints on AI

Thinking about the definition of AI, one should ask 'Where's the AI?' (Schank 1991). There seem to be at least four prevailing answers to that question.

The **first view** sees AI as something magic that emerges out of a computationally effective computer after you put in it enough things. Indeed, it is still often acclaimed that neural networks mimic the behaviour of human brains. That is rather surprising since what they do at present is to mimic a numeric filter at best able to tune predefined parameters. Not surprisingly, computationally more diverse and structurarily more complex statistical methods typically achieve better classification accuracy (Henery & Taylor 1992).

This approach has already contributed to the first dark age at the beginning of AI and is still present in many subfields of AI. On the other hand, one should not underestimate their advantages such as flexibility and robustness.

The **second view** sees AI as a superb inference engine and therefore resembles knowledge engineering. What one has to do is to find an expert, encode his knowledge into lists of rules, add an inference engine with appropriate interface and there you are. This has led many people to think that AI means rule-based expert systems, and then they thought they understand them as well as AI. And since they have also learned the limitations of rule-based systems, they also think that is the limitation of AI, not just of one component of AI (Hayes-Roth & Fikes 1991). While connectionism as well as knowledge engineering and inference engines are important parts of AI research and applications, labelling it intelligent or as AI itself is misleading (Schank 1991).

The **third view** maintains that, if no machine ever did it before, it must be AI. For example, years ago research in computer chess was one of

[1] A similar but shorter paper (Gams 1992) has been presented as the first paper in the AI section at the ERK'92 conference - similar parts are reprinted with permission.

the central themata in AI. People thought that computer programs playing chess well would certainly have to be intelligent. Now, nearly everybody agrees that these programs do not have any deep intelligence at all. Luckily, there are several explanations of this contradiction. For one thing, this viewpoint seems to confuse getting a machine to do something intelligent with getting it to be a model of human intelligence. More important, if you define AI in that way - to be one of frontiers of computer science, once the area that you are looking at is understood, then it is no longer at the frontiers of computer science and therefore no longer AI, and so it is a no-win kind of situation (Schank 1991).

So, where is the intelligence in computer programs? As mentioned earlier, many difficult problems which had long been thought to require real intelligence, have been solved by rather unintelligent methods. Intensifying this argument, even superb intelligent behaviour does not guarantee that real intelligence and understanding have been achieved. For example, Searle (Searle 1982) has constructed a hypothetic Chinese room in which a group of workers performs intelligent translation between two natural languages (English-Chinese) and each of them performs only a subpart of the whole process on the basis of a predefined procedure. Although such a Chinese room could pass Turing's test, the room (and nobody inside it) does not understand the whole process and there is no real intelligence at all.

After a decade of quite intensive debate, there has not been any definitive answer to this paradox since it is actually a philosophical question: Is performance, i.e. a mechanicistic approach really sufficient? From a practical point of view, most things in our world work in the mechanicistic mode. Of course, there are paradoxes and unsolved questions (e.g. are there other universes or is there only ours?) but people have successfully lived with them. Not to mention that other approaches based on ideology or spiritism have not yielded similarly good results.

Therefore, it seems rather surprising that criticism is so strong in the AI area even if the same arguments are being repeated over and over again. A recent example of this kind could be the article "Artificial Intelligence as the Year 2000 Approaches" (Wilkes 1992). It provoked harsh replies (Hayes et. al. 1992) in which several errors and misconceptions were exposed. Nevertheless, in all this argumentation there are at least two points where the AI community still has to prove itself:

- if intelligence (in computers) were simple, fast and powerful computers would have facilitated it a long time ago, and
- many of the ideas in the AI field have produced much more optimism than real improvements.

Here we shall devote attention to the first argument, and the second argument will be analysed in the second part of this paper.

For example, Wilkes (Wilkes 1992) claims that intelligence may come from analogue circuitry since, obviously, it has not come from digital computers so far. Searle (Searle 1982) claims that digital machines can not be intelligent as biological beings since they are essentially different. Although Hayes (Hayes et. al. 1992) claims that no proof is given for such claims, the same is valid also for the reverse claim.

At this point we can only agree that real intelligence in machines has not been achieved yet. Furthermore, we still do not have any good ideas how to make a true intelligent machine. However, two arguments seem plausible:

- Real human intelligence is very complex. If it were simple enough for us to understand it, than we would be too simple to perceive that (as claimed by several authors).
- Intelligence may be just an emerging property of successful information machines or beings. There does not have to be any deeper motive or principle behind it. This approach is very close to the "artificial life" where computational models share many characteristics with biological computation (Brooks 1991).

Furthermore, in computing there are good foundations and clear concepts like Turing's machine or Church's thesis. There is also Turing's test in which real intelligence is achieved when human judges can not distinguish between the performance of a computer and human. Since computer programs are far away from achieving such a level, the contest area is often limited to a domain which still requires intelligence by human counterpart. However, it is important to notice that the communication between the judge and the contenders is an open one. Therefore, a computer program playing superb chess but unable to explain the motives of its moves certainly would not pass the test while even a novice player with normal explanation and reasoning capabilities certainly would. In recent years slightly modified tests, or competitions, are becoming annual events with rewards up to $100,000 (Epstein 1992).

This leads us to the **fourth view** on AI. True intelligence, exhibited by computer programs, would have to have many or even most properties of human intelligent behaviour regardless of how narrow the application area was. One of the main such properties is *learning* since intelligence first of all means getting better over time. In relation to Turing's test, a computer program unable to learn from its mistake would certainly be exposed. Today, hardly any AI programs learn from their mistakes, although - with very good reason, learning is the central area of AI at least in the last decade.

There is some additional reasoning about intelligence:

- Intelligence is in size. It is hard to expect a small program to display intelligence. Intelligence is neither simple nor easy understandable.

- Intelligence is in complexity and heterogeneity. This area is sometimes related to multistrategy learning (Michalski & Tecuci 1992), a multiple-knowledge approach (Gams et. al. 1991) and multi-agents (Minsky 1987).

- Intelligence is in the ability to perform well real-life tasks which require the use of knowledge. For example, Mathematica, a program for symbolic computing is regarded as approximately as intelligent as a numeric library. Contrary to recent interest in logic programming, it is quite probable that intelligence there will be at a similar level to Mathematica until real-life knowledge is incorporated into programs.

Furthermore, there are several aspects of intelligence each of which can be compared if not measured on a scale. For example, motional intelligence can be quite high in many animals. In another aspect, AI research can well be at the frontiers of computer science while AI applications fell into an application area years away from scientific achievements. AI applications do not have to be intelligent, they have to be related to AI research similar to other science/application relations.

In short, while agitated debates about AI raise interest and in both ways affect funds, what really matters is what works and which new discoveries are produced. *It is not that AI needs definitions; it is more that AI needs substance* (Schank 1991). Although general artificial intelligence has not yet been achieved, we know more and more about it. Some basic facts, legends and myths about AI will be represented in the following sections.

3 Facts

The first AI concept is search. Most difficult problems involve choosing between alternative solutions and evaluation processes in which the best solution is found. This basic search schema may not be immediately observed in diverse subareas of AI such as scheduling, games, learning or expert systems. Novice readers in AI might get distressed by different terminology and diverse techniques. However, even one of the oldest definitions of AI promotes it as a fight against combinatorial explosion.

While faster computers certainly help, simple search techniques can not ever deal with the exponential growth rate of the number of possibilities in a search tree. For example, in a single factory having 85 orders, 10 operations, and only one substitutional machine, one could create over 10^{880} alternative schedules (Fox 1990) while the number of all atomic particles in our universe is estimated at 10^{80}. Obviously, the key question is how to reduce search space.

The second AI concept is knowledge representation. It is not that the knowledge representation concept is second to search; it is one of the two. Knowledge is probably even more important than search in biological systems. In real life, response to a specific pattern is usually prestored - learned through experience. This resembles the fourth viewpoint on AI presented earlier. But from a practical point of view, computers as well as AI were more successful in search than in knowledge representation.

Although there are many techniques from semantic nets to frames, the most successful AI applications so far are expert systems. At times, it was thought that the expert-systems approach enables a uniform solution to knowledge representation problems. It has led to overenthusiasm and overselling the technological possibilities. Now we know that expert systems are appropriate only when problems are relatively small and stable or can be decomposed into such subproblems, meaning that experts agree with each other upon a proposed knowledge base.

The main problem, how to represent different kinds of knowledge, complex and heterogeneous knowledge, and combine them into one system has not been solved yet. As a consequence, successful learning from interactions with the environment has not been, and quite probably can not be achieved without it.

AI copes with the search combinatoric explosion by using knowledge. The use of knowledge enables successful pruning of a search tree. For example, in an expert system OPEX (Gams et. al. 1991) for generating appropriate machining operation sequences, designed in cooperation with researchers from the Faculty of Mechanical Engineering and Jožef Stefan Institute, there are three levels of rules:
(a) Rules for applying basic machining operation. Example:
operation drilling : if
gdb:fc is-a-cylinder-in and
gdb:dc included interval(3,40) and
gdb:lc/max(gdb:dc) = 10 and
gdb:nc subset interval(11,12)
then
fc := is-a blank and dc := 0 and nc := undefined
(b) Rules that define various possibilities of linking basic machining operations within an individual feature. Example:
from boring to drilling if true end.
(c) Rules for combining operation sequences that define which operation sequences should be adopted for a combination of features. Example:
combination drilling and drilling if true end.

The task of OPEX is to design operation sequences for a machine and a specified part, and sort them according to predefined criteria. Naive combining of operations quickly leads to combinatorial explosion, but through smarter selection of possibilities, i.e. by utilising domain knowledge, the combinatorics is reduced to a feasible level.

As indicated by previous example, AI enhances search by reformulating problems, through the use of opportunism, heuristics, and by abstraction and differentiation of quantitative models. These are techniques behind the general principle of using knowledge to control search. In essence they perform similar improvements of search as hierarchical search or dynamic programming, however, the use of knowledge can greatly improve performance.

AI systems can increase productivity. Various reports estimate the number of AI systems regularly in use to around 3000 with some of them being in use for more than 10 years. The main problem with such estimates is where to put borders between AI and non-AI applications. For example, is Prolog interpreter an AI application or not? Clearly, there is no intelligence in it. On the other hand, Prolog as well as Lisp and many other products were designed as a by-product in AI research. In our opinion, they should be included as AI applications as well as neural networks. Actually, marketplace AI-software packages fall into at least four major categories: programming languages, programming environments, problem-solving shells (for a class of problems), and application shells (specialised for a given domain).

For example, in our rather small country of Slovenia, in two AI laboratories at Jožef Stefan Institute and the Faculty for Electrical Engineering and Computing, around 60 applications were successfully performed in recent years and 5 original programming systems with several thousands of lines are in regular use (Urbančič & Křižman 1991).

4 Legends

AI systems are easy to build. Indeed, under specific conditions, improvements in speed and productivity are enormous when using AI systems. For example, having stored a history of events, it is possible to design an expert system with the use of inductive learning tools in a couple of days. On the other hand, there are problems which take more or even much more time than by classical methods.

Specifications and prototyping largely enhance productivity. This is partially true. If the problem fits an application shell, knowledge gathered from experts can be put into a system quickly and then tested. Rapid prototyping elicits the requirements and specifications of software for ill-defined problems; in recent years it has been included in software development approaches as another example of AI product finishing in classical computer science and applications. But the limitations of the methodology and conditions for successfulness have also become known.

AI systems are easily verified and maintained. Since AI systems rely on knowledge instead of formulae, e.g. expert knowledge in expert systems, it is often propagated that these systems are highly understandable and, therefore, easy to be verified and maintained. For example, expert systems provide explanation possibilities as a sort of rule tracker instead of 'trace' in conventional programming languages. Practical experience has shown that while it is an important improvement over classical methods, verification and maintenance remain time consuming phases.

5 Myths

Artificial intelligence approach does not need conventional program-engineering and management techniques. This incorrect belief is still quite common due in part to academic ignorance of the requirements for building production-level systems.

Systems working on simple examples can easily be upgraded to full-scale real-life systems. Performing speech understanding for a small vocabulary of, say 50 words, differs greatly from the same task but with thousands of words. Similarly, many problems are difficult only because of their size. The myth of simple scaling is still very alive mainly due to an academic approach where it is most important that idea is fresh and attractive (working on a simple, carefully designed problem). Literature reviews in AI show that about half of all publications belong to this category and only half of the systems actually work on non-toy problems. In the worst scenario, some subareas of AI have for years attracted interest and funds without actually producing a program working on a realistic problem. There seem to be certain similarities to fashion movements in which a new direction promoted by famous people attracts global interest. After a critical mass is obtained, the movement can sustain for several years without any realistic verification. The problem is similar in several other sciences. The "publish or perish" science tends to produce famous writers instead of famous scientists, researchers, engineers or inventors. However slowly, in AI it is changing in favour of more strict verification of results. For example, there are several projects which for years have evaluated different methods (Henery & Taylor 1992). Even at our laboratories we have been testing all available inductive learning systems for 5 years and making the results public.

Small systems can exhibit full-scale human intelligence. In serious AI circles it is known that it is not possible to simulate full-scale human intelligence without huge and complex systems and that searching for a genuine simple algorithm is similar to searching for perpetuum mobile.

If we have an expert, then we can create an expert system. Obviously, a lot more is needed; first of all a feasibility study.

AI does not need business motivation to produce valuable results. Several studies have shown that those initiated by management have a better chance of returning profit.

AI tools can enable novices to develop expert systems. Inexperience and lack of skill can not be compensated in any field.

Expert systems consist of expert systems. Typically, in expert systems there is much more than that, including lots of classical programming.

Expert systems perform as specialised, stand-alone programs. Actually, they access databases, conventional programming languages, operating systems etc.

All AI tools are the same. There are different categories.

All expert systems are rule-based. Many, but there is much more.

Expert systems do not make mistakes. In real life there is no such thing.

AI replaces conventional approaches. Rather, they can both be useful depending on conditions, and are often combined together.

AI knowledge engineering is all we need to know about AI. The more you know the better. Again, AI consists of many diverse subareas.

AI tools are good only for AI applications. AI software supports qualitative and quantitative reasoning equally well.

In simple expert systems an exhaustive search can provide solutions. Yes, for toy problems.

Tools equally support both forward and backward chaining. At the expense of the other.

The more general the tool the better. Task specific tools are actually more productive but on a more narrow area.

There exist universal algorithms for specific subareas such as learning. In theory, not working in practise.

Several subareas of AI have good theoretical foundations. No true intelligence has it so far.

6 Conclusion

AI systems can work well under favourable conditions, and are neither panaceas nor research curiosities. AI is not (just) art or a fashion, it is first of all a scientific discipline. At present, AI can importantly improve productivity and enhance the application areas of computers. As all other technologies, it must be used with a certain precaution and especially when circumstances are favourable. Therefore, more knowledge about AI in general as well as knowing about common legends and myths about AI may improve the success rate and extend the number of AI applications.

References

[1] Brooks R. A. (1991) Intelligence Without Reason. *Proceedings of the IJCAI'91 Conference*, Sydney, Australia, p. 569-595.

[2] Epstein R. (1992) Can Machines Think; The Quest for the Thinking Computer. AI Magazine, 13, 2, p. 80-95.

[3] Fox M. S. (1990) AI and Expert System Myths, Legends, and Facts. *IEEE Expert*, February 1990, p. 8-19.

[4] Gams M. (1992) Facts, Legends and Myths about AI. *Proceedings of the ERK'92*, Portorož, Slovenia, p. 139-142.

[5] Gams M., Drobnič M. & Petkovšek M. (1991) Learning from examples – a uniform view. *Int. Journal for Man-machine Studies*, 34, p. 49-89.

[6] Hayes P. J., Novak G. S. & Lehnert W. G. (1992) ACM Forum - In Defence of Artificial Intelligence. it Communications of the ACM, 35, 12, p. 13-14.

[7] Hayes-Roth F. & Fikes R. (1991) Interview. *IEEE Expert*, 6, 5, p. 3-14.

[8] Henery R. & Taylor C. (1992) Applications of Machine Learning, Statistics and Neural Networks in Prediction and Classification. *Proceedings of the ISSEK Workshop'92*, Bled, Slovenia, p. 22.

[9] Mettrey W. (1992) Expert Systems and Tools: Myths and Realities. *IEEE Expert*, 7, 1, p. 4-12.

[10] Michalski R. S. & Tecuci G. (1991) *Proceedings of the First International Workshop on Multistrategy Learning.* Harpers Ferry, USA.

[11] Minsky M. (1987) *The Society of Mind.* New York: Simon and Schuster.

[12] Schank R. C. (1991) Where's the AI?. *AI magazine*, 12,4, p. 38-49.

[13] Searle J. R. (1982) The Chinese Room Revisited. *Behavioral and Brain Sciences*, 8, p. 345-348.

[14] Sluga A., Butala P., Lavrač N. & Gams M. (1988) An attempt to implement expert system techniques in CAPP. *Robotics & Computer-Integrated Manufacturing*, 4, 1/2, p. 77-82.

[15] Urbančič T. & Križman V. (1991) *A Handbook of AI Applications.* IJS Press, Slovenia.

[16] Wilkes M. W. (1992) Artificial Intelligence as the Year 2000 Approaches. *Communications of the ACM*, 35, 8, p. 17-20.

REGULAR GRAPHS ARE 'DIFFICULT' FOR COLOURING

Janez Žerovnik
University of Maribor, Faculty of Technical Sciences,
Smetanova 17, Maribor,
Slovenia

Keywords: graph colouring, decision problem, NP-completeness, regular graphs

Edited by: Rudi Murn

Received: December 19, 1992 **Revised:** February 16, 1993 **Accepted:** March 1, 1993

Abstract: *Let k be 3 or 4. In this two cases we prove that the decision problem of k-colourability when restricted to Δ-regular graphs is NP-complete for any $\Delta \geq k+1$.*

1 Introduction

In this note we consider the time complexity of the decision problem of (vertex) k-colourability restricted to regular graphs.

It is known that 'almost all k-colourable graphs are easy to colour', namely the proportion of 'difficult' graphs for the usual backtrack algorithm vanishes with growing problem size [9]. Knowing this it is not surprising that there are algorithms with average polynomial time complexity [1], when average is taken over all graphs and even when the average is taken over all 3-colourable graphs with a given number of vertices[5].

If P$\neq$NP, then for every algorithm there has to be a class of 'counterexamples', i.e. graphs on which the algorithm either has superpolynomial time complexity or it fails to produce a correct answer.

For example, Petford and Welsh noticed that one of the situations in which the 3-colourable graphs were not efficiently coloured by their randomised algorithm is when graphs are approximately regular of a low vertex degree [8]. Similarly, approximately regular graphs of a relative low vertex degree are 'difficult' also for the k-colouring generalisation of their algorithm [10]. Petford and Welsh conjectured that 'dense' graphs are easy. Indeed, Edwards showed that, when restricted to class of graphs with lowest vertex degree $\delta > \alpha n$ for arbitrary $\alpha > 0$, the decision problem of 3-colouring is polynomial [4].

This may be understood that the 'difficult' graphs are likely to be found among 'sparse' graphs. It is known that the problem of 3-colouring is NP-complete (even) when restricted to graphs of maximal vertex degree 4 [6]. Here we show that the problem can be further 'simplified', proving that the decision problem of 3-colouring is NP-complete when restricted to Δ-regular graphs (for $\Delta \geq 4$). We also show that the decision problem of 4-colouring is NP-complete when restricted to Δ-regular graphs (for $\Delta \geq 5$).

We assume that the reader is familiar with some standard definitions of graph theory and of computational complexity theory (given, for example, in [2] and [7]).

2 3-colourability of 4-regular graphs is NP-complete

Let us define the problem $\Pi(k, \Delta)$ as follows:
Input: Δ-regular graph G
Question: Is G k-colourable?

Lemma 1 *For any graph G there is a graph G' with no vertex of degree 1 or 2 such that:*
G is 3-colourable iff G' is 3-colourable

Remark: G' in the Lemma is either a graph with minimal vertex degree ≥ 3 or the empty graph, which is the case when G is, for example, a cycle. If G' is empty, it is trivially 3-colourable, and from the proof of the Lemma 1 it follows that also G is 3-colourable.

Proof (of Lemma 1): It is easy to see that the following assertions are true:

(a) If there is a vertex $v \in V(G)$ with degree 1, then G is 3-colourable if and only if the

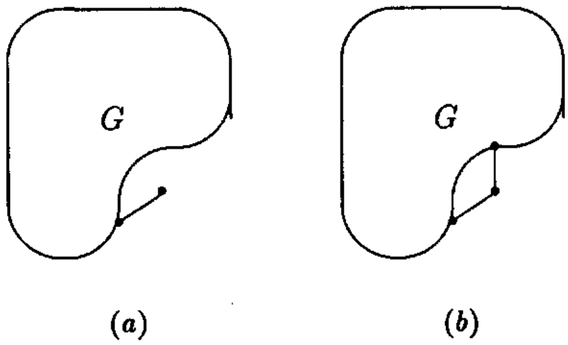

Figure 1: Vertices of degree 1 or 2 may be omitted

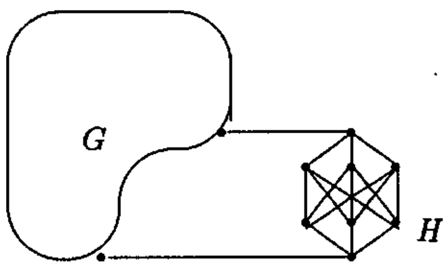

Figure 2: G is 3-colourable iff G' is 3-colourable

induced graph on $V \setminus \{v\}$ is 3-colourable (see Fig. 1(a)).

(b) If there is a vertex $v \in V(G)$ with degree 2, then G is 3-colourable if and only if the induced graph on $V \setminus \{v\}$ is 3-colourable (see Fig. 1(b)).

In this way we can reduce any graph G to a graph G' with minimal vertex degree at least 3 which is 3-colourable exactly when G is. **Q.E.D.**

Remark: Extracting G' successively using (a) and (b) can clearly be done efficiently.

Remark: It is obvious that maximal vertex degree of the graph G' is not greater than maximal vertex degree of the original graph G.

Lemma 2 *G is 3-colourable iff G' is 3-colourable*

where G' is a graph, obtained from G by the construction given in Fig. 2.

The construction, given in Fig. 2, can be done as follows. Take two sets, say M and N, of three vertices each. Connect every pair x,y; $x \in M$ and $y \in N$. Add two vertices, say u and v and connect u to all the vertices of N and v to all the vertices of M. Now choose arbitrary pair of distinct vertices of G, say w and z, and connect u with w and v with z to get the graph G'.

Proof: Since the graph H is bipartite, it is easy to see that 3-colouring of arbitrary graph (G on Fig. 2) can be extended to 3-colouring of graph G'. On the other hand, since G is subgraph of G', G is 3-colourable if G' is. **Q.E.D.**

Now we shall prove

Lemma 3 *The problem* $\Pi(3,4)$ *is NP-complete.*

Proof: We will reduce the problem of 3-colourability of graphs with vertex degree at most 4 (which is known to be NP-complete [6]) to the problem $\Pi(3,4)$.

Let G be arbitrary graph with maximal degree $\Delta \leq 4$. By Lemma 1 there is a graph G_1 (which has at most as many vertices as G) and G_1 is 3-colourable exactly when G is 3-colourable. If G_1 is empty, then we know that G is 3-colourable.

Now consider the case when G_1 is nonempty. By construction, G_1 is a graph with vertex degrees 3 and 4. Since the sum of all the vertex degrees is twice the number of edges ($\sum_{v \in V} \delta_v = 2|E|$), the number of vertices with degree 3 must be even.

Now couple vertices of degree 3 in G_1 arbitrarily. Connect a copy of the graph H to each couple of vertices of degree 3, as defined in Fig. 2. By Lemma 2, this construction gives a graph G_2 which is 3-colourable exactly when G_1 is 3-colourable. **Q.E.D.**

Remark: Graph H has 8 vertices. Since we added at most $\frac{|V|}{2}8$ new vertices, the resulting graph G_2 has at most a constant factor more vertices than G_1.

Remark: The construction can clearly be done efficiently.

Thus 3-colourability of 4-regular graphs is NP-complete. Now we reduce the problem of 3-colourability of Δ-regular graphs to the same problem on $\Delta + 1$-regular graphs.

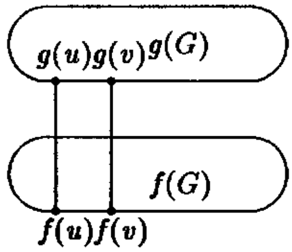

Figure 3: Joining two copies of a Δ-regular graph we get a $\Delta+1$-regular graph

3 3-colourability of Δ-regular graphs is NP-complete

Lemma 4 $\Pi(3,\Delta) \propto \Pi(3,\Delta+1)$

Proof: Let G be arbitrary Δ-regular graph. Now we give a construction of a graph G'.

Take two copies of G, $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Denote with $f : G \to G_1$ and $g : G \to G_2$ the corresponding isomorphisms.

Graph $G' = (V', E')$ is defined with: $V' = V_1 \cup V_2$ and $E' = E_1 \cup E_2 \cup \{\{f(v), g(v)\} \mid v \in V\}$ (see Fig. 3).

If G' is 3-colourable, then also G is 3-colourable, since it is isomorphic to a subgraph in G'.

On the other hand, if we have a 3-colouring b of G, it is easy to construct a 3-colouring b' of G', for example with a 'shift' of colours: $b'(f(v)) = b(v)$ and $b'(g(v)) = (b(v) \bmod 3) + 1$.

Since for any Δ size blow up is a constant factor, the assertion of the Lemma follows. **Q.E.D.**

By induction, from Lemma 3 and Lemma 4 we have:

Proposition 1 *The decision problem of 3-colouring restricted to Δ-regular graphs $\Pi(3,\Delta)$ is NP-complete for $\Delta \geq 4$.*

It is known that for graphs of maximal vertex degree 3 the problem is polynomial [6]. Hence we know for all problems $\Pi(3,\Delta)$ whether they are polynomial or NP-complete.

4 4-colourability of Regular Graphs

Here we discuss an attempt to generalise the proposition 1 on the problem of k-colouring. With analogous proof as for the case of 3-colourings we prove a proposition for 4-colouring, while for $k > 4$ the time complexity of the decision problem of k-colouring of Δ-regular graphs remains open for some Δ.

Two of the previous lemmas are easily generalised:

Lemma 5 *Let G' be any subgraph of G obtained by the following process: if there is a vertex of degree less than k, delete it. Graph G is k-colourable if and only if graph G' is k-colourable.*

Proof: Assume we coloured the graph G' with k-colours. It is easy to see that there is algorithm, which properly extends the proper colouring of G' to a proper colouring of G. (Take, for example, vertices of G in opposite order as they were deleted from G. When a vertex was deleted, it had less than k neighbours, therefore there is at least one free colour for it.) **Q.E.D.**

Lemma 6 *For any graph G with vertex degrees k and $k+1$ there is a $k+1$-regular graph G', such that:*

G is k-colourable iff G' is k-colourable

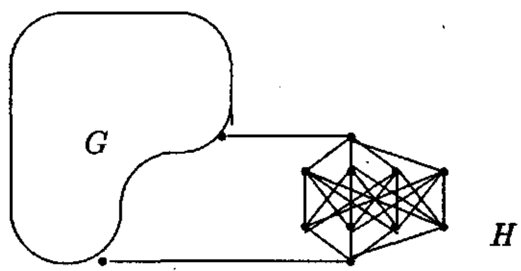

Figure 4: G is 4-colourable iff G' is 4-colourable

Proof: If there are at least two vertices of degree k in G, then we add a copy of graph H. For given k the graph H is defined as follows. Take a complete bipartite graph $K_{k,k}$. Add two vertices and connect one vertex with all the vertices of one independent set of the $K_{k,k}$ and the other vertex with the second independent set of the $K_{k,k}$ (for the case $k = 4$ see Fig. 4). In this way we reduce the number of vertices of degree k by two.

If there is only one vertex of degree k in G, then we construct a new graph as follows: Take two copies of G, connect the two vertices of degree k with an edge. The resulting graph is obviously $k + 1$-regular and it is easy to see that it is k-colourable exactly when G is k-colourable. **Q.E.D.**

For a proof of a generalization of the proposition 1 we need a lemma of the following type: decision problem of k-colouring on arbitrary graph can be reduced to the same problem on a graph of maximal vertex degree $k + 1$.

In the proof of the proposition for 3-colouring we used the result of Garey, Johnson and Stockmayer. Here we give the idea of a proof for $k = 4$. We were not able to generalise the idea for $k > 4$.

Lemma 7 *The decision problem of 4-colouring of graphs of vertex degree ≤ 5 is NP-complete.*

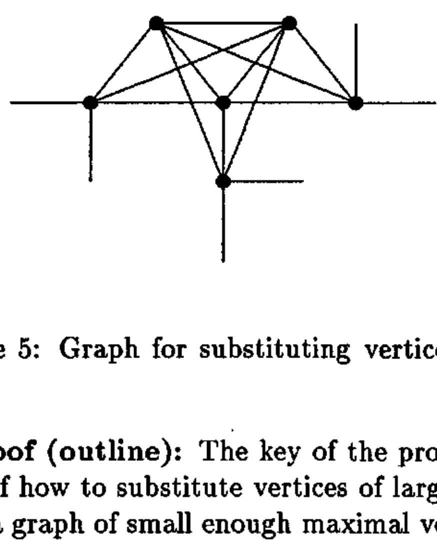

Figure 5: Graph for substituting vertices of degree 6

Proof (outline): The key of the proof is the idea of how to substitute vertices of large degree with a graph of small enough maximal vertex degree and with property that any 4-colouring of the resulting graph $G\prime$ defines a 4-colouring of the original graph G. Such graphs are given in Figures 5,6 and 7. The graphs in Fig. 5 and Fig. 6 are used for substituting vertices of degrees 6 and 7, respectively. For vertices of larger degrees, a longer chain is used, as indicated on Fig. 7. The graphs given have the property, that in any proper 4-colouring all the vertices with 'free edges' have to be coloured with the same colour. (This colour

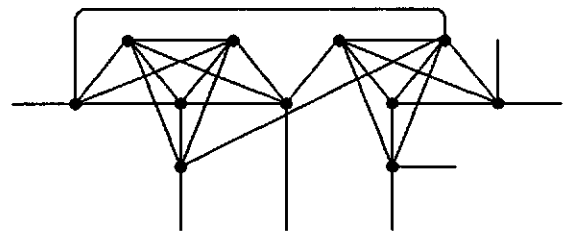

Figure 6: Graph for substituting vertices of degree 7

can be then assigned to the substituted vertex in the original graph. The other vertices of G can then be assigned the same colours as they had in the 4-colouring of $G\prime$.) We omit the details. **Q.E.D.**

With a straightforward generalization of the proof of Lemma 4 we have also:

Lemma 8 $\Pi(k, \Delta) \propto \Pi(k, \Delta + 1)$

Therefore:

Proposition 2 *The decision problem of 4-colouring of Δ-regular graphs $\Pi(4, \Delta)$ is NP-complete for any $\Delta \geq 5$.*

Again, because of the theorem of Brooks [3], the problem $\Pi(4, \Delta)$ has polynomial time complexity for $\Delta \leq 4$. Thus for all the problems $\Pi(4, \Delta)$ we know whether they are polynomial or NP-complete. Let us conclude with a couple of conjectures. Since we were unable to generalise the Lemma 7 we state

Conjecture 1 *The decision problem of k-colouring of graphs with vertex degree $\leq k + 1$ is NP-complete.*

If the first conjecture was true, then we would have a nice classification of time complexity for all the problems $\Pi(k, \Delta)$.

Conjecture 2 *For any $k > 2$, $\Delta > 2$ the decision problem of k-colourability of Δ-regular graphs $\Pi(k, \Delta)$ is NP-complete if $\Delta > k$ and is polynomial otherwise.*

Let us conclude with a simple consequence of the proposition. Assume we have an algorithm

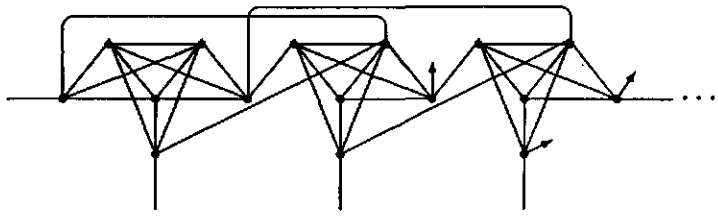

Figure 7: Graph for substituting vertices of degree > 5

$\mathcal{A}$ for 3-colouring and we want to characterise graphs, for which the algorithm does not provide the correct solution in polynomial time. If P$\neq$NP then for any algorithm $\mathcal{A}$ for each $\Delta \geq 4$ there exists an infinite family $F(\mathcal{A}, \Delta)$ of Δ-regular graphs such that the algorithm $\mathcal{A}$ has superpolynomial complexity on each family $F(\mathcal{A}, \Delta)$. If this were not the case for some Δ then $\mathcal{A}$ would be a polynomial algorithm for 3-colouring of Δ-regular graphs, which would imply P=NP!

Acknowledgement. The contents of the paper is based on a part of author's dissertation submitted to University of Ljubljana under supervision of Tomaž Pisanski. Thanks are also due to the anonymous referees, who helped to improve the quality of presentation considerably.

References

[1] E.A. Bender, H.S. Wilf, A Theoretical Analysis of Backtracking in the Graph Coloring Problem, *Journal of Algorithms* **6** (1985) 175-282.

[2] J.A. Bondy, U.S.R. Murty, *Graph Theory with Applications*, Amer. Elsevier, New York, 1976.

[3] R.L. Brooks, On Coloring the Nodes of a Network, *Proc. Camb. Phil. Soc* **37** (1941) 194-197.

[4] K. Edwards, The Complexity of Colouring Problems on Dense Graphs, *Theoretical Computer Science* **43**(1986) 123-147.

[5] J.A. Ellis, P.M. Lepolesa, A Las Vegas Graph Colouring Algorithm, *The Computer Journal* **32** (1989) 474-476.

[6] M.R. Garey, D.S. Johnson, L. Stockmeyer, Some Simplified NP-complete Graph Problems, *Theoretical Computer Science* **1** (1976) 237-267.

[7] M.R. Garey, D.S. Johnson, *Computers and Intractability*, W.H. Freeman and Co., San Francisco, 1979.

[8] A.D. Petford, D.J.A. Welsh, A Randomised 3-Colouring Algorithm, *Discrete Math.*, **74** (1989) 253-261.

[9] J.S. Turner, Almost All k-Colorable Graphs Are Easy to Color, *Journal of Algorithms* **9** (1988) 63-82.

[10] J. Žerovnik, A Randomized Algorithm for k-colorability, *Discrete Mathematics* (to appear).

METAPHYSICALISM OF INFORMING

Anton P. Železnikar
Volaričeva ul. 8, 61111 Ljubljana, Slovenia
anton.p.zeleznikar@ijs.si

Keywords: counterinforming, cyclicity, informational embedding, informing, intelligence, metaphysicalism, parallelism

Edited by: Jiři Slechta

Received: January 26, 1993 **Revised:** February 18, 1993 **Accepted:** March 1, 1993

This paper is an introduction to the phenomena of metaphysical informing occurring within an informational entity [Železnikar 92a, Železnikar 92b]. The basic question is how to structure and how to organize the processes of informing within the metaphysical triplet of informing, counterinforming, and informational embedding, which perform cyclically, in parallel, and spontaneously in a complex entity–metaphysical cycle. The problem of the so–called metaphysical informing (called metaphysicalism) has to be solved conceptually and, then, constructively, that is, in a language–formalized (machined) way. The open cyclic–parallel and spontaneous informing hides the potentiality of intelligence (intelligent information) which through informing, counterinforming, and informational embedding of an informational entity in question comes to the informational surface (of an observer). The aim of this paper is to expose certain possibilities of metaphysical informing within machines, programs, and tools performing in an informationally arising environment. Metaphysicalism, that is, cyclic, spontaneous, entity–intentional and informationally open informing, seems to be the most fundamental problem which has to be solved formally (constructively) on the way to informational machine. To some extent, the possibilities of such proceedings are already visible.

Ich würde hier sagen, daß uns ein Vorstellungsakt als solcher direkt anschaulich wird, wo wir gerade diesen Unterschied zwischen Vorstellung und Vorstellung dieser Vorstellung phänomenologisch *konstantieren.*

—Edmund Husserl [Husserl 00] II/1 508

1 Introduction

Metaphysicalism[1] (called also metaphysically cyclic informing [Železnikar 92b]) is a term denoting the interior phenomenalism of an informational entity. Metaphysicalism of an informational entity means its *own* circular–parallel and spontaneous informing. The metaphysical pertains to an entity's circular and parallel informing, which is spontaneous, that is, speaking generally, unforeseeable, unpredictable to certain extent (sense) or informationally arising, however, entity–intentional, structurally and organizationally oriented or, simply, informationally persevering.

Informing as a phenomenon of an informational entity is the entity process, by which the entity arises, that is, develops, maintains both its structure and its own and from the environment impacted phenomenality, acts in an informational way. Informing as an entity active component performs, as we say, through informing per se, counterinforming, and informational embedding. In the triplet informing–counterinforming–embedding, the informing is a regular informational phenomenon being in accord with the entity normal intention, its phenomenalizing informational stream, keeping the entity identity,

[1]This paper is a private author's work and no part of it may be used, reproduced or translated in any manner whatsoever without written permission except in the case of brief quotations embodied in critical articles.

structure, and organization. In contrary to informing and in regard to it, the counterinforming is a disturbing informing component which arises during the process of informing, as a consequence of interior and exterior informational impacts. At the first glance, as a result within an entity informing, counterinforming is not well-structured and well-organized yet, it is not informationally well-connected in respect to the ruling informing, which determines the character of informational entity.

Informational embedding as the next component in the informing of an entity has to connect properly the arising and from the environment arriving informational items to the informational body of the entity. Embedding as a form of informing is arising according to the phenomena of the arising information within counterinforming as well as the phenomena of the arriving information from the entity environment.

In the pointed sense, metaphysicalism is nothing else than a common term for the informational phenomenality within an informational entity, within which the phenomena of informing, counterinforming, and informational embedding occur. This interior phenomenon of an informing entity is not necessarily evident for the entity exterior observer and, as one may say, remains concealed to a certain informational extent. The aim of this paper is to analyze and to determine these phenomena formally by means of the informational language [Železnikar 92a] and, through this formalization, to capture the conceptuality of informing of an informational machine implementation [Železnikar 92c]. Similar needs can arise within the so-called knowledge archives projects [Knowledge 92], where knowledge and components of knowledge can emerge and have to be determined informationally.

2 Formalizing Some Basic Axioms of Informing

How does an informational entity, marked by α, inform and in which way is it informed? We distinguish four basic types of α's informing called *externalism, internalism*[2], *metaphysicalism,* and *phenomenalism* [Železnikar 92b].

The externalism[3] of the informational operand α means the possibility of α to inform other entities and itself, called also *α's informing(ness) for others and itself.* The informingness of α is its basic (potential) property (predicate, physical phenomenon) marked by the general informational operator $\models$ on its right side. The externalism of entity α is represented by informational formula $\alpha \models$ and reads *α inform(s).* Thus, $\alpha \models$ is an open formula (with the open right side of operator $\models$).

The internalism of the informational operand α means the possibility of α to be informed by other entities and by itself; it is called also *α's informedness by others and by itself.* The informedness of α is its basic (potential) property (predicate, physical phenomenon) marked by the general informational operator $\models$ on its left side. The internalism of entity α is represented by informational formula $\models \alpha$ and reads *α is/are informed* or *α is/are being informed.* Thus, $\models \alpha$ is an open formula (with the open left side of operator $\models$).

The metaphysicalism of the informational operand α means the possibility of α to inform and to be informed by itself; it is called also *α's informingness and informedness in itself.* The interior cyclic and spontaneous informingness and informedness of α, called α's metaphysicalism, is its basic (potential) property (predicate, physical phenomenon) marked by the general informational formula (expression) $\alpha \models \alpha$. This formula reads *α informs and is being informed metaphysically* or, in an informationally general way, *α informs and is being informed cyclically and spontaneously in itself.* However, metaphysicalism $\alpha \models \alpha$ is in no way a closed formula depending solely on α's own (internal) informingness and informedness.

The last of basic informational axioms is called phenomenalism[4]. The phenomenalism of entity α is a consequence of its externalism, internalism, and metaphysicalism, is an informational system

[2] The informational internalism may be comprehended as a subjective informational phenomenalism, phenomenalizing the world and the entity into the entity in question.

[3] Informational externalism is called also *informatio prima* because of the basic informational hypothesis that everything, which *is*, informs.

[4] Informational phenomenalism is the most general principle, by which things inform and are informed in various ways, e.g. physically, biologically, socially, etc. Phenomenalism may not be replaced by phenomenology, which is a philosophical discipline (for instance, [Husserl 00]).

of those phenomena and a systemic expression of two basic formulas (connected in parallel by a semicolon) $\alpha \models; \models \alpha$. Thus, the previous formulas are subformulas of α's phenomenalism, that is

$$(\alpha \models, \models \alpha, \alpha \models \alpha) \subset (\alpha \models; \models \alpha)$$

Informational operator $\subset$ marks the subinforming entities (externalism, internalism, metaphysicalism), separated by commas, within the informing entity (phenomenalism).

The four basic axioms, which pertain to *externalism, internalism, metaphysicalism,* and *phenomenalism*, are forms of the so-called *informationalism* concerning basic modes of an informational entity informing (in Latin, modi informationis). We have formalized them by senseful formulas being derived from basic axioms pertaining to entity α [Železnikar 92b]. These formulas are expressions. Externalism means always an expression, that is, exteriorization [Derrida 67] (output phenomenalism). On contrary, internalism means an impression, that is, interiorization (input phenomenalism). Both externalism and internalism carry meaning (in German, *Bedeutung* [Husserl 00]). This interwoven meaning causes the expression and impression of an entity metaphysicalism to an exterior observer together with the intention of circular and spontaneous metaphysical phenomenality. Similar could be said for α's phenomenalism.

Within this expressional and meaningful scope the following can be concluded: metaphysicalness of α possesses its own externalism, internalism, metaphysicalism, and phenomenalism. And, also all externalism, internalism, and phenomenalism of α possess each own metaphysicalism (metaphysical recursiveness). But by cyclic and parallel decomposition of metaphysicalism $\alpha \models \alpha$, the marker (informational operand) α develops and is being developed through the arising of its meaning (contents, significance, structure, organization, informational broadening). In parallel to the cyclic decomposition, as a consequence of a straightforward metaphysical analysis, parallel meanings can emerge and in this way altogether can be composed into a complexly developing scheme of the initial (symbolic) metaphysicalism $\alpha \models \alpha$. A metaphysical informational system concerning entity α is coming into existence through informing of α and it concerning environment.

3 Problems of Informing of an Informational Entity

The question is how to determine, conceptualize, design, construct, organize and, lastly, implement the process of informing as a regular activity of an informational entity. What to say in accord to informing which represents the active informing component of an informational entity? With the last question, the duality of informingness (active component) and informedness (passive component) is implicitly introduced into the meaning of informational entity. The preceding questions are on the way to possibilities of an informational machine implementation which, in its particular cases, reduces in, for instance, electronic dictionary, knowledge archives [Knowledge 92], expert tools, or intelligent machine.

The general informational operator $\models$ is, from the view of informational operand α, an implicit (to α belonging) expression of informing of (within) operand α. Informing of operand α can also be explicated in the form of an operand entity, marked by $\mathcal{I}_\alpha$, or by the functional (predicative) form $\mathcal{I}(\alpha)$. A more directly corresponding notation for this kind of informing which pertains to α would be simply $\mathcal{A}$. The correspondence between entities $\alpha, \beta, \gamma, \cdots \omega$ and their informing would be $\mathcal{A}, \mathcal{B}, \mathcal{C}, \cdots \mathcal{Z}$, respectively.

3.1 Basic Metaphysical Decomposition

Within an informational entity α, the outmost metaphysical cycle will be $\alpha \models \alpha$. By definition, through the most primitive metaphysical decomposition, there is

$$\alpha \rightleftharpoons_{\mathrm{Def}} \left(\begin{array}{c} ((\alpha \models \mathcal{I}_\alpha) \models \alpha); \\ (\alpha \models (\mathcal{I}_\alpha \models \alpha)) \end{array} \right) \quad (1)$$

As informing $\mathcal{I}_\alpha$ is introduced, it represents an α–inner metaphysical cycle in the form

$$\mathcal{I}_\alpha \rightleftharpoons_{\mathrm{Def}} \left(\begin{array}{c} ((\mathcal{I}_\alpha \models \alpha) \models \mathcal{I}_\alpha); \\ (\mathcal{I}_\alpha \models (\alpha \models \mathcal{I}_\alpha)) \end{array} \right) \quad (2)$$

where $\rightleftharpoons_{\mathrm{Def}}$ is read as *informs* or *means by definition.* In the first step of analysis of informing $\mathcal{I}_\alpha$, we put the question how could $\mathcal{I}_\alpha$ be positioned and attitudinized within α and vice versa or, how do α and $\mathcal{I}_\alpha$ impact each other dynamically in an informational way. The basic informational cycles pertaining to informingness and informedness of both entity α and its informing $\mathcal{I}_\alpha$ are, in fact, manifold, e.g.,

$$\left.\begin{array}{ll} (\alpha \models \mathcal{I}_\alpha) \models \alpha & \alpha\text{–metaphysicalism 1} \\ \alpha \models (\mathcal{I}_\alpha \models \alpha) & \alpha\text{–metaphysicalism 2} \\ & \\ (\mathcal{I}_\alpha \models \alpha) \models \mathcal{I}_\alpha & \mathcal{I}_\alpha\text{–metaphysicalism 1} \\ \mathcal{I}_\alpha \models (\alpha \models \mathcal{I}_\alpha) & \mathcal{I}_\alpha\text{–metaphysicalism 2} \end{array}\right\} \quad (3)$$

The question is, in which way a particular metaphysical cycle could imply the alternative ones. Thus, considering all possibilities of formula system 3, hypothetically,

$$\begin{array}{l} ((\xi \models \eta) \models \xi) \Longrightarrow \left(\begin{array}{c} \xi \models (\eta \models \xi); \\ (\eta \models \xi) \models \eta; \\ \eta \models (\xi \models \eta) \end{array}\right); \\ \xi \neq \eta; \\ \xi, \eta \in \{\alpha, \mathcal{I}_\alpha\} \end{array} \quad (4)$$

where informational operator $\Longrightarrow$ represents the informational implication and reads as *informs in an implicative way* or, simply, *implies*.

In formula system 3, all formulas, that is, $(\alpha \models \mathcal{I}_\alpha) \models \alpha$; $\alpha \models (\mathcal{I}_\alpha \models \alpha)$; $(\mathcal{I}_\alpha \models \alpha) \models \mathcal{I}_\alpha$; and $\mathcal{I}_\alpha \models (\alpha \models \mathcal{I}_\alpha)$, are metaphysical and deduced (decomposed) from the basic metaphysical, that is, informationally cyclic form $\alpha \models \alpha$ and its inner consequence $\mathcal{I}_\alpha \models \mathcal{I}_\alpha$. Sometimes, we understand these basic formulas as the shortcuts for α's and $\mathcal{I}_\alpha$'s metaphysicalism which complexities are hidden in the general informational operator $\models$. Thus, the decomposition of $\alpha \models \alpha$ and $\mathcal{I}_\alpha \models \mathcal{I}_\alpha$ concerns, in fact, the cyclically connecting informational operator $\models$. We shall deal with more complex and parallel basic informational and composed (also perplexed) informational cycles within an entity metaphysicalism.

3.2 Spontaneity of Informing

In the second step of our investigation we rise the question of possibilities of (inner, metaphysical) informational spontaneity of informing $\mathcal{I}_\alpha$, that is, of the phenomenon of informational arising of entity α.

Informational spontaneity does not mean an arbitrary development of an entity contents, structure and organization, that is, of an arbitrary entity–informational broadening, meaningful filtering or notional purification. In an informational situation and attitude, entity α already has its informational structure and organization (course, orientation, intention, behavior) and in its own way filtered informational impactedness of its exterior (informedness) on disposal, both in the form of informing $\mathcal{I}_\alpha$ and entity α as a whole (including also the temporarily passive components of its arising informational structure). Informational spontaneity of α is caused intentionally by itself and it impacting exterior and is in accord with its instantaneously changing (arising) orientation (worldliness).

We have to answer the question of spontaneity pertaining to the entity–informational intention in a constructive way. Which mechanisms for informational spontaneity simulation, modeling, and organization are realistic, machine realizable, and conceptually possible? Answers to the last question are various and depend on particular situation, for instance, multimedial, pictorial, acoustic, and linguistic, pertaining to signals, data, written text, dictionaries [Dictionary 90a, Dictionary 90b], knowledge archives [Knowledge 92], etc.

Spontaneity means, for instance, a free moving along an existing informational net and taking with informational items which correspond, coincide, fit, match, etc. an informational situation and connect, interweave, embed, interpret, etc. them into, in, and within a concrete informational entity. This model of spontaneity could be called a model of free informational association. In fact, the decision, which items in the net to take with, is spontaneous, depending on some distinguished informational attitudes concerning the entity in question and its informational environment.

It is to stress that spontaneity as such is an informational entity by itself, which is a constitutive part of the informational entity and of to the entity pertaining informational environment. Spontaneity is intentional, concerns a goal–oriented information, is in no way something informationally surprising and quite unexpected. It arises from an informational impulse, tendency, but unplanned as an internal force or cause. Informationally spontaneous strategies, procedures, random associative processes, and like that have to be conceptualized and used as system supporting mechanisms for spontaneous informing of entities.

3.3 Some Inner States of Circular Informing

How does an informational entity α begin to inform and how does it inform from one situation into another? The beginning of an entity informing $\mathcal{I}_\alpha$ is caused by the appearance of marker α, which is the most simple expression and carries the most primitive meaning, for instance, an identifier, a headword, symbol, token, sign, etc. In this very beginning situation, informing $\mathcal{I}_\alpha$ informs the basic meaning being different from nothing[5]. In an informational environment, the informing of initial entity α is being supported, for instance, through its physical environment, informational machine, knowledge archive, informational dictionary, living actor, linguistic system, multimedia exterior, associative mechanisms, dispersive algorithmic procedures, artificial surroundings, etc. Several entities can be informed of the occurrence of an initial entity α and can support its informational arising in different informational ways.

How does this inner development of informing proceed and which are the proposed (defined) mechanisms, structure, organization, in short, the entity metaphysicalism? We have to develop a systematic approach to the problem of informing, which could be applied in cases of an informational environment implementation, for instance, in an informational machine or even in a computer, which can model an adequate informational environment.

Informing $\mathcal{I}_\alpha$ is an active part of entity α. Sometimes, by $\mathcal{I}_\alpha$, the whole informational activity of entity α is meant. But the emphasis of informing as a distinguishable entity within an entity as a unit is in its includedness or participation, that is,

$$\mathcal{I}_\alpha \rightleftharpoons_{\mathrm{Def}} (\mathcal{I}_\alpha \subset \alpha) \tag{5}$$

[5]The nothing means the nonappearance of something in an informational context. However, as soon as we speak about the nothing of something, the something appears in the informational context and becomes an informing entity. Through this, the informational existence of something (as nothing) cannot be denied anymore.

In the metaphysical sense, there is

$$(\mathcal{I}_\alpha \models \mathcal{I}_\alpha) \subset (\alpha \models \alpha) \tag{6}$$

irrespective of the possible informational structure of cycle $\mathcal{I}_\alpha \models \mathcal{I}_\alpha$. It is understood that informing $\mathcal{I}_\alpha$ is an informationally subordinated active component of informational entity (unity) α. According to definition 5, as a consequence of the basic metaphysical includedness of $\mathcal{I}_\alpha$ in α, we can introduce the includedness of corresponding $\mathcal{I}_\alpha$-cycles,

$$(\mathcal{I}_\alpha \subset \alpha) \rightleftharpoons_{\mathrm{Def}}$$

$$\left(\begin{array}{l} ((\mathcal{I}_\alpha \models \mathcal{I}_\alpha) \subset (\alpha \models \alpha)) \Longrightarrow \\ \left(\begin{array}{l} ((\mathcal{I}_\alpha \models \alpha) \models \mathcal{I}_\alpha) \subset \\ ((\alpha \models \mathcal{I}_\alpha) \models \alpha); \\ (\mathcal{I}_\alpha \models (\alpha \models \mathcal{I}_\alpha)) \subset \\ (\alpha \models (\mathcal{I}_\alpha \models \alpha)) \end{array} \right) \end{array} \right) \tag{7}$$

for the corresponding short-form metaphysical decompositions. The decomposition procedure can extend further to longer and longer forms of cycles considering the components of informing, counterinforming, and informational embedding, that is, considering the generalized idea of metaphysical informing as discussed in the previous sections. In concrete cases, these general terms will be particularized and, certainly, decomposed in specific (particular) ways.

Within the general context, we can speak about short, medium-sized, and long metaphysical cycles of informing. If we introduce the measure for metaphysical length ℓ_{meta} of an informational formula φ, that is, $\ell_{\mathrm{meta}}(\varphi)$, then $\ell_{\mathrm{meta}}(\mathcal{I}_\alpha) = 0$, $\ell_{\mathrm{meta}}(\mathcal{I}_\alpha \models \mathcal{I}_\alpha) = 1$, $\ell_{\mathrm{meta}}((\mathcal{I}_\alpha \models \alpha) \models \mathcal{I}_\alpha) = 2$, $\ldots$, $\ell_{\mathrm{meta}}(((((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \alpha) \models \mathcal{I}_\alpha) = 6$. In the general case of informing with counterinforming and embedding within entity α, the maximal length has the value 6 (a long metaphysical cycle). In concrete, particularized cases, the metaphysical length can be extended by proceeding into a greater detail of the informational problem.

Which are the medium-sized metaphysical cycles of informing? We determined an entity informing in a basic (formula 2) and in a complex way (formula 8), considering counterinforming $\mathcal{C}_\alpha$ with counterinformation γ_α and informational embedding $\mathcal{E}_\alpha$ with embedding information ε_α. Counterinformational and embedding-informational metaphysical subcycles can inform within the long metaphysical cycle of informing (formula 17).

The long metaphysical cycles concerning informing $\mathcal{I}_\alpha$ are, systematically,

$$\left(\begin{array}{l} \mathcal{I}_\alpha \models_\mu \\ \quad (\mathcal{C}_\alpha \models (\gamma_\alpha \models (\mathcal{E}_\alpha \models (\varepsilon_\alpha \models (\alpha \models \\ \quad \mathcal{I}_\alpha))))); \\ (\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models_\mu \\ \quad (\gamma_\alpha \models (\mathcal{E}_\alpha \models (\varepsilon_\alpha \models (\alpha \models \mathcal{I}_\alpha)))); \\ ((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models_\mu \\ \quad (\mathcal{E}_\alpha \models (\varepsilon_\alpha \models (\alpha \models \mathcal{I}_\alpha))); \\ (((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models_\mu \\ \quad (\varepsilon_\alpha \models (\alpha \models \mathcal{I}_\alpha)); \\ ((((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \\ \quad \models_\mu (\alpha \models \mathcal{I}_\alpha); \\ (((((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \\ \quad \models \alpha) \models_\mu \mathcal{I}_\alpha \end{array}\right) \subset (\mathcal{I}_\alpha \models \mathcal{I}_\alpha) \qquad (8)$$

where $\models_\mu$ marks the so-called point of main (symbol μ) cyclic informing.

4 Problems of Counterinforming of an Informational Entity

To open the possibilities of counterinforming mechanisms, that is, their informational implementation, we have to make a short overview concerning concepts of counterinforming which have their roots in, for instance, linguistic meaning of both words *counter* and *informing*. Within the question of how to implement the process of counterinforming, by which in regard to the informing entity a new, different information is coming into existence, we can use various concepts concerning the meaning of the word *counter* (adverb, verb transitive, and word prefix) while the meaning of the word *informing* is consequently connected with the word *information*, understood in an extended sense.

To counterinform can mean to inform counter to the instantaneous course, intention, ruling, perseverance, phenomenalism of an informing entity. This can happen in a pure informational way, where in parallel to the straightforwardness of an informational entity always to it contrary phenomenalism (e.g., doubtingness, differentia, diversity, associationism, dissociation, frustrations, etc.) comes to the informational surface. An example of counterinforming in a linguistic way is the coming up of antonymous meaning to an existing synonymous meaning of a headword, phrase, or sentence and connecting such phenomena of meaning reasonably to the already recognized meaning (knowledge).of the original informational item. This kind of counterinforming might mean nothing else than an additional interpretation of the original meaning, broadening the original meaning in significant and simultaneously various and varying ways. Such emerging of information concerning a distinct informational entity is common and obvious within human cultures performing discourse, confrontation of beliefs, democratic dialog, brainstorming, etc.

Counterinforming $\mathcal{C}_\alpha$ is a substantial generative part of informing $\mathcal{I}_\alpha$ and, by definition, is the producer of the counterinformational entity γ_α. There is no essential difference in regard to general formulas 1 and 2 and thus

$$\begin{array}{l} \mathcal{C}_\alpha \rightleftharpoons_{\mathrm{Def}} \left(\begin{array}{l} ((\mathcal{C}_\alpha \models \gamma_\alpha) \models \mathcal{C}_\alpha); \\ (\mathcal{C}_\alpha \models (\gamma_\alpha \models \mathcal{C}_\alpha)) \end{array}\right) \\ \gamma_\alpha \rightleftharpoons_{\mathrm{Def}} \left(\begin{array}{l} ((\gamma_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha); \\ (\gamma_\alpha \models (\mathcal{C}_\alpha \models \gamma_\alpha)) \end{array}\right) \end{array} \qquad (9)$$

The essential difference between informing $\mathcal{I}_\alpha$ and counterinforming $\mathcal{C}_\alpha$ in comparison to formula 5 is the following:

$$((\gamma_\alpha \subset \mathcal{C}_\alpha) \subset \mathcal{I}_\alpha) \subset \alpha \qquad (10)$$

We see how counterinforming $\mathcal{C}_\alpha$ is informationally subordinated (included) in informing $\mathcal{I}_\alpha$ irrespective of other possible inclusions pertaining to different informational entities.

Metaphysicalism of counterinforming is a phenomenon with various faces, also in respect to the pure formalism, that is, to different possibilities of formal expressing. Which are the possible forms (all of them in a given situation) of counterinformational metaphysicalism within an informing entity α? We can develop (decompose, compose) metaphysical concepts proceeding from the shortest to the longest cycle, for instance,

$$\begin{array}{l} \mathcal{C}_\alpha \models \mathcal{C}_\alpha; \\ (\mathcal{C}_\alpha \models \gamma_\alpha) \models \mathcal{C}_\alpha; \\ ((\mathcal{C}_\alpha \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \mathcal{C}_\alpha; \\ (((\mathcal{C}_\alpha \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \mathcal{C}_\alpha; \\ ((((\mathcal{C}_\alpha \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \alpha) \models \mathcal{C}_\alpha; \\ (((((\mathcal{C}_\alpha \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \alpha) \models \mathcal{I}_\alpha) \\ \quad \models \mathcal{C}_\alpha \end{array} \quad (11)$$

Formulas of system 11 are not in a final shape since they can be metaphysically decomposed in regard to each in them occurring operand entity. For the last formula of system 11 we can introduce metaphysical markers for γ_α and α, for instance,

$$\begin{array}{l} (((((\mathcal{C}_\alpha \models (\gamma_\alpha \models \gamma_\alpha)) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \\ (\alpha \models \alpha)) \models \mathcal{I}_\alpha) \models \mathcal{C}_\alpha \end{array} \quad (12)$$

Under such circumstances, in the process of metaphysical decomposition, also a direct informational connection between entity α and its counterinformational entity γ_α can come into existence as a parallel formula to the informationally arising metaphysicalism. This process is in no way arbitrarily spontaneous, it keeps the route of informational intentionality of α and it informationally influencing environment.

5 Problems of Informational Embedding of an Informational Entity

Informational embedding is a process by which the arisen counterinformation and from the exterior of an informing entity arriving information is informationally embedded (connected, meaningly associated, interpreted) into the informing entity. At this attempt of investigation, we shall not research into particular details, by which in a certain case the embedding process could be determined according to some concrete informational demands.

Informational embedding $\mathcal{E}_\alpha$ is a part of α's informing and, by definition, the producer of the embedding information ε_α. There is no notional difference in regard to formulas 1 and 2, thus

$$\begin{array}{l} \mathcal{E}_\alpha \rightleftharpoons_{\mathrm{Def}} \left(\begin{array}{l} ((\mathcal{E}_\alpha \models \varepsilon_\alpha) \models \mathcal{E}_\alpha); \\ (\mathcal{E}_\alpha \models (\varepsilon_\alpha \models \mathcal{E}_\alpha)) \end{array} \right) \\ \\ \varepsilon_\alpha \rightleftharpoons_{\mathrm{Def}} \left(\begin{array}{l} ((\varepsilon_\alpha \models \mathcal{E}_\alpha) \models \varepsilon_\alpha); \\ (\varepsilon_\alpha \models (\mathcal{E}_\alpha \models \varepsilon_\alpha)) \end{array} \right) \end{array} \quad (13)$$

The essential difference between informing $\mathcal{I}_\alpha$ and informational embedding $\mathcal{E}_\alpha$ in comparison to formula 5 is the following:

$$(((((\varepsilon_\alpha \subset \mathcal{E}_\alpha) \subset \gamma_\alpha) \subset \mathcal{C}_\alpha) \subset \mathcal{I}_\alpha) \subset \alpha \quad (14)$$

We see how informational embedding $\mathcal{E}_\alpha$ is indirectly, via counterinformational entity γ_α and counterinforming $\mathcal{C}_\alpha$, informationally included in informing $\mathcal{I}_\alpha$.

To interpret concretely informational embedding $\mathcal{E}_\alpha$ and by embedding produced embedding entity (information) ε_α, let us introduce understanding $\mathcal{U}_\alpha$ instead or as a part of $\mathcal{E}_\alpha$ and meaning μ_α of an interior or exterior entity β, that is, $\mu_\alpha(\beta)$. In accord to formula system 13 and formula 14, there is

$$\begin{array}{l} (\mathcal{U}_\alpha \models \mu_\alpha(\beta)) \models \mathcal{U}_\alpha; \\ \mathcal{U}_\alpha \models (\mu_\alpha(\beta) \models \mathcal{U}_\alpha); \\ ((((\mu_\alpha(\beta) \subset \mathcal{U}_\alpha) \subset \gamma_\alpha) \subset \mathcal{C}_\alpha) \subset \mathcal{I}_\alpha) \subset \alpha \end{array} \quad (15)$$

Meaning $\mu_\alpha(\beta)$ is, for instance, an interpretation of meaning pertaining to entity β, produced by understanding $\mathcal{U}_\alpha$. Additionally, entity $\mathcal{U}_\alpha$ can concern other meanings of other informational subjects, e.g., $\mu_\alpha(\xi), \mu_\alpha(\eta)$, etc. *To mean* means to apply different informational modi or rules (modes) of inference, known (in Latin) as *modus ponens, modus tollens* (affirmative and negative mode in traditional logic, respectively), *modus obliquus* (e.g., logic of absurdity, for instance, devious inversion, adjustment, or directionality, which appears together with the direct), *modus rectus, modus procedendi* (logic of intentionality and processing to reach a certain goal, respectively), *modus vivendi* (logic of tolerant coexistence, for example), *modus possibilitatis, modus necessitatis* (modal logic), etc.

6 A Constitution of the Metaphysical Cycle as Informing, Counterinforming, and Informational Embedding

So far, the structure of metaphysical cycles was determined with basic informational components and their length was measured by the number of occurring informational operators $\models$ within the cycle. Let us remind and systemize the previously discussed metaphysical cycles of various lengths. We distinguish several types of metaphysical cycles belonging to an informing entity. These cycles can be classified as, for instance, primitive, basic, medium-sized (counterinformational-embedding), and long (the longest, holistic) metaphysical cycles.

6.1 Primitive Metaphysical Cycles of an Informing Entity

Although primitive metaphysical cycles of an informing entity are trivial, they are the starting points, from which the cyclical decomposition begins. The longest metaphysical cycle belongs to the informational entity α in question. Its informing component $\mathcal{I}_\alpha$ is for one step shorter, etc., thus, at the end, embedding informational entity ε remains, at the present state of decomposition, in its trivial form. At the beginning, the primitive informational cycles are the following:

$$\left.\begin{array}{ll} \alpha \models \alpha & \text{informational entity} \\ \mathcal{I}_\alpha \models \mathcal{I}_\alpha & \text{informing} \\ \mathcal{C}_\alpha \models \mathcal{C}_\alpha & \text{counterinforming} \\ \gamma_\alpha \models \gamma_\alpha & \text{counterinformational entity} \\ \mathcal{E}_\alpha \models \mathcal{E}_\alpha & \text{informational embedding} \\ \varepsilon_\alpha \models \varepsilon_\alpha & \text{embedding–informational} \\ & \text{entity} \end{array}\right\} \quad (16)$$

In the beginning conceptual state of decomposition, the basically extended metaphysical cycles are as follows:

$$\begin{array}{l} (((((\alpha \models \mathcal{I}_\alpha) \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \\ \quad \varepsilon_\alpha) \models \alpha; \\ ((((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \mathcal{I}_\alpha; \\ (((\mathcal{C}_\alpha \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \mathcal{C}_\alpha; \\ ((\gamma_\alpha \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \gamma_\alpha; \\ (\mathcal{E}_\alpha \models \varepsilon_\alpha) \models \mathcal{E}_\alpha; \\ \varepsilon_\alpha \models \varepsilon_\alpha \end{array} \quad (17)$$

However, we shall see, how these initial cycles will become as long as the longest cycle of entity α (formula 25) because of the strict circular nature of α's informing.

6.2 Basic Metaphysical Cycles of an Informing Entity

We distinguished six informational entities as basic components of an metaphysical cycle belonging to an informing entity, namely: α as the informing entity itself in its wholeness and to it belonging informing $\mathcal{I}_\alpha$; α's counterinforming $\mathcal{C}_\alpha$ and by it informed (produced) counterinformational entity γ_α; and, lastly, α's informational embedding $\mathcal{E}_\alpha$ and by it informed (produced) embedding information (entity) ε_α.

All short metaphysical cycles of the involved informational entities $\alpha, \mathcal{I}_\alpha, \mathcal{C}_\alpha, \gamma_\alpha, \mathcal{E}_\alpha$, and ε_α are the following:

$$\begin{array}{l} (\xi \models \eta) \models \xi; \\ \xi \models (\eta \models \xi); \\ \xi \neq \eta; \\ \xi, \eta \in \{\alpha, \mathcal{I}_\alpha, \mathcal{C}_\alpha, \gamma_\alpha, \mathcal{E}_\alpha, \varepsilon_\alpha\} \end{array} \quad (18)$$

Informational operator $\in$ is read as *informs* in *the context of a set of entities.*

6.3 Metaphysical Cycles Pertaining to Informing $\mathcal{I}_\alpha$

Informing $\mathcal{I}_\alpha$ of informational entity α is the inner mechanism of the activity of informational entity α. Within this cycle of informing several other cycles inform, for instance, both the counterinformational and the embedding–informational one.

The primitive metaphysicalism of α's informing (the starting point of metaphysical decomposition) is expressed by formula $\mathcal{I}_\alpha \models \mathcal{I}_\alpha$. The next step of metaphysical decomposition are two possible short cycles, that is,

$$(\mathcal{I}_\alpha \models \alpha) \models \mathcal{I}_\alpha;\ \mathcal{I}_\alpha \models (\alpha \models \mathcal{I}_\alpha) \quad (19)$$

Counterinforming $\mathcal{C}_\alpha$ and informational embedding $\mathcal{E}_\alpha$ are the informing components (parts) of informing $\mathcal{I}_\alpha$. Thus,

$$(\mathcal{C}_\alpha \models \mathcal{C}_\alpha) \subset \mathcal{I}_\alpha;\ (\mathcal{E}_\alpha \models \mathcal{E}_\alpha) \subset \mathcal{I}_\alpha \quad (20)$$

with $(\gamma_\alpha \models \gamma_\alpha) \subset \mathcal{C}_\alpha$ and $(\varepsilon_\alpha \models \varepsilon_\alpha) \subset \mathcal{E}_\alpha$. There may not exist an informational includedness between entities $\mathcal{C}_\alpha$ and $\mathcal{E}_\alpha$.

Several medium-sized metaphysical cycles concerning $\mathcal{I}_\alpha$ can be observed, for instance:

$$\begin{array}{l}((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{I}_\alpha;\\ (\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models (\gamma_\alpha \models \mathcal{I}_\alpha);\\ \mathcal{I}_\alpha \models (\mathcal{C}_\alpha \models (\gamma_\alpha \models \mathcal{I}_\alpha));\\ ((\mathcal{I}_\alpha \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \mathcal{I}_\alpha;\\ (\mathcal{I}_\alpha \models \mathcal{E}_\alpha) \models (\varepsilon_\alpha \models \mathcal{I}_\alpha);\\ \mathcal{I}_\alpha \models (\mathcal{E}_\alpha \models (\varepsilon_\alpha \models \mathcal{I}_\alpha));\\ (((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \mathcal{I}_\alpha;\\ ((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models (\mathcal{E}_\alpha \models \mathcal{I}_\alpha);\\ (\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models (\gamma_\alpha \models (\mathcal{E}_\alpha \models \mathcal{I}_\alpha));\\ \mathcal{I}_\alpha \models (\mathcal{C}_\alpha \models (\gamma_\alpha \models (\mathcal{E}_\alpha \models \mathcal{I}_\alpha)));\\ ((((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \mathcal{I}_\alpha;\\ (((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models (\varepsilon_\alpha \models \mathcal{I}_\alpha);\\ ((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models (\mathcal{E}_\alpha \models (\varepsilon_\alpha \models \mathcal{I}_\alpha));\\ (\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models (\gamma_\alpha \models (\mathcal{E}_\alpha \models (\varepsilon_\alpha \models \mathcal{I}_\alpha)));\\ \mathcal{I}_\alpha \models (\mathcal{C}_\alpha \models (\gamma_\alpha \models (\mathcal{E}_\alpha \models (\varepsilon_\alpha \models \mathcal{I}_\alpha))))\end{array} \tag{21}$$

The next metaphysical cycle of informing, $\mathcal{I}_\alpha \models \mathcal{I}_\alpha$, belongs already to the long metaphysical cycle of $\mathcal{I}\alpha$ and will be treated in subsection 6.5.

6.4 Counterinformational–embedding Metaphysical Cycles

It seems reasonable to discuss counterinforming and informational embedding within common informational cycles. Counterinformational entity γ_α, produced by counterinforming $\mathcal{C}_\alpha$, has to be informationally embedded before it could become lost in the informational realm of the informing entity α. The basic question within this context is what does the arisen counterinformational entity γ_α mean at all. The answer to the meaning of γ_α is its embedding, that is informational connection into the informational realm of informing entity α.

We introduce the following informational includedness hierarchy concerning the basic metaphysical counterinforming–embedding subcycle:

$$\begin{array}{l}((\mathcal{C}_\alpha \models \mathcal{C}_\alpha) \subset \mathcal{I}_\alpha) \subset \alpha;\\[1ex] (\mathcal{C}_\alpha \models \mathcal{C}_\alpha) \rightleftharpoons_{\mathrm{Def}} \\ \quad ((((\mathcal{C}_\alpha \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \mathcal{C}_\alpha);\\[1ex] ((((\mathcal{C}_\alpha \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \mathcal{C}_\alpha) \Longrightarrow \\ \left(\begin{array}{l}(((\gamma_\alpha \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \mathcal{C}_\alpha) \models \gamma_\alpha;\\ (((\mathcal{E}_\alpha \models \varepsilon_\alpha) \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha;\\ (((\varepsilon_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha\end{array}\right)\end{array} \tag{22}$$

6.5 Long Metaphysical Cycles

The long metaphysical cycles[6] are deduced from the most primitive forms, marking the involved informational entities within an informing entity α. The most simple surveying definitional scheme is, for instance,

$$\alpha \rightleftharpoons_{\mathrm{Def}} \left(\begin{array}{l}\alpha \models;\ \models \alpha;\\ (\alpha \models \alpha) \Longrightarrow \left(\begin{array}{l}\mathcal{I}_\alpha;\\ \mathcal{C}_\alpha;\\ \gamma_\alpha;\\ \mathcal{E}_\alpha;\\ \varepsilon_\alpha\end{array}\right)\end{array}\right) \tag{23}$$

We see how the metaphysical component $\alpha \models \alpha$ remains informationally open because of the presence of the system formula $\alpha \models; \models \alpha$ in which α's metaphysicalism is recursively open, that is, can inform and can be informed both interiorly and exteriorly in concern to entity α. The metaphysical openness of entity α means that metaphysicalism of α informs and is informed, that is, $(\alpha \models \alpha) \models$ and $\models (\alpha \models \alpha)$, which is a recursive property of system $\alpha \models;\ \models \alpha$. This definitional scheme can be expressed in the primitive metaphysical form, which is

$$\alpha \rightleftharpoons_{\mathrm{Def}} \left(\begin{array}{l}\alpha \models;\ \models \alpha;\\ (\alpha \models \alpha) \Longrightarrow \left(\begin{array}{l}\mathcal{I}_\alpha \models \mathcal{I}_\alpha;\\ \mathcal{C}_\alpha \models \mathcal{C}_\alpha;\\ \gamma_\alpha \models \gamma_\alpha;\\ \mathcal{E}_\alpha \models \mathcal{E}_\alpha;\\ \varepsilon_\alpha \models \varepsilon_\alpha\end{array}\right)\end{array}\right) \tag{24}$$

Formula 23 and formula 24 are merely the initial schemes for the construction of the so-called long metaphysical cycles of informing within informational entity α. The implicational part of the long

[6]The term *long metaphysical cycle* concerns the longest cyclically structured operand–operator informational formula, which considers all the identified components occurring in a metaphysical cycle, constructed by a constructing informational entity. Within characteristic systemic components (informing, counterinforming, counterinformation, embedding, and embedding information) of an entity, concrete informational entities appear (look at, for example, formula 30), which can concretely lengthen the conceptually basic metaphysical cycle.

metaphysical cycle of entity α's informing is

$$(\alpha \models \alpha)_{\mathrm{long}} \rightleftharpoons_{\mathrm{Def}}$$

$$\left(\begin{array}{l} (((((\alpha \models \mathcal{I}_\alpha) \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \\ \quad \models \varepsilon_\alpha) \models \alpha) \Longrightarrow \\ \\ \left(\begin{array}{l} (((((\mathcal{I}_\alpha \models \mathcal{C}_\alpha) \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \\ \quad \models \alpha) \models \mathcal{I}_\alpha; \\ (((((\mathcal{C}_\alpha \models \gamma_\alpha) \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \alpha) \\ \quad \models \mathcal{I}_\alpha) \models \mathcal{C}_\alpha; \\ (((((\gamma_\alpha \models \mathcal{E}_\alpha) \models \varepsilon_\alpha) \models \alpha) \models \mathcal{I}_\alpha) \\ \quad \models \mathcal{C}_\alpha) \models \gamma_\alpha; \\ (((((\mathcal{E}_\alpha \models \varepsilon_\alpha) \models \alpha) \models \mathcal{I}_\alpha) \models \mathcal{C}_\alpha) \\ \quad \models \gamma_\alpha) \models \mathcal{E}_\alpha; \\ (((((\varepsilon_\alpha \models \alpha) \models \mathcal{I}_\alpha) \models \mathcal{C}_\alpha) \models \gamma_\alpha) \\ \quad \models \mathcal{E}_\alpha) \models \varepsilon_\alpha \end{array}\right) \end{array}\right) \qquad (25)$$

In long metaphysical cycles, entity α observes its constituting parts $\mathcal{I}_\alpha, \mathcal{C}_\alpha, \gamma_\alpha, \mathcal{E}_\alpha$, and ε_α, but these parts can observe the entity as a whole too. In principle, all parts can observe each other and the entity as a unit. This situation is in no way principally contradictory. Of course, the question arises, how do different parts "know" the wholeness of entity α, also this knowledge can be equally exhaustive for all constitutive entities, if it is supported (delivered) from a central informational place, where all information concerning informational entity α is collected (e.g. in an informational machine, in the informational operand dictionary [Železnikar 92c]). In this way, in any metaphysical cycle, within which α occurs, entity α is one and the same entity.

The other approach would be to understand entity α, at any place or situation it appears, as a sample of α. For instance, in a formula system,

$$(\alpha \models \mathcal{I}_\alpha) \models \alpha;\ (\mathcal{I}_\alpha \models \alpha) \models \mathcal{I}_\alpha$$

entity α in both formulas could mean one and the same entity. In a different situation, entity α in the first formula could represent one sample and, in the second formula, the other sample of one and the same thing. This situation is not so surprising as it might be understood by the traditional philosophy (readiness-to-hand). Similar affairs take place in a living mind, where different samples of one and the same thing occur within different mental situations and attitudes.

7 Metaphysicalism of an Informational Entity

Metaphysicalism is a common notion, by which the internal informing of an informational entity is determined. This informing is entity-cyclic and sensitive in respect to the entity environment in a semantic-pragmatical (ontic, ontological) way. The entity-cyclic pertains to the informing, counterinforming, and embedding-informational nature of an entity. The semantic-pragmatical of an entity concerns the internalization of external entities, so that they inform within an entity's cyclicity. This cyclicity is structured in parallel and has its short, medium-sized, and long cycles, which pertain to internalized external entities. In this way, metaphysicalism is not only a simple, direct, and straightforwardly shaped informational cyclicity determined once and for all. It is an entity-ontological process, which develops by entity informing in its own way and by means of entity-sensitive impacting of external entities. We shall show the semantic-pragmatical part of metaphysicalism by an example of intelligent informational entity ι in section 8.

Metaphysicalism is a composed and parallel structured cyclicity of an informing entity, which has its internal intention, but remains open for external informational impacts. In this respect, it is a particular view of possible informing of an entity, of its individual and externally impacted processing, in which the identity of an informational entity emerges in an internal way and, for an external observer, also in an externally influenced way.

In this section we have to show the metaphysicalism of an informational entity in its entirety. We have to join the results obtained in the previous sections and interpret them in a compact form. So, let us make a verbal compilation of discussed metaphysical possibilities.

Decomposition of an entity α metaphysicalism starts by the trivial formula $\alpha \models \alpha$. This formula acts as a title (idea as the linguistic-informational meaning) in the top-down design of metaphysicalism. Within this initial situation, metaphysicalism can be tackled by two basic ideas of informing. The first one is in the cyclically based triplet *informing, counterinforming,* and *informational embedding,* which is the fundamental way of an entity informing. The second idea is semantic-pragmatical, which particularizes the first concept and makes it more concrete. In fact, the second idea (in section 8) is an informational projection to the first one.

The next step further from the trivial situation $\alpha \models \alpha$ is the introduction of informing ($\mathcal{I}_\alpha$), counterinforming ($\mathcal{C}_\alpha$) with counterinformational entity (γ_α), and informational embedding ($\mathcal{E}_\alpha$) with embedding informational entity (ε_α). In this way a long basic metaphysical cycle is obtained and all shorter cycles can be derived as parallel informing cycles. At this step of development more concrete entities, particularizing the previous ones, can be brought into the game. For instance, intelligent informational entity ι in section 8 is a good example of a general concept of an intelligent system, which concerns intelligently an entity α.

The question "What is the informing $\mathcal{I}_\alpha$ of an informational entity α?" particularizes entity $\mathcal{I}_\alpha$ with the next question, which is "What are counterinforming $\mathcal{C}_\alpha$ with counterinformational entity γ_α and embedding $\mathcal{E}_\alpha$ with embedding-informational entity ε_α?" In this point of view, both γ_α and ε_α are certain results of entities $\mathcal{C}_\alpha$ and $\mathcal{E}_\alpha$, respectively.

For instance, we can introduce some strategic functions and their results as counterinformational entities ($\mathcal{C}_\alpha$ and γ_α) in the form of intention, significance, sense, etc. Embedding components can embrace certain sensing, observing, perceiving, etc. situation in concern to an observed interior or exterior entity. Then, informing $\mathcal{I}_\alpha$ produces cyclically a result (e.g. meaning to the understood situation), etc. In this manner, a pragmatical way of α's informational structure and organization remains open for possibilities of further development, improvement, intention, etc., that is, an entity's metaphysicalism.

8 Intelligence as an Informational Entity's Metaphysicalism

The question of intelligence can be tackled by the informational theory of metaphysicalism in an innovative, that is, informationally arising, circular, and creatively spontaneous way. Informational schemes of intelligence become highly parallel and circularly perplexed; this yields together with informational formula systems, an informationally arising formula system, describing parallel, circular, and interwoven intelligent informational phenomena. On this basis, intelligent entities can be treated as emerging systems, which can be modeled (machined) by the proposed metaphysical conceptualism.

Intelligent informational entities concern several other mutually perplexed informational entities, for instance, understanding, meaning as a result of understanding, consciousness as a specifically circularly structured informational phenomenality, phenomena of observing, perceiving, conceiving, concluding, comprehending, etc. producing observation, perception, conception, conclusion, comprehension, meaning, etc. as intelligent informational items, respectively. Further, intelligence concerns knowledge, truth, belief, faith, significance, etc., which meaningly overlap each other and form a redundant informational overlapping, that is interweavement, parallelism, community. Only an informational entity possessing some of these characteristics as commonly recognized properties occurs as intelligent, that is, informationally satisfactory in an intelligent way. Thus, the intelligent means to have a sufficiently dynamically phenomenalizing meaning, contents, externalism, internalism, and, certainly, metaphysicalism.

Let us structure metaphysically intelligent informational entity to some extent, proceeding from basic informational cycles to more cyclically and parallel complex ones. As we said in the previous paragraph, intelligent information must be such and such when conceptualizing it from a basic point of view.

We can build up a hierarchy of intelligent functions as they appear in a circular and spontaneous act of understanding. At the bottom is the capability of sensing, which is followed by functions of observing and perceiving of sensed informational entity. Then, on this basis, the conceiving generates concepts, which concern observation and perception of something. A mutual game between perception and conception entity delivers conclusions in the framework of consciousness, which is a kind of comprehension. On the top of intelligent informing metaphysicalism are cycles and parallels of understanding, which produce topical and detailed meaning of something and behave in their individual and common intelligent ways. Products of such informing are senses, observations, perceptions, conceptions, conclusions, consciousnesses, comprehensions, and meanings, which may embrace other essentially conscious or unconscious entities as there might be intention, significance, sense, identification, etc. within the informational game. All those components can be informationally determined, structured, organized, and formalized.

Now we can sketch some initial attempts of a strategy, by which concepts of intelligence could be informationally implemented. An intelligent informational entity ι informs and is informed in an intelligent metaphysical way and intelligence is always demonstrated in concern to a concrete subject, that is, concrete informational entity, say α. An intelligent entity phenomenalism is marked simply by

$$\iota(\alpha) \rightleftharpoons_{\mathrm{Def}} (\iota(\alpha) \models_{\mathrm{intelligent}}; \models_{\mathrm{intelligent}} \iota(\alpha)) \quad (26)$$

Intelligent entity ι is an informational function of entity α and notation $\iota(\alpha)$ expresses this functionality. The informational arising of intelligent entity ι depends on informing of entity α. Further, $\iota(\alpha)$ is a regular informational entity inclusive with its components, which are informing $\mathcal{I}_\iota(\alpha)$, counterinforming $\mathcal{C}_\iota(\alpha)$, counterinformational entity $\gamma_\iota(\alpha)$, informational embedding $\mathcal{E}_\iota(\alpha)$, and embedding informational entity $\varepsilon_\iota(\alpha)$. We can take a more concretely componential and circular formula, where the so-called intelligence pertains to a certain entity α, by

$$\iota(\alpha) \rightleftharpoons_{\mathrm{Def}} \left(\left(\begin{array}{l} \mathcal{S}^\iota_{\mathrm{sense}}(\alpha) \models \mathcal{S}^\iota_{\mathrm{sense}}(\alpha); \\ \mathcal{O}^\iota_{\mathrm{observe}}(\alpha) \models \mathcal{O}^\iota_{\mathrm{observe}}(\alpha); \\ \mathcal{P}^\iota_{\mathrm{perceive}}(\alpha) \models \mathcal{P}^\iota_{\mathrm{perceive}}(\alpha); \\ \mathcal{C}^\iota_{\mathrm{conceive}}(\alpha) \models \mathcal{C}^\iota_{\mathrm{conceive}}(\alpha); \\ \mathcal{C}^\iota_{\mathrm{conclude}}(\alpha) \models \mathcal{C}^\iota_{\mathrm{conclude}}(\alpha); \\ \mathcal{C}^\iota_{\mathrm{be\_conscious}}(\alpha) \models \mathcal{C}^\iota_{\mathrm{be\_conscious}}(\alpha); \\ \mathcal{C}^\iota_{\mathrm{comprehend}}(\alpha) \models \mathcal{C}^\iota_{\mathrm{comprehend}}(\alpha); \\ \mathcal{U}^\iota_{\mathrm{understand}}(\alpha) \models \mathcal{U}^\iota_{\mathrm{understand}}(\alpha) \end{array} \right) \subset \iota(\alpha) \right) \quad (27)$$

In this definition of informational inclusion, which pertains to intelligent entity $\iota(\alpha)$, its components have a superscript ι while a subscript expresses a more concise pragmatical property. In this definition, the so-called operands of informing, that is, $\mathcal{S}^\iota_{\mathrm{sense}}(\alpha)$, $\mathcal{O}^\iota_{\mathrm{observe}}(\alpha)$, $\mathcal{P}^\iota_{\mathrm{perceive}}(\alpha)$, $\mathcal{C}^\iota_{\mathrm{conceive}}(\alpha)$, $\mathcal{C}^\iota_{\mathrm{conclude}}(\alpha)$, $\mathcal{C}^\iota_{\mathrm{be\_conscious}}(\alpha)$, $\mathcal{C}^\iota_{\mathrm{comprehend}}(\alpha)$, and $\mathcal{U}^\iota_{\mathrm{understand}}(\alpha)$ produce (adequate) results in the form of informational entities, as $\sigma^\iota_{\mathrm{sensation}}(\alpha)$, $o^\iota_{\mathrm{observation}}(\alpha)$, $\pi^\iota_{\mathrm{perception}}(\alpha)$, $\gamma^\iota_{\mathrm{conception}}(\alpha)$, $\gamma^\iota_{\mathrm{conclusion}}(\alpha)$, $\gamma^\iota_{\mathrm{consciousness}}(\alpha)$, $\gamma^\iota_{\mathrm{comprehension}}(\alpha)$, and $\mu^\iota_{\mathrm{meaning}}(\alpha)$, respectively. Arbitrary pragmatical components of informing can be introduced at the informational formalization of philosophical texts[7].

Within a long metaphysical cycle concerning intelligent entity $\iota(\alpha)$, its informing-active pragmatical components in formula 27 can be structured metaphysically as follows:

$$\begin{array}{l} (((((((((\iota(\alpha) \models \mathcal{I}_\iota(\alpha)) \models \mathcal{S}^\iota_{\mathrm{sense}}(\alpha)) \\ \models \mathcal{O}^\iota_{\mathrm{observe}}(\alpha)) \models \quad \mathcal{P}^\iota_{\mathrm{perceive}}(\alpha)) \\ \models \mathcal{C}^\iota_{\mathrm{conceive}}(\alpha)) \models \quad \mathcal{C}^\iota_{\mathrm{conclude}}(\alpha)) \\ \models \mathcal{C}^\iota_{\mathrm{be\_conscious}}(\alpha)) \models \quad \mathcal{C}^\iota_{\mathrm{comprehend}}(\alpha)) \\ \models \mathcal{U}^\iota_{\mathrm{understand}}(\alpha)) \models \iota(\alpha) \end{array} \quad (28)$$

In this cycle we kept informing $\mathcal{I}_\iota(\alpha)$ of intelligent entity $\iota(\alpha)$ to be involved cyclically in informing of chosen pragmatical components. The basic pragmatical metaphysical cycles in formula 28 are

$$\begin{array}{l} (\mathcal{S}^\iota_{\mathrm{sense}}(\alpha) \models \sigma^\iota_{\mathrm{sensation}}(\alpha)) \models \\ \quad \mathcal{S}^\iota_{\mathrm{sense}}(\alpha); \\ (\mathcal{O}^\iota_{\mathrm{observe}}(\alpha) \models o^\iota_{\mathrm{observation}}(\alpha)) \models \\ \quad \mathcal{O}^\iota_{\mathrm{observe}}(\alpha); \\ (\mathcal{P}^\iota_{\mathrm{perceive}}(\alpha) \models \pi^\iota_{\mathrm{perception}}(\alpha)) \models \\ \quad \mathcal{P}^\iota_{\mathrm{perceive}}(\alpha); \\ (\mathcal{C}^\iota_{\mathrm{conceive}}(\alpha) \models \gamma^\iota_{\mathrm{conception}}(\alpha)) \models \\ \quad \mathcal{C}^\iota_{\mathrm{conceive}}(\alpha); \\ (\mathcal{C}^\iota_{\mathrm{conclude}}(\alpha) \models \gamma^\iota_{\mathrm{conclusion}}(\alpha)) \models \\ \quad \mathcal{C}^\iota_{\mathrm{conclude}}(\alpha); \\ (\mathcal{C}^\iota_{\mathrm{be\_conscious}}(\alpha) \models \gamma^\iota_{\mathrm{consciousness}}(\alpha)) \models \\ \quad \mathcal{C}^\iota_{\mathrm{be\_conscious}}(\alpha); \\ (\mathcal{C}^\iota_{\mathrm{comprehend}}(\alpha) \models \gamma^\iota_{\mathrm{comprehension}}(\alpha)) \models \\ \quad \mathcal{C}^\iota_{\mathrm{comprehend}}(\alpha); \\ (\mathcal{U}^\iota_{\mathrm{understand}}(\alpha) \models \mu^\iota_{\mathrm{meaning}}(\alpha)) \models \\ \quad \mathcal{U}^\iota_{\mathrm{understand}}(\alpha) \end{array} \quad (29)$$

and α marks something, for instance, a word, sentence, text paragraph, picture, etc. (in short, an informational entity), which will be in the process of understanding within intelligent entity $\iota(\alpha)$.

[7] Such an attempt was made at informational formalization [Železnikar 92d] of ¶ 31 (Being-there as Understanding) in [Heidegger 62].

As in the discussion pertaining to general informing of an entity, various short, medium-sized, and long metaphysical cycles for intelligent entity ι and its components can occur, making the intelligent structure as complex, perplexed, interwoven, cycled, parallel, spontaneous, etc. as possible. Additional components of intelligent informing can be considered in an intelligent informational game, for instance, $\iota^{\iota}_{\mathrm{intention}}(\alpha)$, $\sigma^{\iota}_{\mathrm{significance}}(\alpha)$, $\sigma^{\iota}_{\mathrm{sense}}(\alpha)$, etc. E.g., while intention may appear already on lower levels of an understanding process, significance can become a strategic role on a higher level of a recognizing process, etc. Also, the object of understanding α becomes meaningly more and more informationally identified. Different parallel metaphysical cycles inform the intelligent entity ι as a whole as well as its components. The reader can imagine how extremely complex schemes and scenarios, that is, informational formulas of understanding will come into existence.

An essential question is how one can choose informational components, which interact in an understanding process. That what is known and comes into the consciousness about understanding of something, concerns certainly the sensing of something. But, sensing of something, by which a sensation of something comes into existence, is a lower (or the lowest) function in an observing system. This system generates the observing information as a consequence of the observing analysis and synthesis, which certainly have some perceptional and conceptional characteristics. We see how the main metaphysical cycle of understanding begins to appear via sensing, producing sensation into observing, producing an observational information with elements of perceiving and conceiving of something. But, this is only the beginning of a cycle, which becomes more and more structured and complex in a componential way. Information of perception and conception is the result so far. After this initial situation, higher informational functions of understanding can enter into the informational game of understanding. Concluding is a highly informationally integrative entity, which takes the arisen and cyclically structured components of sensing, observing, perceiving and conceiving, and produces a sort of the first approximation of that, which we can call a conclusion about something. But, conclusion pertaining to the concluded of something is in no way the final result. First, it can be cycled informationally with the intention to obtain a more sophisticated information about something and, second, it can mediate the concluding results to hierarchically higher positioned entities as, for example, comprehending and understanding are. The process of comprehending has, for example, the function of an informational comprehension of an integral form of conclusion within the being-conscious.

Informational comprehending is an action of informational grasping, seizing, comprising, and including as a consequence of the previous informational consciousness. Everything, which in the cyclic process of understanding was produced in informational ways till this situation, has to be grasped anew and included into the consideration of comprehending. In this way, comprehending functions as an overtaking of something and coming up with the overtaken. This process of grasping reminds on sensing on a higher level and, certainly, conceiving. The result of comprehending is a comprehensional information, which now waits to be understood in a new way, when the action proceeds into new metaphysical cycle (middle-sized or long one) for the informational refinement of that, which was sensed, observed, perceived, conceived, concluded, conscious, comprehended, and understood up to this situation by an intelligent informational entity and its informing.

Understanding something is a function of apprehending the meaning of something, that is, grasping the idea, information, concept of something. On this level of informing, understanding is thoroughly acquainted or familiar with something, so it can deal with something properly when producing the meaning pertaining to something. When the result of this acquaintance is not satisfactory or not final (informationally still relevant), a further cycling of the understanding process is going on, producing a refined or more sophisticated meaning of something. Usually, understanding of something can never reach the point of being final or satisfactory, because the arisen meaning is a structure of an informational (linguistic, semantic, tautological) net with various unexplored possibilities. On this way of informing, especially through its cycling, understanding as an informationally acting entity can proceed to higher informational levels of knowledge, which concerns something. As the highest and all-embracing component within the metaphysical cycle, understanding is in the possession of faculties of lower metaphysical components that concern something. In this sense, together with participating metaphysical components, understanding masters the informing (communication, language, information), which expresses and impresses the meaning of something.

As an additional matter, metaphysical cycles pertaining to understanding are, in their nature, cyclic–parallel.

After this discussion, we can introduce a medium-sized and pragmatically conceptualized parallel–serial metaphysical cycle, which considers all the mentioned components and delivers, when analyzed, a set of evident serial cycles in parallel. At this occasion still modes of informing $\mathcal{I}^{\iota}_{\text{intend}}(\alpha)$, $\mathcal{S}^{\iota}_{\text{signify}}(\alpha)$, and $\mathcal{S}^{\iota}_{\text{make\_sense}}(\alpha)$ are introduced, which are self-explanatory. Thus, instead of formula 28 we have a parallel–serial metaphysical scheme for intelligent entity ι:

$$
\begin{array}{l}
((((\iota(\alpha) \models \mathcal{I}_{\iota}(\alpha)) \models \\[6pt]
\left(\left(\begin{array}{c} \mathcal{I}^{\iota}_{\text{intend}}(\alpha); \\ \mathcal{S}^{\iota}_{\text{signify}}(\alpha); \\ \mathcal{S}^{\iota}_{\text{make\_sense}}(\alpha) \end{array}\right)\right) \models \\[6pt]
\left(\left(\begin{array}{c} \iota^{\iota}_{\text{intention}}(\alpha); \\ \sigma^{\iota}_{\text{significance}}(\alpha); \\ \sigma^{\iota}_{\text{sense}}(\alpha) \end{array}\right)\right) \models \\[6pt]
\left(\left(\begin{array}{c} \mathcal{S}^{\iota}_{\text{sense}}(\alpha); \\ \mathcal{O}^{\iota}_{\text{observe}}(\alpha); \\ \mathcal{P}^{\iota}_{\text{perceive}}(\alpha); \\ \mathcal{C}^{\iota}_{\text{conceive}}(\alpha); \\ \mathcal{C}^{\iota}_{\text{conclude}}(\alpha); \\ \mathcal{C}^{\iota}_{\text{be\_conscious}}(\alpha); \\ \mathcal{C}^{\iota}_{\text{comprehend}}(\alpha); \\ \mathcal{U}^{\iota}_{\text{understand}}(\alpha) \end{array}\right)\right) \models \\[6pt]
\left(\left(\begin{array}{c} \sigma^{\iota}_{\text{sensation}}(\alpha); \\ o^{\iota}_{\text{observation}}(\alpha); \\ \pi^{\iota}_{\text{perception}}(\alpha); \\ \gamma^{\iota}_{\text{conception}}(\alpha); \\ \gamma^{\iota}_{\text{conclusion}}(\alpha); \\ \gamma^{\iota}_{\text{consciousness}}(\alpha); \\ \gamma^{\iota}_{\text{comprehension}}(\alpha); \\ \mu^{\iota}_{\text{meaning}}(\alpha) \end{array}\right)\right) \models \iota(\alpha)
\end{array}
\qquad (30)
$$

The last informational formula is a shortcut for a parallel system of serial formulas of all possible forms, that is,

$$
\begin{array}{l}
(((((\iota(\alpha) \models \mathcal{I}_{\iota}(\alpha)) \models \mathcal{X}(\alpha)) \models \xi(\alpha)) \\
\quad \models \mathcal{Y}(\alpha)) \models \eta(\alpha)) \models \iota(\alpha); \\[8pt]
\mathcal{X} \in \{\mathcal{I}^{\iota}_{\text{intend}}, \mathcal{S}^{\iota}_{\text{signify}}, \mathcal{S}^{\iota}_{\text{make\_sense}}\}; \\
\xi \in \{\iota^{\iota}_{\text{intention}}, \sigma^{\iota}_{\text{significance}}, \sigma^{\iota}_{\text{sense}}\}; \\
\mathcal{Y} \in \{\mathcal{S}^{\iota}_{\text{sense}}, \mathcal{O}^{\iota}_{\text{observe}}, \mathcal{P}^{\iota}_{\text{perceive}}, \\
\quad \mathcal{C}^{\iota}_{\text{conceive}}, \mathcal{C}^{\iota}_{\text{conclude}}, \mathcal{C}^{\iota}_{\text{be\_conscious}}, \\
\quad \mathcal{C}^{\iota}_{\text{comprehend}}, \mathcal{U}^{\iota}_{\text{understand}}\}; \\
\eta \in \{\sigma^{\iota}_{\text{sensation}}, o^{\iota}_{\text{observation}}, \pi^{\iota}_{\text{perception}}, \\
\quad \gamma^{\iota}_{\text{conception}}, \gamma^{\iota}_{\text{conclusion}}, \gamma^{\iota}_{\text{consciousness}}, \\
\quad \gamma^{\iota}_{\text{comprehension}}, \mu^{\iota}_{\text{meaning}}\}
\end{array}
\qquad (31)
$$

Formula 30 shows how a serial–parallel expression can be formally presented by a system of cycles, in which all possible informational permutations come into the foreground. We see how specific informational entities, belonging to specific entities of informing, can become informationally influenced not only by informing entities, but also by entities themselves and vice versa. In this way, each entity can informationally impact and can be informationally impacted by an occurring entity.

With formula 30, we can suggest a long metaphysical cycle, considering all components of intelligent entity $\iota(\alpha)$, which appear in the formula. We can "properly" permute the positions of components, e.g., within four parallel blocks in formula 30. The length of the long metaphysical cycles pertaining to $\iota(\alpha)$ is the number of successive informational operators in a long cycle, that is, $\ell_{\text{meta}}(\iota(\alpha)) = 22$. However, 22 is not a final value, because some of the appearing components can be additionally decomposed (for example, $\mathcal{I}_{\iota}(\alpha)$).

Let us show only one of the long metaphysical cycles of intelligent entity $\iota(\alpha)$, within which we consider the informational pairs, as follows: intending and intention; signifying and significance; making sense and sense; sensing and sensation; observing and observation; perceiving and perception; conceiving and conception; concluding and conclusion; being–conscious and consciousness; comprehending and comprehension; and understanding and meaning concerning something (that is, entity α). Thus, one of the examples (possibilities) of long metaphysical cycles is the following one:

$$\begin{array}{l}(((((((((((((((((((((((((\iota(\alpha) \models \mathcal{I}_{\iota}(\alpha)) \models \\ \mathcal{I}^{\iota}_{\mathrm{intend}}(\alpha)) \models \iota^{\iota}_{\mathrm{intention}}(\alpha)) \models \\ \mathcal{S}^{\iota}_{\mathrm{signify}}(\alpha)) \models \sigma^{\iota}_{\mathrm{significance}}(\alpha)) \models \\ \mathcal{S}^{\iota}_{\mathrm{make\_sense}}(\alpha)) \models \sigma^{\iota}_{\mathrm{sense}}(\alpha)) \models \\ \mathcal{S}^{\iota}_{\mathrm{sense}}(\alpha)) \models \sigma^{\iota}_{\mathrm{sensation}}(\alpha)) \models \\ \mathcal{O}^{\iota}_{\mathrm{observe}}(\alpha)) \models o^{\iota}_{\mathrm{observation}}(\alpha)) \models \\ \mathcal{P}^{\iota}_{\mathrm{perceive}}(\alpha)) \models \pi^{\iota}_{\mathrm{perception}}(\alpha)) \models \\ \mathcal{C}^{\iota}_{\mathrm{conceive}}(\alpha)) \models \gamma^{\iota}_{\mathrm{conception}}(\alpha)) \models \\ \mathcal{C}^{\iota}_{\mathrm{conclude}}(\alpha)) \models \gamma^{\iota}_{\mathrm{conclusion}}(\alpha)) \models \\ \mathcal{C}^{\iota}_{\mathrm{be\_conscious}}(\alpha)) \models \gamma^{\iota}_{\mathrm{consciousness}}(\alpha)) \models \\ \mathcal{C}^{\iota}_{\mathrm{comprehend}}(\alpha)) \models \gamma^{\iota}_{\mathrm{comprehension}}(\alpha)) \models \\ \mathcal{U}^{\iota}_{\mathrm{understand}}(\alpha)) \models \mu^{\iota}_{\mathrm{meaning}}(\alpha)) \models \\ \iota(\alpha)\end{array} \tag{32}$$

The reader can imagine how many other long metaphysical cycles concerning intelligent entity $\iota(\alpha)$ are possible and how each of them represents an alternative case of metaphysical informing. In this variety of syntactic possibilities of long metaphysical formulas, which can inform in parallel (simultaneously, cooperatively), participating components can arise in various informational ways, constituting the spontaneity and cyclicity of an intelligent informational entity.

9 Conclusion

Metaphysicalism is a concept of inner informing of entities. By metaphysicalism, the informational arising of entities in scopes of their informational contents is performed in an informational constructive way. Metaphysicalism is a basic principle (called *informatio tertia*) and is an essential circular–spontaneous property of an informing entity. Entities within an informational system (e.g. informational machine) can obtain metaphysical support by the system, but can also have their own metaphysical "mechanisms". In this sense, metaphysicalism is a constructive approach in a conceptual and a machine–oriented way.

Metaphysicalism may not be paralleled by metaphysics as *philosophia prima* or supernatural power. The term expresses that, which concerns the informational emerging of an entity's possibilities and lies in different presentations (in German, Vorstellungen) of one and the same thing at different observing places with different informational possibilities. Metaphysicalism is an inner creative power of an informing entity. Under such circumstances, it is never completely foreseeable in advance[8] for it can be impacted interiorly and exteriorly in respect to the informing entity.

Within the conclusion pertaining to metaphysicalism, we have to say which are the possibilities of its technological implementation. Informationally supported computing system is the first step to such implementation. Such a system must deliver a basic informational support to informing entities and, within its operating system, there must be informational dictionaries [Dictionary 90a] and, for instance, knowledge archives [Knowledge 92], in which informational entities needed for informational arising of an informing entity can be searched.

The speed of an informational machine implementation and machine's functional (informational) performance will dramatically depend on the sophistication of machine metaphysicalism, that is, on basic informing, counterinforming and informational embedding mechanisms by which informational machine as an informing entity by itself will systemically support the informing of occurring informational entities (operands and formulas, informational programs, informational bases) [Železnikar 92c].

In some former essays [Železnikar 92d] it was shown how semantically reasonably structured written texts, that is, words, word groups, idioms, sentences, paragraphs, sections, etc. inform and are informed in various circular and parallel ways, which are mutually interwoven. It became evident that an informational interpretation (understanding) of a text surpasses the conventional, human style of linguistic comprehension, which is on a global level serial, atomistically structured (consciously particularized), also non–parallel, and not dynamically structured in the way of a system of text and its parts interpreting informational

[8] *To foresee in advance* may not be equaled with *to predict*. By informational terms, we can put, for example, $\mathcal{F}_{\mathrm{foresee\_in\_advance}} \rightleftharpoons ((\mathcal{S}_{\mathrm{see}} \models_{\mathrm{in\_advance}}) \models_{\mathrm{in\_advance}})$ while, for the other case, there is, $\mathcal{P}_{\mathrm{predict}} \rightleftharpoons (\mathcal{S}_{\mathrm{say}} \models_{\mathrm{before}})$. A further difference is in the semantical nature of both cases and concerns, in the first case *to see before in advance* or, also, *to see in advance, in advance* and, in the second case, *to say in advance*. As we understand, the seeing and saying might be completely different informational phenomena. In the common speech, it may be inappropriate to say *to foresee in advance, in advance*. Within the informational discourse, this case can become a matter of informational externalism and internalism.

formulas. Of course, besides unforeseeable pragmatical approaches of a text recognition, an informational system (machine) of informing entities (operands and formulas) can consider various traditional and scientifically organized methods and structures of text interpretation, cognition, informational processing, etc. But, all that may not suffice for a dynamically understood written text, which informs highly parallel in an openly structured informational realm, that is, in the world, where information and informational understanding arise at every time.

Thus, let us close the discourse on metaphysicalism of informing with the following rumination. Perceiving of the physical is metaphysical. Components as sensing, observing, perceiving, conceiving, concluding, being-conscious, comprehending and, at the end, understanding[9] are characteristic metaphysical informational entities. But, metaphysicalism does not mean that these components do not possess their own physical, biological, chemical, genetic, neuronal, social, etc. backgrounds of matter, energy, information (structure, organization), which enable the appearance of "metaphysical" phenomenalism. Or, said in another way ([Husserl 00] II/2, p. 244): *Und sie existieren dabei keineswegs bloß phänomenal und intentional (als erscheinende und bloß vermeinte Inhalte), sondern wirklich.*

References

[Derrida 67] J. Derrida: *La Voix et le Phénomène*, Presses Universitaires de France, Paris, 1967.

[Heidegger 62] M. Heidegger: *Being and Time*, Harper & Row, New York, 1962.

[Husserl 00] E. Husserl: *Logische Untersuchungen*, Max Niemeyer Verlag, Tübingen, 1900-1901.

[Železnikar 90a] A.P. Železnikar: Understanding as Information, Informatica, Vol. 14 (1990), No.3, pp. 8-30.

[9] An initial philosophy of understanding as an informational entity and a first attempt to its informational formalization is presented in [Železnikar 90a] (Part One: A General Philosophy and Theory of Understanding as Information) and in [Železnikar 90b] (Part Two: A Formal Theory of Understanding as Information).

[Železnikar 90b] A.P. Železnikar: Understanding as Information II, Informatica, Vol. 14 (1990), No.4, pp. 5-30.

[Železnikar 92a] A.P. Železnikar: *Towards an Informational Language*, Cybernetica, Vol. 35 (1992), No. 2, pp. 139-158.

[Železnikar 92b] A.P. Železnikar: *Basic Informational Axioms*, Informatica, Vol. 16 (1992), No. 3, pp. 1-16.

[Železnikar 92c] A.P. Železnikar: *An Introduction to Informational Machine*, Informatica, Vol. 16 (1992), No. 4, pp. 8-29.

[Železnikar 92d] A.P. Železnikar: *An Informational Approach of Being-there as Understanding* (in three parts), Informatica, Vol. 16 (1992), No.1, pp. 9-26; No.2, pp. 29-58; and No. 3, pp. 64-75.

[Dictionary 90a] *An Overview of the EDR Electronic Dictionaries*, TR-024, Japan Electronic Dictionary Research Institute, Tokyo, April, 1990.

[Dictionary 90b] *English Word Dictionary*, TR-026, Japan Electronic Dictionary Research Institute, Tokyo, April, 1990.

[Knowledge 92] *A Plan for the Knowledge Archives Project*, The Economic Research Institute, Japan Society for the Promotion of Machine Industry, and Systems Research & Development Institute of Japan, Tokyo, March, 1992.

MISSION AND RESEARCH REPORTS

1 Introduction

The two project frameworks—one the Japanese concerning a plan for the *Knowledge Archives Project* and the other pertaining to the research program of the *Center for the Study of Language and Information* at Stanford University—have much in common. Both concern knowledge of language and information in an extended view of understanding and both tend towards new, the so-called informational theories, methodologies, programs and architectures. The careful reader will observe the crucial perplexity of both undertakings—the first one in the form of a global project of knowledge understanding and technology, and the second one as a set of theoretically, methodologically, and experimentally oriented projects concerning knowledge in linguistic and informational sense.

In columns entitled *Mission and Research Reports* we shall present the most significant research and technology projects running in the world at present time. In this issue of *Informatica* the first parts of both project frameworks are presented. The second parts will be published as sequels in the next issue of *Informatica*. Thus, the reader can compare the new research and technology trends within both frameworks leading into the realm of informational, where the informational becomes an acting, meaning (semantic), and intelligent environment.

2 A Plan for the Knowledge Archives Project I

2.1 Introduction

Again, as in the case of the Fifth Generation Computer Systems Project in 1980's, the Japanese research and development initiative has surprised the professional research and technology communities over the globe by its grandiose plan to unite (accumulate, standardize, collect, define, aggregate, transform, arise, etc.) various kinds of knowledge (languages, sciences, technologies or, lastly, cultures of the world, in general) in a powerful, effective, dynamic, emerging, significant, intentional, etc., that is, intelligently operating knowledge informational system. Three eminent Japanese institutes have launched this plan through a special publication[1] and, as it seems, by an advanced organizational arrangement.

The plan sets some new standards in the terminology (definition) of knowledge related informational entities. Knowledge archives is a very large-scale knowledge base considering an automated acquisition and collection of knowledge, stored systematically; supporting the creation of new knowledge; and translating and transmitting knowledge. These technologies will impact knowledge itself, that is, its determination, dynamic structure, and intelligent organization. The knowledge archives shifts the perspective of knowledge processing, which was concerned with classic development of computers, programming languages, and high level technologies. The Knowledge Archives is not only a mechanism for massive information and electronic library. Some relevant domains of knowledge archives are natural language processing, knowledge engineering technology, multimedia, next generation databases (e.g. deductive and object oriented), and software engineering, by which knowledge will be developed by itself. The last view—developing knowledge by knowledge—is a characteristic shift to the new informational wave that can be marked as *informational*. This shifting into the informational will require new linguistic verbal and formal styles of expression and new ways of dynamic semantics, meaning, intention, and significance—to follow new paradigms of informing, counterinforming, and embedding of knowledge.

Computers as tools have to be re-evaluated from the standpoint of the user through the new approaches to technology, humanities, and social sciences in understanding of information, knowledge, and their mechanisms. In this way, knowledge archives should become the most universal of all application systems, e.g. the most universal expert system. The other view of the project is to be a common basis of knowledge for interna-

[1] *A Plan for the Knowledge Archives Project*, The Economic Research Institute; Japan Society for the Promotion of Machine Industry; Systems Research & Development Institute of Japan, Tokyo, March, 1992, 1-78.

tional and interdisciplinary exchange in research and technology communities, where extremely effective and cooperative international relationships can be built up. This view concerns, for instance, subcultural techniques of each country's language, which have to come together and bring mutual study and learning of different language techniques.

2.2 Technical Background

Within the project plan, knowledge is looked on as information carrying meaning, that is, considering semantics of informational expression. Today information processing technology focuses its attention on the views of form and syntax (e.g. "formulas" in natural and artificial or formal languages). This platform is generally accepted in different information processing systems. But, it becomes also evident that conventional techniques of form and syntax can not produce useful and efficient results. The task is to challenge the contents and semantics of information, that is, to proceed into higher, parallel, interwoven levels of form, syntax, and meaning pertaining to something as information. In this respect, the word *knowledge* remains novel and concerns the informational realm of that what is commonly (culturally) recognized as the factical, true, believable, faithful, etc. Knowledge as a term is novel because of possibilities of its informational emerging in every specific case, where meaning of something can be extended in different informational ways, in a straightforward and circular direction.

"How to handle the contents and semantics of information as knowledge?" is the basic question and, to this one, the second is "How the generated knowledge of something is information, which carries its own contents and semantics?" This process of knowledge generation on the linguistic level can proceed further and further and is in the last consequence always tautological. The AI boom lacked this point of view: it limited the range of informational objects and tried to search meaning in depth under certain limitations. The project previews to deal with informational entities in a broader range and with the meaning in a shallow range. Said literally: "The Knowledge Archives is the technology which can grasp the meaning in a shallow but as wide as possible range and which can broaden and generalize the application area as much as possible."

By all means, two of the essential questions are "Did AI research arrived to a dead end because of having dealt with just a small amount of knowledge or, self-critically, with toy problems?" and "Was AI only a phase of infantilism, which ended in proverbial dead lock on the way to a new informational research and technology perspective?" The new informational research and technology perspective accentuates large-scale information processing, where both the amount of knowledge and the capacity to process knowledge are enlarged. E.g. the Fifth Generation Computer Systems Project is a representative of the latter view. In this respect, new informational conceptualism and technologies are required to acquire and store massive knowledge automatically and as efficiently as possible. New attention has to be spend to, for instance, massive parallel computing technology known as memory based reasoning; neural network computing for large-scale symbol manipulation; and, last but not least, theories and methods handling knowledge as a dynamic informational entity, which arises in every case and in every moment in the social and physical environment of a living being.

The diverse knowledge representation media is the next inevitable demand. Knowledge has to be understood by humans using computers, but in this function, "a part" of knowledge must be understood by computers, which support the functions of human understanding of knowledge. In human culture, media representing knowledge are natural, artificial and graphical languages, diversified images, sounds, and other, yet not identified human sensory and mental information. Multimedia technology is on the way to the highly and flexibly fused demands for informationally mixed and complex informational presentation.

The next question is, in which form knowledge could be presented to human intelligence and, in this respect, the necessity for research based on ecology of knowledge arises. Types of media and types of knowledge will determine the presentation to humans. Knowledge is a regular informational entity[2], which arises, varies, diversifies, and comes in many forms and it is being generated, edited, transformed, stored, retrieved, and

[2] A.P. Železnikar, *Metaphysicalism of Informing*, Informatica, Vol. 17 (1993), No. 1, 65–80.

transmitted. Simple theories of the informational and insignificant experiences will not suffice for the phenomenalism of highly diversified knowledge as informational entity. Knowledge as informational phenomenon has to be clarified together with proper research environment and development of technologies, considering the ecology and dynamics of knowledge.

One of the next task is to standardize knowledge representation media. New types of logic and logic programming (e.g., informational logic and informational language[3]) have to be considered. The next generation data base is necessary for storing the large amounts of knowledge and retrieve it on request.

In this context some Japanese national projects have to be mentioned. The Fifth Generation Computer Systems Project will proceed into the research and development of logic programming, parallel computers for knowledge processing, and application systems for verification. The Japan Electronic Research Institute (EDR) electronic dictionary project implements natural language processing technology. At this project, a natural language is the kernel language of knowledge representation media and a part of the dictionary is a large-scale knowledge base of lexical knowledge. The multilanguage electronic dictionary is provided for translation of sentences. Further, machine translation systems are under development at software houses and computer industry of Japan.

2.3 Social Background

Knowledge Archives Project is the first project ever, which tackles knowledge itself completely and, in this way, meets a wide range of the social needs. A knowledge (culture) base usable among nations (civilizations) is certainly a desire. Human intelligence and knowledge is a socially interwoven phenomenon, which should be tackled together with broadened informational theory, humanities, social sciences, and computer technology. That could restructure the joined information industry.

Knowledge is a turbulent information and the ignoring of its dynamism can lead to the collapse of social systems (e.g., communism, socialism). Today, knowledge must be exchanged on a global level. Unraveling the mystery (revealing) of intelligence and actualizing it as informational phenomenalism belongs to the most hopeful, dreamful and exciting challenges in progressing toward a new human civilization.

Overproduction of information in the information age is becoming a tremendous burden for individuals, who must decide, by help of knowledge, what to accept informationally and what to leave without perception, that is, what to ignore. New terms must be coined to characterize entities like redundant, misused, insignificant, damaging, misunderstood, polluted, unworthy information, etc.

Current research and technology projects must consider the demands of an international society, long-term industrial needs, and the mode of technology. Knowledge Archives Project is a project for projects, by which other projects (e.g. the fifth generation, electronic dictionary, language translation, international projects, etc.) will be covered and coordinated. The technology, which will make computers more intelligent is going beyond small-scale projects conducted by business ventures. The whole industry is to be made more intelligent and the high-quality information in nothing else than a higher intelligence. Knowledge may be mass of textual material (static intelligence) or it may be informational entities floating around in minds of experts.

2.4 Functions and System Structure of the Project

Let us describe in short the knowledge internal structure. Knowledge representation media are natural languages, formal languages, picture languages, images and sounds. Natural languages are Japanese and various foreign languages. Formal languages are algebraic formulas, logical formulas, and programming languages. Picture languages enable the representation of diagrams, tables, architectural design drawings, electronic circuits diagrams, music scores, etc. Images include static images, dynamic images and animations. Sounds are speech, music, and sounds in general. These knowledge representation media are appropriately combined and all media are treated equally.

"Knowledge documents" is appropriately represented, observed, and objectively analyzed knowledge. Knowledge documents act as a processing

[3] A.P. Železnikar, *Towards an Informational Language*, Cybernetica, Vol. 35 (1992), No. 2, 139-158.

and understanding system for humans and computers. Some documents are merely data, some can be understood syntactically, and some can be understood semantically. By the advance of technology, the understanding of knowledge documents will shift more toward computers. Knowledge documents will be normalized especially in the domain of natural languages.

The massive amount of knowledge documents will be provided by learning and self-organization. For instance, a knowledge object is an expression of one aspect of a knowledge document. Objects are mutually related, attributed, inferential, etc. Functions and structure will clarify the final form of the knowledge base by an arising improving. Basic units of knowledge documents are stories (a type of text), paragraphs, sentences, words. Basic types of knowledge objects are surface objects (knowledge documents) and meaning of them are semantic objects. Semantic objects are of a kind, which is a superior (semantic) object, etc. There are relations and rules for composition of surface objects and semantic objects (relations between semantic objects, between surface objects and semantic objects, and between knowledge objects).

Mechanisms of inference include deduction and inferring of semantic objects, semantic relations (equivalence, super and subinclusion, etc.). Knowledge base will learn by using its own inference mechanism and knowledge will be self-organized. However, the knowledge base will not be completed within the duration of the project (until year 2000).

The knowledge in the base will be determined in the following ways:

(i) Characteristics of knowledge for various fields will be determined carefully.

(ii) "Summary knowledge documents" will be attached to knowledge documents, defining relationships (e.g. synonymous, antonymous).

(iii) At creating of word knowledge documents, the expanded EDR electronic dictionaries will be used.

(iv) For every field, the narrative (story) knowledge will be included in the knowledge base. This type of knowledge is the most general for semantic recognition of knowledge documents.

"How to chose the diverse fields of knowledge?" is another problem. Documents can be divided in three types: general texts; knowledge representation and programs; and data.

Texts in natural languages are narratives, newspaper articles, scientific and technical papers, patent documents, legal documents and precedents, and manuals.

Knowledge and programs in formal languages are knowledge documents in system knowledge representation languages, knowledge documents in constraint logic programming languages, knowledge documents using expert systems shells, and knowledge documents in general-purpose programming languages.

Data will have its own database language for extracting knowledge from massive amount of data, e.g. from MITI database.

The functions of Knowledge Archives are the following: knowledge storage and retrieval function; knowledge collection and acquisition function; knowledge creation and utilization function; and knowledge translation and communication function.

The basic functions of the knowledge base are: knowledge representation language and executing environment. A function defines knowledge objects, their attributes, relations between objects and attributes, inference rules, and rules for learning. This function stores the defined relations, rules, etc. and responds efficiently to demands. An expanded mechanism of deductive and object-oriented database seems appropriate.

Knowledge being common to various fields will be stored and retrieved systematically. Knowledge is organized hierarchically in respect to the document structure: words, sentences, paragraphs, texts, and stories (narratives). The hierarchy is described by natural languages. Knowledge documents can be structured only by natural languages, knowledge representation languages, and programming languages. Graphics, images, and sounds will be handled as paragraphs and organized as sentences and words.

Knowledge extraction is a function, which extracts and stores knowledge automatically and effectively. The extraction means the forming of knowledge documents in a condensed contents form (content knowledge documents, which generally correspond to index words, key words, etc.).

Knowledge extraction on the level of words is a kind of automatic index extraction. *Sentence content knowledge documents* correspond to summary sentences and *Paragraph content knowledge documents* correspond to summary texts or excerpts or abstracts. Thus, knowledge extraction is an automatic abstraction.

Knowledge documents will be created effectively from undocumented knowledge and stored in knowledge bases. Only knowledge required will be selected from the enormous amount of knowledge.

A function for the translation of knowledge documents for translation into various languages will be available. Existing translation methods will be improved and expanded. Knowledge will be communicated among knowledge archives located in different countries. Standardization and other forms of agreement will be important, for instance, standardization of protocols, unification of specifications, etc. The protocol language will be one of the representation media, so that specification of various ontologies will be unified.

Knowledge archives is a system structure, is a linked system or network of four types of servers and clients. *Knowledge workbench* is a highly efficient work station possessing the necessary functions for operations related to knowledge extraction, creation, transformation, etc. *Knowledge base center server* stores commonly usable knowledge in large amounts and answers retrieval inquiries coming from different places of the globe. *Knowledge base site server* is located at each research site. It emphatically accumulates knowledge characteristics to each site and responds to retrieval inquiries from other sites. Knowledge communication server is an intelligent gateway to different countries, other projects, and research institutes. Small-scaled servers of this type are set up at each site. The described system structure of knowledge archives is a knowledge archives network.

[To be continued]
—Summarized by *A.P. Železnikar*

3 Center for the Study of Language and Information

Based on the CSLI 1991 Annual Report

3.1 Overview and Background

CSLI is an independent laboratory at Stanford University devoted to research in the emerging science of information, computing, and cognition. This new science had its origins in the late 1970s as computer scientists, linguists, logicians, philosophers, psychologists, and artificial intelligence researchers, seeking solutions to problems in their own disciplines, turned to one another for help.

The problems that brought them together were rooted in issues that crossed the traditional boundaries among the disciplines. A shared interest in how agents, whether biological or artificial, acquire, process, and convey information forced researchers in these different fields to confront many of the same issues concerning communication, perception, action, reasoning, and representation. Many researchers saw that problems targeted by their own discipline were linked to solutions in the others, and, as interaction developed, began to view the common issues as defining a science in its own right.

In 1985, the National Science Foundation sponsored a meeting of representatives from the allied disciplines to investigate the nature and potential of the new science.[4] They found that in the interest of progress, researchers in the individual disciplines had idealized along dimensions that made sense in light of the questions central to their respective fields, but that these idealizations were complementary when viewed across disciplines. For example, mathematical logic had traditionally ignored the resource limitations of information-processing agents, while computer science had restricted itself to data bases describable within simple fragments of first-order logic. The conference participants concluded that the new science had reached a critical point. For future progress, it was now necessary to replace traditional idealizations with the more realistic assumptions that

[4]See Report of Workshop on Information and Representation, ed. Barbara H. Partee, Stanley Peters, and Richmond Thomason. Stanford, Calif.: CSLI, 1985.

would emerge from serious interdisciplinary collaboration.

CSLI's foremost goal is to provide an interdisciplinary setting for research in this new science. All of the research projects currently under way benefit from significant interaction across disciplines. Many are collaborative projects involving senior researchers from different fields. The interaction has born fruit in both expected and unexpected ways. For example, research on speech-act theory, originally the province of philosophers and linguists studying human communication in natural language, has influenced several CSLI projects whose goal is the design of better computer languages, architectures, and protocols. Conversely, attention to computational efficiency and tractability has affected the basic structure of syntactic and semantic theories pursued at CSLI, as well as the models of human rationality, communication, and learning under development in various of the projects.

Besides being interdisciplinary, CSLI is an interinstitutional laboratory. Founded in 1983 by researchers from Stanford, SRI International, and Xerox PARC, CSLI has since its inception promoted collaboration between industrial laboratories and academic departments. In recent years, this collaboration has expanded to include researchers from additional universities, laboratories, and companies, both within the immediate geographical vicinity and around the world. CSLI's Industrial Affiliates Program currently includes fourteen corporate members, many of which send researchers to participate in research projects on site.

This interinstitutional collaboration has had an equally important effect on the nature and progress of many CSLI projects. It has informed the more theoretical projects with an awareness of current technology and potential applications, while providing the more applied projects with access to the latest theoretical advances.

CSLI researchers are pursuing a wide variety of topics, including robotics design, planning and reasoning, speech recognition, machine-aided translation, language acquisition, text understanding, computer languages, and software design strategies, among others. Each project focuses on one or more aspects of the use of information by natural and artificial agents. Roughly half deal with languages, vehicles by which information is communicated between agents. These in turn divide into those concerned with natural (human) languages, and those concerned with computer languages. The other half deal with a variety of questions involving the acquisition and manipulation of information: how agents acquire and use information to guide action; what information-processing architectures are best suited to various tasks; how representational format affects information processing and human comprehension; and so forth.

The Annual Report is intended as a record of the researchers and projects associated with CSLI during 1991, as well as the seminars and colloquia held during the year. In presenting reports from the individual projects, we divide them into the following rough groupings, according to the project's principal concerns:

- Intelligent Agents
- Human/Computer Interaction
- Computer Languages and Architectures
- Natural Language
- Foundations

Readers who would like more detailed information about specific research projects should consult the list of publications and references for that project, or contact the publications department at CSLI.

3.2 Intelligent Agents

Information gets its value as a guide to intelligent action. Whether we are deciding when and where to market a new product, or when and where to invest our money, or simply when and where to cross the street, information is the crucial ingredient in our deliberations. It is what allows us to navigate through a dynamic and changing environment.

Intelligent agents, biological or artificial, must be equipped to gather information about their environment and to bring that information to bear on their behavior. Understanding how humans do this, and how to build artificial agents capable of doing it as well, is no simple task. The projects described in this section are devoted to various aspects of this problem. Their goals range from the highly theoretical to the practical, from developing general models of rational agency to investigating techniques for improving the real-time performance of autonomous robots.

3.2.1 Autonomous Agents[5]

Goals. The goal of the Autonomous Agents project is to build agents, such as robots, capable of learned and planned behavior in dynamic environments.

Significance. The control of real-time, embedded systems has presented severe challenges to computer scientists, yet control theorists have long been able to use continuous feedback as a technique for reactive control of processes. The major significance of our work to date is that it imports some powerful control-theoretic ideas into the core of computer science—permitting programs whose control structure is radically different from that of conventional programs.

3.2.2 Integrating Perception and Reasoning[6]

Goals. We are interested in the role that reasoning plays in perception. More specifically, we are concerned with the boundary between low-level interpretation of sensory input and higher-level cognitive processes such as planning and belief derivation. It is obvious that sensory interpretation is useful as input to high-level cognition; it is less clear how and to what extent information flows in the other direction. This is the area we are exploring, using computational models of perception and cognition.

Significance. Artificial intelligence (AI) has developed powerful computational models of both sensory interpretation and cognitive skills such as planning. There has been relatively little interaction between these areas, although recently the importance of inferential and planning capabilities in visual interpretation has been recognized and given the name "active vision." Starting from the point of view that perception is a form of inference, and further that explicit symbolic reasoning is an integral part of perception, we are trying to integrate representation and theorem-proving techniques into the perceptual process. The application areas include abstract reasoning about perception, map-making for mobile robots, and perception-based text editors.

3.2.3 Rational Agency (RATAG)[7]

Goals. The aim of the Rational Agency (RATAG) project is to provide models and theories of situated, resource-bounded, intelligent action. The scope of the research covers rational activities of human beings, including the use of language, and also the activities of artificial agents with a variety of rational and deliberative abilities.

Significance. The models developed by RATAG should facilitate the development of artificial agents capable of using information to guide their action in real-life environments, as well as enhance our understanding of how biological agents bring information to bear on action.

[4] Project participants: Nils J. Nilsson, Stanford Computer Science (project leader); Andrew Kosoresow, Stanford Computer Science student; plus occasional undergraduates. The project is funded primarily by NASA under grant NCC2-494. The project monitor is Monte Zweben of NASA-Ames.

[5] Project participants: Kurt Konolige, SRI International (project leader); Karen Myers, SRI International and Stanford Computer Science postdoctoral fellow; Daniela Musto, Italian National Research Council (CNR); Chad Walters, Stanford Symbolic Systems student. This project was funded by the Office of Naval Research under contract N00014-89-C-0095 and by CSLI internal research funds.

[6] Project participants: John Perry, Stanford Philosophy (project leader); Michael Bratman, Stanford Philosophy; Philip Cohen, SRI International and Stanford Linguistics consulting professor; Güven Güzeldere, Stanford Symbolic Systems student; Felix Ingrand, SRI International; David Israel, SRI International and Stanford Philosophy consulting professor; Kurt Konolige, SRI International; Hector Levesque, Toronto Computer Science; Betsy Macken, CSLI; Karen Myers, SRI International and Stanford Computer Science postdoctoral fellow; Robert C. Moore, SRI International and Stanford Computer Science consulting professor; Eunok Paek, Stanford Computer Science student; John Perry, Stanford Philosophy; Martha Pollack, SRI International; Yoav Shoham, Stanford Computer Science; Syun Tutiya, Chiba Philosophy; Len Wesley, SRI International. The project was funded by a grant from the System Development Foundation.

[7] Project participants: Grigori Mints, Stanford Philosophy, Stanford Computer Science, and Institute for Cybernetics, Estonian Academy of Sciences (project leader); Tanel Tammet, Institute for Cybernetics, Estonian Academy of Sciences; Vadim Basylev, Kazan University, and Institute for Mathematical Studies in the Social Sciences (IMSSS). Funding for this project was provided by the Institute for Cybernetics, by IMSSS, and by CSLI

3.2.4 Resolution Procedures for Reasoning[8]

Goals. Many projects in artificial intelligence (AI) and computer science presuppose a logic engine capable of doing more or less sophisticated reasoning. This project involves research in problem-oriented logical systems of the resolution type. In particular, the following research areas are to be addressed: (1) improvement and application of the existing systems for classical and intensional logics; (2) construction of new systems for linear and belief logics.

Significance. The system we design should be useful in developing specialized reasoning tools and effective tools for teaching nonclassical logics.

3.2.5 Robotic Machine Learning of Natural Language[9]

Goals. The goal of the Robotic Machine Learning of Natural Language project is to develop a natural-language learning interface for a robotic system that can be taught to execute, in an appropriate environment, commands like "Put the screw left of the nut into the hole between the washer and the black nut!"

Significance. The system we are developing will contribute to the machine-learning theory of natural language. Our approach contrasts with others in that it is semantically rather than syntactically based.

3.2.6 Situated Automata (SA)[10]

Goals. The Situated Automata (SA) project is engaged in a long-term program of research aimed at developing theoretical foundations and design methods for sophisticated embedded computer systems. A significant portion of the work is devoted to real-time machine perception and to the integration of results in working systems to control physical systems such as robots.

Significance. New theories being developed in the SA project promise dramatic improvements in real-time robotics applications, since much of the computational cost of current approaches can be shown to be eliminable in principle. For example, all costs involved in representing and deriving consequences of invariant facts can be eliminated by exploiting the embedding circumstances. The project has also developed practical symbolic languages and other tools to aid in the development of high-performance, parallel artificial intelligence (AI) systems in the situated-automata framework.

3.3 Human/Computer Interaction

The computer is the tool of the information age. It allows us to gather, store, and transform massive amounts of information in ways unimagined thirty years ago. But the development of the tool has outpaced our ability to put it to productive use. One bottleneck is the communication channel that links the computer and its human users. Often, the form in which information is most easily managed by computers is not the form most natural, or most readily understood, by their users. To us, a picture can be worth a thousand words, but to the computer, a line of Pascal is worth a thousand pictures. To bridge such gaps, we need techniques for converting information into a variety of forms and modalities, and a clear understanding of both the computational and psychological effects of the conversion.

This is just one dimension of the problem of human/computer interaction. Large computerized systems, such as those used in banking and air-traffic control, must interact with a heterogeneous group of organizations, workers, and other computers. The difficulty of tailoring such a system to a complex work setting is an order of magnitude greater than the design of single-user software. New tools and methods are needed to aid in the construction of such systems.

The projects described in this section are con-

---

internal research funds.

[8] Project participants: Patrick Suppes, Stanford Philosophy and Institute for Mathematical Studies in the Social Sciences (IMSSS) (project leader); Michael Böttner, Max-Planck-Institute for Psycholinguistics; Lin Liang, Stanford Mechanical Engineering student; Weile Zhu, University of Electronic Science and Technology, China, Electrical Engineering; Michael Donio, Paris student. Funding for the project was provided by IMSSS and by the Deutsche Forschungsgemeinschaft (DFG).

[9] Project participants: Stanley J. Rosenschein, Teleos Research and Stanford Computer Science consulting professor (project leader); David Chapman, Teleos Research; Neil Hunt, Teleos Research; Leslie Pack Kaelbling, Stanford Computer Science student and Teleos Research; H. Keith Nishihara, Teleos Research; Lambert Wixson, University of Rochester student; Nathan Wilson, Teleos Research. Funding for this project was provided by NASA and DARPA.

cerned with these and related topics. They range in scope from projects that address conceptual issues in software design to a project whose goal is to introduce computerized tools into the teaching of cognitive psychology.

3.3.1 Experiments in Cognitive Psychology[11]

Goals. Ideally, courses in cognitive psychology should have a laboratory to introduce students to experimental techniques and results, to demonstrate the classic findings instead of telling about them. In practice, implementing a laboratory has been expensive in terms of equipment and personnel. Recent advances in microcomputers have rendered developing such a laboratory possible. The goals of this project were to develop a laboratory to accompany a course in cognitive psychology that would be: (a) comprehensive, including a broad range of basic and classic phenomena; (b) self-running and self-explanatory, obviating the need for personnel; (c) provide feedback to the student-subjects on their own performance, the group's performance, and typical performance.

Significance. As a whole, our package demonstrates most of the classic techniques and phenomena used in experimental psychology. The programs can be used as a demonstration laboratory for beginning and intermediate level students, and can be used as an experiment generator by more advanced students. That is, the standard experiments can be revised to create new experiments, and then collect data on them. The package was awarded a Distinguished Software Award by NCRIPTAL/EDUCOM in 1990.

3.3.2 Hyperproof[12]

Goals. The Hyperproof project has two goals. The first is to develop a mathematical theory of heterogeneous reasoning—reasoning in which information is presented in more than one modality or representational form. The second is to construct a computer application that allows the user to reason using both propositions and diagrams, one common form of heterogeneous reasoning.

Significance. We believe that the Hyperproof project could have far-reaching significance in a variety of fields.

(1) We anticipate that the first version of the Hyperproof program will be an effective tool for teaching analytical reasoning skills. This is an important pedagogical goal that is not achieved by standard techniques of teaching logic.

(2) Hyperproof II, the successor program, could provide the prototype for a powerful general-purpose reasoning tool, useful in a wide variety of problem-solving contexts such as scheduling and planning.

(3) The Hyperproof architecture should be useful in developing special-purpose reasoning tools, such as intelligent CAD/CAM systems, in which both graphical representations and linguistically stated constraints must be satisfied.

(4) Techniques used in the Hyperproof system should be transferable to more general problems of information management in multimodal data bases (e.g., data bases containing pictures, graphs, and text).

(5) From a psychological perspective, the Hyperproof inference system is suggestive of an alternative view of human reasoning, distinct from both the purely deductive view associated, for example, with Jerry Fodor (Rutgers Psychology), and the purely model-building view associated with Philip Johnson-Laird (Princeton Psychology).

[10] Project participants: Barbara Tversky, Stanford Psychology (project leader); Approximately fifty undergraduates in Symbolics Systems and Psychology participated in creating the first versions of the experimental modules. Two Symbolic Systems interns worked summers, Jim White and Paul Glauthier. Glauthier continued the work, funded by the Dean's Innovation Fund.

[11] Project participants: John Etchemendy, Stanford Philosophy (project leader); Gerrard Allwein, Indiana Computer Science student; Jon Barwise, Indiana Philosophy, Mathematics, and Computer Science; Douglas Felt, Stanford Academic Information Resources; Mark Greaves, Stanford Philosophy student; Michael Lenz, Stanford Symbolic Systems and Computer Science student; Kenneth Norman, Stanford Symbolic Systems student; Sun-Joo Shin, Notre Dame Philosophy. This project has received funding from the System Development Foundation, the Office of Academic Information Resources at Stanford, the Center for Innovative Computer Applications at Indiana University, and NATO.

[12] Project participants: Betsy Macken, CSLI (project leader); Cathy Haas, Stanford Special Language Program. This project was funded by CSLI internal research funds.

3.3.3 Hyperproof and American Sign Language[13]

Goals. We are investigating the potential of Hyperproof for teaching reasoning skills to deaf college students, especially those whose preferred language is American Sign Language (ASL). In parallel, we seek to provide an information-based characterization of ASL inspired by research related to multi-representational computing and heterogeneous reasoning.

Significance. Our characterization of ASL began as a way of developing two related conjectures, which, if correct, would have clear implications for the deaf community and for researchers interested in heterogeneous reasoning.

The first is that Hyperproof, which teaches formal reasoning based on both visual and sentential forms of information, holds special potential for teaching reasoning skills to deaf students and for allowing us to make explicit reasoning skills they already possess but are not currently recognized in traditional logic courses. The second is the complement that theories of heterogeneous reasoning will benefit from an understanding of the crucial features of ASL.

In addition, our characterization would have implications for the teaching of ASL to English speakers and the teaching of English to those whose first language is ASL. It also may be of interest to psychologists interested in spatial perception and spatial mental models.

3.3.4 Language Shapes[14]

Goals. Traditionally a very strong distinction has been drawn between texts and pictures. Applied to computer representations, however, this distinction is less clear. We are exploring this distinction on a number of levels, from ways of textually describing shape, to the development of languages with shaped terms. The long-range aim is to develop a theory of the semantics of shapes—how shapes can carry meaning, both in themselves and by virtue of their relationships to others—and to understand how to apply it to knowledge representation in artificial intelligence (AI).

Significance. Various aspects of this work have practical significance. The shape-description work with Leyton has direct application to medical image perception, and is part of a developing field of automatic visual perception mechanisms with many obvious applications.

More generally, however, the analysis of the semantic foundations of shape has implications for our understanding of cognition. For example, there has been an ongoing debate in psychology between those who believe that mental images are somehow pictorial in nature and those who wish to map them into something essentially propositional. This work suggests that both may be right, or at any rate the differences between them may be less significant than has been supposed.

3.3.5 People, Computers, and Design (PCD)[15]

Goals. The goal of our research is to develop the theories of communication and interaction that underlie the design of computer systems for cooperative work. Our focus is on developing the theoretical and practical background needed to incorporate human contextual elements into the design and analysis of computer systems. The theoretical emphasis is on the development of a "language/action perspective" in which current and potential software and hardware devices are analyzed and designed in the context of their embedding in work and communicative structures. The language/action perspective grew out of earlier work in artificial intelligence, but it shifts the focus of attention away from the mental and the individual, to the social activity by which people generate the space of cooperative actions in which they work—and to the technology that is the medium for those actions.

[13]Project participants: Patrick J. Hayes, Xerox PARC and Stanford Computer Science consulting professor (project leader); Michael Leyton, Rutgers Psychology. This project was funded by CSLI internal research funds.

[14]Project participants: Terry Winograd, Stanford Computer Science (project leader); Eric Babinet, Stanford Computer Science student; Clarisse Sieckenius de Souza, Pontificia Universidade Catolica, Brazil, Information Sciences; Rafael Furst, Stanford Computer Science student; Brad Hartfield, Stanford Computer Science; Annie Kreyenberg, Stanford Computer Science student; Rafael Pardo, University of Madrid; Alice Wu, Stanford Computer Science student; Mountaz Zizi, Paris VI Computer Science student. Funding Sources: NSF Grant CDA9018898, Directorate of Computer and Information Science and Engineering, and CSLI internal research funds.

Significance. The project is developing theoretical models and methodologies that will be central to the design of computer systems that include interactions with people of any kind. The People, Computers, and Design (PCD) project is also funded by an NSF grant to develop a series of innovative courses on human/computer interaction.

3.3.6 Spatial Mental Models[16]

Goals. The goals of the Spatial Mental Models project are to characterize people's mental representations of space for a variety of spatial layouts, and to study how they are acquired and how information is accessed from them. We are interested in situations acquired solely from description as well as those acquired from visual interaction. A secondary goal is to characterize linguistic descriptions of space.

Significance. From this work, we will learn how people form mental representations of space from language, what spatial properties are reflected in those mental representations, and how people use language to describe space. The present results indicate that the spatial mental representations formed are not like the internalized perceptions proposed by most theories of mental imagery. Nevertheless, they are spatial in the sense that they contain information about space and have biases that reflect general conceptions of the spatial world. Spatial knowledge is particularly accessible to investigation because all humans have it and communicate about it. Mental representations of, and communication about, space can serve as a model for representation and communication about more abstract domains of knowledge.

3.3.7 Syntax WorkBench[17]

Goals. Teaching introductory syntax is primarily a matter of teaching the basic ideas of hypothesis construction and testing, rather than the details of any particular theory that has been developed within the field. Syntax WorkBench is a project intended to put in the students' hands a tool that allows them to focus on analyzing data and testing different ideas about it, thereby improving this aspect of the class. Our goal is to replace the standard "textbook" method of teaching with an interactive environment provided by this program, preconstructed exercises within it, and an accompanying text.

Significance. The program represents a precise implementation of one particular set of analytic assumptions (transformational grammar) in a way that makes exploration of classic problems in linguistics accessible to the student in a very short time. We are unaware of any comparable program that is publicly available.

3.3.8 Tarski's World Study (TWS)[18]

Goals. The Tarski's World Study (TWS) project has three goals: to study human interaction with the Tarski's World first-order logic microworld, to design a cognitive model that accounts for human reasoning within Tarski's World, and to model the means by which learning takes place in this environment.

Significance. The project has both practical and theoretical significance. Practically, it is important to understand processes of learning in open-ended exploratory environments. Theoretically, we need a better understanding of situated reasoning processes and mental representations involved in human/computer interactions.

3.4 Computer Languages and Architectures

The first computers were used to compute—to carry out numerical calculations. Because of this, most theories of computation, as well as most

[15] Project participants: Barbara Tversky, Stanford Psychology (project leader); David Bryant, Northeastern Psychology; Nancy Franklin, New York Stony Brook Psychology; Caren Jones, Stanford Psychology student; Joe Kim, Stanford Psychology student; Jay Leahy, Stanford Psychology student; Scott Mainwaring, Stanford Psychology student; Deborah Tatar, Stanford Psychology student; Holly Taylor, Stanford Psychology student. The research was funded by the Air Force Office of Scientific Research, Air Force Systems Command, USAF under grant AFOSR 89-0076, and by CSLI internal research funds.

[16] Project participants: Peter Sells, Stanford Linguistics (project leader); Michael Calcagno, Stanford Symbolic Systems student; David Whelan, Stanford Symbolic Systems student. This project has received funding from the Dean's Innovation Fund, matched by CSLI internal research funds.

[17] Project participants: James G. Greeno, Stanford Education (project leader); Eric Cooper, Stanford Education student; John Etchemendy, Stanford Philosophy; Hiroshi Kato, NEC Corporation. This project was supported by CSLI internal research funds.

programming languages, were based on an image of the computer as a programmable calculator. But since then, computers have become full-fledged information-processing devices. We find them controlling the brakes in our cars, regulating load distribution on the power grid, tracking and guiding missiles, running elevators. They still compute, but computation in the old sense forms only a small kernel of what they do, an intervening step in the process of acquiring, manipulating, and putting to use information about the world in which they are embedded.

Computers have become engines of information, and this change requires a new understanding of computation, new programming languages and paradigms, new techniques for ensuring the adequacy of a program. The projects described in this section are devoted to this cluster of topics. They are exploring a wide variety of ideas—from linear logic to speech-act theory to situation semantics—and applying them to issues affecting the current and future state of computation.

3.4.1 Agent-oriented Programming (AOP)[19]

Goals. The Agent-oriented Programming (AOP) project aims to develop a new programming paradigm that exploits the computational advantages of attributing mental state to machines, which employs ideas from speech act theory, and which relies on computational versions of social laws.

Significance. Agent-oriented programming provides a new approach to programming distributed systems, emphasizing explicit representation of time, beliefs, and commitments, which includes speech-act-like communicative commands. Our work included the theoretical investigation of mental state, interpreter design and implementation, and applications such as traffic control and robotics.

3.4.2 AMALA[20]

Goals. The goal of the AMALA project is to find better ways to design, understand, and reason about computer programs that must interact with the world external to the program. This will lead to improved programming languages, better frameworks for specifying their semantics, and more effective techniques for reasoning about particular interactive programs.

Significance. In addition to its direct relevance to the design and theory of programming languages, the AMALA project has also led to a better appreciation of just what kinds of information facilitate understanding and practical reasoning about programs that interact with the external world. This is expected to lead to improved programming methodologies helpful to programmers using existing imperative programming languages.

Furthermore, our new view of programming language semantics brings it more into line with the notion of semantics in natural-language understanding, mathematical logic, and knowledge representation. This approach promises to provide a more comprehensive and more realistic view of complete computational agents interacting with the world in accordance with the semantic interpretation of their components.

3.4.3 Connectionist Models of Linguistic Information Processing (CMLIP)[21]

Goals. The goal of the Connectionist Models of Linguistic Information Processing (CMLIP) project is to develop connectionist models of various aspects of linguistic information processing—including various modalities of language production and comprehension/recognition of such utterances. These models are being built around the

[18] Project participants: Yoav Shoham, Stanford Computer Science (project leader); Alvaro del Val, Stanford Philosophy student; Nita Goyal, Stanford Computer Science student; Ron Kohavi, Stanford Computer Science student; Fangzhen Lin, Stanford Computer Science research associate; Eyal Mozes, Stanford Computer Science student; Anton Schwartz, Stanford Computer Science student; Moshe Tennenholtz, Stanford Computer Science postdoctoral fellow; Becky Thomas, Stanford Computer Science student.

[19] Project participants: Michael Dixon, Stanford Computer Science student and Xerox PARC (project leader); Michael Ashley, Indiana Computer Science student; Jim des Rivières, Xerox PARC. This project was supported by Xerox Corporation and by a grant from the System Development Foundation.

[20] Project participants: David E. Rumelhart, Stanford Psychology (project leader); Ben Martin, Stanford Psychology student. The project was partly funded by DARPA and Ricoh Corporation.

two strengths of connectionist systems—namely, their ability to learn and their ability to solve large constraint-satisfaction problems with many "soft" constraints. We have been developing models in the areas of morphophonology, lexical representation, syntax, acquisition of semantic representations, and some of the pragmatic aspects of conversation and communication. This project began in 1989.

Significance. The work on connectionist computation has a large potential range of payoffs. On the one hand, since brains are inherently parallel computational devices, it may be possible to develop parallel computational systems modeled on the algorithms and methods found effective in real biological systems. Moreover, some of the algorithms have proven to be as good as, or better than, existing methods in such areas as automatic speech recognition, recognition of cursive handwriting and other image processing tasks. A further understanding of how to apply connectionist systems to the general language-processing system will allow for the development of a more total parallel language-processing system. At another level, the development of these systems should feed back to biology and allow the development of models of actual brain systems, which will improve our understanding of how relatively large-scale brain systems function to produce the kinds of intelligent behavior we observe in biological systems.

3.4.4 Declarative Programming[22]

Goals. The main goal of the Declarative Programming project is to extend declarative programming beyond its present static realizations to cover many dynamic and concurrent applications that at present are considered beyond the scope of declarative programming. In particular, the project aims to bring concurrent programming and object-oriented programming within the fold of declarative programming, and also to unify different programming paradigms in a declarative way.

Significance. Extending the scope of declarative programming will make a wider range of tasks easier by supporting high-level reasoning about the problem to be solved, and by allowing the direct formal expression of such reasoning in a declarative program. An important added benefit is the greater ease with which parallelism can be exploited once a program has been expressed at a high level of abstraction.

3.4.5 Embedded Computation (EC)[23]

Goals. The long-term goal of the Embedded Computation (EC) project is to develop a foundational theory of computation as a physical process that both interacts with and represents its environment. During the course of development, intermediate results and insights are applied to a variety of projects—in computer science, to the design of new computer languages and flexible architectures; in linguistics, to the development of a situated language for interaction between people and machines; and in cognitive science, to the understanding and development of systems of perception and cognition.

Significance. The theoretical significance of this project is twofold. In terms of computer science, it will provide a rigorous framework for understanding interactive programs—an analysis as deep as current theories of purely functional and numerical computation. In terms of cognitive science, it rests on the view that computational processes constitute one variety of representational or semantic phenomenon. Asking about the nature of computation, on this view, involves issues in semantics, representation, and intentionality—

[21] Project participants: José Meseguer, SRI International (project leader); Patrick Lincoln, Stanford Computer Science student and SRI International Computer Science Laboratory; Narciso Martí-Oliet, Madrid Computer Science student and SRI International; Timothy Winkler, SRI International. Funding Sources: Office of Naval Research Contracts N00014-88-C-0618 and N00014-90-C-0210, and NSF grant CCR-8707155.

[22] Project participants: Brian Cantwell Smith, Xerox PARC and Stanford Philosophy consulting professor (project leader); Kathleen Akins, Xerox PARC and Illinois Philosophy and Neuroscience Cognitive Science Group; John Batali, San Diego Cognitive Science; Jim des Rivières, Xerox PARC; Michael Dixon, Xerox PARC and Computer Science student; Vinod Goel, Berkeley Special Program in Cognitive Science student; Patrick J. Hayes, Xerox PARC and Stanford Computer Science consulting professor; Gregor Kiczales, Xerox PARC; John Lamping, Xerox PARC; Geoffrey Nunberg, Xerox PARC and Stanford Linguistics consulting professor; Susan Stucky, Institute for Research on Learning; John Woodfill, Stanford Computer Science student; Ramin Zabih, Stanford Computer Science student. This research was funded by Xerox Corporation and by a grant from the System Development Foundation.

all problems of critical concern to the cognitive sciences as a whole.

The practical significance of this project (and of a group of collaborative projects being carried on at Xerox PARC) stems from the development of a series of specific architectures. Three that are presently under way are:

(a) AMALA, a new programming language for interactive systems based on a reworked conception of semantics, reference, and state (see below, tenet (1), and also its independent project report).

(b) Intrigue, a metalevel compiler for Scheme (in the tradition of 3-lisp, CLOS, and other reflective systems) that, by exploiting internal semantic relationships, provides an elegant way in which a program can be tuned for high performance without compromising the simplicity or modularity of its original functional conception. The basic insight is to use metalevel techniques to provide orthogonal control of the different aspects of a computation. As a first example, Intrigue is focusing on performance and implementation, one step towards the long-term goal of making intended physical realization an integral part of program design.

(c) Pidgin, a designed language for human/computer interaction, based not only on the tenets of the EC project, but also on many of the ideas of efficiency, context-dependence, indexicality, and discourse structure that are being studied throughout CSLI.

3.4.6 Logic and Language for Computation[24]

Goals. This project is concerned with logics for reasoning about computation and with languages for describing computational processes. There are two complementary long-term goals. One is to devise logics for stating and proving properties of programs. In this area, we are primarily interested in simple logics with limited expressiveness. Such logics may not provide a basis for full "program verification," but may lead to tractable methods for preventing some common kinds of programmer error. The second goal is to develop programming languages that more accurately express distinctions of interest while hiding irrelevant concerns. We aim to develop foundations for systematic analysis of current programming-language features and provide a logical basis for current and future programming tools.

Significance. Type theory, the general study of type systems, provides a basis for reasoning about programs and suggests extensions to current programming languages. Type systems in current use prevent simple but common forms of programmer error. In addition, type systems can guarantee the correctness of certain implementation decisions. In a field that has lacked a general paradigm and standard terminology, type theory has had significant influence.

The primary shortcomings of current type theory are that there is little application to such important topics as computational efficiency, imperative programming, and concurrency. The recent development of linear logic raises the possibility that current type-theoretic methods could be extended to algorithms for automatic complexity analysis and new perspectives on concurrency.

A number of researchers are currently investigating connections between linear logic and Petri nets (a basic model of concurrent program execution), logic programming and nonmonotonic logic. Our current work aims to generalize type systems to include simple logics for reasoning about the resource requirements of a computation, and other important program characteristics not presently covered by type systems.

[23]Project participants: John Mitchell, Stanford Computer Science (project leader); Patrick Lincoln, Stanford Computer Science student and SRI International Computer Science Laboratory; Brian Howard, Stanford Computer Science student; Dinesh Katiyar, Stanford Computer Science student. Funding Sources: NSF Presidential Young Investigator Award to Mitchell, with matching funds from Digital Equipment Corporation, Powell Foundation, Xerox Corporation, a gift from the Mitsubishi Corporation, Wallace F. and Lucille M. Davis Faculty Scholarship. Lincoln was supported by an AT&T Bell Laboratories Doctoral Scholarship and SRI International internal funding.

[24]Project participants: Stanley Peters, Stanford Linguistics (project leader); Per-Kristian Halvorsen, Xerox PARC and Stanford Linguistics consulting professor; Hideyuki Nakashima, Electrotechnical Laboratories (ETL) Cognitive Science Section; Hinrich Schütze, Stanford Linguistics student; Hiroyuki Suzuki, Matsushita Electric Industrial Co., Ltd. This project was supported by funds from the System Development Foundation.

3.4.7 PROSIT[25]

Goals. The PROSIT project is designing and implementing a programming/knowledge-representation language based on situation theory. The language is intended to improve facilities for computing with context-dependent information—as is essential in natural-language processing, problems involving cooperation among multiple agents, and interpretation of sensory data.

Significance. Putting situated inference under the control of programmers could significantly increase the power of logic-programming languages, whose primitive notion is inference. The PROSIT project moves in this direction by replacing Horn clauses with situation-theoretic constraints and allowing concrete reference to situations that respect these constraints, thereby implementing a useful notion of situated inference.

Hypothetical reasoning can then be programmed using inference in a hypothesized situation. Reasoning mixing private knowledge with shared or common knowledge is carried out using PROSIT's facilities for treating situation hierarchies; efficiency is gained by virtue of inheritance within the hierarchy, often replacing inference within a single situation. An emerging style of situated programming provides new tools for solving the computational problems that are the goals motivating the PROSIT project.

3.5 Natural Language

Language is the most distinctively human vehicle of communication. Using it, we exchange information in remarkably efficient bursts of sound. But while sound technology—techniques for reproducing, storing, and transmitting auditory signals—is highly advanced, language technology is in its infancy. A human's ability to recognize an acoustic signal as a piece of language, to parse it into its component words and phrases, and then to access its information content, is a remarkable feat. Engineering artificial systems that can accomplish almost any step in this process has proven a formidable task.

Language technology encompasses a broad range of research efforts, from speech recognition and synthesis to text understanding and machine translation. Many early forays into these areas followed a similar pattern. Limited success was achieved by applying seat-of-the-pants techniques, but then a sudden barrier was encountered. More often than not, such barriers arise from an impoverished theoretical understanding of the phenomenon in question.

Just as mechanized flight eluded inventors until aeronautical theory was sufficiently developed, so too must efforts in the area of language technology be supported by a strong theoretical base. The projects described in this section address some of the immediate problems in language technology, as well as the broader theoretical issues that underlie them.

3.5.1 Computational Models of Representational Signals (CMORS)[26]

Goals. The goal of the Computational Models of Representational Signals (CM ORS) project is to provide a scientific base for the machine perception of natural, spoken communication and visual symbols.

Significance. The theories and computational technologies developed will result in improved understanding of language and its communicative function.

3.5.2 Dialogue as a Joint Activity[27]

Goals. The goal of the Dialogue as a Joint Activity project is to develop a formal theory of dialogue that takes seriously the jointly interactive nature of the dialogue process, predicts discourse behavior, and can serve as the basis for computational dialogue partners.

Significance. Understanding the structure of dialogue will help increase the naturalness of lin-

[25] Project participants: Meg Withgott, Xerox PARC (project leader); Francine R. Chen, Xerox PARC; Ronald M. Kaplan, Xerox PARC and Stanford Linguistics consulting professor; Julian Kupiec, Xerox PARC; Ramin Zabih, Stanford Computer Science student. This project was funded by Xerox PARC and by a grant from the System Development Foundation.

[26] Project participants: Philip Cohen, SRI International and Stanford Linguistics consulting professor (project leader); Hector Levesque, Toronto Computer Science. Funding for this project was provided by ATR Interpreting Telephony Research Laboratories and a grant from the System Development Corporation.

guistic interaction between humans and computers, as well as improve the performance of machine translation and interpretation systems.

3.5.3 Grammatical Processing[28]

Goals. The goal of the Grammatical Processing project is to discover computational principles for the efficient and psycholinguistically plausible interpretation of grammatical formalisms that are expressive enough to characterize natural languages.

Significance. Rich grammatical formalisms have been developed to permit the informational dependencies in natural-language syntactic and semantic relations to be expressed in an accurate and insightful way. The descriptive power of these formalisms is purchased at a computational cost, however, in that there are no known algorithms for parsing and generating grammatical sentences in less than worst-case exponential time. This worst-case behavior is not just a theoretical possibility, since exponential explosion is frequently observed with the programs that are typically written to process these formalisms. This undermines the hypothesis that such grammatical systems can play a direct role in practical applications or plausible psycholinguistic models. The Grammatical Processing project is investigating alternative methods of computing with complex constraint-based formalisms that may show average-case polynomial behavior when applied to real sentences and real grammars. Such algorithms would resolve the apparent incompatibility of linguistically motivated and computationally tractable grammatical descriptions.

3.5.4 Knowledge-based Text Understanding (TACITUS)[29]

Goals. The long-range goal of the Knowledge-based Text Understanding (TACITUS) project is to investigate the problem of how knowledge is used in the interpretation of discourse. This very large problem decomposes into two more very large problems: (1) how should our commonsense knowledge of the world be encoded? and (2) how should this knowledge be manipulated by inference processes to interpret discourse? In the present phase of the project, we are building computational models of text understanding, based on abductive inference. This is done primarily by implementing a system to analyze and extract the appropriate information from naturally generated texts of some complexity.

Significance. Text-understanding systems have a substantial economic potential for helping organizations collect information and route that information to the people who need it.

3.5.5 Language Use in Interactional Settings[30]

Goals. This project is concerned with the study of discourse as a joint activity among coordinating participants.

Significance. This work contributes to three research efforts. One is to develop theories of planning, intentions, and acting. A second is to bring in the notion of collective action: how two or more people accomplish goals by acting in ensemble. And a third is to expand the notions of information beyond the traditional concern with words and other symbols.

3.5.6 Lexical Acquisition in Language Development (LALD)[31]

Goals. The goal of this project is to account for

[27] Project participants: Ronald M. Kaplan, Xerox PARC and Stanford Linguistics consulting professor (project leader); John Maxwell, Xerox PARC. This project was funded by Xerox PARC.

[28] Project participants: Jerry Hobbs, SRI International (project leader); Douglas Appelt, SRI International; John Bear, SRI International; Megumi Kameyama, Stanford Computer Science student; Mark Stickel, SRI International; Mabry Tyson, SRI International. This research was funded by the Defense Advanced Research Projects Agency under Office of Naval Research contract N00014–85–C–0013.

[29] Project participants: Herbert H. Clark, Stanford Psychology (project leader); Bridget Bly, Stanford Psychology student; Jean Fox Tree, Stanford Psychology student; Elizabeth Wade, Stanford Psychology student. Funding for this project, which was provided by the Max-Planck-Institute for Psycholinguistics, Nijmegen, The Netherlands, supported both Clark and Fox Tree for the academic year.

[30] Project participants: Eve V. Clark, Stanford Linguistics (project leader); Trisha Svaib, Stanford Linguistics student; Ruth A. Berman, Tel Aviv Linguistics. This research was supported in part by the Max-Planck-Gesellschaft during a sabbatical year spent at the Max-Planck-Institute for Psycholinguistics, Nijmegen, The Netherlands.

how children build up their lexicon during acquisition.

Significance. This issue is central to any theory of language acquisition since it requires an account of what children know about the lexicon and word-formation, what constraints they may or must observe as they learn new words and word-forms, and how they put their knowledge to use in talking and understanding.

3.5.7 Machine-aided Translation (XL)[32]

Goals. The primary goal of the Machine-aided Translation (XL) project is to develop a theoretical and computational foundation for research in computational linguistics in general, but with particular emphasis on machine translation. Its scope thus includes any representation and processing issues that might be relevant to systems that perform parsing, generation, morphological analysis, and so forth. One of the project's specific goals is to provide computational tools for investigating the Resolution Problem (see report of the Resolution project). Another specific goal is to build a prototype machine-translation (MT) system that explores techniques for context-dependent negotiation.

Significance. The computational tools we are developing are intended to play an important supporting role for future computational linguistic research at CSLI, including providing new tools for investigating the Resolution Problem. The theoretical and conceptual research conducted by this project, for example, the development of notions like "negotiation," is likely to make an important contribution both to the theory of translation and to MT applications.

3.5.8 Parallel Constraint Grammar (PCG)[33]

Goals . The goal of the Parallel Constraint Grammar (PCG) project is to explore the organization of linguistic structure into parallel systems of constraints—structural, functional, semantic, and prosodic.

Significance. The parallel constraint grammar architecture has computational and theoretical advantages over current alternative designs. These include advantages of *order-free computation* through the removal of information dependencies that require ordering of computational processes, *extensibility* through the ease of adding structures and correspondence principles representing new domains, and *conceptual clarity* through increasing understanding of interactions of different domains that have been cleanly factored.

[31] Project participants: Martin Kay, Xerox PARC and Stanford Linguistics (project leader); Michael Calcagno, Stanford Symbolic Systems student; Suk-Jin Chang, Seoul National Linguistics; Dan Fish, Stanford Symbolic Systems student; Jerry Hobbs, SRI International; Masayo Iida, Stanford Linguistics student and Hewlett-Packard Laboratories; Michio Isoda, WACOM Co., Ltd.; Megumi Kameyama, Stanford Computer Science student; Makoto Kanazawa, Stanford Linguistics student; Charles Lee, Stanford Linguistics student; Yo Matsumoto, Stanford Linguistics student; Hideo Miyoshi, Sharp Corporation; Hiroshi Nakagawa, Yokohama National Linguistics; Naohiko Noguchi, Matsushita Electric Industrial Co., Ltd.; Ryo Ochitani, Fujitsu Laboratories Ltd.; Stanley Peters, Stanford Linguistics; Livia Polanyi, Rice Linguistics; Alan Ramaley, Stanford Symbolic Systems student; Amnon Ribak, Tel Aviv Linguistics student; Ivan A. Sag, Stanford Linguistics; Hinrich Schütze, Stanford Linguistics student; Peter Sells, Stanford Linguistics; Hadar Shem-Tov, Stanford Linguistics student; Hidetosi Sirai, Chukyo Computer and Cognitive Sciences; Yoshihiro Ueda, ATR Interpreting Telephony Research Laboratories; Chris Weyand, Stanford Symbolic Systems student; Shuichi Yatabe, Stanford Linguistics student. Funding for this project was provided by CSLI internal research funds and Center for East Asian Studies (CEAS) funds.

[32] Project participants: Joan Bresnan, Stanford Linguistics and Xerox PARC (project leader); Alex Alsina, Stanford Linguistics student; Jan Borota, Charles Linguistics; Miriam Butt, Stanford Linguistics student; Mary Dalrymple, Xerox PARC; Ki-Sun Hong, Stanford Linguistics student; Chu-Ren Huang, Institute of History and Philology, Academia Sinica, and Tsing Hua University; Michio Isoda, WACOM Co., Ltd.; Smita Joshi, Stanford Linguistics student; Jonni Kanerva, Indiana Linguistics; Ronald M. Kaplan, Xerox PARC and Stanford Linguistics consulting professor; Paul Kroeger, Stanford Linguistics student; Tibor Laczkó, Kossuth English; Malillo Machobane, National University of Lesotho Linguistics; Christopher Manning, Stanford Linguistics student; Yo Matsumoto, Stanford Linguistics student; John Maxwell, Xerox PARC; Sam Mchombo, Berkeley Linguistics; Lioba Moshi, Georgia Linguistics; Tetsuro Nishino, Tokyo Denki Information Sciences; Gillian Ramchand, Stanford Linguistics student; Peter Sells, Stanford Linguistics; Whitney Tabor, Stanford Linguistics student; Annie Zaenen, Xerox PARC and Stanford Linguistics consulting professor. The project was funded by a grant from the System Development Foundation and by NSF grant BNS-8919880.

[33] Project participants: Paul Kiparsky, Stanford Linguistics (project leader); Young-Mee Yu Cho, Stanford Asian Languages student; Jennifer Cole, Stanford Linguistics stu-

3.5.9 Phonetics and Phonology (P&P)[34]

Goals. The Phonetics and Phonology (P&P) project focuses on the representation and interpretation of spoken language, with the goal of characterizing more explicitly the information transmitted in the acoustic signal.

Significance. Since utterances are physically located acoustic signals, parsed by listeners into discrete symbolic units that serve as the basis for higher-level linguistic processing, a theory of the mapping between signal and symbol is an essential part of any theory of the relation between utterances and the information they convey.

3.5.10 The Resolution Project[35]

Goals. The goal of this research project is to develop a basic scientific theory of the "resolution problem," which may be defined as the problem of how diverse kinds of information—linguistic, contextual, and encyclopedic—are integrated in real-time language use; the problem of how communication can proceed rapidly and efficiently (or at all) in light of the fact that interpretation is radically underdetermined by language.

Significance. The resolution problem is perhaps the most important issue in the theory of language processing and arguably the most significant obstacle facing the development of useful natural-language processing technology.

3.5.11 Spoken Language in Interpreted Telephone Dialogues[36]

Goals. The goal of the Spoken Language in Interpreted Telephone Dialogues project is to analyze dialogue and performance characteristics of Japanese/English telephone conversations conducted through an interpreter.

Significance. This study allowed us to provide preliminary target requirements for automatic telephony interpretation systems, that is, systems in which speech input in one language is automatically translated into speech output in another.

3.5.12 Theory of Aitiational Frames (TAF)[37]

Goals. The Theory of Aitiational Frames (TAF) project is developing a word semantics, or lexical semantics that cuts across syntactic theories, and endeavors to formulate a theory with ties to formal semantics as practiced by philosophers and generative grammar as practiced by linguists. Our aim is to provide a conceptual framework for analyzing our understanding of nouns, verbs, and other parts of speech that can be used by work in psycholinguistics, such as that of S. Pinker (MIT), and cognitive psychologists such as R. Case (Stanford CERAS).

Significance. This theory offers a uniform treatment of the semantic representation of words that can be used in predicate expressions. It aims to unify insights from philosophy of language, linguistics, and psychology. It can represent a

---

dent; Eunjoo Han, Stanford Linguistics student; Michael Inman, Stanford Linguistics student; Goh Kawai, Stanford Linguistics student and SRI International; András Kornai, Stanford Linguistics student; William Leben, Stanford Linguistics; William Poser, Stanford Linguistics; Linda Uyechi, Stanford Linguistics student. Funding for the project was provided by a grant from the System Development Foundation.

[34] Project participants: Ivan A. Sag, Stanford Linguistics (project leader); Herbert H. Clark, Stanford Psychology; Lewis Creary, Hewlett-Packard Laboratories; Anthony Davis, Stanford Linguistics student; Jane Edwards, Berkeley Psychology; Aaron Halpern, Stanford Linguistics student; Koiti Hasida, Institute for New Generation Computer Technology (ICOT); Jerry Hobbs, SRI International; Martin Kay, Xerox PARC and Stanford Linguistics; David Magerman, Stanford Computer Science student; Daniel Morrow, Stanford Psychology and School of Medicine; David E. Rumelhart, Stanford Psychology; Hinrich Schütze, Stanford Linguistics student; Whitney Tabor, Stanford Linguistics student; Michael Tanenhaus, Rochester Psychology. The project was funded by CSLI internal research funds and a contract from the Boeing Corporation.

[35] Project participants: Philip Cohen, SRI International and Stanford Linguistics consulting professor (project leader); Sharon L. Oviatt, SRI International; Ann Podlozny, SRI International. Funding for this project was provided by ATR Interpreting Telephony Research Laboratories.

[36] Project participants: Julius Moravcsik, Stanford Philosophy (project leader); Jane Aronson, Stanford Philosophy student; Michael Calcagno, Stanford Symbolic Systems student; Paul Glauthier, Stanford Symbolic Systems student; Stephen Irish, Stanford Philosophy student; Martin McDermott, Stanford Philosophy student; Ray Ravaglia, Stanford Philosophy student; Katherine Uum, Stanford Philosophy student. This project was funded by a grant from the System Development Foundation.

variety of semantic relations including not only homonymy, synonymy, and polysemy, but also partial overlaps of a variety of sorts. Practical applications include facilitating translation from language to language, and dealing with recalcitrant semantic facts such as metaphor.

3.5.13 Verbmobil: A Translation System for Face-to-Face Dialog[38]

Goals. This project arose as the result of a request from the Bundesministerium für Forschung und Technologie (BMFT), the German Federal Ministry of Research and Technology. They wish to embark on a program of research to develop an experimental prototype for "Verbmobil," a portable simultaneous interpreter. They asked some of the researchers at CSLI to examine the fields of science and engineering bearing on Verbmobil and to assess what they could contribute now and what advances of importance to the program could be expected soon. In addition, they asked us to draw up a plan for the Verbmobil program for the period beginning now and ending in the year 2000, saying how close it might be possible to come to the goal in that time and what steps should be taken to assure the best outcome.

Significance. Verbmobil is an extremely ambitious program. It depends on finding solutions to many difficult problems. For example, Verbmobil must be able to pick sentences out of the stream of sounds that impinge on its microphone. But the sounds are run together and confused by background noise. Nothing in the stream corresponds to the spaces that separate written letters and words, or the punctuation marks that set off major phrases. There is also much that is not understood about the way language is used in everyday conversation, such as interpretation of intonation and rhythmic variation, how mistakes are corrected, and how one person yields the floor to another. Above all, Verbmobil must be able to translate what it hears into another language and, while this problem has motivated much research in recent years, much more must be learned before even the first prototype can be built.

Thus, while Verbmobil is clearly important as an end in itself, for the immediate future its importance will lie in the way in which it will focus attention on central scientific issues and channel energy into key enabling technologies.

3.6 Foundations

The concept of information—like those of energy and mass in physics, species and gene in biology, inflation and deflation in economics—derives much of its meaning from our everyday interactions with the world. But the demands placed on these concepts by the working scientist require a degree of precision and refinement that goes beyond that of the ordinary notion.

Many issues that arise in the study of action, communication, and computation require a precise characterization of the information content of sentences or signals or representations. The specific goal may be to translate sentences from one language to another, to present large amounts of data in a humanly comprehensible form, to search and summarize a body of text, or to compare the contents of differently structured data bases. But whatever the goal, such notions as information content and informational equivalence loom large in the investigation.

The project described in this section is devoted to the abstract study of information. Its goal is to develop a mathematically precise theory of information, and to apply that theory in a variety of areas within the study of language and computation.

[37] Project participants: Martin Kay, Stanford Linguistics and Xerox PARC (project leader); Jared Bernstein, SRI International; John Etchemendy, Stanford Philosophy; Jean Mark Gawron, SRI International; Jerry Hobbs, SRI International; Megumi Kameyama, Stanford Computer Science student; Betsy Macken, CSLI; Peter Norvig, Sun Microsystems; Ryo Ochitani, Fujitsu Laboratories Ltd.; Stanley Peters, Stanford Linguistics; William Poser, Stanford Linguistics; Ivan A. Sag, Stanford Linguistics. This project was funded by the BMFT contract "International Trends in Machine Translation and Speech Understanding."

[38] Project participants: John Etchemendy, Stanford Philosophy (project leader); Peter Aczel, Manchester Computer Science; Jon Barwise, Indiana Philosophy, Mathematics, and Computer Science; Robin Cooper, Edinburgh Cognitive Science; Keith Devlin, Colby Mathematics; Elizabet Engdahl, Edinburgh Linguistics; Timothy Fernando, Stanford Symbolic Systems student and Amsterdam Institute for Logic, Language, and Information; Jean Mark Gawron, SRI International; David Israel, SRI International and Stanford Philosophy consulting professor; Yasuhiro Katagiri, NTT; Paul J. King, CSLI postdoctoral fellow; Kuniaki Mukai, Institute for New Generation Computer Technology (ICOT); Hideyuki Nakashima, Electrotechnical Laboratories (ETL) Cognitive Science Section;

3.6.1 Situation Theory and Situation Semantics (STASS)[39]

Goals. The Situation Theory and Situation Semantics (STASS) project serves as a clearinghouse for worldwide research on situation theory. The goals of situation theory are to develop a unified mathematical theory of meaning and information content, and to apply that theory to specific areas within the study of language, computation, and cognition.

Significance. Most work in information theory adopts a purely quantitative approach to the subject, dealing only with the information-carrying capacity of a signal or information channel. From this perspective, a meaningless string of characters or bits is equivalent to a meaningful string, so long as the two signals diverge equally from random background noise. This approach to information is valuable for certain purposes, particularly when our concern is the error-free transmission of information by devices that are not themselves users of information.

But quantitative information theory ignores issues that arise when signals must be dealt with at the level of information content, for example, in the context of machine translation (when the content of sentences from different languages must be compared), machine perception (when usable information must be extracted from sensory input), or, potentially, multimedia (when the same or related information is represented in different modalities or formats). As information technology has matured, the need for a general mathematical framework in which to address these issues has become increasingly evident. The theory of meaning and information content being developed in situation theory is meant to provide such a framework.

John Nerbonne, Saarbrücken Computational Linguistics and Deutsches Forschungsinstitut für Künstliche Intelligenz (DFKI); John Perry, Stanford Philosophy; Stanley Peters, Stanford Linguistics; Gordon Plotkin, Edinburgh Computer Science; Jerry Seligman, Edinburgh Cognitive Science and Indiana Logic Group; Sun-Joo Shin, Notre Dame Philosophy; Hidetosi Sirai, Chukyo Computer and Cognitive Sciences; Brian Cantwell Smith, Xerox PARC and Stanford Philosophy consulting professor; Hiroyuki Suzuki, Matsushita Electric Industrial Co.; Syun Tutiya, Chiba Philosophy; Dag Westerståhl, Stockholm Philosophy; Edward N. Zalta, Stanford Philosophy acting assistant professor. Funding for this project was provided by a grant from the System Development Foundation.

NEWS

CACM, Vol. 36, No. 1, January 1993; Newstrack:

According to the General Accounting Office, 10 of the 11 key industrial sectors in USA fell to competitors.
President (at the time candidate) Clinton announced his plans for promoting USA high technology, such as robotics, biotechnology, fiber optics, networks, digital imaging, data storage, CAD and AI.
Daily News reports that the project to computerize student attendance records is millions over budget while less than half of the intended 1,000 city public schools are wired.
Benchmark tests have shown that on three selected benchmark problems traditional vector supercomputers were faster than massively parallel machines.
A new magazine Future Sex is devoted to cybersex.

CACM, Vol. 36, No. 2, February 1993; Newstrack:

IFIP appeals to everybody worldwide to censure harmful games, to raise awareness of the issues involved, and to support only computer games that respect human dignity.
Under President Clinton and Vice President Gore, an AI-based system Resumix helped to sort 100,000 appliances for 4,000 federal jobs thus saving about a million pieces of paper.
Carnegie Mellon University has created a five-year programme leading to a bachelor's degree and a master's degree in software engineering in coordination with firms such as Apple, Intel, Microsoft and Motorola.

CACM, Vol. 36, No. 1, January 1993; Contents:

Devoted to multimedia.
In the President's letter, the ACM President Gwen Bell introduces bridges between worlds. As claimed, the ACM membership reaches approximately 85,000, however, only 20% of them are outside North America. Similar proportions seems valid for several other important special interest groups.
In the ACM Forum, Gregory Aharonian notes that in 1960 – 1992 over 9,000 software patents were issued, and in 1992 alone about 1,300. Robert Milner, a professor at the University of Edinburgh and a Turing Award winner, presents his work on concurrency, and his viewpoints on several aspects of computing in an interview.

CACM, Vol. 36, No. 2, February 1993; Contents:

Devoted to Digital's Alpha Chip Project (see article in this issue of Informatica).
In the Forum, debate concerns the proposed motive by governmental institutions to broaden the computer science by encompassing computer engineering and applications. Science is not to be done just because it is science, but it must either produce results or be at least very promising in the future. The principal issue raised is who is to set priorities for computing research, and if it is reasonable to advocate linear (mutually supportive) transmissions of discoveries into applications.

ACM Membernet:

The 5-hour ACM sponsored television series **'The Machine That Changed The World'** was rated among the top six PBS prime time productions in USA, and received generous commends. Regular retail prices are about $4000 (contact P.O. Box 2053, Princeton, NJ 08543). Personally, and from comments of colleagues, we share acclamations by American critics with a small remark that it should be shipped to Europe in European VHS recording standards.

Artificial Intelligence, Vol. 59, No. 1-2, February 1993

Special volume: artificial intelligence in perspective.
Dedicated to the memory of Allen Newell who died on June 19, 1992, one of the founders of AI.
Based on the SCI's databases, authors of the papers in the magazine which had over 24 citations were invited to write a short paper about their work, area and prospectives. There were around 30 of them including Newell, which was able to write his work although probably knowing he won't be able to read it.
Indeed, we should pay him greatest respects to several of his important achievements, especially Soar in the last decade or so, to his great intellectual power, his dreams and devotion.

Minds and Machines, Vol. 3, No. 1, February 1993; Contents:

Larry Hausler and William J. Rapaport exchange arguments regarding the question 'is a pocket calculator a thinking thing'. Hauser starts argumenting that there is no sufficient proof (at least without a reasonable doubt) that a calculator is not a thinking thing. It seems that he is deliberately posing the frontier of his arguments at the level even he does not believe in, just to show that there are no real arguments that computers can not be (are not?) intelligent. Although Rapaport demolishes Hauser's claims, he seems to nearly come into terms with the claim that an integrated program (system) could be considered as intelligent to a certain degree.

IJCAI-93 WORKSHOP

Machine learning and knowledge acquisition: common issues, contrasting methods, and integrated approaches.

29 August 1993, Chambery, France.

Machine learning and knowledge acquisition share the common goal of acquiring and organizing the knowledge of a knowledge-based system. However, each field has a different focus, and most research is still done in isolation from each other. The focus of knowledge acquisition has been to improve and partially automate the acquisition of knowledge from human experts. In contrast, machine learning focuses on mostly autonomous algorithms for acquiring or improving the organization of knowledge, often in simple prototype domains. Also, in knowledge acquisition, the acquired knowledge is directly validated by the expert that expresses it, while in machine learning, the acquired knowledge needs an experimental validation on data sets independent of those on which learning took place. As machine learning moves to more 'real' domains, and knowledge acquisition attempts to automate more of the acquisition process, the two fields increasingly find themselves investigating common issues with complementary methods. However, lack of common research methodologies, terminology, and underlying assumptions often hinder a close collaboration. The purpose of this symposium is to bring together machine learning and knowledge acquisition researchers in order to facilitate cross-fertilization and collaboration, and to promote integrated approaches which could take advantage of the complementary nature of machine learning and knowledge acquisition.

Topics of interest include, but are not limited to, the following: Case Studies, Comparative Studies, Hard Problems, Knowledge Representation, Key Issues, Overviews, Position Papers.

It is recommended that the papers make explicit the research methodology, the underlying assumptions, definitions of technical terms, important future issues, and potential points of collaboration. They should not exceed 15 pages. The organizers intend to publish a selection of the accepted papers as a book or the special issue of a journal. They encourage the authors to take this into account while preparing their papers. The format of the workshop will be paper sessions with discussion at the end of each session, and a concluding panel on the integrated approaches, guidelines for successful collaboration, and concrete action items. The number of the participants to the workshop is limited to 40.

Each workshop attendee must also register for the IJCAI conference and must pay an additional 300FF (about $60) fee for the workshop. One student attending the workshop and being in charge of taking notes will be exempted from the additional 300 FF fee. Volunteers are invited.

WORKSHOP CO-CHAIRS

Smadar Kedar, NASA Ames & Inst. for Learning Sciences, (kedar@ils.nwu.edu)
Yves Kodratoff, CNRS & Universite de Paris-Sud, (yk@lri.lri.fr)
Gheorghe Tecuci, George Mason Univ. & Romanian Academy, (tecuci@aic.gmu.edu)

PROGRAM COMMITTEE

Ray Bareiss, Institute for the Learning Sciences
Catherine Baudin, NASA Ames
Guy Boy, European Inst. of Cognitive Sc. and Eng.
Brian Gaines, University of Calgary
Matjaz Gams, Jozef Stefan Institute
Jean-Gabriel Ganascia, Univ. Pierre and Marie Curie
Nathalie Mathe, European Space Ag. and NASA Ames
Ryszard Michalski, George Mason University
Raymond Mooney, University of Texas at Austin
Katharina Morik, Dortmund University
Mark Musen, Stanford University
Michael Pazzani, Univ. of California at Irvine
Luc De Raedt, Catholic University of Leuven
Alan Schultz, Naval Research Laboratory
Mildred Shaw, University of Calgary
Maarten van Someren, University of Amsterdam
Walter Van de Velde, University of Brussels

ADDRESS FOR CORRESPONDENCE

Gheorghe Tecuci
Artificial Intelligence Center, Computer Science Department
George Mason University, 4400 University Dr., Fairfax, VA 22030
email: mlka93@aic.gmu.edu, fax: (703)993-3729

Those who would like to attend without a presentation should send a one to two-page description of relevant research interests and a list of selected publications.

(Informations about events should be sent by e-mail to matjaz.gams@ijs.si, possibly already in LaTeX or at least in ASCII)

ERK'93

Electrotechnical and Computer Conference
Elektrotehniška in računalniška konferenca

27.–29. September 1993

Conference Chairman
Baldomir Zajc
University of Ljubljana
Faculty of Electr. Eng. and Comp. Science
Tržaška 25, 61000 Ljubljana, Slovenia
Tel: (061) 265 161, Fax: (061) 264 990
E-mail: baldomir.zajc@fer.uni-lj.si

Conference Vicechairman
Bogomir Horvat
University of Maribor
Technical Faculty,
Smetanova 17, 62000 Maribor, Slovenia
Tel: (062) 25 461, Fax: (062) 212 013
E-mail: horvat@uni-mb.ac.mail.yu

Program Committee Chairman
Saša Divjak
University of Ljubljana
Faculty of Electr. Eng. and Comp. Science
Tržaška 25, 61000 Ljubljana, Slovenia
Tel: (061) 265 161, Fax: (061) 264 990
E-mail: sasa.divjak@fer.uni-lj.si

Programe Committee
Tadej Bajd
Zmago Brezočnik
Janko Drnovšek
Matjaž Gams
Ferdo Gubina
Marko Jagodič
Jadran Lenarčič
Drago Matko
Miro Milanovič
Andrej Novak
Nikola Pavešić
Franjo Pernuš

Publications Chairman
Franc Solina
University of Ljubljana
Faculty of Electr. Eng. and Comp. Science
Tržaška 25, 61000 Ljubljana, Slovenia
Tel: (061) 265 161, Fax: (061) 264 990
E-mail: franc@fer.uni-lj.si

Advisory Board
Ludvik Gyergyek
Dali Djonlagić
Karel Jezernik
Peter Jereb
Alojz Kralj
Marjan Plaper
Jernej Virant
Lojze Vodovnik

Call for Papers

for the second **Electrotechnical and Computer Conference ERK'93**, which will be held on 27–29 September 1993 in Portorož, Slovenia. All presentations during the first day of the conference (invited lectures and selected sessions) will be held in English.

The following areas will be represented at the conference:

- *electronics,*
- *telecommunications,*
- *measurement,*
- *automatic control and robotics,*
- *computer and information science,*
- *artificial intelligence and pattern recognition,*
- *biomedical engineering,*
- *power engineering.*

The conference is organized by the **Slovenia Section IEEE** and other Slovenian professional societies:

- Slovenian Society for Automatic Control,
- Slovenian Measurement Society (ISEMEC 93),
- SLOKO-CIGRE,
- Slovenian Society for Medical and Biological Engineering,
- Slovenian Society for Robotics,
- Slovenian Artificial Intelligence Society,
- Slovenian Pattern Recognition Society.

Authors who wish to present a paper at the conference should send three copies of their abstract (500 words) to the chairman of the Programe Committee prof. S. Divjak. The abstract should include also:

1. the title of the paper,
2. author's address,
3. telephone, telefax and e-mail of the contact author,
4. the paper's subject area.

Authors of accepted papers will have to prepare a four page camera-ready copy of their paper for inclusion into the proceedings of the conference.

Time schedule:	Abstracts due	*1 June 1993*
	Notification of acceptance	*30 June 1993*
	Camera-ready paper	*1 September 1993.*

For all additional information please contact the conference chairmen.

JOŽEF STEFAN INSTITUTE

Jožef Stefan (1835-1893) was one of the most prominent physicists of the 19th century. Born to Slovenian parents, he obtained his Ph.D. in Vienna University, where he was later Director of the Physical Institute, Vice-President of the Vienna Academy of Sciences and member of several scientific institutions in Europe. Stefan explored many areas from 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 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 growth 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, totalling about 800, has 500 researchers: about 250 of them are postgraduates, over 200 have doctorates (Ph.D.), and around 150 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 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 independent country Slovenia (or S♡nia). The capital today is considered as a crossroad between the East, West and Mediterranean Europe, offering excellent productive capabilities and consolidate business opportunities with strong international connections. Ljubljana is connected to important centers such as Praha, Budapest, Wien, Zagreb, Milano, Roma, Monaco, Nice, Bern, München all inside the circle of 600 km.

In the last year at the location of the Jožef Stefan Institute, the Technology park "Ljubljana" is proposed as a part of the national strategy for technological development to foster synergies between research and production 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 developing cycle of innovative products.

At the present time a part of the Institute is being reorganized in several high-tech units supported and connected within the Technology park at "Jožef Stefan" Institute, established as a beginning of an regional Technology park "Ljubljana". The project is being developed at a particular historical moment characterized by a process of state reorganization, privatisation and private initiative. The national Park will take the form of a shareholding company and will host an independent financial institution for venture kapital.

Promoters and operative entities of the project are the Republic of Slovenia, Ministry of Science and Technology and the Jožef Stefan Institute. The frame of the operation includes also University of Ljubljana, Institute of Chemistry, Institute for Electronics and Vacuum Technique, Institute for Materials and Construction Research and some others. Furthermore the project is supported by Ministry of Small Business, National Chamber of Commerce and the City of Ljubljana.

Jožef Stefan Institute
Jamova 39, 61000 Ljubljana, Slovenia
Tel.:+38 61 159 199
Fax.:+38 61 161 029
Tlx.:31 296 JOSTIN SI
E-mail: matjaz.gams@ijs.si

Contact person for the Park: Iztok Lesjak, M.Sc.

Public relations: Ines Černe

INFORMATION FOR CONTRIBUTORS

1 Submissions and Refereeing

Please submit three copies of the manuscript with good copies of the figures and photographs to one of the editors from the Editorial Board or to the Contact Person. At least two referees outside the author's country will examine it and they are invited to write as many remarks as possible directly on the manuscript, from typing errors to global philosophical disagreements. The chosen editor will send the author copies with remarks, and if accepted also to the Contact Person. The Executive Board will inform the author that the paper is accepted, meaning that it will be published in less than one year after receiving original figures on separate sheets and the text on an IBM PC DOS floppy disk or through e-mail – both in ASCII and the Informatica LaTeX format. Style (attached) and examples of papers can be obtained through e-mail from the Contact Person.

2 News, letters, opinions

Opinions, news, calls for conferences, calls for papers, etc. should be sent directly to the Contact Person.

```
\documentstyle[twoside,informat]{article}

\begin{document}
  \title{TITLE OF THE ARTICLE}
   \author{First Author \\ Address \\ E-mail \\
      AND \\
      Second Author \\ Address \\ E-mail}
  \titleodd{TITLE IN HEADER}
  \authoreven{Author's name in header}
  \keywords{article keywords}
  \edited{editor in charge}
  \received{date}
  \revised{date}
  \accepted{date}
  \abstract{abstract -- around 200 words}
  \maketitle

  \section{Introduction}
    Text of Introduction.

  \section{Subject}
    Text of the Subject.

  \begin{figure}
    An example of a figure over 1 column.
    \caption{Caption of the figure 1}
  \end{figure*}

  \begin{figure*}
    An example of a figure over 2 columns.
    \caption{Caption of the figure 2}
  \end{figure*}

  \begin{thebibliography}{99}
    \bibitem{} First Author: {\sl Title},
      Magazine, Vol. 1, No. 1.
  \end{thebibliography}

\end{document}
```

```
\def\journal{
  Informatica Vol. 17, No. 1, March 1993}
\immediate\write16{'Informatica' V1.1  by B."Z}
\newif\iftitle \titlefalse
\hoffset=11mm \voffset=8mm
\oddsidemargin=-21mm \evensidemargin=-14mm
\topmargin=-33mm
\headheight=17mm \headsep=10mm
\footheight=8.4mm \footskip=52.5mm
\textheight=242mm \textwidth=170mm
\columnsep=5mm \columnseprule=0pt
\twocolumn \sloppy \flushbottom
\parindent 1em
\leftmargini 2em \leftmargin\leftmargini
\leftmarginv .5em \leftmarginvi .5em
\def\labelitemi{\bf --} \def\labelitemii{--}
\def\labelitemiii{--}
\setcounter{secnumdepth}{3}
\def\maketitle{\twocolumn[%
  \vbox{\hsize=\textwidth\Large\bf\raggedright
  \@title}\vss\bigskip\bigskip \vbox{
  \hsize=\textwidth \@author}\bigskip\smallskip
  \vbox{\hsize=\textwidth {\bf Keywords:}
    \@keywords}
  \bigskip \hbox{{\bf Edited by:} \@edited}
  \smallskip
  \hbox{{\bf Received:}
    \hbox to 10em{ \@received\hss}
    {\bf Revised:}\hbox to 10em{ \@revised\hss}
    {\bf Accepted:}\hbox to 10em{ \@accepted\hss}}
  \bigskip \vbox{\hsize=\textwidth
    \parskip=\baselineskip \leftskip=3em
    \rightskip=3em \sl \@abstract}
  \bigskip\bigskip]\titletrue}
\def\maketitleauthor{\twocolumn[%
  \vbox{\hsize=\textwidth \Large\bf\raggedright
  \@title}\vss \bigskip\bigskip
  \vbox{\hsize=\textwidth \@author}
  \bigskip\bigskip] \gdef\@title{}\titletrue}
\def\makeonlytitle{\twocolumn[%
  \vbox{\hsize=\textwidth \Large\bf\raggedright
  \@title}\vss\bigskip\bigskip]
  \gdef\@title{}\titletrue}
\def\@title{} \def\@author{}
\def\@titleH{} \def\@authorH{}
\def\@keywords{} \def\@edited{} \def\@abstract{}
\def\@received{} \def\@revised{} \def\@accepted{}
\def\authoreven#1{\gdef\@authorH{#1}}
\def\titleodd#1{\gdef\@titleH{#1}}
\def\keywords#1{\gdef\@keywords{#1}}
\def\edited#1{\gdef\@edited{#1}}
\def\received#1{\gdef\@received{#1}}
\def\revised#1{\gdef\@revised{#1}}
\def\accepted#1{\gdef\@accepted{#1}}
\long\def\abstract#1{\gdef\@abstract{#1}}
\def\section{\@startsection {section}
  {1}{\z@}{-3.5ex plus -1ex minus -.2ex}
  {2.3ex plus .2ex}{\Large\bf\raggedright}}
\def\subsection{\@startsection{subsection}
  {2}{\z@}{-3.25ex plus -1ex minus -.2ex}
  {1.5ex plus .2ex}{\large\bf\raggedright}}
\def\subsubsection{\@startsection{subsubsection}
  {3}{\z@}{-3.25ex plus -1ex minus -.2ex}
  {1.5ex plus .2ex}{\normalsize\bf\raggedright}}
\def\@evenhead{\hbox to 3em{\bf\thepage\hss}
  {\small\journal\hfil \iftitle\else
  \@authorH \fi}\global\titlefalse}
\def\@oddhead{{\small\iftitle\else \@titleH \fi
  \hfil\journal}\hbox to 3em{\hss\bf\thepage}
  \global\titlefalse}
\def\@evenfoot{\hfil} \def\@oddfoot{\hfil}
\endinput
```

INFORMATICA

AN INTERNATIONAL JOURNAL OF COMPUTING AND INFORMATICS

INVITATION, COOPERATION

Since 1977 Informatica has been a major Slovenian scientific journal of computing and informatics, including telecommunications, automatics and other related areas. In its 17th year it is becoming 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 the European computer science and informatics 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, pedagogical and governmental institutions. Others should subscribe (institutional rate 50 DM, individual rate 25 DM, and for students 10 DM). Send a check in the appropriate amount to Slovene Society Informatica, Vožarski pot 12, Ljubljana, account no. 50101-678-51841.

Please, return this questionnaire.

QUESTIONNAIRE

☐ Send Informatica free of charge

☐ Yes, we subscribe

☐ We intend to cooperate (describe):

☐ Propositions for improvements (describe):

Informatica is published in one volume (4 issues) per year.
If possible, send one copy of Informatica to the local library.
Please, fulfil the order card and send to dr. Rudi Murn, Informatica, Institut Jožef Stefan, Jamova 39, 61111 Ljubljana, Slovenia.

ORDER CARD – INFORMATICA

Name: ....
Title and Profession (optionally): ....
....
Home Address and Telephone (optionally): ....
....
Office Address and Telephone (optionally): ....
....
E-mail Address (optionally): ....
Signature and Date: ....

INFORMATICA
REVIJA ZA RAČUNALNIŠTVO IN INFORMATIKO
VABILO K SODELOVANJU

Od leta 1977 je Informatica najpomembnejša slovenska strokovna revija na področju računalnitva in informatike pa tudi telekomunikacij, avtomatike in drugih sorodnih področij. V 1993 vstopa Informatica z uglednim mednarodnim uredniškim odborom in novo, panevropsko usmeritvijo. V njej bodo štirikrat letno objavljeni strokovni članki, ki jih bosta recenzirala vsaj dva recenzenta izven države avtorja članka. V drugem delu revije bodo objavljene strokovne debate, ocene, mnenja, pomembne organizacijske in tržne spremembe in produkti na znanstvenem, pedagoškem in gospodarskem področju.

Prosimo Vas, da na ustrezen način predstavite revijo svoji okolici, npr. tako da jo postavite na viden prostor, omogočite dostop do nje in jo uvrstite v svojo knjižnico.

Revija Informatica bo brezplačna za nekatere najpomembnejše znanstvene, pedagoške in ostale institucije v Sloveniji, drugi pa jo bodo lahko dobili z enoletno naročnino, ki znaša za podjetja 3000 SIT (50 DEM), za zasebnike 1500 SIT (25 DEM) in za študente 500 SIT (10 DEM). Izpolnite prijavnico in nakažite ustrezni znesek na Slovensko društvo Informatika, Vožarski pot 12, Ljubljana, žiro račun št. 50101-678-51841.

Če se aktivno udejstvujete na ožjem ali širšem področju računalništva in informatike, Vas vabimo, da pošljete svoje prispevke (npr. članke ali propagandne materiale) komurkoli izmed "Executive Editors". Informatika bo skrbela za pošiljanje revije v vse pomembnejše raziskovalne, pedagoške, gospodarske in infrastrukturne institucije po Sloveniji in Evropi pa tudi Ameriki, zato je to enkratna priložnost za predstavitev Vašega dela.

Prosimo Vas, da vrnete ta list z izpolnjeno anketo.

ANKETA

☐ Pričakujemo brezplačno prejemanje

☐ Naročamo revijo

☐ Želimo si sodelovati

☐ Način sodelovanja (opisno):

☐ Predlogi izboljšav (opisno):

Izpolnjeno naročilnico pošljite na naslov: dr. Rudi Murn, Informatica, Institut Jožef Stefan, Jamova 39, 61111 Ljubljana.

Naročilnica na revijo INFORMATICA

Ime in priimek: ....

Izobrazba: ....

Domač naslov in telefon: ....

....

Služben naslov in telefon: ....

....

Elektronski naslov: ....

EMŠO (neobvezno): ....

V ...., dne ....

Podpis: ....

EDITORIAL BOARDS, PUBLISHING COUNCIL

Informatica is a journal primarily covering the European computer science and informatics 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. Each paper bears the name of the editor who appointed the referees. Each editor can propose new members for the Editorial Board or Board of Referees. Editors and referees inactive for a longer period can be automatically replaced. Changes in the Editorial Board and Board of Referees 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 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.

Board of Advisors
Ivan Bratko, Marko Jagodič,
Tomaž Pisanski, Stanko Strmčnik

Publishing Council
Tomaž Banovec, Ciril Baškovič,
Andrej Jerman-Blažič, Dagmar Šuster,
Jernej Virant

Board of Referees
C. Anglano, F. Abbattista, S. Bezjak, D. Bojadžiev, G. Božič, P. Giolito, I. Kononenko, F. Lanubile, N. Lavrač, J. P. Merlet, R. Sabatino, M. Sapino, J. Taylor, A. Werbrouk

Executive Editors

Editor in Chief
Anton P. Železnikar
Volaričeva 8, Ljubljana, Slovenia
E-mail: anton.p.zeleznikar@ijs.si

Associate Editor (Contact Person)
Matjaž Gams
Jožef Stefan Institute
Jamova 39, 61000 Ljubljana, Slovenia
Phone: +38 61 159 199, Fax: +38 61 161 029
E-mail: matjaz.gams@ijs.si

Associate Editor (Technical Editor)
Rudi Murn
Jožef Stefan Institute
Jamova 39, 61000 Ljubljana, Slovenia
Phone: +38 61 159 199, Fax: +38 61 161 029

Editorial Board
Suad Alagić (Bosnia and Herzegovina)
Vladimir Batagelj (Slovenia)
Andrej Bekeš (Japan)
Leon Birnbaum (Romania)
Marco Botta (Italy)
Pavel Brazdil (Portugal)
Andrej Brodnik (Canada)
Janusz Brozyna (France)
Luca Console (Italy)
Hubert L. Dreyfus (USA)
Jozo Dujmović (USA)
Johann Eder (Austria)
Vladimir A. Fomichov (Russia)
Janez Grad (Slovenia)
Noel Heather (UK)
Francis Heylighen (Belgium)
Bogomir Horvat (Slovenia)
Jadran Lenarčič (Slovenia)
Angelo Montanari (Italy)
Peter Mowforth (UK)
Igor Mozetič (Austria)
Oliver Popov (Macedonia)
Sašo Prešern (Slovenia)
Luc De Raedt (Belgium)
Giacomo Della Riccia (Italy)
Claude Sammut (Australia)
Jiři Slechta (UK)
Branko Souček (Italy)
Harald Stadlbauer (Austria)
Gheorghe Tecucci (USA)
Robert Trappl (Austria)
Terry Winograd (USA)
Stefan Wrobel (Germany)

Volume 17 Number 1 March 1993 ISSN 0350-5596

Informatica

An International Journal of Computing and Informatics

Contents: