ITHEA

International Journal INFORMATION THEORIES & APPLICATIONS

2004 Volume 11 Number 4

International Journal
INFORMATION THEORIES & APPLICATIONS
ISSN 1310-0513
Volume 11 / 2004, Number 4

IJ ITA is official publisher of the scientific papers of the members of
the Association of Developers and Users of Intellectualized Systems (ADUIS).

IJ ITA welcomes scientific papers connected with any information theory or its application.
Original and non-standard ideas will be published with preferences.

IJ ITA rules for preparing the manuscripts are compulsory.
The **rules for the papers** for IJ ITA as well as the **subscription fees** are given on *www.foibg.com/ijita*.
The camera-ready copy of the paper should be received by e-mail: foi@nlcv.net.
Responsibility for papers published in IJ ITA belongs to authors.

General Sponsor of IJ ITA is the **Consortium FOI Bulgaria** (www.foibg.com).

International Journal "INFORMATION THEORIES & APPLICATIONS" Vol.11, Number 4, 2004
Printed in Bulgaria
Edited by the **Institute of Information Theories and Applications FOI ITHEA**, Bulgaria,
in collaboration with the V.M.Glushkov Institute of Cybernetics of NAS, Ukraine, and
the Institute of Mathematics and Informatics, BAS, Bulgaria.
Publisher: FOI-COMMERCE - Sofia, 1000, P.O.B. 775, Bulgaria. www.foibg.com, e-mail: foi@nlcv.net

ISSN 1310-0513

MULTI-DOMAIN INFORMATION MODEL

Krassimir Markov

Abstract: The Multi-Domain Information Model for organisation of the information bases is presented.

Keywords: Multi Domain Information Model, Information Bases, Knowledge Representation.

1. Introduction

The "Multi-Domain Information Model" (MDIM) has been established twenty years ago. For a long period it has been used as a basis for organisation of various information bases. The first publication containing some details of MDIM is [Markov, 1984] but the model has not been fully presented till now. In addition, over the years, the model has been extended with some new concepts like "information space", "metaindex", "polyindexation", etc. which we will introduce in this paper.

The present paper aims to present MDIM as a coherent whole.

2. Information Domain

Definition 1. ***Basic information element*** "e" of MDIM is an arbitrary long string of indivisible information fragments (bytes in the version for IBM PC; symbols; etc.). ■

Let E_1 is a set of basic information elements:

$$E_1 = \{e_i \mid e_i \in E_1, i=1,\ldots, m_1\}.$$

Let μ_1 is a function which defines a biunique correspondence between elements of the set E_1 and elements of a set C_1 of natural numbers: $C_1 = \{c_i \mid c_i \in N, i:=1,\ldots, m_1\}$, i.e.

$$\mu_1 : E_1 \leftrightarrow C_1$$

Definition 2. The elements of C_1 are said to be ***co-ordinates*** of the elements of E_1. ■

Definition 3. The triple $S_1 = (E_1, \mu_1, C_1)$ is said to be an ***information domain*** of range one ***(one-dimensional information space)*** . ■

Remark: In the previous publications, the information domain S_1 was denoted by D and the co-ordinates c_i were called "codes" of the corresponded information elements.

3. Information Spaces

Definition 4. The triple $S_n = (E_n, \mu_n, C_n)$, $n \geq 2$, is said to be an (***complex*** or ***multi-domain***) ***information space*** of range n iff E_n is a set which elements are information spaces of range n-1 and μ_n is a function which defines a biunique correspondence between elements of E_n and elements of the set C_n of natural numbers (co-ordinates of range n):

$$C_n = \{c_k \mid c_k \in N, k:=1,\ldots,m_n\}, \text{ i.e.}$$

$$\mu_n : E_n \leftrightarrow C_n \;■$$

Definition 5. Every basic information element "e" is considered as an ***information space*** S_0 of range 0. ■

It is clear that the information space $S_0 = (E_0, \mu_0, C_0)$, is constructed in the same manner as all others:

- the indivisible information fragments (bytes) b_i, $i=1,\ldots,m_0$ are considered as elements of E_0,
- the position p_i (natural number) of b_i in the string e is considered as co-ordinate of b_i, i.e. $C_0 = \{p_k \mid p_k \in N, k:=1,\ldots,m_0\}$,
- function μ_0 is defined by the physical order of b_i in e and we have: $\mu_0 : E_0 \leftrightarrow C_0$

When it is necessary the string S_0 may be considered as a set of ***sub-elements (sub-strings)*** which may contain one or more indivisible information fragments (bytes). The number and length of the sub-elements may be variable. This option is very helpful but it closely depends on the concrete realizations and is considered as none standard characteristic of MDIM.

Definition 6. The information space S_n of range n is called ***information base*** of range n. ■

Usually, the concept information base without indication of the range is used as generalized concept to denote all information spaces in use during given time period.

4. Indexes

Definition 7. The sequence $A = (c_n,c_{n-1},\ldots,c_1)$ where $c_i \in C_i$, $i=1,\ldots,n$, is called ***space address*** of range n of an basic information element. ■

Every space address of range m, $m<n$, may be extended to space address of range n by adding leading n-m zero co-ordinates.

Definition 8. Every sequence of space addresses $A_1,A_2,\ldots,A_k$, where k is arbitrary natural number, is said to be an (address) ***index***. ■

Definition 9. Every ordered subset I_i, $I_i \subset C_i \subset N$ of co-ordinates (i – arbitrary natural number) is said to be a (space) ***index***. ■

It is clear that space index is a kind of address index.

5. Polyindexation

Every index may be considered as basic information element (i.e. as a string) and may be stored in a point of any information domain. In such case, it will have a space address which may be pointed again.

Definition 10. Every ***index*** which point only to indexes is said to be a ***metaindex***. ■

Every metaindex may be considered as basic information element (i.e. as a string) and may be stored in a point of any information domain, too. So, it will have a space address which may be pointed again, etc. This way, we may build a hierarchy of metaindexes.

Definition 11. The approach of representing the interconnections between elements of the information domains as well as between spaces using hierarchies of metaindexes is called ***polyindexation***. ■

6. Aggregates

Let $G = \{S_i \mid i=1,...,m\}$ is a set of information spaces.

Let $\tau=\{\nu_{ij} \mid \nu_{ij} : S_i \rightarrow S_j,\ i=const,\ j=1,\ldots m\}$ is a set of mappings of one "main" information space $S_i \subset G$, $i=const$, into the others $S_j \subset G$, $j=1,\ldots m$, and, in particular, into itself.

Definition 12. The couple $Đ = (G, \tau)$ is said to be an "***aggregate***". ■

It is clear we can build m aggregates using the set *G* because every information space $S_i \subset G$, $j=1,\ldots m$, may be chosen to be a main information space.

Remark: In the previous publications, the aggregate Đ was called generalized domain.

7. Operations in MDIM

After defining the information structures we need to present the operations which are admissible in the model.

It is clear; the operations are closely connected to the defined structures. So, we have operations with:

- basic information elements (**BIE**)
- spaces
- indexes
- metaindexes

In MDIM, we assume that all information elements of all information spaces exist. If for any S_i : $E_i=\varnothing \wedge C_i=\varnothing$, than it is called *empty.* Usually, most of the information elements and spaces are empty. This is very important for practical realizations.

7.1. Operation with basic information elements

Because of the rule for existing of the all structures given above we have need of only two operations:

- updating the BIE
- getting the value of BIE

For both types of operations we need two service operations:

- getting length of BIE
- positioning in the BIE

Updating, or simply – writing the element, has several modifications with obvious meaning:

- writing of a BIE as a whole
- appending a BIE
- inserting in a BIE
- cutting a part of BIE
- replacing a part of BIE
- deleting a BIE

The operation for getting the value of BIE is only one – **Read** a portion from BIE starting from given position. We may receive the whole BIE if the starting position is the beginning of BIE and the length of the portion is equal to the BIE length.

7.2. Operation with spaces

With a single space we may do only one operation – clearing (deleting) the space, i.e. replacing the all BIE of the space with ∅. After this operation the BIE of the space will have zero length.

With two spaces we may provide two operations with two modifications every:

- copying the first space in the second
- moving the first space in the second

The modifications concern the type of processing the BIE of the recipient space. We may have:

- copy with clear
- move with clear
- copy with merge
- move with merge

The "clear" modifications first clear the recipient space and after that provide copy or move operation.

The merge modifications may have two types of processing:

- destructive
- constructive

The *destructive merging* may be "conservative" or "alternative". In the conservative approach the recipient space BIE remain in the result if it is with none zero length. In the other approach – the donor space BIE remain in the result.

In the *constructive merging* the result is any composition of the corresponded BIE of the two spaces.

Of course, the move operation deletes the donor space after the operation.

7.3. Operation with indexes and metaindexes

The indexes are the main approach for describing the interconnections between the BIE.

At the first place, we may operate with and in the indexes C_i, $i=1,2,\ldots,n$ of the spaces. We may receive the co-ordinates of the next or previous empty or none empty elements of the space starting from any given co-ordinate. The possibility to count the number of none empty elements is useful for practical realisations.

The operations with indexes and metaindexes may be classified in two main types:

- logical operations
- information operations

The first type is content independent operations based on usual logical operations between sets. The difference from usual sets is that the information spaces are build by interconnection between two main sets:

- set of co-ordinates
- set of information elements

The logical operations defined in the MDIM are based on the classical logical operations – intersection, union and supplement, but these operations are not so trivial. Because of complexity of the structure of the information spaces these operations have at least two principally different realizations based on:

- co-ordinates
- information elements

The operations based on co-ordinates are determined by the existence of the corresponding space information elements. So, the values of the co-ordinates of the existing information elements determine the operations.

In the other case, the values of the BIE determine the logical operations.

In both cases the result of the logical operations is any index, respectively – metaindex.

The information operations are context depended and need special realizations for concrete purposes.

The main information operation is creating the indexes and metaindexes. This may be very complicated processes and could not be given in advance. The main purpose of the MDIM is to give up possibility for access to the practically unlimited information space and easy approach for building interconnection between its elements. The goal of the concrete applications is to build tools for creating and operating with the indexes and metaindexes and to implement these tools in the realization of user requested systems.

For instance such tools may realize the transfer from one structure to another, information search, sorting, making reports, more complicated information processing, etc.

The information operations can be grouped into four sets corresponding to the main information structures:

- basic information elements
- information domains
- information spaces
- index or metaindex structures

8. Discussion

Usually, the submission of any new information model needs to be discussed in connection to already existing models and theories. We have no place in this paper to analyze all known models. Because of this we will point only two of them we assume as more important:

- theory of the named sets [Burgin, 1984]
- relation model of Codd [Codd, 1970]

Our proposition is that the MDIM has the same and more modeling possibilities than named sets and relation model.

8.1. Theory of the named sets

For our further discussion we need some information from [Burgin and Gladun, 1989] .

If α is a relation of X with Y i.e. $\alpha \subseteq X \times Y$, $A \subseteq X$, $B \subseteq Y$ then

$\alpha(A)=\{y|\exists x\in A\ ((x,y)\in\alpha)\}$, $\alpha^{-1}(B)=\{x|\exists y\in B\ ((x,y)\in\alpha)\}$,

$\alpha|_{(A,B)}=\{(x,y)\in\alpha\ |\ x\in A\ y\in B\ \}$.

The empty set is denoted by $\varnothing$.

Definition B&G-1. A named by $\mathcal{M}$ set (an N-set) is a triple

$$\boldsymbol{X}=(X,\alpha,I)$$

where X is a set from some fixed class of sets and is called the support set of the named set $\boldsymbol{X}$. I is a set from some (may be another) fixed class of sets and is called the set of names of the named set $\boldsymbol{X}$. $\alpha:X\rightarrow Y$ is a map or a correspondence (a relation) from X to I and belongs to a given class of relations $\mathcal{M}$.

A name $a\in I$ is called empty if $\alpha^{-1}(B) = \varnothing$.

Named sets as special cases include: usual sets, fuzzy sets, multisets, enumerations, sequences (countable as well as uncountable), etc. A lot of examples of named sets we may find in linguistics studying semantic aspects that are connected with applying different elements of a language (words, phrases, texts) with their meaning. [Burgin and Gladun, 1989, p.121-122].

The Theory of named sets (TNS) has been established about 1982 [Burgin 1984]. Independently, the MDIM has been developed in the period from 1980-1982 and its first publication was [Markov 1984] .

We may find many common ideas in the two approaches. Here we will point at two main characteristic of MDIM.

Proposition 1. Every information space is a named set.

Proof: By definition, the set E is the support set, C is the set of the names and μ is a function of naming.

Proposition 2. Every named set may be represented by an aggregate.

Proof: It is simple to build the named set by an aggregate using:

- two information spaces: one for the names and one for the elements of the named set,
- aggregation mapping which is identical to the named set mapping. ■

This way all possibilities of the TNS exist in the MDIM. In other hand, the polyindexation does not exist as theoretical base in the TNS. The aggregates are more general constructs than named sets. At the end, MDIM is designed to support practical realizations whereas the TNS is a theoretic logical construction for reasoning.

The conclusion is that the MDIM has the same and more modeling possibilities than named sets.

8.2. Relation model of Codd

The Cood's Relation theory [Codd 1970] is so popular that we do not need to explain it here. For our discussion we will proof one very important proposition.

Proposition 3. The relation in the sense of the model of Codd may be represented by an aggregate.

Proof: It is easy to see that if the aggregation mappings of the generalized domain are one-one mappings it will be relation in the sense of the model of Codd. ■

In the same time many possibilities of MDIM could not be represented by the relation model or this is very expensive work. Especially, the polyindexation could not be represented by relations. The representation of the information spaces of range more than three is very expensive for the practical realizations.

So, we may say that MDIM is more universal and convenient for practical realizations than the relation model.

9. Conclusion

The Multi-Domain Information Model (MDIM) for organisation of the information bases has been presented in this paper. The information structures and operations of MDIM have been presented.

The correspondences between MDIM and named sets (Propositions 1 and 2) as well as the relation model (Proposition 3) were shown. Our conclusion is that the MDIM has the same and more modeling possibilities than named sets and relation model.

At the end, we need to discuss some more general conclusions.

We consider *the real world* as a space of *entities.* The entities are built by other entities, connected with *relationships*. The entities and relationships between them form the internal *structure* of the entity they build. To create the entity of a certain structural level of the world, it is necessary to have:

- the entities of the lower structural level;
- establishing of the forming relationship.

The entity can dialectically be considered as a relationship between its entities of all internal structural levels. [Markov et al 2003]. Every entity may be considered as relationship between "*atoms*" which are entities on the lowest structural level where there exists another relationship and so on.

This way we may distinguish three types of relationships: explicit (forming relationships), implicit (forming relationships at lower levels) and mixed (in case we distinguish the relationships from lower levels as elements of the forming relationship of given level).

In our model, the information atoms are the basic information elements. It is easy to see that they may contain more complex structures such as domains, spaces, generalized domains, indexes, metaindexes, etc.

This means: *the complexity of the real word can be reflected by the complexity of the MDIM realizations.*

This inference gives us one very fruitful idea – to use MDIM as a model for memory structuring in intelligent systems [Gladun 2003].

Finally, we need to point out that for more than twenty years the MDIM realizations have shown the power of this model. The concrete systems based on MDIM information bases now work on more than one thousand installations all over the Bulgaria.

Acknowledgments

Author is indebted to Ms. Krassimira Ivanova and Mr. Ilia Mitov for support and collaboration during the latest 15 years. Due to theirs hard work the model presented in this paper has been implemented in practice.

Special thanks to Mr. Boicho Kokinov for fruitful remarks and advices.

Author owes a dept of gratitude to Prof. S.S. Lavrov for his important support and recommendations given in the beginning of work presented in this paper.

Bibliography

[Burgin 1984] Burgin M.S. Named Sets and Information representation. VII Allunion conference on mathematical logic, Novosibirsk, 1984. (in Russian)

[Burgin and Gladun, 1989] Mathematical Foundations of Semantic Networks Theory. Lecture Notes in Computer Science, No. 364, pp. 117-135.

[Codd 1970] Codd E.F. A Relation Model of Data for Large Shared Data Banks. CACM 13, No.6 (June 1970).

[Gladun 2003] Gladun V. Intelligent Systems Memory Structuring // International Journal: Information Theories and Applications.-2003, V.10, №1.

[Markov 1984] Kr.Markov. A Multi-domain Access Method. // Proceedings of the International Conference on Computer Based Scientific Research. Plovdiv, 1984. pp. 558-563.

[Markov et al 2003] Kr.Markov, Kr.Ivanova, I.Mitov. General Information Theory. Basic Formulations. // FOI-COMMERCE, Sofia, 2003.

Author Information

Krassimir Markov – Institute of Mathematics and Informatics, BAS, P.O. Box: 775, Sofia-1090, Bulgaria; e-mail: foi@nlcv.net

ALGORITHM BIDIMS FOR AUTOMATED SYSTEMATIZATION OF DATA ARRAY. CASE STUDY: REDISCOVERING MENDELEEV'S PERIODIC TABLE OF CHEMICAL ELEMENTS.

Andrey Zagoruiko and Nikolay Zagoruiko

Abstract: The method (algorithm BIDIMS) of multivariate objects display to bidimensional structure in which the sum of differences of objects properties and their nearest neighbors is minimal is being described. The basic regularities on the set of objects at this ordering become evident. Besides, such structures (tables) have high inductive opportunities: many latent properties of objects may be predicted on their coordinates in this table. Opportunities of a method are illustrated on an example of bidimentional ordering of chemical elements. The table received in result practically coincides with the periodic Mendeleev table.

Keywords: bidimensional structure, data mining, ordering, prediction, approximation.

Introduction

One of Data Mining purposes consists in reduction of the available data to such kind at which the person easily perceives the basic contents of the analyzed information. The information advanced in such a way becomes a little more substantial for the person. The specified purpose is achieved by different means. The important role plays machine graphics, allowing the person to perceive the information through the powerful visual analyzer. Various ways of ordering at which one-dimensional or bi-directional data file will be coordinated well with the simple concepts (models) have a wide circulation. One-dimensional line of smoothly growing or decreasing numerical values is evident, for example. Even more information is contained in bi-directional tables with monotonic change of data on the first and second coordinate. It is not surprising that many fundamental laws of a nature - the law of the Ohm, Newton, Mendel etc. are well illustrated by bi-directional data tables.

D.I. Mendeleev, studying dependencies between various properties of chemical elements, relied on many results of the predecessors. In particular, the grouping of some elements on a generality of their chemical properties was known, on similarity of nuclear weights etc. Mendeleev put the task before itself to make such bi-dimensional ordering of all 63 elements known at that time at which the neighbouring elements of the table would be similar to each other on maximum big set chemical and physical properties. Such their arrangement will be coordinated with the concept of local smoothness, which provides easy perception of the general laws for the table.

The use of concepts beforehand prepared or the models is useful not only for an explanation of the data analysis result, but also for the process of this result receiving. The rich history of scientific discoveries speaks about it, in many of which traces of attempts of such type are obviously visible: " - It seems these objects have a S-structure. And what if to try use such a model? And what if to order the objects by such rule?"

Let's illustrate the utility of the initial empirical data association with the simple concepts during revealing the laws latent in data. We'll do it on an example of automatic rediscovering of the periodic law of chemical elements. Such attempt was already done by us [1], but in a little bit idealized conditions. In particular, true nuclear weights and valences on hydrogen, instead of those known to Mendeleev were used; the program was adjusted to the fixed number of properties etc. In the given work those data and knowledge, which D.I. Mendeleev had at his disposal during the creation of the periodic law of chemical elements, are used.

Description of Systematization Algorithm

Let us imagine that we have some data array, consisting of n elements A, each of them characterized by a set of k properties. In the current formulation the task is to systematize this array in a form of a two-dimensional table with internal uniformity in changing of element properties in both dimensions.

Let us also propose that we have some «embryo» of the table, i.e. credible group of small amount of elements, constructed on the base of researcher intuition. Having such a group it is possible to try to predict the properties of neighbouring elements. Assuming the uniformity of elements properties changing inside the «embryo» in both dimensions, this procedure may be performed using linear approximations. Possible types of such approximations are given below:

Internal approximation may be applied if it is necessary to predict the properties of the element, situated between two elements provided that their positions in the table are already known. In this case if known elements are situated in the table at coordinates, for example, *(i+1,j)* and *(i-1,j)*, then the value of *m-th* property in position (i,j) P_{ij}^m may be predicted using the equation

$$P_{ij}^m = \frac{P_{i-1j}^m + P_{i+1j}^m}{2} \quad (1)$$

Similar estimations may be obtained as well for combinations *(i+1,j+1)* и *(i-1,j-1), (i+1,j-1)* и *(i-1,j+1), (i,j+1)* и *(i,j-1)*, i.e. via the horizontal, vertical and two diagonals (total – 4 variants).

External approximation is used when it is necessary to construct the forecast for the table cell adjacent to the pair of situated elements with known positions. For example, if known elements are placed in the cells with coordinates *(i-2,j)* and *(i-1,j)*, then value of P_{ij}^m may be predicted as follows:

$$P_{ij}^m = 2P_{i-1j}^m - P_{i-2j}^m \quad (2)$$

Here the predictions also can be made via horizontals, verticals and diagonals (total – 8 variants).

Corner approximation is applied when known elements are placed in the table in form of “corner”, for example, in the cells with coordinates *(i+1,j), (i,j+1)* and *(i+1,j+1)*. In this case it is necessary to use the equation

$$P_{ij}^m = P_{i+1j}^m + P_{ij+1}^m - P_{i+1j+1}^m \quad (3)$$

Four variants of “corner” positions are possible. The final prediction of the property P_{ij}^m is defined as averaged value of all forecasts according to equations, which total number may reach 16.

The next step of the procedure is selection of optimal “pretending” element from the set of remaining elements, which are not positioned in the table. The running through of all remaining elements in relation to every empty cell of the table is made with definition of the positioning quality of every element/cell combination. The modulus of deviation between predicted and real element property values may be used as quality criterion:

$$X_{ij}^h = \sum_{m=1}^{k} abs(P_{ij}^m - R_h^m) \quad (4)$$

where X_{ij}^h - quality criterion of *h-th* element in the cell with coordinates *(i,j),* R_h^m - real value of *m-th* property for this element. After running of every elements versus every table cell, the table is filled with <u>only one</u> element, which at all set of *h, i* and *j* is characterized with minimum value of X_{ij}^h. This procedure is repeated until completion of positioning of all initial elements.

The problem may be complicated by two factors. Firstly, range of parameter values may be significantly different for different properties, resulting in different contribution of each property to the value of X_{ij}^h criterion and, therefore, leading to their «inequality of rights». Secondly, it is not evident that every element is described by a full set of properties, so property array actually may include missing values. Correspondingly, the reversed situation is possible as well, when the definite element property is really present, but cannot be predicted, because it is not present in the property sets of elements used for prediction.

First complication is easily solved by normalization of data for each of properties. In the second case it becomes necessary to define for each cell and each element how many properties area really predicted and then normalize criterion X_{ij}^h value as follows:

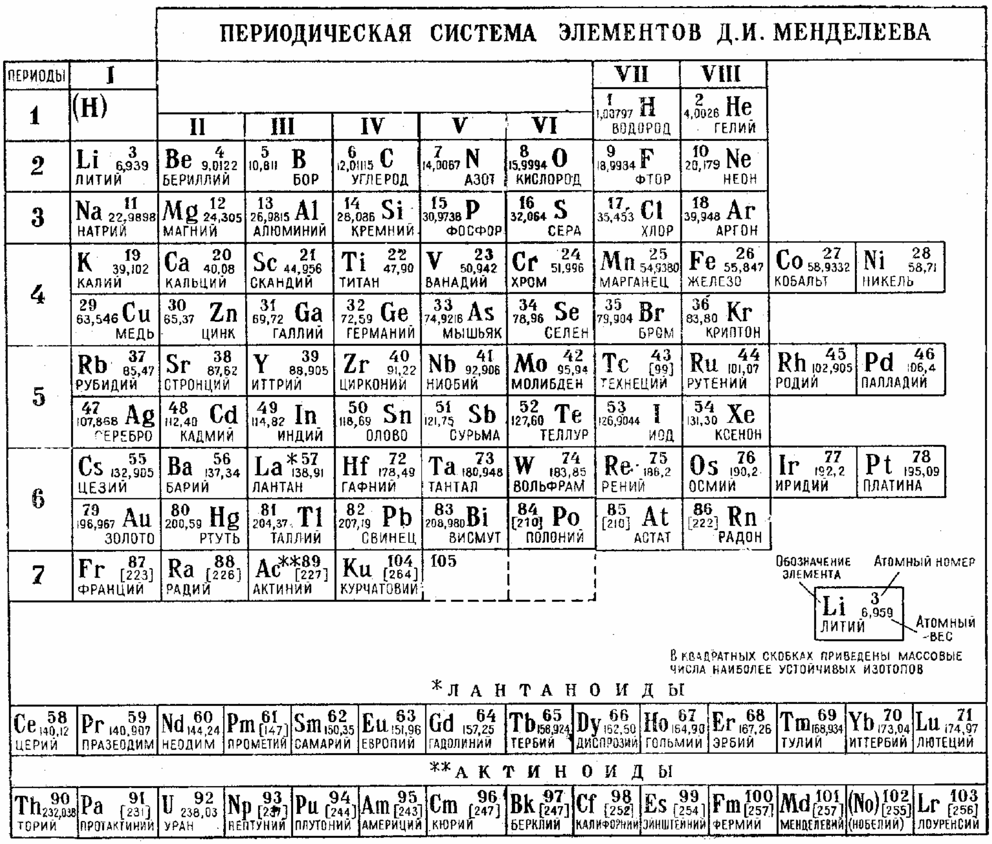
$$X_{ij}^{h} = \frac{\sum_{m=1}^{k} abs(P_{ij}^{m} - R_{h}^{m})}{Z_{ij}^{h}} \quad (5)$$

where Z_{ij}^{h} - number of coincidence «successfully predicted property» / «presence of that property in the element property set» under attempt to place the *h-th* element in the cell with coordinates *(i,j)*. Moreover, as it was demonstrated by test calculations, to improve systematization quality it is better to give preference to elements with higher value of Z_{ij}^{h}, as more reliably determined. Such preference may be realized by different ways, but in this work we used empirical method, based on application of Z_{ij}^{h}, raised to a power higher than one. Particularly, the optimal order value was found to be 3, i.e equation (5) was transformed into:

$$X_{ij}^{h} = \frac{\sum_{m=1}^{k} abs(P_{ij}^{m} - R_{h}^{m})}{(Z_{ij}^{h})^{3}} \quad (6)$$

Systematization of a Full Set of Chemical Elements

As a case study we used a complete set of chemical elements (see Fig.1), e.i. the essential aim of the work was reopening of periodic law, discovered by Dmitry Mendeleev in 1869.



ПЕРИОДИЧЕСКАЯ СИСТЕМА ЭЛЕМЕНТОВ Д.И. МЕНДЕЛЕЕВА

ПЕРИОДЫ	I	II	III	IV	V	VI	VII	VIII		
1	(H)						H 1 1,00797 ВОДОРОД	He 2 4,0026 ГЕЛИЙ		
2	Li 3 6,939 ЛИТИЙ	Be 4 9,0122 БЕРИЛЛИЙ	B 5 10,811 БОР	C 6 12,01115 УГЛЕРОД	N 7 14,0067 АЗОТ	O 8 15,9994 КИСЛОРОД	F 9 18,9984 ФТОР	Ne 10 20,179 НЕОН		
3	Na 11 22,9898 НАТРИЙ	Mg 12 24,305 МАГНИЙ	Al 13 26,9815 АЛЮМИНИЙ	Si 14 28,086 КРЕМНИЙ	P 15 30,9738 ФОСФОР	S 16 32,064 СЕРА	Cl 17 35,453 ХЛОР	Ar 18 39,948 АРГОН		
4	K 19 39,102 КАЛИЙ	Ca 20 40,08 КАЛЬЦИЙ	Sc 21 44,956 СКАНДИЙ	Ti 22 47,90 ТИТАН	V 23 50,942 ВАНАДИЙ	Cr 24 51,996 ХРОМ	Mn 25 54,9380 МАРГАНЕЦ	Fe 26 55,847 ЖЕЛЕЗО	Co 27 58,9332 КОБАЛЬТ	Ni 28 58,71 НИКЕЛЬ
	Cu 29 63,546 МЕДЬ	Zn 30 65,37 ЦИНК	Ga 31 69,72 ГАЛЛИЙ	Ge 32 72,59 ГЕРМАНИЙ	As 33 74,9216 МЫШЬЯК	Se 34 78,96 СЕЛЕН	Br 35 79,904 БРОМ	Kr 36 83,80 КРИПТОН		
5	Rb 37 85,47 РУБИДИЙ	Sr 38 87,62 СТРОНЦИЙ	Y 39 88,905 ИТТРИЙ	Zr 40 91,22 ЦИРКОНИЙ	Nb 41 92,906 НИОБИЙ	Mo 42 95,94 МОЛИБДЕН	Tc 43 [99] ТЕХНЕЦИЙ	Ru 44 101,07 РУТЕНИЙ	Rh 45 102,905 РОДИЙ	Pd 46 106,4 ПАЛЛАДИЙ
	Ag 47 107,868 СЕРЕБРО	Cd 48 112,40 КАДМИЙ	In 49 114,82 ИНДИЙ	Sn 50 118,69 ОЛОВО	Sb 51 121,75 СУРЬМА	Te 52 127,60 ТЕЛЛУР	I 53 126,9044 ИОД	Xe 54 131,30 КСЕНОН		
6	Cs 55 132,905 ЦЕЗИЙ	Ba 56 137,34 БАРИЙ	La *57 138,91 ЛАНТАН	Hf 72 178,49 ГАФНИЙ	Ta 73 180,948 ТАНТАЛ	W 74 183,85 ВОЛЬФРАМ	Re 75 186,2 РЕНИЙ	Os 76 190,2 ОСМИЙ	Ir 77 192,2 ИРИДИЙ	Pt 78 195,09 ПЛАТИНА
	Au 79 196,967 ЗОЛОТО	Hg 80 200,59 РТУТЬ	Tl 81 204,37 ТАЛЛИЙ	Pb 82 207,19 СВИНЕЦ	Bi 83 208,980 ВИСМУТ	Po 84 [210] ПОЛОНИЙ	At 85 [210] АСТАТ	Rn 86 [222] РАДОН		
7	Fr 87 [223] ФРАНЦИЙ	Ra 88 [226] РАДИЙ	Ac **89 [227] АКТИНИЙ	Ku 104 [264] КУРЧАТОВИЙ	105					

*ЛАНТАНОИДЫ

Ce 58 140,12 ЦЕРИЙ	Pr 59 140,907 ПРАЗЕОДИМ	Nd 60 144,24 НЕОДИМ	Pm 61 [147] ПРОМЕТИЙ	Sm 62 150,35 САМАРИЙ	Eu 63 151,96 ЕВРОПИЙ	Gd 64 157,25 ГАДОЛИНИЙ	Tb 65 158,924 ТЕРБИЙ	Dy 66 162,50 ДИСПРОЗИЙ	Ho 67 164,930 ГОЛЬМИЙ	Er 68 167,26 ЭРБИЙ	Tm 69 168,934 ТУЛИЙ	Yb 70 173,04 ИТТЕРБИЙ	Lu 71 174,97 ЛЮТЕЦИЙ

**АКТИНОИДЫ

Th 90 232,038 ТОРИЙ	Pa 91 [231] ПРОТАКТИНИЙ	U 92 238,03 УРАН	Np 93 [237] НЕПТУНИЙ	Pu 94 [244] ПЛУТОНИЙ	Am 95 [243] АМЕРИЦИЙ	Cm 96 [247] КЮРИЙ	Bk 97 [247] БЕРКЛИЙ	Cf 98 [252] КАЛИФОРНИЙ	Es 99 [254] ЭЙНШТЕЙНИЙ	Fm 100 [257] ФЕРМИЙ	Md 101 [257] МЕНДЕЛЕВИЙ	(No) 102 [255] (НОБЕЛИЙ)	Lr 103 [256] ЛОУРЕНСИЙ

Fig.1. Short-period table of elements

At the first stage we performed test calculations with application of full set of chemical elements, known at the current moment. In case when the set of three basic properties (atomic mass, group number and period number) the described algorithm provided fast and correct solution. Of course, application of group and period numbers was equivalent to inclusion of already known correct solution into initial data. Therefore, this variant was for software testing purposes only.

On the second stage we used a set of three basic properties, that were known to Mendeleev and which were used by him in his work on periodic law construction: atomic mass, oxygen and hydrogen valence. As "embryos" we used intuitive combinations which look, nevertheless, quite obvious from chemical point of view, such as:

Na			O	F
K	Ca	or	S	Cl

In this case it was possible to successfully construct the "framework" of the table, where periodicity and uniformity of properties changing are evident, namely – 2nd and 3rd periods. The following table filling met significant complications, mainly for transitional element and, especially, for triads Fe-Co-Ni, Ru-Rh-Pd, Os-Ir-Pt and inert gases. Correct positioning of lanthanides and actinides was found to be completely impossible.

At the same time some interesting regularities were discovered. It was found that even one erroneous positioning of an element leads to the chain of further errors, the sooner the error is made the more significant distortions are contributed to the final result. It was also detected that table construction quality (quite logically) depends very much upon the choice of initial "embryo".

Explanations of all these problems are rather simple. Periodicity and uniformity of changing of atomic mass looks evident (Fig.2), at least if will not consider natural mass gap in the area of lanthanides placement (this gap is absent in a long-period table). At the same time the picture for oxygen and hydrogen valences is much more complicated (Figs. 3 and 4).

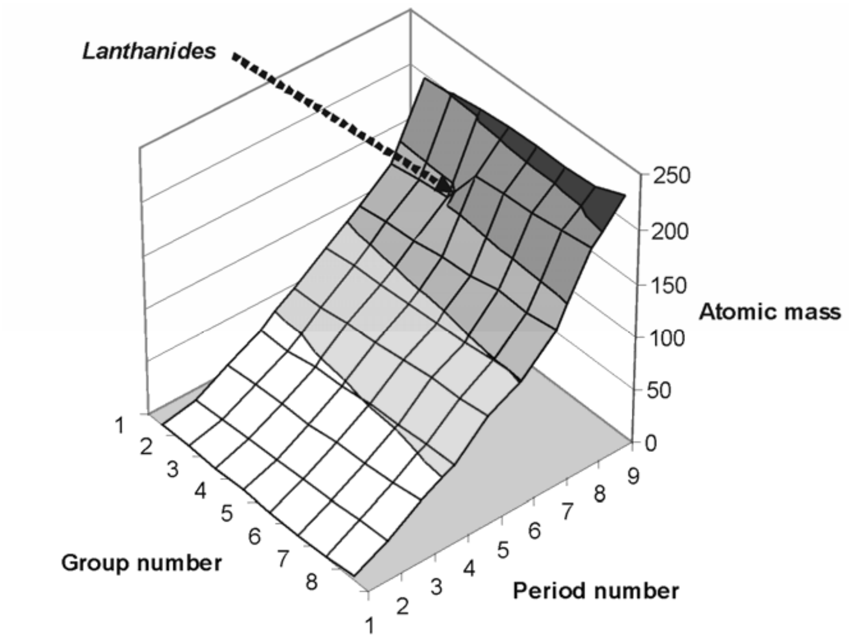


Fig.2. Changing of atomic mass of elements in groups and periods of "short" periodic table.

Good periodicity is observed for hydrogen valence (Fig.4), but this data is present for less than a half of elements. Furthermore, continuity of data is present in 2nd and 3rd periods only, and in higher periods the periodicity is broken by transitional elements (i.e. is repeated "a string after").

Majority of oxygen valence data (Fig.3) is fit into irreproachable flat plane, but with significant anomalies at the table periphery, notably:

- decrease of observed valence in triads Fe (6+) - Co(3+) - Ni(2+), Ru(8+) – Rh (4+) – Pd(2+), Os (8+) – Ir(4+) – Pt(4+);

- zero valence for inert gases (except Xe(8+) and Kr(2+));

- high valences of copper (2+) and gold (3+) instead of expected (1+).

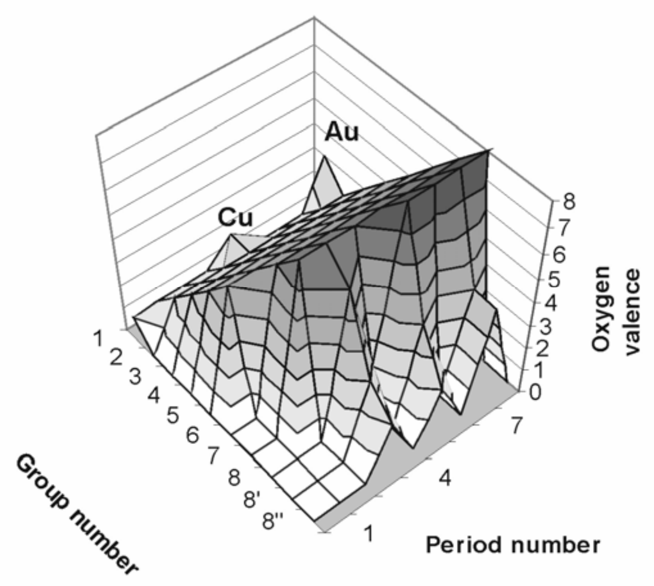


Fig.3. Changing of oxygen valence.

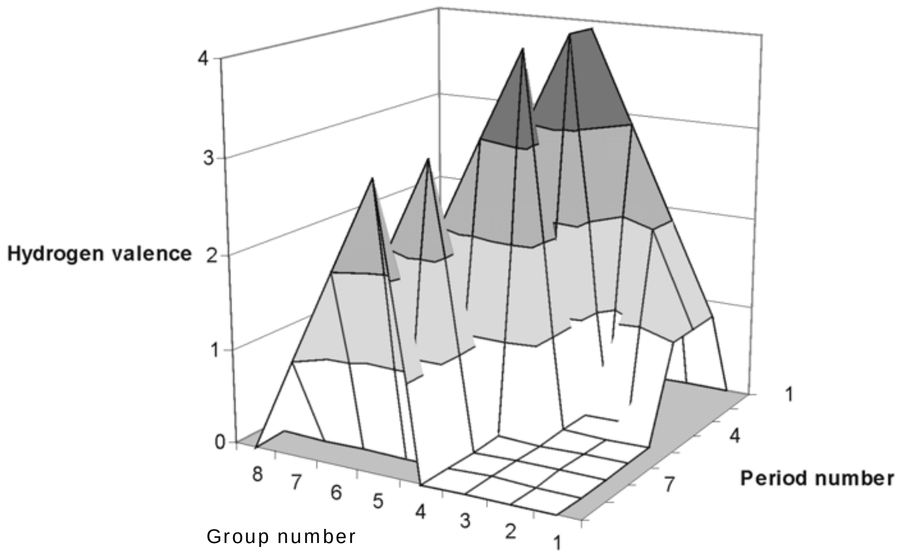


Fig..4. Changing of hydrogen valence.

In general it may be stated that in relation to mentioned properties the data array in Mendeleev's table is not uniform, as it was proposed at the stage of problem formulation. Nevertheless, the attempts were made to "smooth" these nonuniformities by application of greater number of properties in element descriptions. The properties that were definitely known to Mendeleev during his work on Periodic Law (densities, melting and boiling temperatures for elements and their oxides and chlorides; acid/base properties of oxides etc) were chosen to expand data array.

Surprisingly, this attempt was even less successful. The reason for this fault may be demonstrated on the base of element density changing (Fig.5).

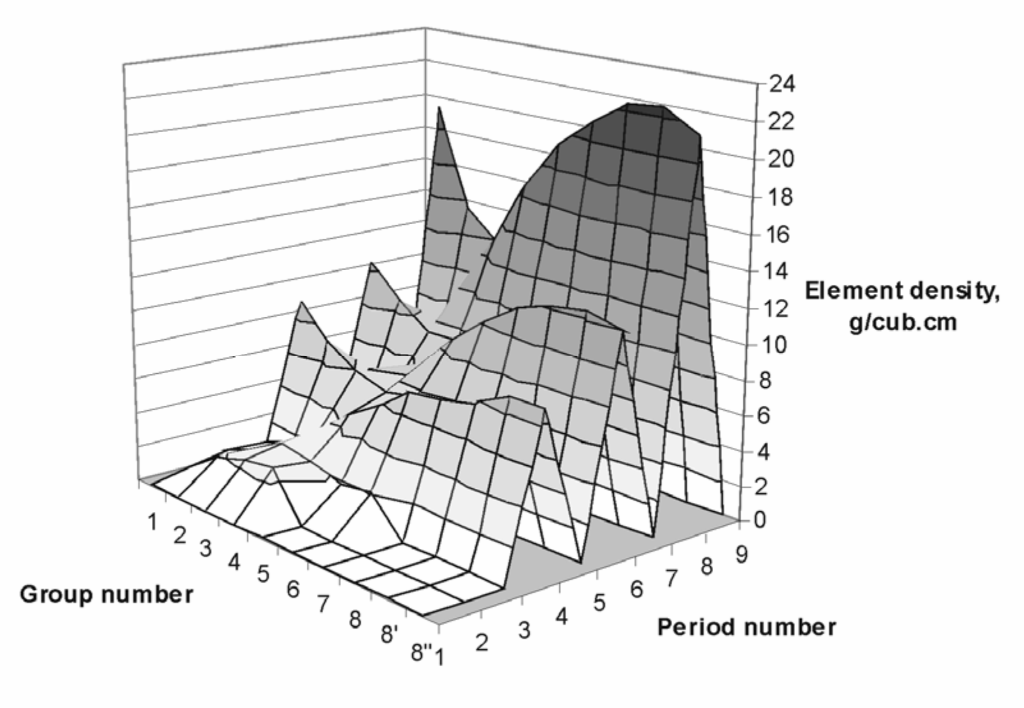


Fig.5. Changing of density of elements in groups and periods of “short” periodic table.

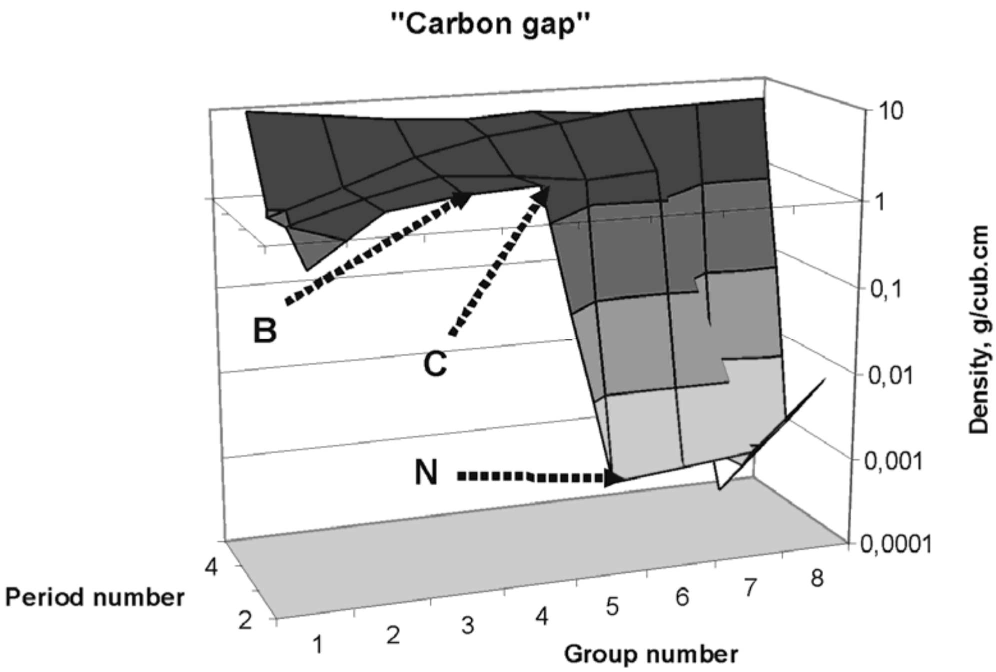


Fig.6. Elements density gap in the 2nd period.

Fig.5 shows strong nonlinearity and nonuniformity of plotted surface. Notably, such oscillations are typical not for densities only, but for all other mentioned properties as well. In principle, if we will switch to long-period table, then these oscillations will become smoother, but such switching is incorrect, because it will require choice of elements from 4th and higher periods for formation of the "embryo", what looks unobvious. Construction of table from more reasonable "embryo" elements of 2nd and 3rd periods will in any case lead to short-period type of the table.

Moreover, there also exists the problem that cannot be resolved even by transfer to long-period table. Let us consider the fragment of Fig.5 related to 2nd period. Here (Fig.6) it is seen that between starting elements in this period (Li, Be, B, C) and further sequence (starting from N) there is a gap in densities (by a few orders of magnitude – values in vertical axis are given in logarithmic scale). Similar anomaly behaviour is also observed for other physical properties. Gap position may shift from C/N to C/B zone (e.g. for melting and boiling temperatures of oxides), but is always connected with carbon. Existence of such anomaly ("carbon gap") finally leads to impossibility to construct correctly even the 2nd period of the table and to completely absurd construction of following periods.

Therefore, expansion of data array by introduction of additional physics-chemical properties only decreases the quality of systematization process. Of course, it is possible to try using modern elements characteristics (atomic electron configuration, ionization potential, electro negativity etc.), but it looks incorrect, because this information was not available to Dmitry Mendeleev, and, besides, in this case the data array will artificially contain correct solution, thus "killing intrigue" of this study.

Systematization of "Mendeleev's" Set of Elements

Though the systematization of complete set of elements was unsuccessful, in this pert of the study we tried to systemize the elements set that was known to Mendeleev during his work. This set is different from complete one in following:

- inert gases are completely absent (they were discovered later);
- majority of lanthanides and actinides, as well as heavy elements (heavier this bismuth) are absent;
- few elements from middle periods are also undiscovered (Sc, Ga, Ge).

Furthermore, for some elements Mendeleev has doubtful and incorrect data for atomic weights and valences. Due to aforementioned reasons here we used only data on atomic mass and valences. It should be noted that Mendeleev also used this information as basic in table construction process.

Surprisingly, it was found that in this case systematization is much more simple than that for complete set of elements. First of all, it is explained by absence of inert gases, which actually are placed in the table quite illogically (none of them, except xenon, demonstrate 8+ oxygen valence, which is predicted for this group). Additional advantage is absence of lanthanides, because their position in short-period table also looks quite unusual.

In this case we've managed to reproduce the major part of the table, but here as well we met the effect of "wrong" behaviour in triads of transitional elements. For example, cobalt (quite logically) was "trying" to fill the cell of absent gallium due to coincidence of maximum oxygen valence (3+) with relatively low error in prediction of atomic mass. From chemical point of view it is absolutely evident that cobalt is an analogue of iron and nickel, but not aluminum (as gallium), but such "chemical" understanding cannot be described in data array within existing data structure. Anyway, incorrect cobalt positioning led to distortions in further construction of the table. The same may be told about triads of noble metals.

Therefore, we've attempts to modify initial data array, based on exclusion of the most "odious" elements, particularly:

- all elements from transitional triads, except first ones (Fe, Ru, Os), were excluded;
- present lanthanides were excluded (except La and Ce only).

Moreover, for copper and gold the basic oxygen valence 1+ was stated, though their actual maximum valences are higher (2+ for Cu and 3+ for Au). It was done to reveal the fact that Cu and Au are analogues of silver.

The result of systematization in this case is shown in Fig.7.

Period	Group							
							H	
	Li	Be	B	C	N	O	F	
	Na	Mg	Al	Si	P	S	Cl	
	K	Ca		Ti	V	Cr	Mn	Fe
	Cu	Zn			As	Se	Br	
	Rb	Sr	Y	Zr	Nb	Mo		Ru
	Ag	Cd	In	Sn	Sb	Te	I	
	Cs	Ba	La	*Ce*	Ta	W		Os
	Au	Hg	Tl	Pb	Bi	*U*		
				Th				

Fig.7. Result of systematization of “Mendeleev’s” set of elements.

It is seen that systematization quality is quite high. Practically all elements are placed in cells, where they should be. Exclusion is made by uranium which actually should be situated in V-7 cell after thorium, but this error is not important (Th and U actually should be placed in separate subgroup of actinides, which is stipulated in this type of the table). Placement of thorium and cerium also does not look formally correct, but actually it is quite usual for them to demonstrate 4+ valences, what gives the ground to position them in the IV-th group of basic table. Such their dual behaviour is well known and is defined by objective specifics of their electronic structure, so such placement may be accepted as appropriate. It is curious, that D.I. Mendeleev the same as also our program, has placed in the initial kind of the table Thorium and Cerium in 4-th group. Moreover, in the same group he has placed and Lantan [2,3]. We shall note that our program has placed Lantan on a correct place in third group.

Prediction of Undiscovered Elements Properties

Special attention should be paid to prediction abilities of the constructed table. As it is seen from Fig.7, after systematization few cells inside the table were left unfilled. These cells strongly correspond to existing elements, that were undiscovered at the time of Mendeleev’s study. To predict the properties of missing elements we used the described algorithm. In this case only obtained data values being inside the normalized range ([0,1]) were chosen.

Result of such prediction is quite impressive. First of all, 5 elements that must be positioned inside the table body were clearly shown (their positions in the table are shown at Fig.7 by crossed cells). Description of predicted values is given in Table 1.

Table.1

Position column/string	Atomic mass		Oxygen valence		Hydrogen valence		Real element
	forecast	fact	forecast	fact	forecast	fact	
3/4	43,90	44,95	3	3	-	-	Sc
4/5	69,00	72,59	4	4	4	4	Ge
3/5	65,20	69,72	3	3	-	-	Ga
7/6	101,10	98,91	7	7	1	-	Tc
7/8	176,80	186,20	7	7	-	-	Re

It is seen that coincidence between predicted and actual property values is quite good. Moreover, during analysis of predictions that were excluded from consideration, because the predicted values were found to be outside normalized range, we selected the group of similar elements, which formally should have been positioned in 8th group and have formal valences 8+ for oxygen and 0 for hydrogen (shown by shadowed cells at Fig.7). The reason of their exclusion was zero hydrogen valences, what was considered as inappropriate property value. Actually these predictions are strongly equivalent to the group of inert gases and good coincidence between predicted and actual properties is seen here as well (see Table.2). Furthermore, "wrong" prediction of zero hydrogen valences in this case achieves real physical sense – inert gases do not form hydrogen compounds in reality.

Table.2

Position column/string	Atomic mass		Oxygen valence		Real element
	forecast	fact	forecast	fact	
8/1	6,00	4,00	8	-	He
8/2	20,05	20,18	8	-	Ne
8/3	35,80	39,95	8	-	Ar
8/5	80,30	83,80	8	2	Kr
8/7	138,80	131,30	8	8	Xe

It is interesting that after selection of predictions no forecasts were made non-existent elements, i.e. the algorithm has made no attempts to fill empty cells of 1st period and cells to the right and to the left of the table body.

Conclusion

In general we may state that proposed algorithm BIDIMS (under definite assumptions and modifications) successfully managed to systemize "Mendeleev's" set of elements and, in fact, repeated the discovery of Periodic Law in a form, which was possible in Mendeleev's work period. Performed study, nevertheless, is not diminishing Mendeleev's achievements in any extent. First of all, used assumptions and modifications were based on intuitive and forced decisions, having no formally strong grounds. In second, the basic decisive properties (atomic mass, valences) were chosen the same as ones used by Mendeleev. And the most important – the essence of genius Mendeleev's discovery is proposition, that existing element may be systemized in form of two-dimensional table. We used this proposition as acknowledged fact in our study.

Acknowledgments

Russian Fund of Basic Researches supported the work, grant № 02-01-00082

Bibliography

1. A.N. Zagoruiko and N.G. Zagoruiko. Experiments on Rediscovering of Mendeleev's Low by Using Computer // Strukturnyi Analiz Simvol'nikh Posledovatel'nostei (Vychislitel'nye Sistemy 101, Ak. Nauk SSSR, Novosibirsk, 1984), p. 82-90. (In Russian).
2. A.A. Makarenia. D.I. Mendeleev and physicochemical sciences. Atomizdat. Moscow, 1972, 256 p. (In Russian).
3. http://chemlab.pc.maricopa.edu/periodic/foldedtable.html
4. N.G. Zagoruiko. Applied Methods of Data and Knowledge Mining. Ed. Inst. of Mathematics. Novosibirsk, 1999. 268 p. (In Russian).

Authors' Information

Andrey N. Zagoruiko – Institute of Catalysis of SD RAS, Russia, 630090, Novosiborsk, pr. Lavrentieva, 5
e-mail: zagor@catalysis.nsk.su
Nikolay G. Zagoruiko – Institute of Mathematics SD RAS, Russia, 630090, Novosibirsk, pr. Koptyug, 4
e-mail: zag@math.nsc.ru

SENSITIVITY AND BIAS WITHIN THE BINARY SIGNAL DETECTION THEORY, BSDT

Petro Gopych

Abstract: *Similar to classic Signal Detection Theory (SDT), recent optimal Binary Signal Detection Theory (BSDT) and based on it Neural Network Assembly Memory Model (NNAMM) can successfully reproduce Receiver Operating Characteristic (ROC) curves although BSDT/NNAMM parameters (intensity of cue and neuron threshold) and classic SDT parameters (perception distance and response bias) are essentially different. In present work BSDT/NNAMM optimal likelihood and posterior probabilities are analytically analyzed and used to generate ROCs and modified (posterior) mROCs, optimal overall likelihood and posterior. It is shown that for the description of basic discrimination experiments in psychophysics within the BSDT a 'neural space' can be introduced where sensory stimuli as neural codes are represented and decision processes are defined, the BSDT's isobias curves can simultaneously be interpreted as universal psychometric functions satisfying the Neyman-Pearson objective, the just noticeable difference (jnd) can be defined and interpreted as an atom of experience, and near-neutral values of biases are observers' natural choice. The uniformity or no-priming hypotheses, concerning the 'in-mind' distribution of false-alarm probabilities during ROC or overall probability estimations, is introduced. The BSDT's and classic SDT's sensitivity, bias, their ROC and decision spaces are compared.*

Keywords: *binary signal detection theory, sensitivity, bias, ROC, mROC, overall likelihood and posterior, neural space, psychometric function, just noticeable difference (jnd), uniformity or no-priming hypotheses.*

1. Introduction

Since D.Green & J.Swets' pioneering book [1], classic Signal Detection Theory (SDT) is widely used in psychology for describing different discrimination experiments concerning the study of human/animal sensory and memory abilities. Further developments were summarized by N.A.Macmillan & C.D.Creelman whose monograph [2] reviews the state of the art in this field. Since 1960th the SDT's productivity was successfully demonstrated in numerous experiments performed using different experimental paradigms and for this reason it became extremely popular as a tool for analysis and interpretation of data in sensory and cognitive psychology.

Of course, not all SDT's applications are equally successful and this fact plays the role of an impetus for further SDT development and for designing its new, sometimes technically sophisticated although not always perfect, versions. But, perhaps, the main SDT's disadvantage is conceptual rather than technical: its decision rules act in a so called psychological decision space – the hypothetical space, quite separate from the world of stimuli and having unclear relations to it; decision space is deliberately introduced to define internal (mental) stimulus representations, objects of the SDT. Moreover, it is unclear whether specific relations between stimulus space (world) and the SDT's decision space can be one day discovered even in principle.

In present work using complete numerical examples basic notions and parameters of the optimal Binary (Binomial) SDT (BSDT) [3] are investigated and compared with corresponding notions and parameters of classic SDT [2]. The main two distinctions between BSDT and SDT approaches are emphasized. The first is technical: in contrast to the SDT's continues (Gaussian) probability distributions, BSDT operates with discrete (binomial) probability distributions and for this reason all its predictions are discrete. The second is conceptual: in contrast to the SDT's two separate spaces (the stimulus space and psychological decision space), BSDT defines for stimuli and decisions their common 'neural space' where stimuli are represented as unified neural codes (N-dimensional binary vectors) and decisions as operations over these codes. Furthermore, it may be expected that in the future all objects in the neural space could be related to sensory stimuli using the methods of neuroscience.

2. About the BSDT

As it has already been demonstrated [3], for a binary data coding initially proposed in ref. [4] there exist three data decoding algorithms (neural network, convolutional, and Hamming distance) which have equivalent, and the best in the sense of statistical patterns recognition quality, performance. For such a decoding algorithm that is equivalent to an intact two-layer neural network (NN) operating with a neuron threshold θ, its quality performance were derived [3] as analytical expressions for exact calculation of the probability (likelihood) $L(d,\theta)$ of the best correct decoding N-dimensional binary vectors $x = x(d)$ with components ±1 and a given intensity of cue $q = 1 - d$, $d = m/N$, where m is the number of noise components of x and $N - m$ is the number of intact components of x_0 among the components of x [reference vector x_0 contains the information stored or that should be stored in the NN and, consequently, $L(d,\theta)$ is also the probability of correct recognition of x_0 in $x(d)$] [3]. It is important that BSDT and recent Neural Network Assembly Memory Model (NNAMM) are mathematically similar (NNAMM is in fact a direct implementation of the BSDT for solving the problem of memory storing/retrieval) and some their basic parameters are the same [5]. The similarity between the BSDT and NNAMM (between their mathematical tools) is important as in many cases it allows do not watch for distinctions between patterns' coding/decoding and storing/retrieving and consider these processes using their common, BSDT/NNAMM, point of view.

We refer to ref. 3 and 5 for some BSDT/NNAMM details and pay here the main attention only to those BSDT's parameters which are needed to derive the BSDT's counterparts to sensitivity and bias of the classic SDT. For simplicity, to exclude the consideration of splitting the probability functions $L(d,\theta)$ [3], in this work we shall discuss only the case of odd N, i.e. the case of an odd number of the NN's entrance- or exit-layer neurons. Now we only rewrite the expression for the probability of correct decoding $L(d,\theta) = L(m,N,\theta)$ [3] using a new parameter, the threshold interval index Θ, introduced in Section 3 and compared with other BSDT's parameters in Table 1:

$$L(m,N,\Theta) = \sum_{k=0}^{k\max} C_k^m / 2^m \qquad (1)$$

where if $kmax \le kmax_0$ then $kmax = m$ else $kmax = kmax_0$; for odd N $kmax_0 = (N - \Theta - 1)/2$ where Θ is even, $-(N + 1) \le \Theta \le N - 1$, and $\Delta\Theta = 2$ with a special case if $\Theta = N + 1$ then $L(m,N,\Theta) = L(d, \theta) = 0$.

3. Relations between Some BSDT Parameters

For the case of odd N Table 1 and Figures 1 and 2 illustrate relations between some BSDT's parameters (rather large amount of them is in particular caused by the fact that the decoding algorithm has three different forms).

Table 1

Relations between some BSDT parameters for the case N = 9, as in our works [3,6][*)]

i [a)]	$\Delta\theta_i$ [b)]	Q_i [c)]	D_i [d)]	ρ_i [e)]	Θ_i [f)]	F_i [g)]	ΔF_i [h)]
–1	[9, +∞)	9	0	9/9	10	0/512 = 0.00000	–
0	[7, 9)	7, 8	1	7/9, 8/9	8	1/512 = 0.00195	1/512 = 0.00195
1	[5, 7)	5, 6	2	5/9, 6/9	6	10/512 = 0.01953	9/512 = 0.01758
2	[3, 5)	3, 4	3	3/9, 4/9	4	46/512 = 0.08984	36/512 = 0.07031
3	[1, 3)	1, 2	4	1/9, 2/9	2	130/512 = 0.25391	84/512 = 0.16406
4	[–1, 1)	–1, 0	5	–1/9, 0/9	0	256/512 = 0.50000	126/512 = 0.24606
5	[–3, –1)	–3, –2	6	–3/9, –2/9	–2	382/512 = 0.74609	126/512 = 0.24606
6	[–5, –3)	–5, –4	7	–5/9, –4/9	–4	466/512 = 0.91016	84/512 = 0.16406
7	[–7, –5)	–7, –6	8	–7/9, –6/9	–6	502/512 = 0.98047	36/512 = 0.07031
8	[–9, –7)	–9, –8	9	–9/9, –8/9	–8	511/512 = 0.99805	9/512 = 0.01758
9	(–∞, –9)	–	–	–	–10	512/512 = 1.00000	1/512 = 0.00195

*) N is simultaneously the dimension of binary vectors $x = x(d)$, the number of the NN's entrance- and exit-layer neurons, maximal amount of bits of information may be conveyed by vectors x, the NN's information capacity in bits, and the length of convolutional interval for the NN convolutional decoding algorithm [6].

a) The number of a neuron threshold interval $\Delta\theta_i$, false alarm F_i, and etc, $i = -1,0,1,2,\dots,N$.

b) The ith neuron threshold interval $\Delta\theta_i = [N-2(i+1), N-2(i+1)+2)$, $i = 0,1,2,\dots,N-1$; $\Delta\theta_{-1} = [N, +\infty)$, $\Delta\theta_N = (-\infty, -N)$; $[\theta_{left}, \theta_{right})$ means $\theta_{left} \le \theta < \theta_{right}$, $(\theta_{left}, \theta_{right})$ means $\theta_{left} < \theta < \theta_{right}$ (θ_{left} and θ_{right} are left-most and right-most points of an interval $\Delta\theta$); magnitudes of neuron thresholds are continuos, $-\infty < \theta < +\infty$; each $\Delta\theta_i$ is so defined that for all $\theta \in \Delta\theta_i$ probabilities $L(d,\theta)$ are constant, $L(d,\theta) = L(q,\Theta_i)$.

c) The convolution, Q_i, between $x(d)$ and x_0 for the ith neuron threshold interval $\Delta\theta_i$, $i = -1,0,1,2,\dots,N-1$, $-N \le Q_i \le N$; Q and all other parameters in the table, except θ, are discrete variables; within their common range Q and θ are equivalent, $\theta = Q$.

d) The ith Hamming distance, $D_i = (N - Q_i)/2$; as D_i is integer, for each $\Delta\theta_i$ it may be defined unambiguously.

e) The ith correlation coefficient, $\rho_i = Q_i/N$; in the neuron threshold interval $\Delta\theta_{-1}$ parameters ρ_i, D_i, and Q_i exist only in a single point ($\theta = N = 9$), in $\Delta\theta_N$ they are not defined at all.

f) The ith neuron threshold interval index (for short, threshold interval index), $\Theta_i = N - 2i - 1$, $i = -1,0,1,2,\dots,N$; the distance between any two neighbor values of Θ_i is $\Delta\Theta_i = 2$; depending on the parity of N and taking into account that for each $\Delta\theta_i$ its Q_i and $Q_i + 1$ values produce the same value of $L(d,Q_i) = L(d,\Theta_i)$, the series of indices Θ_i is defined in such a way that $\Theta_i = 0$ is always among its items; indices Θ_i provide also a possibility to calculate the probability $L(d,\theta)$ in neuron threshold interval $\Delta\theta_{-1}$ where Q, D, and ρ are defined only in one point $\theta = N$, in $\Delta\theta_N$ where they are not defined at all and show $L(d,\theta = 0)$ explicitly.

g) The ith false-alarm probability $F_i = \sum C^N_k/2^N$ where $k = 0,1,\dots,i$, $C^N_k = N!/(N-k)!/k!$, $i = 0,1,2,\dots,N$, $F_{-1} = 0$; the value $F_N = 1$ is assigned for $\theta \in \Delta\theta_N$ where Q, D, and ρ are not defined.

h) The ith false-alarm probability interval $\Delta F_i = F_i - F_{i-1} = C^N_i/2^N$, $C^N_i = N!/(N-i)!/i!$, $i = 0,1,2,\dots,N$.

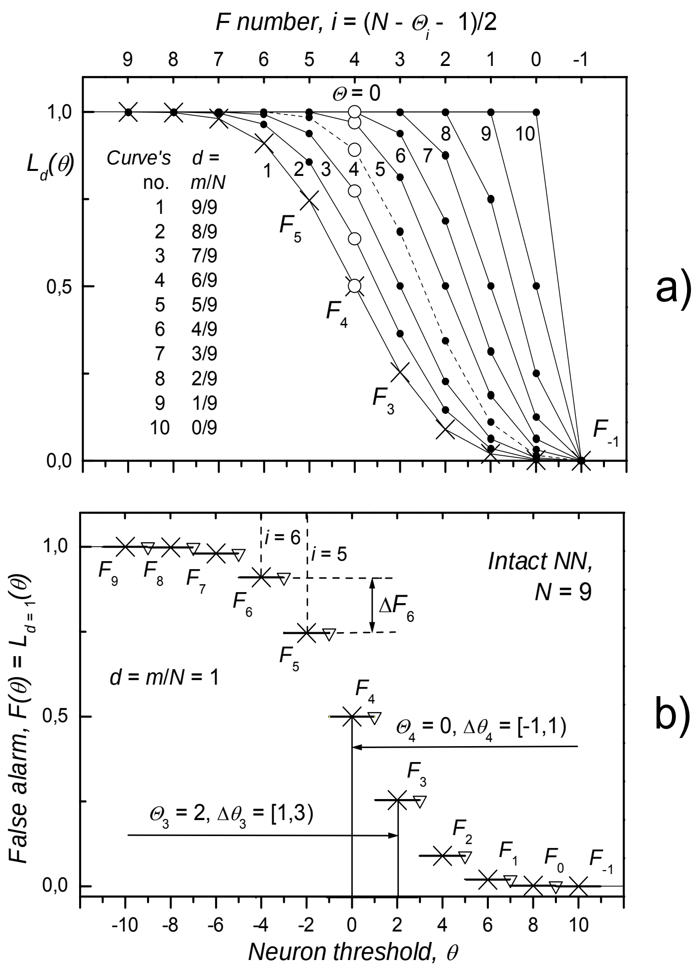


Figure 1. a) Correct decoding probability of vectors $x = x(d)$ (the probability of correct decoding of x under condition that it is x_0 damaged) or the likelihood $L(d,\theta) = P(A|H_1)$ (the probability of the event A under condition that hypotheses H_1 is valid, see Section 4) vs. the neuron threshold θ (lower scale) and index i, the number of F_i, Θ_i, or $\Delta\theta_i$ (upper scale). All values of $L_d(\theta) = L(d,\theta)$ were calculated according to Equation 1: crosses denote false-alarm probabilities (values of F_{-1}, F_3, F_4, and F_5 are marked), open circles denote $L_d(\theta)$ for near-zero neuron thresholds $\theta \in \Delta\theta_4 = [-1,1)$, $\Theta_4 = 0$ [designation $L_d(\theta)$ means that in $L(d,\theta)$ parameter d is fixed]. Probabilities $L_d(\theta)$ specified by a constant value of d are connected in straight lines. Here and in all next Figures values $L_d(\theta)$, $d = 6/9$, are connected in dashed lines. b) The same in more details (see text) but only for the function representing false-alarm probability, $F(\theta)$; crosses denote $F_i = F(\Theta_i)$, all F_i are marked; $\Delta F_6 = F_6 - F_5$ is the interval between two neighbors, F_5 and F_6; vertical lines designate indices $\Theta_3 = 2$ and $\Theta_4 = 0$.

To calculate $F(\theta)$ for different θ, we should posit in Equation 1 $m = N$, i.e. $d = 1$. In Figures 1a and 1b crosses are the same but from the panel *b*) it is seen that they correspond to the middles of neuron threshold intervals $\Delta\theta_i$ (i = 0, 1,…, N – 1) where for all $\theta \in \Delta\theta_i$ Equation 1 gives the same magnitude of the likelihood probability $L_d(\theta)$. As Figure 1b shows, $F(\theta)$ is a stepwise discontinues function where its horizontal line segments denote the constant value of $F(\theta)$ for θ belonging to corresponding threshold interval $\Delta\theta_i$, $\theta \in \Delta\theta_i$. Triangles at the right-most points of all, except $\Delta\theta_{-1}$ = [9,+∞), segments mean that the values of $F(\theta)$ in these points are not defined. On the number axis, $F(\theta)$ is a discontinuous single-valued total function; its unambiguity is provided by the fact that in each point of discontinuity for each of two neighbor $F(\theta)$ line segments its left frontier point is defined while its right frontier point is not (see also Table 1). Since all $\theta \in \Delta\theta_i$ produce only a single value of the probability $L(d,\theta)$, it is convenient to assign to the *i*th $\Delta\theta_i$ its neuron threshold interval index Θ_i which produces the same $L(d,\theta) = L(d,\Theta_i)$. We define the series of even Θ_i with $\Delta\Theta = 2$ in such a way that without fail it contains its zero-element, e.g., Θ_4 = 0 in Figure 1.

Figure 2 illustrates the way in which all probability values in Figure 1 were calculated under Equation 1. We see that this Equation may be written as a sum of probabilities shown in Figure 2, $L(m,N,\Theta) = \sum p_b(i,m)$, with appropriate summation rules. Summation results obtained equal probabilities shown in Figure 1 (e.g., to calculate $F(\theta)$ the distribution m = 9 from Figure 2 should be adopted). We also specially emphasize that the expansion of the standard binomial distribution is needed to calculate $L(d,\theta)$ at $\theta < -9$ and $\theta > 9$ (they provide probabilities 1 and 0).

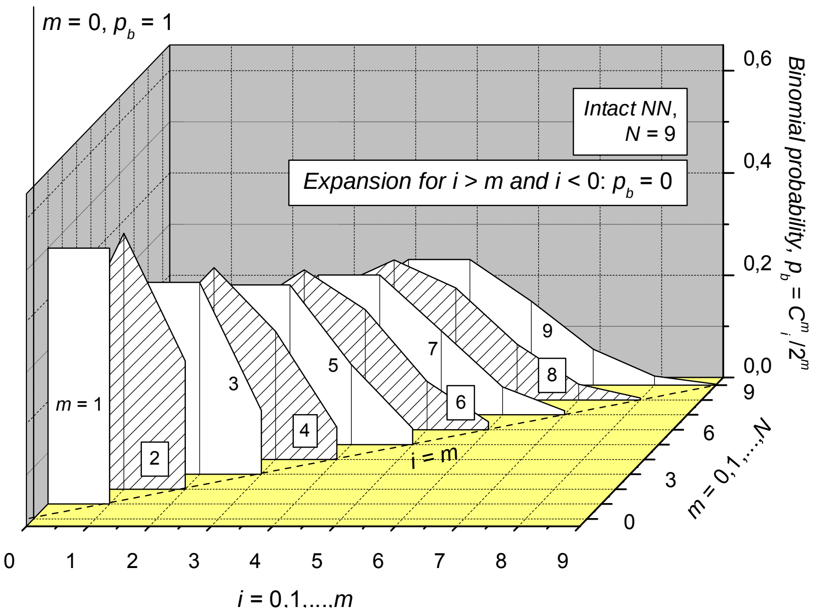


Figure 2. Probability densities $p_b(i,m)$ of Hamming distances between vectors $x(d)$ and x_0 under condition that $x(d)$, $d = m/N$, contains m its noise components ±1 chosen randomly with uniform probability, ½; p_b is a binomial distribution expanded at the ranges $i > m$ and $i < 0$ where we posit that $p_b = 0$. For each m, $0 < m \le N$, exact values of p_b are connected in straight lines bounding corresponding areas, dashed or not. The case $m = 0$ is special as contains only one point $p_b = 1$ which is on the apex of a separate vertical line.

4. Bayes Inferences within the BSDT

Let us define the event A = 'identification of x_0 in $x(d)$, $d = m/N$, by an NN with the neuron threshold θ' (or A = 'decoding vectors $x(d)$, $d = m/N$, by an NN with the neuron threshold θ') and two alternative hypothesis: H_0 implying that $x(d)$ is a sample of pure binary noise and H_1 implying that $x(d)$ is x_0 damaged to the damage degree d by such a noise. Using these designations and famous Bayes formula we can write

$$P(A)P(H_0|A) = P(H_0)P(A|H_0), \quad P(A)P(H_1|A) = P(H_1)P(A|H_1), \qquad (2)$$

$$P(A) = P(H_0)P(A|H_0) + P(H_1)P(A|H_1) \qquad (3)$$

where $P(A)$, $P(H_0)$, and $P(H_1)$ are prior probabilities of the event A, hypothesis H_0, and H_1; $P(A|H_1) = L(d,\theta)$ and $P(A|H_0) = L(d = 1,\theta) = F(\theta)$ are conditional likelihood probabilities of correct and false decoding; Equation 3 reflects the obvious fact that if the event A occurs then H_0 and H_1 are valid with probabilities $P(H_0)$ and $P(H_1)$, respectively. Combining Equations 2 and 3 we have

$$P_{FD}(d,\theta) = [1 + \kappa(d)L(d,\theta)/F(\theta)]^{-1},\ \ P_{CD}(d,\theta) = \{1 + 1/[\kappa(d)L(d,\theta)/F(\theta)]\}^{-1}, \quad (4)$$

where $P_{FD}(d,\theta) = P(H_0|A)$ and $P_{CD}(d,\theta) = P(H_1|A)$ are conditional posterior probabilities respectively of false and correct decoding (cf. ref. 3),

$$\kappa(d) = P(H_1)/P(H_0) = (1 - d)/d \quad (5)$$

is the ratio of prior probabilities of H_0 and H_1 defined within the BSDT explicitly, $P(H_0) = d$ and $P(H_1) = q = 1 - d$. As at $d = 0$ (the case $x(d) = x_0$) $\kappa(d)$ does not exist and at $d = 1$ (the case of pure noise) $1/\kappa(d)$ does not exist, in these special cases we posit that at $d = 0$ $P_{CD} = 1$, $P_{FD} = 0$ and at $d = 1$ $P_{CD} = 0$, $P_{FD} = 1$ in accordance with our expectations. Hence, now using Equations 1, 4 and 5 $P_{CD}(d,\theta)$ and $P_{FD}(d,\theta)$ can analytically be calculated for any possible values of d and θ. Since $P_{CD}(d,\theta) + P_{FD}(d,\theta) = 1$, it is enough to consider only one of these two posteriors. Below we shall discuss $P_{CD}(d,\theta)$ writing it without its subscripts, $P(d,\theta)$. Also we emphasize that in contrast to ref. 3 within this work likelihood and posterior are always designated respectively as '*L*' and '*P*,' regardless of lists of their subscripts or arguments; such designations directly point to distinctions between conditional probabilities of two types, likelihood and posterior, and are convenient when they are considered together.

In a 3D orthogonal space with axes d, θ, and L (or d, θ, and P) likelihood $L(d,\theta)$ [or posterior $P(d,\theta)$] produces a lattice of discrete points representing a complete set of all possible values of $L(d,\theta)$ [or $P(d,\theta)$]. For short, here we do not display corresponding 3D figures although in Figure 1a one can see a projection of the $L(d,\theta)$-lattice on the coordinate plane (L,θ); projections of $L(q,\Theta)$- and $P(q,\Theta)$-lattices on coordinate planes (L,q) and (P,q) see in Figure 5 ($q = 1 - d$, for relations between θ and Θ see Table 1).

5. ROC, mROC, Overall Likelihood and Posterior

As the values of $L(d,\theta)$, $F(\theta)$, $P(d,\theta)$ and relations between BSDT parameters are known (Equations 1, 4, 5 and Table 1), for different values of d (or more 'physical' parameter $q = 1 - d$ meaning the intensity of cue) it is possible to calculate likelihood, $L_q(F)$, and posterior, $P_q(F)$, as functions of false-alarm probability, F. The dependence $L_q(F)$ is called Receiver Operating Characteristic (ROC) curve; by analogy we refer to the corresponding dependence $P_q(F)$ as modified or posterior ROC curve, mROC; for all q, $0 \le q \le 1$, they are shown in Figures 3a and 4a, respectively. In addition to ROCs and mROCs, we can also define overall, do not depending on F, likelihood and posterior. For this purpose in ref. 3 a simple averaging of probabilities related to particular mROC was used. But taking into account that all ΔF_i are known and constitute a complete binomial probability distribution $\Delta F_i = C^N_i/2^N$, $\Sigma\Delta F_i = 1$, $i = 0,1,\ldots,N$ (see Table 1), it is natural to define overall likelihood, $L_0(q)$, and overall posterior, $P_0(q)$, as binomial averaging of corresponding sets of likelihoods, $L_q(F_i)$, and posteriors, $P_q(F_i)$:

$$L_0(q) = \sum L_q(F_i)\Delta F_i = \sum L_q(F_i)C^N_i/2^N, \quad (6)$$

$$P_0(q) = \sum P_q(F_i)\Delta F_i = \sum P_q(F_i)C^N_i/2^N \quad (7)$$

where all summations are made over $i = 0,1,\ldots,N$ [above $L_q(F_i)\Delta F_i$ and $P_q(F_i)\Delta F_i$ are areas of rectangles with the base ΔF_i and heights $L_q(F_i)$ and $P_q(F_i)$; $L_0(q)$ and $P_0(q)$ are areas under stepwise curves connected discrete ROC and mROC values, respectively]. The choice of values of ΔF_i as weights in Equations 6 and 7 means that if any value of F, e.g. F_x, is randomly chosen with uniform probability within the range $0 \le F_x \le 1$ then the probability (fre-

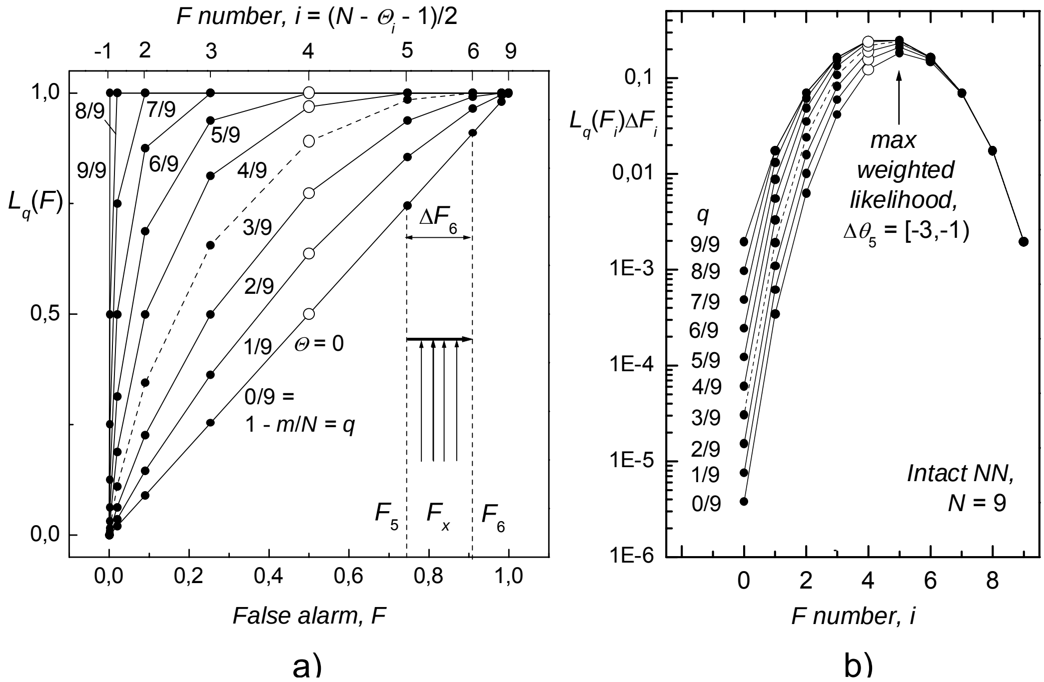


Figure 3. a) ROCs, $L_q(F_i)$, and b) weighted likelihoods, $L_q(F_i)\Delta F_i$; values $L_q(F_i)$ were calculated under Equation 1. In panel a) straight lines connect signs related to a specific value of the intensity of cue, $q = 1 - d$, and constitute a specific ROC; dashed lines designate ROC with $q = 3/9$ ($d = 6/9$); open circles reflect values of $L_q(F_4) = L_q(1/2)$ corresponding to the threshold interval index $\Theta = \Theta_4 = 0$; ΔF_6 is the interval between two neighbors, F_5 and F_6; vertical arrows represent schematically a fraction of values of F belonging to ΔF_6, $F_x \in \Delta F_6$. In panel b) vertical arrow points to maximum items of sums $\sum L_q(F_i)\Delta F_i$, they correspond to $i = 5$ and $\theta \in \Delta\theta_5$.

quency) of the event $F_x \in \Delta F_i$ equals ΔF_i and the probability of the emerging values of $L_q(F_i)$ and $P_q(F_i)$ also equals ΔF_i [the same may relate to $L_q(F_{i-1})$ and $P_q(F_{i-1})$ although this case will not be considered in this work]. Hence, if our assumption that values of F are randomly chosen within the range $0 \le F \le 1$ with uniform probability is valid (we shall call this assumption the *uniformity* or *no-priming hypotheses*) then overall probabilities $L_0(q)$ and $P_0(q)$ are optimal, i.e. the best among other ones calculated according to other possible averaging rules.

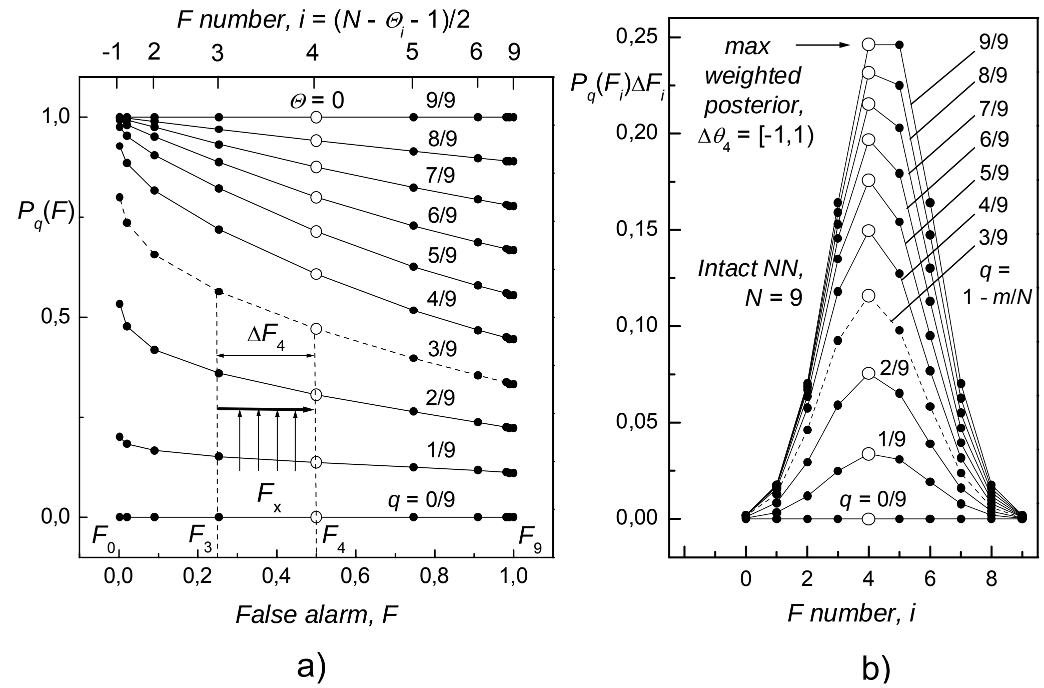


Figure 4. a) mROCs, $P_q(F_i)$, and b) weighted posteriors, $P_q(F_i)\Delta F_i$; posteriors $P_q(F_i)$ were calculated under Equations 1, 4, and 5 [$P_q(F_{-1})$ in panel a) and $P_q(F_{-1})\Delta F_{-1}$ in panel b) are not shown]. In panel a) straight lines connect signs related to a specific value of the intensity of cue, $q = 1 - d$, and constitute a specific mROC; dashed lines designate mROC with $q = 3/9$ ($d = 6/9$); open circles reflect values of $P_q(F_4) = P_q(1/2)$ corresponding to the threshold interval index $\Theta = \Theta_4 = 0$; ΔF_4 is the interval between two neighbors, F_3 and F_4; F_0 and F_9 are also marked; vertical arrows represent schematically a fraction of values of F belonging to ΔF_4, $F_x \in \Delta F_4$. In panel b) vertical arrow points to maximum items of sums $\sum P_q(F_i)\Delta F_i$, they correspond to $i = 4$ and $\theta \in \Delta\theta_4$.

For all q, $0 \leq q \leq 1$, in Figures 3b and 4b components of sums $L_0(q) = \sum L_q(F_i)\Delta F_i$ and $P_0(q) = \sum P_q(F_i)\Delta F_i$ are shown as functions of their summation index, i. It is remarkable that all corresponding curves have a common maximum, for weighted likelihoods at $i = 5$ and for weighted posteriors at $i = 4$. Hence, if our uniformity hypotheses concerning the choice of F for ROC and mROC values estimation is valid then an observer/computer code, who/that does not use any prior information about probabilities of hypothesis H_0 and H_1, naturally (most probably) choices values of the neuron threshold θ which are slightly smaller than zero, $\theta \in \Delta\theta_5 = [-3, -1)$; another observer, who in contrast uses completely the prior information mentioned, naturally choices for θ its near-zero values, $\theta \in \Delta\theta_4 = [-1,1)$. The same Figures demonstrate also that during the estimation of overall probabilities, $L_0(q)$ and $P_0(q)$, in right-hand sums of Equations 6 and 7 their items with their numbers i near to 0 and near to N (them correspond 'small' ΔF_i) are not so essential as their 'central' items (them correspond 'large' ΔF_i). Moreover, $L_0(q)$ and $P_0(q)$ are defined without the use of probabilities $L_q(F_{-1}) = 0$ and $P_q(F_{-1}) = 1$ in corresponding sums of Equations 6 and 7 and, consequently, these probabilities are at all not requested for estimating particular values of overall probabilities (in other words, left-most points of ROCs and mROCs may in practice be not claimed).

6. BDPs, mBDPs, and Psychometric Functions

Each of probability functions $L(d,\theta) = L(d,\Theta) = L(d,F)$ and $P(d,\theta) = P(d,\Theta) = P(d,F)$ has two arguments (for relations between θ, Θ, F, and etc see Table 1; $d = 1 - q$). If for the likelihood $L(d,F)$ one of them, e.g. d, is fixed then we obtain ROC curves, $L_d(F)$ or $L_q(F)$; if for the posterior $P(d,F)$ the same parameter is fixed then we obtain mROC curves, $P_d(F)$ or $P_q(F)$. Similarly, if in $L(d,F)$ or $L(q,F)$ the argument F is fixed then we obtain the function $L_F(q)$ or $L_\Theta(q)$ which we shall call Basic Decoding Performance (BDP) curve; if in $P(d,F)$ or $P(q,F)$ the same argument is fixed then we obtain a modified (posterior) BDP or mBDP curve, $P_F(q)$ or $P_\Theta(q)$. Examples of BDP and mBDP curves are shown in Figures 5a and 5b, respectively.

Intensity of cue $q = 1 - d$ defines a fraction of undamaged signal components among m noise components of N-dimensional vectors $x(d)$, $d = m/N$, or the quality of data analyzed: the more the q the better the quality is. Functions describing the signal's detection probability against the quality of data analyzed (e.g., the signal's intensity, amplitude, or area) are called *psychometric functions* [2, chapter 8], PMFs. Consequently, BDF and mBDP curves may respectively be interpreted as PMFs (Figure 5a) and modified or posterior PMFs, mPMFs (Figure 5b).

In the classic SDT arguments of PMFs are continuos and ranged from zero to positive infinity [2] while within the BSDT q is discrete, with the discreteness degree $\Delta q = 1/N$, and changes in the limited range, $0 \leq q \leq 1$. As in practice magnitudes of all variables are always limited and measurement results are usually discrete, the finiteness and discreteness of q are not its disadvantages as the PMF's or mPMF's argument. Indeed, if discrete values of a variable V are from the range $0 \leq V \leq V_{max}$ then $V = (k/N)V_{max}$ ($k = 0,1, \ldots,N$) and by changing N arbitrary small discreteness of V, $\Delta V = V_{max}/N$, may be achieved. Hence, for any V_{max} such signal detection experiment may be designed that the psychometric function (PMF) measured [6] will have the form as one of curves shown in Figure 5a. To confirm this claim it is simply enough to transform the variable V into a new dimensionless variable $V/V_{max} = k/N$ and assume that $k/N = q$, i.e. $V/V_{max} = q = (N - m)/N$ where $N - m = k$ is the number of undamaged signal components of a vector $x(d)$. Consequently, discrete PMFs as in Figure 5a may be considered as a universal (dimensionless) PMFs, UPMFs, matching to any positive variable V with its arbitrary large maximum value V_{max} (the number of points along particular UPMF defines its fit parameter N may be chosen arbitrary large). Finally, let us pay an attention to a corollary arising: all points along a UPMF or PMF are equidistant ($\Delta q = 1/N$, $\Delta V = V_{max}/N$) and ΔV may be considered as a *just noticeable difference* (*jnd*) [2, p.25], the minimal difference between two stimuli, $V_k = kV_{max}/N$ and $V_{k+1} = (k + 1)V_{max}/N$, that leads to a change in experience, $\Delta L_0(V_{k+1}) = L_0(V_{k+1}) - L_0(V_k)$, $k = 0,1,\ldots, N - 1$. For likelihood and posterior PMFs their jnd's values are the same although the values of $\Delta L_0(V_k)$ depend essentially on V_k and for this reason in different experiments different 'seeming' values of the jnd can be observed. The jnd (i.e. ΔV) could be an atom of experience and this atom-of-experience hypotheses still proposed by Gustav Fechner [2, p. 26] is consistent with our model of an atom of consciousness [5].

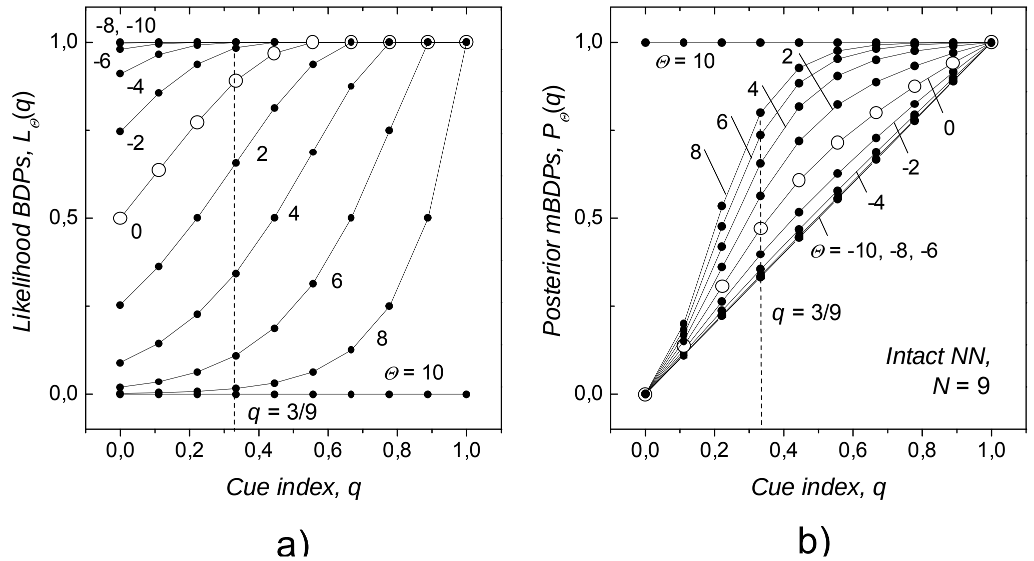


Figure 5. a) BDPs or universal psychometric functions, UPMFs, and b) mBDPs or modified (posterior) UPMFs, mUPMFs. $F_\Theta = L_\Theta(q = 0)$ is the value of F for a UPMF specified by the parameter Θ. Consequently, for each UPMF the Neyman-Pearson objective is achieved; the same concerns to mUPMFs in spite of the fact that for them $P_\Theta(q = 0) = 0$ at all Θ. For the special case $L_{10}(q) = F_{10} = 0$ posteriors $P_{10}(q)$ cannot be calculated (Equations 4) and for this reason for all q it is needed to posit: $P_{10}(q) = 1$. Other designations were explained previously.

7. Sensitivity and Bias, a Comparison between the SDT and BSDT

For psychophysics experiments, where likelihoods are measured, within the SDT sensitivity and bias are defined using ROCs [2]. Within the BSDT not only ROCs, $L_q(F)$, but also mROCs, $P_q(F)$, exist and, consequently, using

Table 2

Discrimination experiment descriptions within the SDT and BSDT, a comparison

Notion	Classic SDT approach [2]		BSDT approach [3,5]	
	Definition	Comments	Definition	Comments
1	2	3	4	5
Decision space	One-dimensional[a)], its psychological decision variable is a single stimulus attribute called familiarity x, $-\infty < x < +\infty$; its objects are underlying familiarity distributions or probability densities $f(x\|S_i)$[b)] related to samples of trials S_i; M_i is the mean for the ith $f(x\|S_i)$, $-\infty < M_i < +\infty$.	Familiarity x is continuous, values of x define criterion locations or biases c, differences of x define perception distances or sensitivities d'; the world of stimuli and psychological decision space are separate with unclear relations between them.	Two-dimensional, its variables are cue index (intensity of cue) $q = 1 - m/N$ and neuron threshold θ, $0 \le q \le 1$, $-\infty < \theta < +\infty$; its objects are N-dimensional binary (±1) vectors $x = x(d)$ with m uniformly distributed noise components, $0 \le m \le N$, $d = m/N$.	q is discrete, θ is continuous and both are statistically independent; a 'neural space' where stimuli are represented as neural codes (binary vectors x) [c)] is also the decision space.

ROC space	Two-dimensional, its variables (measured or calculated) are hit rate $H = \Phi(d'/2 - c)$ and false-alarm rate $F = \Phi(-d'/2 - c)$, $0 \le H \le 1$, $0 \le F \le 1$ [d)]; for transformed ROC space its variables, $z(H)$ and $z(F)$, are H and F transformed into a z-score (z is the inverse of the normal distribution function), $-\infty < z(H) < +\infty$, $-\infty < z(F) < +\infty$.	H and F are continuous and statistically independent [e)]; to generate an ROC, d' should be stable while c changeable; in H,F-coordinates ROCs are curvilinear with changing shape across d' values; in z-scores ROCs have a straight-line form.	Two-dimensional, its variables (measured or calculated) are hit rate $H = L(q,\theta)$ and false-alarm rate $F = L(q = 0,\theta)$, $0 \le H \le 1$, $0 \le F \le 1$, $F \le H$ [d)]; as each stimulus is represented by its vector x whose binomial probability density is specific for each m, a transformed common z-ROC space cannot be defined.	F and H are discrete and functionally related; to generate an ROC, probabilities $H = L(q,\theta) = L_q(F)$ are calculated at a fixed q with changing θ (or with changing F as θ, Θ, F, Q, ρ, and D are related, see Table 1).
Sensitivity	Perception distance, $d' = z(H) - z(F)$ or $d' = M_2 - M_1$, $0 \le d' < +\infty$ [f)]; proportion correct, $p(c) = p(S_2)H + p(S_1)(1 - F)$ where $p(S_i)$ is the probability that S_i is presented; area under the ROC, A', A_g, or A_z [g)].	d' is continuous; a fixed d' and changing c define an isosensitivity or ROC curve, if $d' = 0$ then ROC provides chance-level performance; for unbiased ($c = 0$) observers $p(c)$ is a nonparametric overall sensitivity.	Cue index q; proportion correct, $p(q,\theta)$, may also be defined [h)] and its definition is valid for any possible values of q and θ; overall likelihood, $L_0(q)$, an 'area' estimation under the discrete ROC.	q is discrete; a fixed q and changing θ define an isosensitivity or ROC curve, if $q = 0$ then ROC provides chance level performance.
Bias	Criterion location, $c = -[z(H) + z(F)]/2$, $-\infty < c < +\infty$; relative criterion location, $c' = c/d'$; likelihood ratio or the slope of transformed ROC for a given c, $\beta_G = f(c\|S_2)/f(c\|S_1) = H(c)/F(c) = \exp(cd')$; $\log(\beta_G) = [z(H)^2 - z(F)^2]/2$ [i)].	d' and c are independent and constitute a pair of variables alternative to the pair H and F; a fixed c (c' or β_G) and changing d' define an isobias curve; on isobias curves ($c = 0$) as F increases H must decrease [j)]; observers do not naturally use a neutral value of the bias (confidence level).	Neuron threshold θ, $-\infty < \theta < +\infty$ (or convolution Q, threshold interval index Θ, correlation coefficient ρ, false-alarm probability F, or Hamming distance D as θ, Q, Θ, ρ, F, and D are related; see Table 1).	θ is continuous; a fixed θ (F, Q, Θ, ρ, or D) and changing q define an isobais curve, $H = L_\Theta(q)$, which is a UPMF (see Figure 5a); observes naturally use near-neutral values of the bias, θ.

[a)] There exist experimental paradigms for which two- and many-dimensional versions of the SDT were developed [2, chapter 10]; recently, a new original two-dimensional version of the SDT has also been proposed [7] but here it is not discussed as only classic SDT [1,2] is the comparison subject in this work.

[b)] Here and below only normal densities are considered and for this reason $f(x|S_i)$ is always a Gaussian.

[c)] Within the BSDT a set of neural codes $x(d)$ representing particular stimuli (a 'neural space') is simultaneously a decision space where operations over these codes are defined. The world of stimuli and psychological space within the BSDT are directly not requested but it is supposed that rules for transformation of external/internal sensory stimuli (the world of stimuli or stimuli space) into their corresponding neural representations (the neural space) may be discovered by methods of neuroscience.

[d)] If an ROC curve passes through the points $(F,H) = (0, 0)$ and $(1,1)$ then it is called a regular ROC curve.

[e)] Formally, H and F are defined as statistically independent but while $d' \ge 0$ they can take independently only values $F \le H$ which are on and upper the ROC's main diagonal, $H = F$. If additionally values $d' < 0$ are admitted then H and F may be independent in all ROC space but in this case events $H < F$ become possible.

[f)] d' is defined under condition that underlying distributions for samples of trials S_1 and S_2 have common standard deviation; if that is not the case then for S_1 and S_2 their distinct standard deviations d'_1 and d'_2 are introduced as well as their root-mean-square average, d_a.

[g)] $A' = 0.5 + (H - F)(1 + H - F)/[4H(1 - F)]$ gives an area estimation under the one-point ROC; $A_g = 0.5\sum(F_{i+1} - F_i)(H_{i+1} + H_i)$ provides area under the multipoint ROC (the summation is made over all ROC points numbered from the left lower corner), within the BSDT the counterpart of A_g is overall likelihood $L_0(q)$; $A_z = \Phi(D_{YN}) = \Phi(d_a/\sqrt{2})$ gives area under the non-regular ROC curve (D_{YN} is the distance between the origin and the non-unit-slope z-ROC, $\Phi(x)$ is a cumulative distribution function or the integral of a Gaussian $f(x)$ taken from $-\infty$ to x).

[h] By analogy to proportion correct, $p(c)$, $p(q,\theta) = p(S_2)\,L(q,\theta) + p(S_1)[1 - L(q = 0,\theta)]$ where $L(q,\theta) = H(q,\theta) = H$, $L(q = 0,\theta) = F(\theta) = F$ and $p(S_i)$ is the probability that S_i is presented; for intact NN, odd N, and $\theta = 0$ $p(q,0) = p(S_2)L(q,0) + p(S_1)½$ as here $F = ½$ [3].

[i] Within the singe high-threshold theory [2, chapter 4], an SDT's alternative, false alarm F is introduced as natural bias (confidence level) index but such bias definition cannot be accepted as it leads to isobias curves with constant value of F, in contradiction to the SDT's so called monotonically condition demanding that along isobias curves as F increases H must decrease and vice versa [2, p.93].

[j] Such definition of isobias curves does not satisfy to the Neyman-Pearson objective as F and H are changing simultaneously (due to the monotonically condition); within the SDT to generate an isobias curve satisfying this objective, c and M_1 should be constant while M_2 changeable.

mROCs a possibility arises to define similar parameters as well for posteriors, $P_q(F)$. In Table 2 sensitivity, bias, their decision and ROC spaces defined using the ROCs within the SDT and BSDT are compared. For examples of experiments where posterior probabilities could be measured and mROCs derived see ref. 3, corresponding posterior sensitivity and bias should be discussed separately as till now they have no the SDT's counterparts.

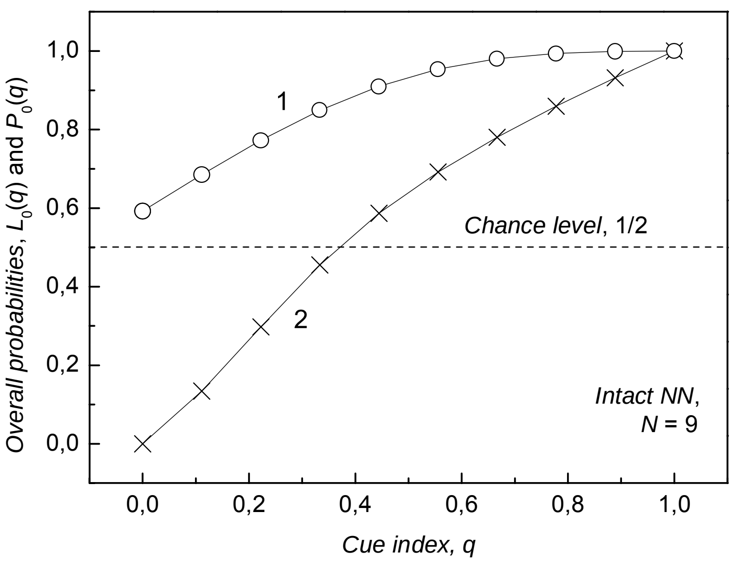


Figure 6. The overall likelihood $L_0(q)$ (open circles, curve 1) and overall posterior $P_0(q)$ (crosses, curve 2) calculated under Equations 6 and 7 as functions of the intensity of cue, q. To each circle on curve 1 corresponds a curve in Figure 3b specified by the same q, to each cross on curve 2 corresponds a curve in Figure 4b specified by the same q.

Finally, Figure 6 demonstrates overall probabilities $L_0(q)$ and $P_0(q)$ for subjects (computer codes) do not using prior information about probabilities of hypothesis H_0 and H_1 and for subjects (computer codes) having and completely using this prior information (in both cases it is supposed that our no-priming hypotheses about 'in-mind' distribution of F is valid). We see that curve 1 begins in a point above the chance-level line while curve 2 in the origin; both curves end in the point (1,1). Overall likelihoods $L_0(q)$ (circles on curve 1) may be compared with areas $A(d')$ ranged from ½ at $d' = 0$ to 1 at $d' \to \infty$ [$A(d')$ is area under the SDT's particular regular ROCs]. Hence, $L_0(q = 0) > ½$ while $A(d' = 0) = ½$ and $L_0(q = 1) = A(d' \to \infty) = 1$ [d' and q are sensitivities of the SDT and BSDT, respectively; $A_g(d') \to A(d')$ if the number of points measured on the SDT's regular ROC goes to infinity, see Table 2 and its footnote [g]].

8. Conclusions

Above for a simple example, the BSDT's decoding algorithm has been studied using its exact quality performance functions analytically calculated. In this way an attempt was made to reveal some similarities and distinctions between the classic SDT [1,2] and recent BSDT [3,5]. We saw that the main similarity consists in the fact that SDT and BSDT can produce the same functions for the description their decoding algorithms' quality performance, ROCs and psychometric functions (PMFs), and the same basic parameters, sensitivity and bias.

Hence, using the BSDT those measurement results can in principle be described that the SDT describes. The source of distinctions between them is, in our opinion, in the BSDT's original binary data coding [3,4] and its corresponding decoding algorithm existing simultaneously in NN, convolutional, and Hamming distance forms having equivalent and the best (in the sense of statistical patterns recognition quality) performance. For this reason in contrast to the classic SDT, BSDT is based on discrete final (binomial) probability distributions and all its predictions and parameters (except the neuron threshold θ) are discrete. In summary: essential features and inferences of the BSDT follow in fact directly from analyzes of the original mathematical form of its performance functions. For example, the notion of a neural space introduced is a direct consequence of the BSDT/NNAMM's binary coding/decoding approach though here it is also implied implicitly that vectors $x(d)$ represent neural codes of sensory stimuli in the brain; psychometric functions satisfying the Neyman-Pearson objective are simply projections of likelihood probability function $L(q,\theta)$ on the coordinate plane (L,q); the just noticeable difference (jnd) is a counterpart to the discreteness, Δq, of the cue index, q (and due to its discreteness Δq can be naturally related to an atom of experience); our new uniformity or no-priming hypotheses is simply a requirement needed to define optimally ROCs, mROCs, and overall probabilities; and, finally, our conclusion that subjects (computer codes) naturally (most probably) choice near-zero thresholds follows from the existence of a maximum among items of weighted sums of particular likelihoods and posteriors.

Our computations of likelihood, $L(q,\theta)$, posterior, $P(q,\theta)$, overall likelihood, $L_0(q)$, and overall posterior, $P_0(q)$, probabilities of correct decoding confirm that the BSDT's two basic parameters (sensitivity or intensity of cue, q, and bias or neuron threshold, θ) are sufficient to parametrize the decoding algorithm's quality performance functions, traditional (ROCs and PMFs) as well newly introduced [mROCs, mPMFs, $L_0(q)$, and $P_0(q)$]. The only limitation consists in use of the decoding algorithm in the form of an intact NN and that is why only regular ROCs and their related functions were discussed so far. This limitation can be evaded yet even now it does not hinder to begin to reinterpret some psychophysics results in terms and notions of the BSDT which, we believe, in many cases are more natural and attractive than terms and notions of the classic SDT.

I am grateful to my family and my friends for their help and support.

Bibliography

1. D.Green & J.Swets. Signal Detection Theory and Psychophysics. New York, Wiley, 1966.
2. N.A.Macmillan & C.D.Creelman. Detection Theory: A User's Guide. New York, Cambridge University Press, 1991.
3. P.M.Gopych. ROC Curves within the Framework of Neural Network Assembly Memory Model: Some Analytic Results. International Journal Information Theories & Applications. 2003, 10(2), 189-197; http://arXiv.org/abs/cs.AI/0309007.
4. P.M.Gopych. Determination of Memory Performance. JINR Rapid Communications, 1999, 4[96]-99, 61-68 (in Russian).
5. P.M.Gopych. A Neural Network Assembly Memory Model Based on an Optimal Binary Signal Detection Theory. Programming Problems, 2004, no.2-3, 473-479; http://arXiv.org/abs/cs.AI/0309036.
6. P.M.Gopych. Identification of Peaks in Line Spectra Using the Algorithm Imitating the Neural Network Operation. Instruments & Experimental Techniques, 1998, 41(3), 341-346.
7. C.M.Rotello, N.A.Macmillan, J.A.Reeder. Sum-Difference Theory of Remembering and Knowing: A Two-Dimensional Signal Detection Model. Psychological Review, 2004, 111, 588–616.

Author Information

Petro Mykhaylovych Gopych – V.N.Karazin Kharkiv National University; Svoboda Sq., 4, Kharkiv, 61077, Ukraine; e-mail: pmg@kharkov.com.

ADAPTIVE CONTROL AND MULTI-AGENT INTERFACE FOR INFOTELECOMMUNICATION SYSTEMS OF NEW GENERATION

Adil Timofeev

Abstract: Problems for intellectualisation for man-machine interface and methods of self-organization for network control in multi-agent infotelecommunication systems have been discussed. Architecture and principles for construction of network and neural agents for telecommunication systems of new generation have been suggested. Methods for adaptive and multi-agent routing for information flows by requests of external agents-users of global telecommunication systems and computer networks have been described.

1. Introduction

Important role for support of "world dialogue" between people and development of man-machine interaction in XXI century will be played by not only global network Internet, but also more modern computer and telecommunication systems (TCS) of new generation. Last years such global TCS are researched and developed in the projects "Internet 2", "Abilene", "NGI" etc [1–4].

Design for TCS of next generations requires development of principally new approaches to man-machine interface and network control for information flows on the base of theory of adaptive and intelligent systems, multi-agent and neural technologies and multi-modal systems for virtual reality.

Paper deals with general problems for intellectualization of man-machine interface in multi-agent global TCS and some new methods of multi-agent routing, adaptive control for data flows and network self-organization in dynamical infotelecommunication environment.

2. Architecture for Global Multi-agent Telecommunication Systems

Global multi-agent TCS serve for providing to external agent-users informational and computing resources, distributed in computer networks (CN) around the world. These telecommunication service and information resources are providing to users through multi-agent man-machine interface.

Global TCS architecture is presented in the fig.1. It consists of 4 basic subsystems:

1. Distributed communication system (DCS);
2. Network control system (NCS);
3. Distributed information system (DIS);
4. Distributed transport system (DTS).

All these subsystems are connected between each other and intended for controlled transfer of

Information and computer resources, stored in distributed CS, to agents-users (subscribers, network administrators etc.) of global TCS. Therefore important role in infotelecommunication networks is played by man-machine interface and problems of its modernization.

DCS consists of distributed tools for access and user interface, and also ports and data bus, providing direct and inverse communication between agents-users of global TCS, and connected with it distributed CS, consisting of remote on significant distances computers, local CS, robotic systems etc.

External agents-users of global TCS may be subscribers, administrators, operators and providers of TCS. For its effective interaction with global TCS and CS it is necessary to advance man-machine interface, which is the main part of DCS.

NCS obtains through man-machine DCS subscribers queries and commands of TCS network administrators and processes internal information about current state of DCS and external information about informational and computing resources in CS, coming from DIS.

On this information NCS forms control for data flows in DTS, providing satisfaction for queries of agents-users by address passing to them necessary informational and computing resources of CS.

DIS obtains signals for internal and external feedback about current state of DTS as control plant and accessible informational and computing resources, stored in global CS. It transfers these signals to NCS for forming or correction for control data flow which is going through DTS.

DTS consists of communication nodes (which may be specialized communication processors) and communication channels between them. It plays role of distributed controlled plant and serves for controlled address transfer of data flows from agents-users to CS through TCS and inversely.

All shown subsystems, including man-machine interface of global TCS, have distributed character, are interconnected and interact actively between each other in the process of providing for agents-users informational and computing resources, stored in global CS.

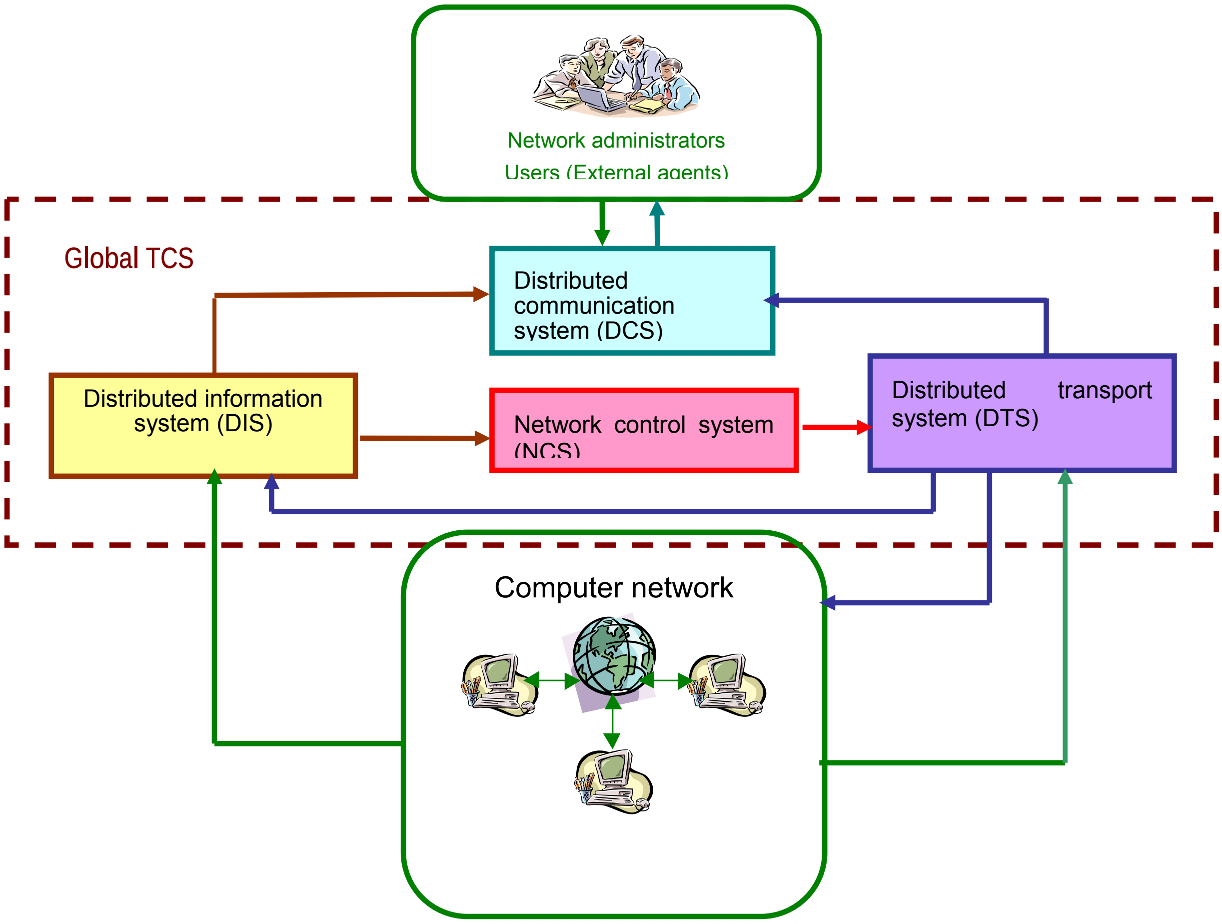


Figure 1. Architecture of global TCN of new generation

3. Self-organization and Adaptation in Network Control

Main role in aiming directed self-organization, adaptive processing of information and address transfer of data flows on the queries of external agents as users of global TCS is played by NCS. Information queries and replies on them are formed with the help of man-machine interface, connecting external agents-users with distributed resources of global TCS and CS.

The main problem for NCS, working on big speeds of data transfer, is self-organization and automatic forming of such adaptive control for data flows in DTS, which supports changing traffic of heterogeneous and multimedia data of great volume with reliable guarantees of high quality of service (Quality of Service, QoS) for external agents-users of TCS.

Decision of this global task for network of adaptive control and self-organization in TCS is divided on local problems of data flows control, adaptation to changing traffic overloading avoidance, network collisions resolution etc. Practical realization of these problems is executed with the help of special network protocols and internal network agents, intended for address transfer of not only informational and controlled signals, but also heterogeneous and multimedia data flows.

In general case shown problems of network control and self-organization should be solved for two main platforms of modern global TCN:

- united IP-networks, interacting through routers of data flows on IP protocol (Internet Protocol) from protocol set TCP/IP (Transmission Control Protocol/Internet Protocol);
- ATM-networks, using protocols ATM (Asynchronous Transfer Mode).

Today these platforms are developed and compete actively on the market of network infotelecommunication service, what is reflected in so called “fight between IP and ATM”. In this connection the significant importance is given to NCS evolution, which will provide convergence and integration of IP- and ATM-networks in global TCS of new generation and their further development.

Traditionally for organization of network control for data flows and DTS equipment network principles and architectures for centralized or decentralized control are used. Every principle has certain advantages and disadvantages.

NCS centralized architecture is based on segregation of central computer, connected through man-machine interface with TCS administrator and executing functions of “control global center” for data flows transfer through nodes and DTS communication channels.

Advantage of such architecture is globality of control from single “centre”. Disadvantages of centralized control are absence of self-organization and low reliability and fault-stability. It reflects in that failure of central controlling computer causes full or partial disappearance of DTS controllability. Therefore reservation of NCS central computer is provided usually. New suggestion is in reservation of also TCN communication channels for multi-flow transferred information.

Decentralized NCS architecture distributes functions for information processing and control between a series of local computers, controlling different segments of DTS or data flows in them.

Advantage of such architecture is that relative independence of distributed “local control centres” increases reliability of address transfer of data flows. Disadvantages of decentralized control are locality and incompleteness of control aims that requires coordination and according work of distributed local controlling computers.

Considering these disadvantages of described traditional network architectures, it is necessary to develop “hybrid” self-organizing architecture of NCS for global TCS of new generation, combinating in itself advantages of centralized and decentralized architectures. Let name such compromise “hybrid” self-organizing architecture of multi-agent architecture of NCS of global TCS.

This new architecture of NCS requires development of theory of internal (network and neural) agents and intellectualization of man-machine interface for external agents-users of TCS.

4. Network and Neural Agents of Global TCS

Basic functions of information processing, self-organization and data flow control on queries of external agents-users of global TCS of new generation are distributed between internal agents. Their role is executed by interconnected network or neural agents of global TCS.

Architecture of these internal agents is presented on fig. 2. Comparing fig. 1 and fig. 2, it may be noted that architecture of global (and also autonomous and local) TCS is analogical (self-similar) to architecture of internal agents. There has appeared fractality of network and neural agents by relation to global TCS in a whole, and also to its most important segment – autonomous and local TCS.

Every internal network or neural agent has own local DB and KB or corresponding neural network (NN) with self-organized architecture and communication tools (communication channels, protocols etc.) with other agents for information exchange in process of joint (cooperative) decision making, self-organization “by interests” and automatic forming of network control of DTS, providing addressed delivering of informational computing resources of CS on the queries of external agents-users of global TCS.

Network or neural agents of TCS may be communication computers or neural routers of NCS, connected with DTS nodes, and also software or software-hardware Bagents of DCS and DIS, connected with intelligent man-machine interface of global TCS. Such internal agents of global TCS of new generation differ significantly from external agents-users of TCS (subscribers, network administrators and operators etc.), using tools for access and network man-machine interface for own informational queries to computer nodes (hosts) of distributed CS and obtaining replies on these queries.

Agents accumulate or generate in themselves local DB and KB, necessary for making of effective (particularly, optimal) decisions and executing of corresponding local operations in the limits of own (local) “competentness”. For communication between each other agents use corresponding “communication language”, including certain “dictionary store”, presenting formats and protocols for data transfer etc. Agents are able to solve independently local decisions and provide their execution.

So they can solve arising tasks both autonomously and collectively. For collective solution of tasks the agents may cooperate and self-organize in working groups “by interests”. Such group will be named agencies, having certain specialization, defined by agent interests, in corresponding problem (plant) area.

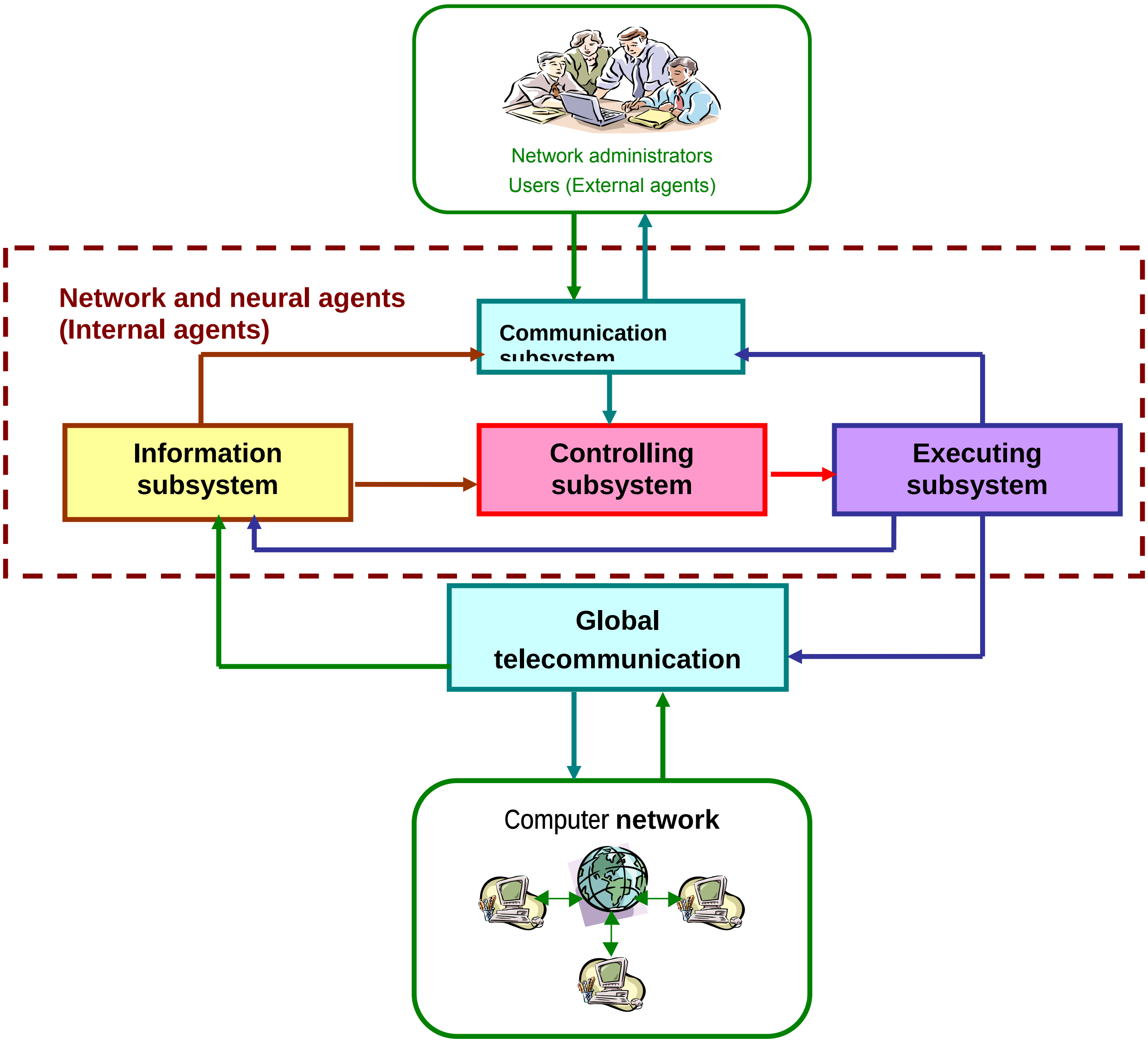


Figure 2. Architecture of network and neural agents in global TCS of new generation

Structure (architecture) and functions (operations) of agent are able to support initially interaction of its four basic subsystems between each other and with environment. However during concrete problem decision agents may be learned and extend their local DB, be adapted to changing or uncertain function conditions and exchange knowledge with other agents. Such learning agents are in fact adaptive developing intelligent systems. Their functional possibilities and intelligent abilities are extended during “vital cycle”, i.e. as learning and experience accumulation.

Neural agents are intended mostly for parallel transfer and processing of complex signals and images. They are mostly 2D- or 3D-images and vector signals. Set of such signals with information about their belonging to different classes of patterns makes experimental data bases (DB). These DB are named learning DB, because they are used for learning and self-organization of neural agents. In result there is adjusting of architecture (topology of network neurons) and parameters (synaptical weights) of neural agents to solved problem by a set of learning precedents.

Non-linearity of functions for neurons activation plays important role in it. These functions may be threshold or sigmoidal, polynomials or conjunctions etc. If these functions are linear, all neural network of agent makes only linear transformations of vector or matrix of internal signals, that corresponds to single-layer neural network. However superposition of non-linear neurons

extends significantly computing and intelligent possibilities of neural agents both local (concentrated), and global (distributed) learning DB.

During design for NCS and intelligent man-machine interface on the base of theory of agents and principles of self-organization new problems of multi-flow routing and multi-agent dialogue between internal agents of global TCS of new generation, external agents-users and server agents-informators of global TCS as distributed world store for data, knowledge and applications arise.

Decision of these problems requires development for methods of self-organization and adaptation, including in itself tools for avoidance or automatic resolution of network collisions between agents under control of intelligent NCS of global TCS with multi-agent architecture.

For controlled address transfer and navigation of data flows, functional diagnosis and recognition for states of global TCS of new generation it is necessary to introduce special internal agents-coordinators (for example, on level of data flow routing) and, possibly, other global agents, providing self-organization and adaptation in process of man-machine interaction and decision making.

Singularity of these coordinating agents of high level is that their DB and KB are formed on the base of local DB and KB of agents of lower level. Therefore they have global (multi-agent) character and allow to evaluate network situation and provide self-organization “as a whole” by queries of external agents-users of global TCS.

Thus, development of man-machine interface and advancement of self-organizing architectures of NCS of global TCS of new generation should be done not only and not so much “in a width”, i.e. “by horizontal” territory envelope, but mostly “in a depth”, i.e. “by vertical” of evolution for hierarchy of network control and self-organization.

Processes of adaptation, self-organization and intellectualization play important role both in NCS and TCS, and in multi-agent man-machine interface.

5. Multi-agent Interface and Problems of Adaptive Routing of Information Flows

Multi-agent interface of global TCS of new generation is necessary for organization of effective interaction of external agents-users of TCS and internal network or neural agents. Here important role is played by intelligent man-machine interface, providing interaction and cooperation “by interests” of external agents as users of global TCS

This man-machine interface is based both on own DB and KB of agents-users, and distributed local and global DB and KB of TCS and CS. It serve for forming of set and sequence of addresses of sources and receivers of the information, which play role of concrete aims of network control of multi-agent address transfer of data flows on queries of external agents-users of TCS.

Let discuss basic singularities of network control and importance of processes of adaptation and self-organization on the example of adaptive multi-agent routing of information flows and global TCS [5–9].

Necessity in adaptive routing arises at unpredictable changes of structure (nodes and communication channels) of TCS or at overloading of node buffers or channels of TCS. Routing and self-organization of information flows in non-stationery global TCS with variable structure and known beforehand load is discussed actually.

Causes of TCS structure changing may be both addition or failure of different nodes and communication channels and network overloadings, which prevent transfer of data flows through forbidden (overloaded) nodes and channels. That is why router should plan and correct optimal routes of transfer of data packages, adapting them to possible TCS changes, happening in real time. For it feedback about current state of nodes and TCS communication channels, which may be organized by information exchange between TCS nodes, is necessary.

Distinctive features of adaptive routing in comparison with traditional routing static or dynamic routing are the following peculiarities [6-8]:

- algorithms for adaptive routing requires consideration and processing of current information about TCS, that makes them more complex and increase optimal route definition time;
- transfer of information about state or structural changes in TCS to adaptive routers loads additionally a network and causes delays (lags);
- increasement of network load and time of delay may cause oscillations or auto-oscillations and increase a number of steps at determination of optimal route.

Adaptive routing of data flows in global TCS has a series of advantages relatively to non-adaptive (static or dynamic) routing and precisely these:

- provides workability and reliability of TCS at unpredictable changes of their structure or parameters
- causes more uniform load of nodes and TCS communication channels by “smoothing” of load;
- simplifies control for transfer of data flows and make more easy adaptation to network loads;
- increase time for infallible time and productivity of TCS at high level of rendered service in unpredictable conditions of changing of network parameters and structure, that is important essentially for external agents-users of TCS

Reaching of these advantages depends significantly from used principles and algorithms of adaptive routing and self-organization of data flows in TCS with unpredictable structure and traffic, unknown beforehand [4–10]. It is important note that “adaptive routing is a problem, which is rather difficult for proper solution” [1].

6. Methods for Adaptive and Multi-agent Routing of Information Flows

Principles of adaptive routing and self-organization of data flows may be divided on three classes in dependence from used information about real (current) state of global TCS, i.e. from character of feedback signals [4–6]:

- local information (feedback) from one node of TCS;
- local information (feedback) from node and its “neighbours” in TCS;
- global information (feedback) from all three node of TCS.

Simplest principle of adaptive routing with local feedback form one node is that data package transfers to communication channel with the most short queue or with the biggest probability of channel preference. Local load smoothing in output channels of global TCS may be happen. However in this case it is possible to deviate from optimal rout.

More effective principles of adaptive routing are based on transfer to initial node a local information (feedback) from neighbour nodes or global information from TCS nodes. As this information data about failures or delays in nodes or communication channels in TCS may be used.

In dependence on used ways of processing for local or global information (feedback) principles of adaptive routing may be divided on three classes:

- centralized (hierarchical) routing;
- decentralized (distributed) routing;
- multi-agent (multi-address) routing.

Principle of centralized routing is that every node of TCS transfers in first an information about own state (delays or external channels capacities etc) to central router. Then this router computes optimal rout on the base of obtained global information about current state and passes it back to all TCS nodes. Then controlled transfer of data packages from node-source to node-subscriber of TCS by planned optimal rout.

Principle of decentralized (distributed) routing is based on information exchange between TCS nodes and using of this information about current state of nodes communication channels of TCS for optimal rout calculating. As calculating of sequent plots of this rout distributed-controlled package transfer from node-source to node-receiver of TCS is executed.

Principle for multi-agent routing and self-organization of data flows is distinctive compromise between principles of centralized and decentralized routing. It is based on multi-agent man-machine interface and multi-address and multi-flow routing and analysis of possible network collisions with aim to eliminate them or to resolve during optimal data transfer by a set of optimal routes from nodes-sources to nodes-receivers of global TCS. More thoroughly this principle and concrete methods of multi-agent routing have been discussed in works [2–9].

7. Multi-flow Routing as Tool of Increase of Reliability of Global Telecommunication Networks

Main disadvantages of single-flow routing in global dynamic TCS are its following peculiarities :

– fault or failure of at least one node or TCS communication channel, through which optimal rout for data package transfer passes, require hard replanning (recalculating) for optimal rout (or its part) with consideration of faulted nodes or communication channels;
– planned rout between any defined node-source and node-receiver of TCS may cause network overloadings in the time, when other (for example, neighbour) nodes and communication channels may be free or not fully loaded.

First disadvantage causes great delays at controlled transfer of data flows, connected with information renewal about TCS state and recalculating of new rout. Such delays (lags) are not accessible for high-quality QoS–service of TCS users' queries or transfer of multi-media real time traffic.

Second disadvantage causes also delays because of overloading of nodes or communication channels, which are in optimal rout. At it a network traffic is distributed non-smoothly, so many intermediate nodes and communication channels of TCS are not loaded or simply are not used.

To overcome difficulties, connected with noted disadvantages, it is useful to use multi-flow routing. It is planned and used simultaneously not one (for example, optimal) rout of data package transfer, but $K \geq 2$ routes. More K is, more probability of data package delivering to node-source to node-receiver is. Consequently reliability and fault-stability of TCS are increased.

At centralized multi-flow routing an apriori planning of $K \geq 2$ optimal or suboptimal routes by existing (fixed) or renewal information about TCS state is made. Parallel use of these uncrossed routes provides more reliable delivering of data packages form node-source to node-receiver. At this case network traffic is distributed on TCS more smoothly, that decreases influence of possible overloadings in separate nodes or communication channels.

At decentralized (distributed) routing in the newt node of rout, it is planned $K \geq 2$ optimal or suboptimal routes of data packages transfer to node-receiver. Such method of aposteriori planning of K-routes "from reached node" requires special mechanism (closed routes) at package data transfer.

The main advantage of the method of aposteriori K-routing is that it provides automatic "avoidance" of failured of fault nodes or TCS communication channels. Other advantage is connected with local detection of faulted nodes or communication channels. It allows to renewal fast an information about TCS current state and insert necessary corrections to tables and routing maps.

Work has been done at partial support of Ministry for Education and Science of Russia (project N 37.029.11.0027), RFBR (project N 03-01-00224a) and RHSF (project N 03–06–12019в).

Bibliography

[1] Tanenbaum A.S. Computer Networks. – Prentice Hall, 1996.

[2] A.V.Timofeev Problems amd methods for adaptive control for data flows in telecommunication systems – Informatization and communication, N 1-2, 2003, p. 68-73.

[3] A.V.Timofeev Models of multi-agent dialogue and information control in global telecommunication network – Proceeding of 10-th International conference "Knowledge-Dialogue-Solution" (16-26 June, 2003, Varna), 2003, p. 180-186 (In Russian).

[4] Timofeev A.V. Methods of high-quality control, intellectualization and functional diagnosis for automatic systems. – Mechatronics, automation , control 2003, N 2, p. 13-17. (In Russian).

[5] Timofeev A.V. Multi-Agent Information Processing and Adaptive Control in Global Telecommunication and Computer Networks. – International Journal "Information Theories and Their Applications", 2003 v. 10, N 1, pp.54–60.

[6] Syrtzev A.V., Timofeev A.V. Neural Approach in Multi-Agent Routing for Static Telecommunication Networks. – International Journal "Information Theories and Their Applications", 2003 v. 10, N 2, pp. 167–172.

[7] Timofeev A.V. Architecture and principles of building of multi-agent telecommunication networks of new generation – Proceedings of XI-th Russian conference "Telematics'2004" (7–10 July 2004, Saint-Petersburg), v 1, pp. 172–174.

[8] A.V.Timofeev, Ostiouchenko I.V. Multi-agent control for flows in telecommunication systems. – Proceedings of XI-th Russian conference "Telematica'2004" (7–10 June 2004, Saint-Petersburg), v. 1, p. 177–178.

[9] Timofeev A.V., Syrtzev A.V. Neural methods for optimal dynamic routing for data flows.– Proceedings of IX Saint-Petersburg international conference "Regional informatics–2002"(22-28 November2002, Saint-Petersburg, pp. 146–149 (In Russian).

[10] Timofeev A.V. Multi-agent control and architecture of global infotelecommunication networks of new generation. – Proceedings of IX Saint-Petersburg international conference "Regional informatics–2004"(22-24 June 2004, Saint-Petersburg), p. 102–103.

[11] Timofeev A.V. Intelligent Control Applied to Non-Linear Systems and Neural Networks with Adaptive Architecture. – International Journal on Intelligent Control, Neurocomputing and Fuzzy Logic, 1997, Vol.1, N1, pp. 1-18.

[12] Timofeev A.V. Method for synthesis of diophantine networks of minimal complexity – AS Reports, 1995, v.301, N3, p. 1106-1109.

[13] Timofeev A.V. Optimal synthesis and minimization of complexity of gene-neural networks on genetic data bases. – Neurocomputers: their development and application, N 5-6, 2002, p. 34-39.

[14] Timofeev A.V. Polynomial Neural Network with Self-Organizing Architecture. – International Journal on Optical Memory and Neural Networks, 2004, N 2.

[15] A.V.Timofeev Intelligent and multi-agent control for robotic systems with the use of models of a virtual reality. – Mechatronics, 2000, № 3. pp. 26-31. (In Russian).

[16] A.V.Timofeev, Ostiouchenko I.V. Architecture of telemedical network and tasks of man-machine interaction in multi-agent neurosurgical system - Conference "Clinical stereotaxis. Experience and prospects of application of the domestic manipulator OREOL". - Theses of the of a conference reports. (Saint-Petersburg, December 5-6, 2001) ISBN 5-9000780-35-X. (In Russian)

[17] A.V.Timofeev, Litvinov M.V., Popov M.B. Development of the man-machine interface on the basis of virtual model neurosurgical of the robot. – Proceedings of international conference "Adaptive robots and general theory of systems ", (Saint-Petersburg, June 8-11, 2004), pp. 16-19 (In Russian).

[18] C.Burghart, O.Schorr, S.Yigit, N.Hata, K.Chinzei, A.Timofeev, R.Kikinis, H.Woern, U.Rembold. A Multi-Agent-System Architecture for Man-Machine-Interaction in Computer Aided Surgery. – Proceedings of the 16th IAR Annual Meeting IAR/ICD Workshop (November 22-23,2001, Strasbourg, France), pp.117-122.

Author Information

Adil V. Timofeev – Saint-Petersburg Institute for Informatics and Automation of Russian Academy of Sciences, Head of the Neural Informatics and Intelligent Control Laboratory; 199178, Russia, Saint-Petersburg, 14-th Line,39, e-mail: tav@iias.spb.su

GENERAL ASPECTS OF CONSTRUCTING AN AUTONOMOUS ADAPTIVE AGENT

Arthur Pchelkin

Abstract: There are a great deal of approaches in artificial intelligence, some of them also coming from biology and neirophysiology. In this paper we are making a review, discussing many of them, and arranging our discussion around the autonomous agent research. We highlight three aspect in our classification: type of abstraction applied for representing agent knowledge, the implementation of hypothesis processing mechanism, allowed degree of freedom in behaviour and self-organizing. Using this classification many approaches in artificial intelligence are evaluated. Then we summarize all discussed ideas and propose a series of general principles for building an autonomous adaptive agent.

Keywords: reinforcement learning, neural networks, functional systems theory, inductive automaton.

Introduction

One of the directions in artificial intelligent (AI) is adaptive autonomous agents research (AAAR). This research direction started actively growing since 1985 [Maes95,Wil85], however, there were proposed to make researches in similar directions also before it [Bong75].

In this paper we analyse general problems that appear developing autonomous adaptive agents learning algorithm. The goal was to analyse different approaches, directly or indirectly connected with autonomous agent research, and to develop constructive principles for autonomous adaptive agent architecture.

We highlight three aspect in our classification: (1) type of abstraction applied for representing agent knowledge, (2) the implementation of hypothesis processing mechanism, (3) allowed degree of freedom in behaviour and self-organizing, and the structure of the paper is arranged in corresponding order.

Type of Abstraction

There are many different learning algorithms [Hunt75,Mit99,Fau94] well known in AI that model different aspects of intelligence. In the general case the agent needs ability to model practically each aspect of intelligence, because there is no possibility to specialize on some specific aspect of cerebration. In this case, a question appears: how to integrate so many approaches into single architecture?

Analyzing the question about the integration of different approaches, modeling different aspects of thinking, we concluded that there is a need to discuss what is common and what is different in these approaches. The common is that each of them is based on some type of abstraction, but the different is that in each specific case it could be applied specific kind of abstraction. To analyze this question, we selected several typical approaches (see tab. 1).

Table 1. The concrete and the abstract in different approaches

An approach	The concrete	The abstract
Neural networks with supervisor, e.g. error backpropagation algorithm [Fau94]	A training set of vector pairs	Neural network itself, i.e. synaptic weights
Neural network without supervisor, Kohonen feature map [Fau94]	A training set of vectors	Synaptic weights vectors – cluster centers
Clustering algorithms [Ryz77]	A set of vectors	Vectors – cluster centers
Decision tree learning, e.g. algorithm ID3 [Mit99]	A training set of object with known attribute values and classification	Decision tree
Finite automata learning [Ang87, RS93, She95]	An observation table or a sequence of pairs “action-perception”	Finite automata itself – a graph of states and transitions
Hidden Markov models learning [Rab89,Chr92]	A sequence of perceptions	A graph of probabilistic transitions between states
Suffix trees (for hidden Markov models)	A sequence of pairs “action-	Suffix tree

[Ron94, McCallum95]	perception"	
Statistical properties discovery, e.g. correlation	A set of vectors	Correlation matrix (between parameters)
Statistical parameters	A set of numbers	Average, variance and so on

Each approach has two levels: abstract and concrete information. Each approach could be described this way: there is some input information, i.e. the concrete, and it is made some assumption about the structure of this information, - the assumption is used for extracting the abstract from the concrete. Then the abstract is used for decision-making automatically or manually.

Autonomous adaptive agent needs the abstract for the similar purpose, i.e. for decision-making using analogies. Inserting specific kind of abstraction into the architecture of the agent could speed up and simplify the learning process, and in some cases it could be simplified even till several parameters adjustment. However, it must bound adaptability and flexibility of architecture that are very important properties of autonomous adaptive agent.

Inductive automaton: Several researchers have come through another way, e.g. Yemelianov-Yaroslavsky proposed a new neural network model, called "inductive automaton" [Yem90]. According to the author idea, specific aspects of intelligence, i.e. specific types of abstraction, must not be mechanically included into the inductive automaton architecture, but different types of abstraction must appear in the inductive automaton as a by-product of solving by it one single task – minimization on energy consumption by its neurons. Our analysis showed that this neural network is able to work as Kohonen [Fau94] map or as finite automata learning algorithms [Ang87] (of course, as some analogy).

The analysis of the last work brings us at the conclusion that it is more perspective not to integrate specific preprogrammed kinds of abstraction into the agent architecture, but to try to discover the main reason for appearing of abstraction in intelligent system, and inductive automaton [Yem90] is an interesting solution of the discussed problem.

Hypotheses Processing

As it was mentioned, intelligent system needs the abstract for decision-making. However, any abstraction is an assumption about properties of external environment, i.e. hypothesis. Let us analyze hypotheses processing in different approaches (see tab. 1).

Finite automata learning: Finite deterministic automata learning algorithms [Ang87, RS93, She95] employ a hypothesis about each of states known to algorithm. Such hypothesis is an assumption that the corresponding state is unique, and at each time moment learning algorithm work only with some approximation of learning automaton. This approximation is represented by graph where nodes are states and edges are transitions between the states. If the number of states in the learning automaton is not known then it is not possible to check: does the state in the model correspond to one state in the automaton, or it really corresponds to several states. That's why, in practice these algorithms work using such principle.

A model, containing such hypotheses, is used for solving different tasks in the environment, e.g. for calculating of optimal policy. However, if fault occurs, i.e. the response of the automaton conflicts with the model, then the learning algorithm automatically identify fallacious hypothesis and splits the corresponding state in the model. So, after the conflict identification the approximation is made corrected. For decision-making it is enough to have an abstract model (in this case the model of automaton), but to identify and correct a fallacious hypothesis, there is necessary to have links between the abstract and the concrete, used to extract the abstract. For example, in the paper [Ang87] an observation table is used, that contains all the concrete information, that is used for identification and correction of fallacious hypotheses.

Hidden Markov models: The similar principle is applied for hidden Markov models (HHM) learning [Rab89, Chr92]. It is made the same assumption that each state in the model is unique in HHM, and concrete information (a sequence of perceptions) is used for calculating of matrix that stores transition probabilities and for splitting of one state in order to improve likehood of the model.

Suffix trees: Some approaches [McCallum95] efficiently uses suffix trees [Ron94] in reinforcement learning with hidden state. In this case, the states of the environment are represented not by graph nodes but by suffix tree leaves. The algorithm uses hypotheses about each of leaves that it is a unique state in the environment.

However, if using a statistical test it is discovered the one of hypothesis is fallacious then the algorithm splits the corresponding leaf, adding new leaves below it. The concrete also is used to calculate transitions' probabilities.

Decision trees: Decision trees [Mit99] could be applied for pattern recognition. Decision trees learning algorithms, e.g. ID3, could be interpreted this way. At each time moment, a decision tree could be considered sufficient for making correct classification of objects from some known set. However, if a new object appears, that could not be classified correctly, the tree can be improved by adding new leaves. It should be noted, that the abstract, i.e. the tree itself, was sufficient for classification, but, to make a correction in this abstract, the concrete is needed, i.e. a training set.

Inductive automaton: Self-organizing of inductive automaton [Yem90] could be interpreted from the hypotheses processing point of view. Neurons, that frequently activates together, could be joined by excitatory links into assemblies. Each of neurons in such a group can have individual links to and from another neurons inside or outside of the assembly. The circumstance, that al the neurons are connected only by excitatory links, could be considered as a hypothesis, i.e. an assumption, that all the neurons in this group performs the same function, but maybe in different contexts.

The hypothesis about neurons functions similarity could be discovered to be wrong. In this case, there is a possibility to split assembly by inhibitory links. It occurs through a transformation of weak excitatory link into strong inhibitory link. As in the previous cases, to correct a wrong hypothesis, i.e. to split an assembly, it is necessary to have the concrete information, i.e. individual links between neurons, because the individual links decides how and where the assembly must be split. Neural assemblies in inductive automaton have also an interesting semantic interpretation, e.g. an assembly can correspond to some concept, and each neuron can represent some aspect of it. In this case, the splitting of an assembly could mean the formation of subconcepts or subclasses. Interesting ideas connected to this topic could be found in paper [Kus2000], however, in contrast with inductive automaton there is used fixed unchangeable structure of assemblies.

To sum up this part, we need to conclude that the abstract model building usually is made through integration of concrete information into the abstract model. During such integration, there is a need to make assumptions about correspondences between units of concrete information and units of the abstract representation. Therefore, the learning algorithm must be provided with the ability of efficient wrong hypotheses processing. Usually, decision-making needs only an abstract model, but in the case of fault it is necessary back to the concrete information for identifying and correcting of a wrong hypothesis in the abstract.

Degree of Freedom

In the phrase "autonomous adaptive agent" [Maes95] the word "autonomous" means the agent has high degree of freedom in decision-making, and word "adaptive" - high degree of freedom in self-organizing. Building an autonomous agent a question arises: where, when and how much freedom to give an autonomous agent? It is difficult to decide because the deficiency of freedom may dramatically bound abilities to adapt the custom environment, but redundant degree of freedom can produce chaos and instability. To illustrate this dilemma, let us consider traditional conflict between classifier systems and connectionist approaches.

Classifier systems: The idea of classifier systems [Hol86] was to provide an intelligent systems with a maximal degree of freedom in self-organizing in order to improve its adaptation abilities. Really, classifier systems (in original Holland framework) don't have potential boundaries in self-organizing possibilities, but at the same time there is not any paper presenting the full exploitation of these possibilities. Using classifier systems, it is possible to obtain good results only in specially simplified environments, e.g. the paper [Wil85] presented good results for a purely reactive classifier system, i.e. without a temporal memory. However, after improving this system by adding a temporal memory a series of negative results has been obtained [CR94]. The improved system was not able to guarantee stable behavior and self-organizing. Then a new improved version was developed [Wil95], but it also was able to show good results only in very simple environments.

Connectionist approaches: Connectionist approaches, employing artificial neural networks [Fau94] as an engine for information processing, allow much more limited degree of freedom for self-organizing in order to provide more stable behavior and self-organizing. In many cases the self-organizing of such a system could be considered as a parameter adjustment. At the same time, classifier system can be self-organizing through different variations in behavior and successful constructions inventing. Therefore, it is possible to say that classifier systems have too much freedom but connectionist approaches employ too limited degree of freedom.

Inductive automaton: Inductive automaton [Yem90] can be considered as an compromises between two previous approaches. On the one hand, inductive automaton processes information as a neural network, but on the other hand, self-organizing is performed through successful construction fixing. Each neuron in this network has its own degree of freedom that depends on the age of the neuron. If a neuron frequently activates and takes active participation in the network processes then its age decreases, and, simultaneously, its degree of freedom also decreases. In opposite case, if a neuron is inactive for a very long time its degree of freedom increases. In this example, degree of freedom is hardly preprogrammed for a component of system, but depends on its functioning. Similar approaches, employing different degrees of freedom for different neurons, could be found in works [Amo73, Wick99].

Functional systems theory: Inductive automaton has a very serious disadvantage – it is not able to control degrees of freedom of its components, i.e. neurons. For example, redundant degrees of freedom of elements could produce only chaos and instability in system work in the case when the agent is performing certain sequence of actions using a well-tested plan. Therefore, from this point view we highlight functional systems theory [Ano74] as the most perspective approach that could be described this way.

Anokhin' s opinion is that the interaction between elements, taken by its own, is not able to form a system from a set of these elements, therefore, according to his opinion, the system-formation factor is only useful for the system adaptive result. So, instead of concept "interaction" must be exploited concepts "collaboration" or "co-operation". Additionally, the system should make the problem statement, including the criterion of a solution and the program of actions, before any acting performed by it. In the case of insufficiency of the obtained result the system should stimulate its activating mechanisms, performing active selecting of new components, changing degree of freedom of acting components, and at the end after several "trials and errors" the system should obtain fully sufficient useful result. In such a way, tasks solving is performed through efficient managing of systems components degree of freedom, and in case of success exploited degrees of freedom should be fixed.

Ordering in interacting between elements should be set using their degrees of contribution in collaboration performed in order to obtain by the system a preprogrammed useful result. If a degree of freedom does not help to obtain the useful result, it should be eliminated from the use. The system should efficiently manage degree of freedom of its elements, and simultaneously only a small subset of all elements could be active.

The described conflict between principal possibilities of self-organizing and stability in system functioning could be solved as follows: hardness in internal problem statement by a system – the guarantee of its stability, but flexibility in problem solution obtained through testing different degrees of freedom – guarantee of principally unlimited possibilities in adaptation and self-organizing. Therefore, we propose to use functional systems theory as framework for building an autonomous adaptive agent.

Internal Tasks

Functional systems theory considers a useful preprogrammed result as a single system-forming factor. It means that self-organizing can occur only through tasks solving. So, a question arises "how to produce sufficient amount of tasks?" because achieving the global goal defined by external environment could not occur sufficiently often, especially, in the beginning of learning. For example, searching for a certain object could take increasable much time if the agent exploits random walk strategy [Whit91]. Besides, usually the external environment provides the agent with a very small set of hard-to-reach goals that need the completion of self-organizing. Therefore, a set of externally defined goals could not server as a sufficient source tasks for self-organizing, so, the agent needs to exploit an additional internally defined set of tasks as a sufficient source for self-organizing. Solving internal tasks the system can obtain skills sufficient for achieving the global externally defined goal [Pch2003a].

We propose three main sources for internal tasks:

- Faults identification and correcting;
- Achieving hard-to-reach states;
- Memorizing the environment behavior.

Faults identification and correcting: Faults identifying and then correcting is an important task for an intelligent autonomous self-organizing system. During learning, the agent has autonomously to make many decisions in the lack of human control, and many of these decisions, producing a serious impact on self-organizing, can appear to be erroneous. Faults and breakdowns may occur very often in such self-organizing system, so, the system components should have ability to "feel", identify and correct the place of a fault. Inductive automaton is a good example of such property implementation [Yem90].

In inductive automaton each neuron has not two as usual, but thee stable states: inactive, half active and fully active. The intermediate half active state could be considered as an indicator of a breakdown of a neuron assemblies functioning consistency [Pch2003c]. Therefore, correcting performed by a system of its own structure should be considered as a normal response produced by the system against the appeared breakdown. However, this response, according to Anokhin's theory, should emerge not as an automatic predefined reaction, but only as an automatic problem statement. Inductive automaton has a homeostatic architecture, and, unfortunately, it has a huge list of automatic internal reactions, employing for self-regulation, that take place in pre-programmed way, and the system has no ability to review their utility.

Besides of examples in AI, it is possible to find the confirmation of faults identification relevancy in distant examples, e.g. modern programming languages usually have specially defined abstract class – an exception for faults and breakdowns processing. Thereby, a fault situation is made normal, because of specially designed mechanism to handle it, and the system keeps ability to function normally also in case of faults and breakdowns.

Achieving hard-to-reach states: Usually to achieve the global external goal, the agent must have a set of skills for it. How to obtain these skills before reaching the global goal? – It is possible to employ such a heuristic: *skills and knowledges, needed to reach the global goal, could be obtained through reaching usual hard-to-reach states of the environment*. Really, the goal state is the same state as all other states, therefore, the agent can use any states to obtain needed skills. For example, in the paper [Pch2003a] the agent through training in reaching of local goals, i.e. usual hard-to-reach states temporally defined as internal goals, obtains knowledge that helps to achieve the global goal.

In real situation the agent doesn't have direct access to a global state of the environment, instead, it has only partial information about it. Besides, the agent can form its model of the environment in order to identify its current position in it. Therefore, all recognizable by the agent states could be represented by its internal states, e.g. a set of currently activated neurons. So, reaching of an internal state could be defined as a reaching of certain states of its components.

Therefore, it is proposed to employ the task of achieving hard-to-reach states for self-organizing. For example, the agent can try to make active certain group of neurons by performing different sequences of actions in the environment. The process of solving such tasks could be considered as an exploration activity in the environment performed in order to gain new knowledge.

Memorizing the environment behavior: The third proposed kind of tasks is memorizing behavior of the environment, i.e. memorizing observed sequences of events in the external environment through formation of corresponding skills that help to reproduce them internally.

In the first and second parts there were analyzed a series of learning algorithms, modeling different aspects of intelligence. Each of them employs the concrete to obtain the abstract. Such functioning could be interpreted as iterative process, where through each iteration the learning algorithm "tries to memorize" the current portion of the concrete, integrating it into the abstract. For shortness, we will refer this process "memorizing".

Practically all the artificial neural networks are constructed for tabular static data processing, e.g. a set of vectors. However, in natural neural network, the memorization unit is not a vector, but it is a dynamical stereotype [Pavl49]. Therefore, we also make an accent on memorizing of logical sequence of observed events, as an elementary unit of the environment behavior.

Obviously, each component of the system could be participated in memorizing of a great deal of sequences. Therefore, it is possible to obtain a conflict situation when new information is not integrating with the old one, damaging memory about it. Moreover, mechanically recorded new information may damage the consistency of the information stored in system components. Therefore, memorizing of information must be focused not on recording of new information, but on searching for a way to generate this information by the system. The system of component should to “born“ needed dynamical stereotype.

We propose to solve the task of memorizing of the environment behavior following way. The agent directly records an observing sequence of events into short-term memory. Then the agent states a problem – to reproduce this sequence in the correct order employing its components. To achieve the planned result, the agent should find and setup degrees of freedom of its components sufficient for memorizing of the previously recorded sequence. This process can occur in correspondence to functional systems theory proposed by Anokhin. The agent can stimulate different mechanisms for searching for a sufficient set of components and then applying different degrees of freedom for them. Thus, the agent through trials and errors should to reproduce by its components a previously recorded sequence.

To protect consistency of the system of components, it is proposed to following criterion: *the memorization of a dynamical stereotype must be done through minimal increasing of degrees of freedom of components, and, additionally, the agent must test its ability to reproduce a memorized dynamical stereotype*.

Discussion

In this part we will analyze additional problems that must be taken into account building autonomous agent.

Multitasking: The total number of tasks could be very large. Therefore, these tasks solution may conflict between each other. So, there is a need for mechanism that selects the most important tasks from the set of all tasks. This mechanism must choose one task, control its solving time and compare obtained result with the planned one. In the case of failure, it must manage degrees of freedom of components or choose another task. Functional systems theory [Ano74] proposes a principle of dominating of an active functional system under alternative systems, so, this theory proposes an adequate solution.

Physical and logical layers: Many approaches, *e.g.* [Amo79, Wick99], employ an assumption that a simple relation between physical and logical layer could be employed. In contrast, our position is that the physical and logical layers must be principally separated. For example, the man is able to operate simultaneously with 5-7 objects. However, much more thousands of neurons could be activated simultaneously. It means that the brain work at much more primitive level, *i.e.* complex behavior is obtained not thought complex semantic meaning of components, but through the huge number of possible way of collaboration in the set of simple elements.

Elements must have a primitive problem-oriented sense, but at the same time they can have advanced system forming semantic. For example, neurons in inductive automaton have additional intermediate half-active state that makes a neuron able to recognize its faults and inconsistent work. It is necessary to focus not on problem domain oriented semantic meaning but on calculating abilities, *i.e.* must have a principal possibility (there must be a principal possibility to obtain it during self-organizing) to generate practically any behavior.

Formation of abstraction: As it was mentioned, preprogrammed types of abstraction must not be embedded in the architecture of the agent, because it may limit freedom and principal of system self-organizing. However, how to obtain different abstractions, if they are not embedded into the system? We propose to formulate the main reason as follows: *the formation of different types of abstraction must be obtained as a by-product of dynamical stereotype memorizing by distributed architecture*.

Solving different tasks, a system will learn to reproduce the huge number of dynamical stereotypes, and each of components could be reused in many stereotypes. It means that the function of a component will be abstract regarding the stereotype. So, it is a reason for abstraction formation. Growth of the logical layer, consisting of different stereotypes, based on fixed physical layer, *i.e.* a fixed set of components, should mean the formation of more and more abstract knowledge representations. This principle applied to [Yem90] has been analyzed in [Pch2003b].

Problem of duplicates: The problem of information duplicates is important enough. Real self-organizing has a serious drawback – in different places of the component system similar substructure may be formed. Additionally, formed duplicates are able to speed up explosion of duplicates formation. Interesting, that the same phenomena could be observed in far areas, e.g. large software products developing, and the development of abstract classes and methods is a widely employed approach against duplicates in the code of programs. Therefore, it is important answer a question: how the similar parts will be able to find each other? We distinguish to base approaches:

- *Simultaneous counteraction.* For example, in Kohonen feature map [Fau94] only one neuron could be activated, or fixed-size group of neurons could be activated in associative-projective neural networks [Kus2000] and M-networks [Amo73]. Limiting a number of simultaneously active neurons, it is possible to stanch the growth of duplicates, because the neurons of duplicated substructures must be activated simultaneously.
- *Delayed counteraction.* To our mind, the most efficient method against duplicates is a formation of neural assemblies. The neurons in duplicated substructures with analogous functions must be activated simultaneously, so, applying Hebb's learning rule [Fau94], corresponding neurons must be joined into assemblies. These ideas could be find in inductive automaton [Yem90] that has been analyzed in paper [Pch2003c].

Conclusion

Basing on this investigation as the most perspective approaches have been selected works [Ano74, Yem90], and the discussed ideas could be summarized as a series of following principle:

- The architecture of the autonomous adaptive agent must be based on functional systems theory [Ano74]. Before any action the agent must state the task and choose a program of actions, and the system must have ability to identify an erroneous decision in the case of fault.
- The single self-organizing and system-forming factor is obtaining of useful preprogrammed result, *i.e.* a solution of some problem. For this purpose, it has been proposed to employ following sources of tasks: faults identification and correcting, achieving hard-to-reach states and memorizing the environment behavior.
- Preprogrammed types of abstraction must not be embedded into the agent architecture. However, the composition of components must have a principal ability to be sufficient for any specific behavior implementation. Abstraction formation must be derived as a by-product of memorizing of huge number of dynamical stereotypes by distributed architecture. Logical layer growth based on fixed-size physical layer must imply the formation of more and more abstract forms of knowledge representation.
- The principal ability to efficiently process hypotheses must be integrated into the architecture. The abstract model usually is sufficient for decision-making, but in the case of fault it is necessary to have ability for backing to concrete original information in order to identify an erroneous hypothesis and for making correction in the abstract knowledge representation. In the most perspective way this idea is implemented inductive automaton [Yem90].

To conclude, it is necessary to remember works occurring in the similar direction, that have made an impact on the current investigation, for example the paper [Ku2003].

Acknowledgement. This research was supported in part by the Latvian Science Foundation under grant No.02-86d.

Bibliography

[Shannon, 1949] C.E.Shannon. The Mathematical theory of communication. In: The Mathematical Theory of Communication. Ed. C.E.Shannon and W.Weaver. University of Illinois Press, Urbana, 1949.

[Amo73] Amosov, N.M., Kasatkin, A.M., Kasatkina, L.M., Talayev, S.A. Automata and the mindfull behaviour. Kiev: Naukova Dumka, 1973. (in Russian)

[Amo79] Amosov, N. M. Algorithms of the Mind. Kiev: Naukova Dumka, 1979. (in Russian)

[Ang87] Angluin D. Learning regular sets from queries and counterexamples, Information and Computation, 1987. Nr. 75, pp.87.-106.

[Ano73] Анохин П.К. Принципиальные вопросы общей теории функциональных систем. Принципы системной организации функций. – М.: Наука, 1973.

[Ano74] Анохин П.К. "Системный анализ интегративной деятельности нейрона.", «Успехи физиол. наук», 1974, т.5, №2, с.5-92,.

[Bong75] Бонгард М.М., Лосев И.С., Смирнов М.С. Проект модели организации поведения – Животное // Моделирование обучения и поведения. – М.: Наука, 1975.

[Chr92] Chrisman, L. Reinforcement Learning with Perceptual Aliasing: The Perceptual Distinctions Approach, In Tenth National Conference on Artificial Intelligence, 1992, pp.183-188.

[CR94] Cliff D., Ross S. Adding temporary memory to ZCS. Adaptive Behavior, Nr. 3(2), 1994, pp.101-150.

[Fau94] Fausett, L. V. Fundamentals of neural networks: architectures, algorithms and applications, Prentice-Hall, Inc. New Jersey, 1994, 461 p.

[Hol86] Holland J.H., Escaping Brittleness: the Possibilities of General-Purpose Learning Algorithms applied to Parallel Rule-Based Systems, In: Machine Learning, an Artificial Intellegence Approach, Nr. II., 1986.

[Hunt75] Hunt E.B.: Artificial intelligence, Academic Press, Inc., NewYork, 1975, p.558.

[KLM96] Kaelbling, L.P., Littman, L.M., Moore, A.W. Reinforcement learning: a survey, Journal of Artificial Intelligence Research, Vol. 4, 1996, pp.237-285.

[Ku2003] Kursin, A., Neural Network: Input Anticipation May Lead To Advanced Adaptation Properties, In: Kaynak, O. et al. (eds.): Artificial Neural Networks and Neural Information Processing. Springer-Verlag, 2003, pp.779-785.

[Kus2000] Rachkovskij, D.A., Kussul, E.M. Building large-scale hierarchical models of the world with binary sparse distributed representations, 2000.

[Maes95] Maes P., Modeling Adaptive Autonomous Agents, Artificial Life: An overview, MIT Press, 1995.

[McCallum95] McCallum, A., Reinforcement learning with selective perception and hidden state (Ph.D. dissertation). Department of Computer Science, University of Rochester, Rochester, NY, 1995.

[Mit99] Mitchel, T.H., Machine learning, The McGraw-Hill Companies. Inc., 1999.

[Pavl49] Павлов И.П. Полное собрание трудов. Т. 3. М..Л., 1949, с. 496.499.

[Pch2003a] Pchelkin A. Efficient exploration in reinforcement learning based on Utile Suffix Memory, In: Informatica, Lithuanian Academy of Sciences, Vol.14., 2003.

[Pch2003b] Pchelkin A. Self-organizing in neural networks based on memorizing, In Scientific Proceedings of Riga Technical University: Computer Science, Information Technology and Management Science, RTU, Riga, Vol. 5.

[Pch2003c] Pchelkin A. Inductive automaton: optimal spike frequency, In Scientific Proceedings of Riga Technical University: Computer Science, Information Technology and Management Science, RTU, Riga, Vol. 5.

[Ron94] Ron, D., Singer, Y., Tishby, N. Learning probabilistic automata with variable memory length, In Proceedings of the Seventh Annual ACM Conference on Computational Learning Theory, 1994, pp.35-46.

[Rab89] Rabiner, Lawrence R. 1989. A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE 77(2).

[Ryz77] Ryzin J.V.: Classification and clustering, Academic Press, Inc., NewYork, 1977. p.389.

[RS93] Rivest R. L., Schapire R. E. Inference of finite automata using homing sequences, Information and Computation, Nr.103(2), 1993, pp.299.-347.

[She95] Shen W.-M. Learning finite automata using local distinguishing experiments. Computational Learning Theory and Natural Learning Systems, MIT Press, Nr. III: Selecting Good Models, 1995.

[Sut98] Sutton, R.S., Barto, A.G. Reinforcement learning: An introduction. Cambridge, MA: MIT Press, 1998.

[Whit91] Whitehead, S.D. Complexity and cooperation in Q-learning, In Proceedings of the Eighth International Workshop on Machine Learning, 1991, pp.363-367.

[Wick99] Wickelgren, W.A.: Webs, Cell Assemblies, and Chunking in Neural Nets. In Canadian Journal of Experimental psychology. Vol. 53. 1, 1999, pp.118-131.

[Wil85] Wilson S. W. Knowledge growth in an artificial animal, Proceeding of the First International Conference on Genetic Algoritms and Their Applications. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 1985, pp.16-23.

[Wil95] Wilson S. W. Classifier fitness based on accuracy, Evolutionary Computation, 3(2), 1995, pp.149.-175.

[Yem90] Yemelianov-Yaroslavsky L.B. Intelligent quasi-biological system. Inductive automaton. Nauka, Moscow, 1990, 112p. (in Russian)

Author Information

Arthur Pchelkin – Ph.d. student; Decision Support Systems Group, Institute of Information Technology, Technical University of Riga; 1, Kalkyu St., Riga LV-1658, Latvia; e-mail: arturp@inbox.lv

REPRESENTING "RECURSIVE" DEFAULT LOGIC IN MODAL LOGIC

Frank Brown

Abstract: The "recursive" definition of Default Logic is shown to be representable in a monotonic Modal Quantificational Logic whose modal laws are stronger than S5. Specifically, it is proven that a set of sentences of First Order Logic is a fixed-point of the "recursive" fixed-point equation of Default Logic with an initial set of axioms and defaults if and only if the meaning of the fixed-point is logically equivalent to a particular modal functor of the meanings of that initial set of sentences and of the sentences in those defaults. This is important because the modal representation allows the use of powerful automatic deduction systems for Modal Logic and because unlike the original "recursive" definition of Default Logic, it is easily generalized to the case where quantified variables may be shared across the scope of the components of the defaults.

Keywords: Recursive Definition of Default Logic, Modal Logic, Nonmonotonic Logic.

1. Introduction

One of the most well known nonmonotonic logics [Antoniou 1997] which deals with entailment conditions in addition to possibility conditions in it's defaults is the so-called Default Logic [Reiter 1980]. The basic idea of Default Logic is that there is a set of axioms Γ and some non-logical default "inference rules" of the form:

$$\frac{\alpha : \beta_1 \ldots \beta_m}{\chi}$$

which is intended to suggest that χ may be inferred from α whenever each $\beta_1,\ldots,\beta_m$ is consistent with everything that is inferable. Such "inference rules" are not recursive and are circular in that the determination as to whether χ is derivable depends on whether β_i is consistent which in turn depends on what was derivable from this and other defaults. Thus, tentatively applying such inference rules by checking the consistency of $\beta_1,\ldots,\beta_m$ with only the current set of inferences produces a χ result which may later have to be retracted. For this reason inferences in a nonmonotonic logic such as Default Logic are essentially carried out not in the original nonmonotonic logic, but rather in some (monotonic) metatheory in which that nonmonotonic logic is monotonically defined. [Reiter 1980] explicated the above intuition by defining Default Logic "recursively" in terms of the set theoretic proof theory metalanguage of First Order Logic (i.e. FOL) with (more or less) the following fixed-point expression[1]:

$$'\kappa=(\text{dr } '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)$$

where dr is defined as:

$(\text{dr } '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i) =_{df} \cup_{t=1,\omega}(\text{r } t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)$

$(\text{r } 0\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i) =_{df} (\text{fol } '\Gamma)$

$(\text{r } t{+}1\ 'k\ '\alpha_i{:}'\beta_{ij}/'\chi_i) =_{df} (\text{fol}((\text{r } t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)\cup\{'\chi_i{:}\ ('\alpha_i\varepsilon(\text{r } t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\wedge_{j=1,m_i}('(\neg\beta_{ij})\not\in'\kappa)\}))$

where $'\alpha_i$, $'\beta_{ij}$, and $'\chi_i$ are the closed sentences of FOL occurring in the ith "inference rule" and $'\Gamma$ is a set of closed sentences of FOL. A closed sentence is a sentence without any free variables. fol is a function which produces the set of theorems derivable in FOL from the set of sentences to which it is applied. The quotations

---

[1] [Reiter 1980] actually used a recursive definition whereby the r sets do not necessarily contain all their FOL consequences:

$(\text{dr } '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i) =_{df} \cup_{t=1,\omega}(\text{r } t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)$

$(\text{r } 0\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i) =_{df}\ '\Gamma$

$(\text{r } t{+}1\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i) =_{df} (\text{fol}(\text{r } t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\cup\{'\chi_i{:}('\alpha_i\varepsilon(\text{r } t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\wedge_{j=1,m_i}('(\neg\beta_{ij})\not\in'\kappa)\}$

If this definition were used then all the theorems in this paper should have $(\text{r } t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)$ replaced by $(\text{fol}(\text{r } t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$ and $(\text{dr } '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)$ replaced by $(\text{fol}(\text{dr } '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$.

appended to the front of these Greek letters indicate references in the metalanguage to the sentences of the FOL object language. Interpreted doxastically this fixed-point equation states:

The set of closed sentences which are believed is equal to
the union of all sets of closed sentences which are believed at any time.
That which is believed at time 0 is the set of closed sentences derived by the laws of FOL from 'Γ.
That which is believed at time t+1 is the set of closed sentences derived by the laws of FOL
from the union of both
the set of beliefs at time t
and the set of all 'χ_i for each i such that
the closed sentence 'α_i is believed at time t and for each j, the closed sentence 'β_{ij} is believable.

The purpose of this paper is to show that all this metatheoretic machinery including the formalized syntax of FOL, the proof theory of FOL, the axioms of set theory, and the set theoretic fixed-point equation is not needed and that the essence of the "recursive" definition of Default Logic in representable as a necessary equivalence in a simple (monotonic) Modal Quantificational Logic. Interpreted as a doxastic logic this necessary equivalence states:

That which is believed is logically equivalent to what is believed at any time.
That which is believed at time 0 is Γ.
That which is believed at time t+1 is
that which is believed at time t and for each i, if α_i is believed at time t and for each j, β_{ij} is believable then χ_i.

thereby eliminating all mention of any metatheoretic machinery.

The remainder of this paper proves that this modal representation is equivalent to the "recursive" definition of Default Logic. Section 2 describes a formalized syntax for a FOL object language. Section 3 describes the part of the proof theory of FOL needed herein (i.e. theorems FOL1-FOL10). Section 4 describes the Intensional Semantics of FOL which includes laws for meaning of FOL sentences: M0-M7, theorems giving the meaning of sets of FOL sentences: MS1, MS2, MS3, and laws specifying the relationship of meaning and modality to the proof theory of FOL (i.e. the laws R0, A1, A2 and A3 and the theorems C1, C2, C3, and C4). The modal version of the "Recursive" definition of Default Logic, called DR, is defined in section 5 and explicated with theorems MD1-MD8 and SS1-SS2. In section 6, this modal version is shown by theorems R1, DR1 and DR2 to be equivalent to the set theoretic fixed-point equation for Default Logic. Figure 1 outlines the relationship of all these theorems in producing the final theorems DR2, FOL10, and MD8.

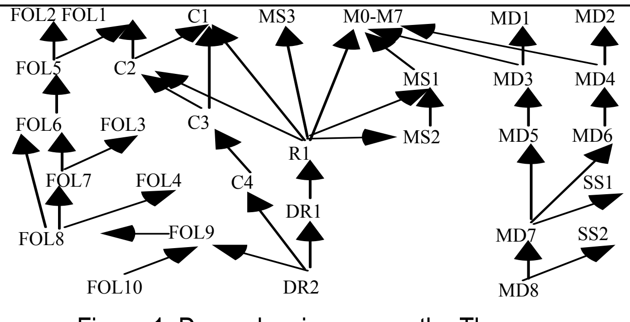


Figure 1: Dependencies among the Theorems

2. Formal Syntax of First Order Logic

We use a First Order Logic (i.e. FOL) defined as the six tuple: ($\rightarrow$, #f, $\forall$, *vars*, *predicates*, *functions*) where $\rightarrow$, #f, and $\forall$ are logical symbols, *vars* is a set of variable symbols, *predicates* is a set of predicate symbols each of which has an implicit arity specifying the number of associated terms, and *functions* is a set of function symbols each of which has an implicit arity specifying the number of associated terms. The sets of logical symbols, variables, predicate symbols, and function symbols are pairwise disjoint. Lower case Roman letters possibly indexed with digits are used as variables. Greek letters possibly indexed with digits are used as syntactic metavariables. γ, $\gamma_1,...\gamma_n$, range over the variables, ξ, $\xi_1...\xi_n$ range over sequences of variables of an appropriate arity, $\pi,\pi_1...\pi_n$ range over the predicate symbols, ϕ, $\phi_1...\phi_n$ range over function symbols, δ, $\delta_1...\delta_n$, σ range over terms, and α, $\alpha_1...\alpha_n$, β, $\beta_1...\beta_n$, χ, $\chi_1...\chi_n$, $\Gamma_1,...\Gamma_n$, φ range over sentences. The terms are of the forms γ and (ϕ δ1...δn), and the sentences are of the forms ($\alpha\rightarrow\beta$), #f, ($\forall\gamma$ α), and (π $\delta_1...\delta_n$). A nullary predicate π or function ϕ is written as a sentence or a term without parentheses. $\varphi\{\pi/\lambda\xi\alpha\}$ represents the replacement of all occurrences of π in φ by $\lambda\xi\alpha$ followed by lambda conversion. The primitive symbols are shown in Figure 2 with their intuitive interpretations.

Symbol	Meaning
$\alpha \rightarrow \beta$	if α then β.
#f	falsity
$\forall\gamma\ \alpha$	for all γ, α.

Figure 2: Primitive Symbols of First Order Logic

The defined symbols are listed in Figure 3 with their definitions and intuitive interpretations.

Symbol	Definition	Meaning		Symbol	Definition	Meaning
$\neg\alpha$	$\alpha \rightarrow$ #f	not α		$\alpha\wedge\beta$	$\neg(\alpha \rightarrow \neg\beta)$	α and β
#t	$\neg$ #f	truth		$\alpha\leftrightarrow\beta$	$(\alpha\rightarrow \beta) \wedge (\beta\rightarrow \alpha)$	α if and only if β
$\alpha\vee\beta$	$(\neg\ \alpha)\rightarrow \beta$	α or β		$\exists\gamma\ \alpha$	$\neg\forall\gamma\ \neg\alpha$	for some γ , α

Figure 3: Defined Symbols of First Order Logic

The FOL object language expressions are referred in the metalanguage (which also includes a FOL syntax) by inserting a quote sign in front of the object language entity thereby making a structural descriptive name of that entity. Generally, a set of sentences is represented as: {'Γ_i} which is defined as: {'Γ_i: #t} which in turn is defined as: {s: $\exists$i(s='Γ_i)} where i ranges over some range of numbers (which may be finite or infinite). With a slight abuse of notation we also write 'κ, 'Γ to refer to such sets.

3. Proof Theory of First Order Logic

FOL is axiomatized with a recursively enumerable set of theorems as its axioms are recursively enumerable and its inference rules are recursive. The axioms and inference rules of FOL [Mendelson 1964] are given in Figure 4.

MA1: $\alpha \rightarrow (\beta\rightarrow \alpha)$

MA2: $(\alpha\rightarrow (\beta\rightarrow \rho)) \rightarrow ((\alpha\rightarrow \beta)\rightarrow (\alpha\rightarrow \rho))$

MA3: $((\neg\ \alpha)\rightarrow (\neg\ \beta))\rightarrow (((\neg\ \alpha)\rightarrow \beta)\rightarrow\alpha)$

MA4: $(\forall\gamma\ \alpha)\rightarrow \beta$ where β is the result of substituting an expression (which is free for the free positions of γ in α) for all the free occurrences of γ in α.

MA5: $((\forall\gamma(\alpha \rightarrow \beta)) \rightarrow (\alpha\rightarrow(\forall\gamma\ \beta)))$ where γ does not occur in α.

MR1: from α and $(\alpha\rightarrow \beta)$ infer β

MR2: from α infer $(\forall\gamma\ \alpha)$

Figure 4: Inferences Rules and Axioms of FOL

In order to talk about sets of sentences we include in the metatheory set theory symbolism as developed along the lines of [Quine 1969]. This set theory includes the symbols ε, $\not\in$, $\supseteq$, $=$, $\cup$ as is defined therein. The derivation operation (i.e. fol) of any First Order Logic obeys the Inclusion (i.e. FOL1), Idempotence (i.e. FOL2), Monotonic (i.e. FOL3) and Union (i.e. FOL4) properties:

FOL1: (fol 'Γ)$\supseteq$'Γ Inclusion

FOL2: (fol 'κ)$\supseteq$(fol(fol 'κ)) Idempotence

FOL3: ('$\kappa\supseteq$'Γ) $\rightarrow$ ((fol 'κ)$\supseteq$ (fol 'Γ)) Monotonicity

FOL4: For any set ψ, if $\forall$t((ψ t)=(fol(ψ t))) and $\forall$t((ψ t+1)$\supseteq$(ψ t)) then ($\cup_{t=0,\omega}$(ψ t))=(fol($\cup_{t=0,\omega}$(ψ t))) Union

From these four properties we prove the following theorems of the proof theory of First Order Logic:

FOL5: $(fol\ '\kappa)=(fol(fol\ '\kappa))$ proof: By FOL1 and FOL2.

FOL6: $(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)=(fol(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$ proof: By induction on t it suffices to prove:

(1) $(r\ 0\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)=(fol(r\ 0\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$ Unfolding r twice gives: $(fol\ '\Gamma)=(fol(fol\ '\Gamma))$ which is FOL5.

(2) $(r\ t{+}1\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)=(fol(r\ t{+}1\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$

Unfolding r twice gives: $(fol((r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)\cup\{'\chi_i{:}\ ('\alpha_i\varepsilon(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin'\kappa)\}))$

$= fol(fol((r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)\cup\{'\chi_i{:}\ ('\alpha_i\varepsilon(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin'\kappa)\}))$

which likewise is an instance of FOL5. QED.

FOL7: $(r\ t{+}1\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)\supseteq(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)$

proof: By FOL6 this is equivalent to: $(r\ t{+}1\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)\supseteq(fol(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$. Unfolding r of t+1 gives:

$(fol((r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)\cup\{'\chi_i{:}\ ('\alpha_i\varepsilon(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin'\kappa)\}))\supseteq(fol(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$

By FOL3 it suffices to prove: $((r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)\cup\{'\chi_i{:}\ ('\alpha_i\varepsilon(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin'\kappa)\})\supseteq(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)$ which holds in set theory. QED.

FOL8: $(\cup_{t=0,\omega}(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)) =(fol(\cup_{t=0,\omega}(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)))$

proof: $\forall t((r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)=(fol(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)))$ holds by FOL6. $\forall t((r\ t{+}1\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)\supseteq(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$ holds by FOL7. Instantiating the hypotheses in FOL4 to these theorems proves this theorem. QED.

FOL9: $(dr\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)=(fol(dr\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$ proof: Unfolding dr twice gives: $\cup_{t=1,\omega}(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)= fol(\cup_{t=1,\omega}(r\ t\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$ which holds by FOL8. QED.

FOL10: $('k=(dr\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\rightarrow('\kappa=(fol\ '\kappa))$ proof: From the hypothesis and FOL9 $'k=(fol(dr\ '\kappa))$ is derived. Using the hypothesis to replace $(dr\ '\kappa)$ by $'\kappa$ in this result gives: $('\kappa=(fol\ '\kappa))$ QED.

4. Intensional Semantics of FOL

The meaning (i.e. mg) [Brown 1978, Boyer & Moore 1981] or rather disquotation of a sentence of FOL is defined in Figure 5 below[1]. mg is defined in terms of mgs which maps each FOL object language sentence and an association list into a meaning. mgn maps each FOL object language term and an association list into a meaning. An association list is a list of pairs consisting of an object language variable and the meaning to which it is bound.

M0: $(mg\ '\alpha)$ =df $(mgs\ '(\forall\gamma 1...\gamma n\ \alpha)'())$ where $'\gamma 1...'\gamma n$ are all the free variables in $'\alpha$

M1: $(mgs\ '(\alpha\rightarrow\beta)a)\leftrightarrow((mgs\ '\alpha\ a)\rightarrow(mgs\ '\beta\ a))$

M2: $(mgs\ '\#f\ a)\leftrightarrow\#f$

M3: $(mgs\ '(\forall\ \gamma\ \alpha)a)\leftrightarrow\forall x(mgs\ '\alpha(cons(cons\ '\gamma\ x)a))$

M4: $(mgs\ '(\pi\ \delta_1...\delta_n)a)\leftrightarrow(\pi(mgn\ '\delta_1\ a)...(mgn\ '\delta_n\ a))$ for each predicate symbol $'\pi$.

M5: $(mgn\ '(\phi\ \delta_1...\delta_n)a) = (\phi(mgn\ '\delta_1\ a)...(mgn\ '\delta_n\ a))$ for each function symbol $'\phi$.

M6: $(mgn\ '\gamma\ a) = (cdr(assoc\ '\gamma\ a))$

M7: (assoc v L)= (if(eq? v(car(car L)))(car L)(assoc v(cdr L))) where: cons, car, cdr, eq?, and if are as in Scheme.

Figure 5: The Meaning of FOL Sentences

The meaning of a set of sentences is defined in terms of the meanings of the sentences in the set as:

$(ms\ '\kappa)$ =df $\forall s((s\varepsilon'\kappa)\rightarrow(mg\ s))$.

[1] The laws M0-M7 are analogous to Tarski's definition of truth except that finite association lists are used to bind variables to values rather than infinite sequences. M4 is different since mg is interpreted as being meaning rather than truth.

MS1: $(ms\{'\alpha: \Gamma\}) \leftrightarrow \forall\xi((\Gamma\{s/'\alpha\})\rightarrow\alpha)$ where ξ is the sequence of all the free variables in $'\alpha$ and where Γ is any sentence of the intensional semantics. proof: $(ms\{'\alpha:\Gamma\})$ Unfolding ms and the set pattern abstraction symbol gives: $\forall s((s\varepsilon\{s: \exists\xi((s='\alpha)\wedge\Gamma)\})\rightarrow(mg\ s))$ where ξ is a sequence of the free variables in 'a. This is equivalent to: $\forall s((\exists\xi((s='\alpha)\wedge\Gamma))\rightarrow(mg\ s))$ which is: $\forall s\forall\xi\ (((s='\alpha)\wedge\Gamma)\rightarrow(mg\ s))$ which is: $\forall\xi(\Gamma\{s/'\alpha\}\rightarrow(mg\ '\alpha))$. Unfolding mg using M0-M7 then gives: $\forall\xi((\Gamma\{s/'\alpha\})\rightarrow\alpha)$ QED

The meaning of a set is the meaning of all the sentences in the set (i.e. MS2):

MS2: $(ms\{'\Gamma_i\}) \leftrightarrow \forall i\forall\xi_i\Gamma_i$ proof: $(ms\{'\Gamma_i\})$ Unfolding the set notation gives: $(ms\{'\Gamma_i: \#t\})$. By MS1 this is equivalent to: $\forall_i\forall\xi_i((\#t\{s/'\alpha\})\rightarrow\Gamma_i)$ which is equivalent to: $\forall i\forall\xi_i\Gamma_i$ QED.

The meaning of the union of two sets of FOL sentences is the conjunction of their meanings (i.e. MS3):

MS3: $(ms('\kappa\cup'\Gamma)) \leftrightarrow ((ms\ '\kappa)\wedge(ms\ '\Gamma))$ proof: Unfolding ms and union in: $(ms('\kappa\cup'\Gamma))$ gives:

$\forall s((s\varepsilon\{s: (s\varepsilon'\kappa)\vee(s\varepsilon'\Gamma)\})\rightarrow(mg\ s))$ or rather: $\forall s(((s\varepsilon'\kappa)\vee(s\varepsilon'\Gamma))\rightarrow(mg\ s))$ which is logically equivalent to: $(\forall s((s\varepsilon'\kappa)\rightarrow(mg\ s)))\wedge(\forall s((s\varepsilon'\Gamma)\rightarrow(mg\ s)))$. Folding ms twice then gives: $((ms\ '\kappa)\wedge(ms\ '\Gamma))$ QED.

The meaning operation may be used to develop an Intensional Semantics for a FOL object language by axiomatizing the modal concept of necessity so that it satisfies the theorem:

C1: $('\alpha\varepsilon(fol\ '\kappa)) \leftrightarrow ([]\ ((ms\ '\kappa)\rightarrow(mg\ '\alpha)))$

for every sentence $'\alpha$ and every set of sentences $'\kappa$ of that FOL object language. The necessity symbol is represented by a box: []. C1 states that a sentence of FOL is a FOL-theorem (i.e. fol) of a set of sentences of FOL if and only if the meaning of that set of sentences necessarily implies the meaning of that sentence. One modal logic which satisfies C1 for FOL is the Z Modal Quantificational Logic described in [Brown 1987; Brown 1989] whose theorems are recursively enumerable. Z has the metatheorem: $(<>\Gamma)\{\pi/\lambda\xi\alpha\}\rightarrow (<>\Gamma)$ where Γ is a sentence of FOL and includes all the laws of S5 Modal Logic [Hughes & Cresswell 1968] whose modal axioms and inference rules are in Figure 6. Therein, κ and Γ are arbitrary sentences of the intentional semantics.

R0: from κ infer $([]\ \kappa)$	A2: $([](\kappa\rightarrow \Gamma)) \rightarrow (([]\kappa)\rightarrow ([]\Gamma))$
A1: $([]\kappa) \rightarrow \kappa$	A3: $([]\kappa) \vee ([]\neg[]\kappa)$

Figure 6: The Laws of S5 Modal Logic

These S5 modal laws and the laws of FOL given in Figure 6 constitute an S5 Modal Quantificational Logic similar to [Carnap 1946; Carnap 1956], and a FOL version [Parks 1976] of [Bressan 1972] in which the Barcan formula: $(\forall\gamma([]\kappa))\rightarrow([]\forall\gamma\kappa)$ and its converse hold. The R0 inference rule implies that anything derivable in the metatheory is necessary. Thus, in any logic with R0, contingent facts would never be asserted as additional axioms of the metatheory. The defined Modal symbols are in Figure 7 with their definitions and interpretations.

Symbol	Definition	Meaning		Symbol	Definition	Meaning
$<>\kappa$	$\neg\ []\ \neg\kappa$	α is logically possible		$[\kappa]\ \Gamma$	$[]\ (\kappa\rightarrow\Gamma)$	β entails α
$\kappa\equiv \Gamma$	$[]\ (\kappa\leftrightarrow\Gamma)$	α is logically equivalent to β		$<\kappa>\ \Gamma$	$<>\ (\kappa\wedge\Gamma)$	α and β is logically possible

Figure 7: Defined Symbols of Modal Logic

From the laws of the Intensional Semantics we prove that the meaning of the set of FOL consequences of a set of sentences is the meaning of that set of sentences (C2), the FOL consequences of a set of sentences contain the FOL consequences of another set if and only if the meaning of the first set entails the meaning of the second set (C3), and the sets of FOL consequences of two sets of sentences are equal if and only if the meanings of the two sets are logically equivalent (C4):

C2: $(ms(fol\ '\kappa))\equiv(ms\ '\kappa)$ proof: The proof divides into two cases:

(1) $[(ms\ '\kappa)](ms(fol\ '\kappa))$ Unfolding the second ms gives: $[(ms\ '\kappa)]\forall s((s\varepsilon(fol\ '\kappa))\rightarrow(mg\ s))$

By the soundness part of C1 this is equivalent to: $[(ms\ '\kappa)]\forall s(([(ms\ '\kappa)](mg\ s))\rightarrow(mg\ s))$

By the S5 laws this is equivalent to: $\forall s(([(ms\ '\kappa)](mg\ s))\rightarrow [(ms\ '\kappa)](mg\ s))$ which is a tautology.

(2) $[(ms(fol\ '\kappa))](ms\ '\kappa)$ Unfolding ms twice gives: $[\forall s((s\varepsilon(fol\ '\kappa))\rightarrow(mg\ s))]\forall s((s\varepsilon'\kappa)\rightarrow(mg\ s))$
which is: $[\forall s((s\varepsilon(fol\ '\kappa))\rightarrow(mg\ s))]((s\varepsilon'\kappa)\rightarrow(mg\ s))$ Backchaining on the hypothesis and then dropping it gives: $(s\varepsilon'\kappa)\rightarrow(s\varepsilon(fol\ '\kappa))$. Folding $\supseteq$ gives an instance of FOL1. QED.

C3: $(fol\ '\kappa)\supseteq(fol\ '\Gamma) \leftrightarrow ([(ms\ '\kappa)](ms\ '\Gamma))$ proof: Unfolding $\supseteq$ gives: $\forall s((s\varepsilon(fol\ '\Gamma))\rightarrow(s\varepsilon(fol\ '\kappa)))$

By C1 twice this is equivalent to: $\forall s(([(ms\ '\Gamma)](mg\ s))\rightarrow([(ms\ '\kappa)](mg\ s)))$

By the laws of S5 modal logic this is equivalent to: $([(ms\ '\kappa)]\forall s(([(ms\ '\Gamma)](mg\ s))\rightarrow(mg\ s)))$

By C1 this is equivalent to: $[(ms\ '\kappa)]\forall s((s\varepsilon(fol\ '\Gamma))\rightarrow(mg\ s))$. Folding ms then gives: $[(ms\ '\kappa)](ms(fol\ '\Gamma))$

By C2 this is equivalent to: $[(ms\ '\kappa)](ms\ '\Gamma)$. QED.

C4: $((fol\ '\kappa)=(fol\ '\Gamma)) \leftrightarrow ((ms\ '\kappa)\equiv(ms\ '\Gamma))$ proof: This is equivalent to $(((fol\ '\kappa)\supseteq(fol\ '\Gamma))\wedge((fol\ '\Gamma)\supseteq(fol\ '\kappa))) \leftrightarrow ([(ms\ '\kappa)](ms\ '\Gamma))\wedge([(ms\ '\Gamma)](ms\ '\kappa))$ which follows by using C3 twice.

5. "Recursive" Default Logic Represented in Modal Logic

The fixed-point equation for Default Logic may be expressed as a necessary equivalence in an S5 Modal Quantificational Logic using a recursive definition, as follows:

$$\kappa\equiv(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$$

where DR is defined as:

$(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =df\ \forall t(R\ t\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$

$(R\ 0\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =df\ \Gamma$

$(R\ t{+}1\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =df\ (R\ t\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) \wedge\forall_i((([(R\ t\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge\wedge_{j=1,mi}(<\kappa>\beta_{ij}))\rightarrow\chi_i)$

When the context is obvious $\Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$ is omitted and just $(DR\ \kappa)$ and $(R\ t\ \kappa)$ are written. Given below are some properties of DR. The first two theorems state that DR entails Γ and any conclusion χ_i of a default whose entailment condition holds in DL and whose possible conditions are possible with κ.

MD1: $[(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\Gamma$ proof: Unfolding DR gives: $[\forall t(R\ t\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\Gamma$

Letting t be 0 shows that it suffices to prove: $[(R\ 0\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\Gamma$. Unfolding R gives a tautology. QED.

MD2: $([(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\ (([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge\wedge_{j=1,mi}(<\kappa>\beta_{ij}))\rightarrow \chi_i)$

proof: By R0 it suffices to prove: $(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)\rightarrow ((([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$

Unfolding DR gives: $\forall t(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\ \chi_i)\rightarrow((([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$

Letting the quantified t be t+1, it suffices to prove:

$(R\ t{+}1\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\ \chi_i)\rightarrow((([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$. Unfolding R t+1 gives:

$((R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)\wedge(\forall_i((([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\ \chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i))$

$\wedge([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i$

Letting the quantified i be i gives a tautology. QED.

The concept (i.e. ss) of the combined meaning of all the sentences of the FOL object language whose meanings are entailed by a proposition is defined as follows:

$$(ss\ \kappa) =df\ \forall s(([\kappa](mg\ s))\rightarrow(mg\ s))$$

SS1 shows that a proposition entails the combined meaning of the FOL object language sentences that it entails. SS2 shows that if a proposition is necessarily equivalent to the combined meaning of the FOL object language sentences that it entails, then there exists a set of FOL object language sentences whose meaning is necessarily equivalent to that proposition:

SS1: $[\kappa](ss\ \kappa)$ proof: By R0 it suffices to prove: $\kappa\rightarrow(ss\ \kappa)$. Unfolding ss gives: $\kappa\rightarrow\forall s(([\kappa](mg\ s))\rightarrow(mg\ s))$ which is equivalent to: $\forall s(([\kappa](mg\ s))\rightarrow(\kappa\rightarrow(mg\ s)))$ which is an instance of A1. QED.

SS2: $(\kappa\equiv(ss\ \kappa))\rightarrow\exists s(\kappa\equiv(ms\ s))$ proof: Letting s be $\{s: ([\kappa](mg\ s))\}$ gives: $(\kappa\equiv(ss\ \kappa))\rightarrow(\kappa\equiv(ms\{s: ([\kappa](mg\ s))\}))$. Unfolding ms and lambda conversion gives: $(\kappa\equiv(ss\ \kappa))\leftrightarrow(\kappa\equiv\forall s(([\kappa](mg\ s))\rightarrow(mg\ s)))$. Folding ss gives a tautology. QED.

The theorems MD3 and MD4 are analogous to MD1 and MD2 except that DR is replaced by the combined meanings of the sentences entailed by DR.

MD3: $[ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)]\forall i\Gamma_i$ proof: By R0 it suffices to prove: $(ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i))\rightarrow\forall i\Gamma_i$ which is equivalent to: $(ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i))\rightarrow\Gamma_i$. Unfolding ss gives: $(\forall s(([(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)](mg\ s))\rightarrow(mg\ s)))\rightarrow\Gamma_i$ which by the laws M0-M7 is equivalent to: $(\forall s(([(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)](mg\ s))\rightarrow(mg\ s)))\rightarrow(mg\ '\Gamma_i)$. Backchaining on $(mg\ '\Gamma_i)$ with s in the hypothesis being $'\Gamma_i$ in the conclusion shows that it suffices to prove: $([(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)](mg\ '\Gamma_i)$ which by the meaning laws: M0-M7 is equivalent to: $[(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)]\Gamma_i)$ which by S5 Modal Logic is equivalent to: $([(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)]\forall i\Gamma_i)$ which is an instance of theorem MD1. QED.

MD4: $[ss(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\ ((([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$

proof: By R0 it suffices to prove: $(ss(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i))\rightarrow((([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$

Unfolding ss: $(\forall s(([(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)](mg\ s))\rightarrow(mg\ s)))\rightarrow((([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\ \chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$

Instantiating s in the hypothesis to $'\chi_i$ and then dropping the hypothesis gives:

$((([(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)](mg\ '\chi_i))\rightarrow(mg\ '\chi_i))\rightarrow((([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\ \chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$

Using the meaning laws M0-M7 gives:

$((([(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\chi_i)\rightarrow\chi_i)\rightarrow((([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\ \chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$

Backchaining on χ_i shows that it suffices to prove:

$((([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/(mg\ '\chi_i))]\chi_i)$

By the laws of S5 modal logic this is equivalent to:

$[(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\ ((([(R\ t\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$ which is MD2. QED.

Theorems MD5 and MD6 show that R is entailed by the meanings of the sentences entailed by DR:

MD5: $[ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)](R\ 0\ k(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i))$

proof: Unfolding R 0 gives: $(ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i))\rightarrow(\forall i\Gamma_i)$ which holds by MD3. QED.

MD6: $([ss(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)](R\ t\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i))\rightarrow([ss(DR\ \kappa\ \alpha_i{:}\beta_{ij}/\chi_i)](R\ t{+}1\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i))$

proof: Unfolding R t+1 in the conclusion gives:

$([ss(DR\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]((R\ t\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)\wedge\forall_i((([(R\ t\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge\wedge_{j=1,m_i}(<\kappa>\beta_{ij}))\rightarrow\chi_i)))$

Using the hypothesis gives: $[ss(DR\ \kappa\ \alpha_i{:}\beta_{ij}/\chi_i)]\ \forall_i((([(R\ t\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge\wedge_{j=1,m_i}(<\kappa>\beta_{ij}))\rightarrow\chi_i)$

which holds by MD4. QED.

Finally MD7 and MD8 show that talking about the meanings of sets of FOL sentences in the modal representation of Default Logic is equivalent to talking about propositions in general.

MD7: $(ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i))\equiv(DR\ \kappa(\forall i\Gamma_i)\ \alpha_i{:}\beta_{ij}/\chi_i)$

proof: In view of SS1, it suffices to prove: $[ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)](DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$

Unfolding the second occurrence of DR gives: $[ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)]\forall t(R\ t\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$

which is equivalent to: $\forall t([ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)](R\ t\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i))$

By induction on t the proof divides into a base case and an induction step:

(1)Base Case: $([ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)](R\ 0\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i))$ which holds by theorem MD5.

(2)Induction Step: $([ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)](R\ t\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)) \rightarrow ([ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)](R\ t{+}1\ k\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i))$ which holds by theorem MD6. QED.

MD8: $(\kappa\equiv(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i))\rightarrow\exists s(\kappa\equiv(ms\ s))$ proof: From the hypothesis and MD7 $\kappa\equiv(ss(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i))$ is derived. Using the hypothesis to replace $(DR\ \kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$ by κ in this result gives: $\kappa\equiv(ss\ \kappa)$ By SS2 this implies the conclusion. QED.

6. Conclusion: The Relationship between "Recursive" Default Logic and the Modal Logic

The relationship between the "recursive" set theoretic definition of Default Logic [Reiter 1980] and the modal representation of it is proven in two steps. First theorem R1 shows that the meaning of the set r is the proposition R. Theorem DR1 shows that the meaning of the set dr is the proposition DR. DL2 shows that a set of FOL sentences which contains its FOL theorems is a fixed-point of the fixed-point equation of Default Logic with an initial set of axioms and defaults if and only if the meaning (or rather disquotation) of that set of sentences is logically equivalent to DR of the meanings of that initial set of sentences and those defaults.

R1: $(ms(r\ t(fol\ '\kappa)\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\equiv(R\ t(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$

proof: Inducting on the numeric variable t gives a base case and an induction step:

(1) The Base Case: $(ms(r\ 0(fol\ '\kappa)\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\equiv(R\ 0(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$. Starting from $(ms(r\ 0(fol\ '\kappa)\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))$ unfolding r gives: $(ms(fol\{'\Gamma i\}))$. By C2 this is equivalent to: $(ms\{'\Gamma i\})$. By MS2 this is equivalent to: $(\forall i\Gamma_i)$. Folding R then gives: $(R\ t(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$ which proves the base case.

(2) The Induction Step: $((ms(r\ t(fol\ '\kappa)\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\equiv(R\ t(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i))$

$\rightarrow((ms(r\ t{+}1(fol\ '\kappa)'\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\equiv(R\ t{+}1(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i))$

Setting aside the induction hypothesis, we start from: $(ms(r\ t{+}1(fol\ '\kappa)\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))$

Unfolding r gives: $(ms(fol((r\ t\ '\kappa\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i)\cup\{'\chi_i{:}\ ('\alpha_i\varepsilon(r\ t\ '\kappa\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin'\kappa)\})))$

By C2 this is equivalent to: $(ms((r\ t\ '\kappa\{'\Gamma i\}\ '\alpha_i{:}'\beta_{ij}/'\chi_i)\cup\{'\chi_i{:}\ ('\alpha_i\varepsilon(r\ t\ '\kappa\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin'\kappa)\}))$

By MS3 this is equivalent to: $((ms(r\ t\ '\kappa\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge(ms\{'\chi_i{:}\ ('\alpha_i\varepsilon(r\ t\ '\kappa\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin'\kappa)\}))$

By MS2 this is : $(ms(r\ t\ '\kappa\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\forall_i((('\alpha_i\varepsilon(r\ t\ '\kappa\{'\Gamma i\}\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin'\kappa))\rightarrow(mg\ '\chi_i))$

Using C1 twice gives and folding $<\kappa>$ gives:

$(ms(r\ t\ '\kappa\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\forall_i((([ms(r\ t\ '\kappa\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i)](mg\ '\alpha_i))\wedge\wedge_{j=1,mi}(<(ms\ '\kappa)>(mg\ '\beta_{ij})))\rightarrow(mg\ '\chi_i))$

Using the M0-M7 gives: $(ms(r\ t\ '\kappa\{'\Gamma i\}''\alpha_i{:}'\beta_{ij}/'\chi_i))\wedge\forall_i((([ms(r\ t\ '\kappa\{'\Gamma i\}''\alpha_i{:}'\beta_{ij}/'\chi_i)]\alpha_i)\wedge\wedge_{j=1,mi}(<(ms\ '\kappa)>\beta_{ij}))\rightarrow\chi_i)$

Using the induction hypothesis twice gives:

$(R\ t(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)\wedge\forall_i((([(R\ t(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i)\wedge\wedge_{j=1,mi}(<(ms\ '\kappa)>\beta_{ij}))\rightarrow\chi_i)$

Folding R then gives: $((R\ t{+}1(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$ which proves the Induction Step. QED.

DR1: $(ms(dr(fol\ '\kappa)\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\equiv(DR(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$

proof: $(ms(dr(fol\ '\kappa)'\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))$ Unfolding the definition of dr gives: $ms(\cup_{t=1,\omega}(r\ t(fol\ '\kappa)'\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))$

Unfolding $\cup$ gives: $ms\{s{:}\ \exists t(s\varepsilon(r\ t(fol\ '\kappa)\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\}$. Unfolding ms gives: $\forall s((s\varepsilon\{s{:}\ \exists t(s\varepsilon(r\ t(fol\ '\kappa)\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\})\rightarrow(mg\ s))$ which is equivalent to: $\forall s((\exists t(s\varepsilon(r\ t(fol\ '\kappa)\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i)))\rightarrow(mg\ s))$ which is equivalent to: $\forall t\forall s((s\varepsilon(r\ t(fol\ '\kappa)\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))\rightarrow(mg\ s))$. Folding ms gives: $\forall t(ms(r\ t(fol\ '\kappa)\{'\Gamma i\}'\alpha_i{:}'\beta_{ij}/'\chi_i))$

By R1 this is equivalent to: $\forall t(R\ t(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$. Folding DR then gives $(DR(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$ QED.

DR2: $((fol\ '\kappa)=(dr(fol\ '\kappa)\{'\Gamma i\}\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\leftrightarrow((ms\ '\kappa)\equiv(DR(ms\ '\kappa)(ms\ '\Gamma)\alpha_i{:}\beta_{ij}/\chi_i))$

proof: By FOL9 (fol 'κ)=(dr(fol 'κ){'Γi}'α_i:'β_{ij}/'χ_i) is: (fol 'κ)=(fol(dl(fol 'κ){'Γi}'α_i:'β_{ij}/'χ_i)). By C4 this is equivalent to: (ms 'κ)≡(ms(dr(fol 'κ){'Γi}'α_i:'β_{ij}/'χ_i)). By DR1 this is equivalent to: (ms 'κ)≡(DR(ms 'κ)(∀iΓ_i)α_i:β_{ij}/χ_i) QED.

Theorem DR2 shows that the set of theorems: (fol 'κ) of a set 'κ is a fixed-point of a fixed-point equation of Default Logic if and only if the meaning (ms 'κ) of 'κ is a solution to the necessary equivalence. Furthermore, by FOL10 there are no other fixed-points (such as a set not containing all its theorems) and by MD8 there are no other solutions (such as a proposition not representable as a sentence in the FOL object language). Therefore, the Modal representation of Default Logic (i.e. DR), faithfully represents the set theoretic description of the "recursive" definition of Default Logic (i.e. dr). Finally, we note that (∀iΓ_i) and (ms 'κ) may be generalized to be arbitrary propositions Γ and κ giving the more general modal representation: κ≡(DR κ Γ α_i:β_{ij}/χ_i).

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant Numbers: MIP-9526532, EIA-9818341, and EIA-9972843.

Bibliography

[Anatoniou 1997] Antoniou, Grigoris 1997. *NonMonotonic Reasoning*, MIT Press.

[Bressan 1972] Bressan, Aldo 1972. *A General Interpreted Modal Calculus,* Yale University Press.

[Boyer&Moore 1981]. Boyer R. S, and J. Strother Moore, "Metafunctions: proving them correct and using them efficiently as new proof procedures," *The Correctness Problem in Computer Science,* R. S. Boyer and J. Strother Moore, eds., Academic Press, New York, 1981.

[Brown 1986] Brown, Frank M. 1986. "Reasoning in a Hierarchy of Deontic Defaults", *Proceedings of the Canadian Artificial Intelligence Conference CSCI 86*, Montreal, Canada, Morgan-Kaufmann, Los Altos.

[Brown 1987] Brown, Frank M. 1987. "The Modal Logic Z", In *The Frame Problem in AI"; Proc. of the 1987 AAAI Workshop*, Morgan Kaufmann, Los Altos, CA .

[Brown 1989] Brown, Frank M. 1989. "The Modal Quantificational Logic Z Applied to the Frame Problem", advanced paper *First International Workshop on Human & Machine Cognition*, May 1989 Pensacola, Florida. Abbreviated version published in *International Journal of Expert Systems Research and Applications, Special Issue: The Frame Problem. Part A.* eds. Kenneth Ford and Patrick Hayes, vol. 3 number 3, pp169-206 JAI Press 1990. Reprinted in *Reasoning Agents in a Dynamic World: The Frame problem*, editors: Kenneth M. Ford, Patrick J. Hayes, JAI Press 1991.

[Carnap 1946] Carnap, Rudolf 1946. "Modalities and Quantification" *Journal of Symbolic Logic*, vol. 11, number 2.

[Carnap 1956] Carnap, Rudolf 1956. *Meaning and Necessity: A Study in the Semantics of Modal Logic*, The University of Chicago Press.

[Fine 1970] Fine, K. 1970. "Propositional Quantifiers in Modal Logic" *Theoria* 36, p336--346.

[Hughes & Cresswell 1968] Hughes, G. E. and Cresswell, M. J., 1968. *An Introduction to Modal Logic*, Methuen & Co. Ltd., London.

[Leasure 1993] Leasure, David E., 1993. The Modal Logic Z Applied to Lifschitz's Benchmark problems for Formal Nonmonotonic Reasoning, University of Kansas Dissertation, University of Kansas Library.

[Mendelson 1964] Mendelson, E. 1964. *Introduction to Mathematical Logic*, Van Norstrand, Reinhold Co., New York.

[Parks 1976] Parks, Z. 1976. "An Investigation into Quantified Modal Logic", Studia Logica 35, p109-125.

[Quine 1969] Quine, W.V.O., Set Theory and Its Logic, revised edition, Oxford University Press, London, 1969.

[Reiter 1980] Reiter, R. 1980. "A Logic for Default Reasoning" *Artificial Intelligence*, 13.

Author Information

Frank M. Brown – University of Kansas, Lawrence, Kansas, 66045, e-mail: brown@ku.edu.

ON THE RELATIONSHIPS AMONG QUANTIFIED AUTOEPISTEMIC LOGIC, ITS KERNEL, AND QUANTIFIED REFLECTIVE LOGIC

Frank Brown

Abstract: A Quantified Autoepistemic Logic is axiomatized in a monotonic Modal Quantificational Logic whose modal laws are slightly stronger than S5. This Quantified Autoepistemic Logic obeys all the laws of First Order Logic and its L predicate obeys the laws of S5 Modal Logic in every fixed-point. It is proven that this Logic has a kernel not containing L such that L holds for a sentence if and only if that sentence is in the kernel. This result is important because it shows that L is superfluous thereby allowing the ori ginal equivalence to be simplified by eliminating L from it. It is also shown that the Kernel of Quantified Autoepistemic Logic is a generalization of Quantified Reflective Logic, which coincides with it in the propositional case.

Keywords: Quantified Autoepistemic Logic, Quantified Reflective Logic, Modal Logic, Nonmonotonic Logic.

1. Introduction

Quantified Autoepistemic Logic (i.e. QAEL) is a generalization of Autoepistemic Logic [Moore, Konolige87, Konolige87b], where both universally and existentially quantified variables are allowed to cross the scope of the L predicate. In a recent paper [Brown 2003b, 2003d] showed that Autoepistemic Logic could be represented in an extension of S5 Modal Logic. This modal representation may be generalized to provide a Quantified Autoepistemic Logic with the following necessary equivalence:

$$\kappa\equiv(\text{QAEL }\kappa\ \Gamma)$$

where QAEL is defined as follows:

$$(\text{QAEL }\kappa\ \Gamma) =\text{df } \Gamma\wedge\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow([\kappa]\chi_i))$$

$$(L\ '\chi_i) =\text{df } (L'\chi_i\ a_i),$$

where χ_i is the ith sentence with or without free variables of a First Order Logic (i.e. FOL) and a_i is an association list associating the free variables in $'\chi_i$ to values specified by the sequence of variables ξ_i. The $\forall i$ quantifier ranges across the natural numbers. This Quantified Autoepistemic Logic is important because unlike some other attempts [Konolidge1989] to generalize Autoepistemic Logic, its quantifiers obey both the Barcan Formulae, the converse of the Barcan formula, and also all the laws of S5 Modal Logic and First Order Logic (i.e. FOL). Interpreted doxastically this necessary equivalence states that:

that which is believed is equivalent to: Γ and for all i and for all ξ_i $(L\ '\chi_i)$ if and only if χ_i is believed.

The purpose of this paper is to show that the L predicate is not essential to solving for κ and can be eliminated thereby allowing the above necessary equivalence to be replaced by a simpler necessary equivalence which when interpreted as a doxastic logic states:

that which is believed is equivalent to: Γ (with each *L'* replaced by $[\kappa]$) .

thereby eliminating every occurrence of the L predicate, all the (quoted) names of sentences, and the bi-implication containing *L*.

The remainder of this paper proves that the L predicate can be eliminated. Section 2 describes the First Order Logic (i.e.FOL) used herein. Section 3 describes the Modal Logic used herein. QAEL is defined in more detail in section 4. The *L* eliminated form of Quantified Autoepistemic Logic herein called the Quantified Autoepistemic Kernel (i.e. QAEK) is defined in section 5 and is explicated with theorems LEXT1 and LEXT2. In section 6, QAEK is shown to be related to QAEL by theorems QAEK1, QAEK2, QAEK3. The relationship between QAEK and Quantified Reflective Logic (i.e. QRL) [Brown 2003a] is given in section 7. Finally, in section 8, some consequences of all these results are discussed. Figure 1 outlines the relationship of all these theorems in producing the final theorems LEXT2, QAEK3, and AR2.

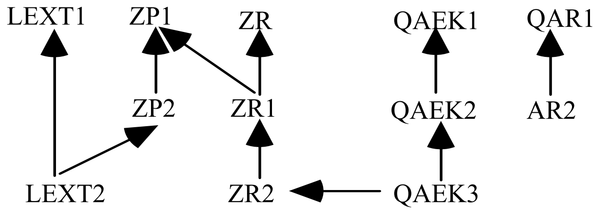


Figure 1: Dependencies among the Theorems

2. First Order Logic

We use a First Order Logic (i.e. FOL) defined as the six tuple: ($\rightarrow$, #f, $\forall$, *vars*, *predicates*, *functions*) where $\rightarrow$, #f, and $\forall$ are logical symbols, *vars* is a set of variable symbols, *predicates* is a set of predicate symbols each of which has an implicit arity specifying the number of associated terms, and *functions* is a set of function symbols each of which has an implicit arity specifying the number of associated terms. Lower case Roman letters possibly indexed with digits are used as variables. Greek letters possibly indexed with digits or lower case roman letters are used as syntactic metavariables. γ, $\gamma_1...\gamma_n$, range over the variables, ξ, $\xi_1...\xi_n$ range over sequences of variables of an appropriate arity, $\pi,\pi_1...\pi_n$ range over the predicate symbols, ϕ, $\phi_1...\phi_n$ range over function symbols, δ, $\delta_1...\delta_n$, σ range over terms, and α, $\alpha_1...\alpha_n$, β, $\beta_1...\beta_n$,χ, $\chi_1...\chi_n$,Γ,Γ_1,...Γ_n,κ,κ_1,...κ_n, φ, range over sentences (including sentences with free variables). The terms are of the forms γ and (ϕ $\delta 1...\delta n$), and the sentences are of the forms ($\alpha\rightarrow\beta$), #f, ($\forall\gamma$ α), and (π $\delta_1...\delta_n$). A nullary predicate π or function ϕ is written without parentheses. $\varphi\{\pi/\lambda\xi\alpha\}$ represents the replacement of all occurrences of π in φ by $\lambda\xi\alpha$ followed by lambda conversion. The primitive symbols are shown in Figure 2 with their interpretations. The particular FOL used herein includes the binary predicate symbol L and a denumerably infinite number of 0-ary function symbols representing the names (i.e. 'α) of the sentences (i.e. α) of this FOL.

Symbol	Meaning
$\alpha\rightarrow\beta$	if α then β.
#f	falsity
$\forall\gamma\ \alpha$	for all γ, α.

Figure 2: Primitive Symbols of First Order Logic

The defined symbols are listed in Figure 3 with their definitions and intuitive interpretations.

Symbol	Definition	Meaning		Symbol	Definition	Meaning
$\neg\alpha$	$\alpha\rightarrow$ #f	not α		$\alpha\wedge\beta$	$\neg(\alpha\rightarrow\neg\beta)$	α and β
#t	$\neg$ #f	truth		$\alpha\leftrightarrow\beta$	$(\alpha\rightarrow\beta)\wedge(\beta\rightarrow\alpha)$	α if and only if β
$\alpha\vee\beta$	$(\neg\alpha)\rightarrow\beta$	α or β		$\exists\gamma\ \alpha$	$\neg\forall\gamma\ \neg\alpha$	for some γ , α

Figure 3: Defined Symbols of First Order Logic

3. Modal Logic

We extend First Order Logic with a necessity symbol as given in Figure 4 below:

Symbol	Meaning
[]α	α is logically necessary

Figure 4: Primitive Symbols of Modal Logic

and with the laws of an S5 Modal Logic [Hughes & Cresswell 1968] as given in Figure 5 below:

R0: from α infer ([] α)	A2: ([]($\alpha\rightarrow\beta$)) $\rightarrow$ (([]α)$\rightarrow$ ([]β))
A1: ([]α) $\rightarrow\alpha$	A3: ([]α) $\vee$ ([]$\neg$[]α)

Figure 5: The Laws of S5 Modal Logic

These S5 modal laws and the laws of FOL constitute an S5 Modal Quantificational Logic similar to [Carnap 1946; Carnap 1956], and a FOL version [Parks 1976] of [Bressan 1972] in which the Barcan formula: $(\forall\gamma([]\alpha))\rightarrow([]\forall\gamma\alpha)$ and its converse hold. The defined Modal symbols used herein are listed in Figure 6 with their definitions and intuitive interpretations.

Symbol	Definition	Meaning		Symbol	Definition	Meaning
$<>\alpha$	$\neg[]\neg\alpha$	α is logically possible		$\alpha\equiv\beta$	$[](\alpha\leftrightarrow\beta)$	α is synonymous to β
$<\beta>\alpha$	$<>(\beta\wedge\alpha)$	α and β is logically possible		$\delta=\sigma$	$(\pi\delta)\equiv(\pi\sigma)$	δ is logically equal to σ
$[\beta]\alpha$	$[](\beta\rightarrow\alpha)$	β entails α		$\delta\neq\sigma$	$\neg(\delta=\sigma)$	δ is not logically equal to σ.

Figure 3: Defined Symbols of Modal Logic

Next, we extend the FOL + S5 Modal Quantificational Logic with the A4 axiom scheme given in Figure 7.

A4: $<>\Gamma\{\pi/\lambda\xi\alpha\}\rightarrow<>\Gamma$

where $\Gamma\{\pi/\lambda\xi\alpha\}$ is the simultaneous replacement in Γ of all unmodalized occurrences of π by α.

Figure 7: The Possibility Axiom Scheme

Intuitively, A4 specifies that a sentence Γ is logically possible whenever the result obtained by "interpreting" all the unmodalized occurrences of a predicate within it, is logically possible. If A4 is successively applied to all the unmodalized predicates then it follows that a sentence Γ is logically possible if the result of interpreting all the unmodalized predicates is logically possible. The possibility axiom A4 extends the trivial possibility axiom (i.e. some proposition is neither #t nor #f) given in [Lewis 1936] and [Bressan 1972], the S5c possibility axiom schema (i.e. every conjunction of distinct negated or unnegated propositional constants is logically possible) given in [Hendry & Pokriefka 1985], and is implied by the possibility axiom schema used in the Z Modal Quantificational Logic described in [Brown 1987; Brown 1989]. The following metatheorems are derivable:

ZP1: The Possibility of a Separable Predicates: If (1) Γ, α, and β are sentences of FOL extended whereby any modalized sentence may occur in the place of predicates and (2) π does not occur unmodalized in any of Γ, α, and β then: $(<>(\Gamma\wedge(\forall\xi(\alpha\rightarrow(\pi\ \xi)))\wedge(\forall\xi(\beta\rightarrow\neg(\pi\ \xi)))))\leftrightarrow(<>(\Gamma\wedge(\neg\exists\xi(\alpha\wedge\beta))))$

ZP2: The Possibility of a Defined Predicate: If (1) Γ and α are sentences of FOL extended whereby any modalized sentence may occur in the place of predicates and (2) π does not occur unmodalized in any of Γ and α then: $(<>(\Gamma\wedge(\forall\xi((\pi\ \xi)\leftrightarrow\alpha))))\leftrightarrow(<>\Gamma)$ proof: Let β be $\neg\alpha$ in ZP1 and simplify. QED.

ZR: The Reduction Lemma: If (1) κ occurs in Γ and Ψ only in the context: $<\kappa>\varphi$ for some φ (or in the context $[\kappa]\mu$ which is essentially of the same modal form: $\neg<\kappa>\neg\mu$ and (2) for all such φ: $\forall p(((<(\Gamma\wedge\Psi)>\varphi)\leftrightarrow(<(\Gamma>\varphi))\{\kappa/p\})$ then : $(\kappa\equiv(\Gamma\wedge\Psi))\leftrightarrow\exists p((\kappa\equiv(p\wedge(\Psi\{\kappa/p\})))\wedge(p\equiv(\Gamma\{\kappa/p\})))$

ZR1: Reducing a Reflection with a Separable Predicate: If (1) κ occurs in Γ, α, and β only in the context: $<\kappa>\varphi$ for some φ (or in the context $[\kappa]\mu$ which is essentially of the same modal form: $(\neg<\kappa>\neg\mu)$, (2) Γ, α, β, and φ are sentences of FOL extended whereby any modalized sentence may occur in the place of predicates, (3) π does not occur unmodalized in any of Γ, α, β, and φ then: $(\kappa\equiv(\Gamma\wedge(\forall\xi(\alpha\rightarrow(\pi\ \xi)))\wedge(\forall\xi(\beta\rightarrow\neg(\pi\ \xi)))))$

$\leftrightarrow\exists p((\kappa\equiv(p\wedge(((\forall\xi(\alpha\rightarrow(\pi\ \xi)))\wedge(\forall\xi(\beta\rightarrow\neg(\pi\ \xi))))\{\kappa/p\})))\wedge(p\equiv((\Gamma\wedge(\neg\exists\xi(\alpha\wedge\beta)))\{\kappa/p\})))$

ZR2: Reducing a Reflection with a Defined Predicate: If (1) κ occurs in Γ, and α only in the context: $<\kappa>\varphi$ for some φ (or in the context $[\kappa]\mu$ which is essentially of the same modal form: $(\neg<\kappa>\neg\mu)$, (2) Γ, α, , and φ are sentences of FOL extended whereby any modalized sentence may occur in the place of predicates, (3) π does not occur unmodalized in any of Γ, α, and φ then: $\kappa\equiv(\Gamma\wedge\forall\xi((\pi\ \xi)\leftrightarrow\alpha))\leftrightarrow\exists p\ \kappa\equiv(p\wedge\forall\xi((\pi\ \xi)\leftrightarrow\alpha\{\kappa/p\}))\wedge p\equiv\Gamma\{\kappa/p\}$. proof: Let β be $\neg\alpha$ in ZR1 and simplify. QED.

4. Quantified Autoepistemic Logic

Quantified Autoepistemic Logic (i.e. QAEL) is defined in Modal Logic by a necessary equivalence of the form:

$$\kappa\equiv(\text{QAEL}\ \kappa\ \Gamma)$$

where QAEL is defined as follows: (QAEL κ Γ) =df $\Gamma\wedge\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow([\kappa]\chi_i))$ where $(L\ '\chi_i)$ =df $(L\ '\chi_i\ a_i)$, χ_i is the ith sentence with or without free variables of FOL and a_i is an association list binding the free variables in $'\chi_i$ to values specified by the sequence of metalanguage variables ξ_i. The $\forall i$ quantifier ranges across the natural numbers. Any FOL proposition κ which makes this necessary equivalence true is a solution. QAEL addresses the problem of how quantified variables whose scopes cross the L predicate may be represented. Furthermore these quantifiers obey not only the Barcan formula but unlike the generalization of Autoepistemic Logic given in [Konolige 1989] its converse and therefore does not suffer the anomalies therein discussed. For example we could then state in QAEL that everything is a bird and that all things believed to be birds for which flying is believable do in fact fly as follows: $\kappa\equiv$(QAEL $\kappa((\forall x(\text{Bird } x))\wedge\forall x(((L\ '(\text{Bird } x))\wedge(\neg L\ '(\neg(\text{Fly } x))))\rightarrow(\text{Fly } x))))$

5. Quantified Autoepistemic Kernel

The Quantified Autoepistemic Kernel [Brown 1989] is defined in Modal Quantificational Logic by the necessary equivalence:

$$\varphi\equiv(\text{QAEK } \varphi\ \Gamma)$$

where QAEK is defined as: (QAEK φ Γ) =df $\Gamma\{L\ '/[\varphi]\}$

The *L* predicate does not occur unmodalized in QAEK. However, the kernel may be used to define an extension containing facts involving *L* as follows:

(L-EXT φ) =df $(\varphi\wedge\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow(([\varphi]\chi_i)\{L\ '/[\varphi]\})))$

The kernel φ possesses two important properties with respect to L-extensions, namely that the L-extension of φ entails φ, and φ entails every FOL sentence not containing an occurrence of *L* which the L-extension entails

LEXT1: $[(\text{L-ext } \varphi)]\varphi$ proof: Unfolding L-ext gives a tautology. QED.

LEXT2: If *L* is not in s and if φ contains no unmodalized occurrence of L, then: $\forall s([(\text{L-ext } \varphi)]s)\leftrightarrow([\varphi]s)$

proof: Pushing negation through gives the equivalent sentence: $\forall r((\langle(\text{L-ext } \varphi)\rangle r)\leftrightarrow(\langle\varphi\rangle r))$

Unfolding L-ext gives: $\forall r((\langle(\varphi\wedge\forall i\forall\xi_i((L\ \chi_i)\leftrightarrow(([\varphi]\chi_i)\{L\ '/[\varphi]\}))\rangle r)\leftrightarrow(\langle\varphi\rangle r))$ or rather:

$\forall r((\langle\varphi\rangle(\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow(([\varphi]\chi_i)\{L\ '/[\varphi]\}))\wedge r)\leftrightarrow(\langle\varphi\rangle r))$ which is an instance of theorem ZP2. QED

LEXT1 and LEXT2 show that the kernel determines all the non-kernel sentences in the L-extension. Representing problems in the Quantified Autoepistemic Kernel simplifies their solution since the pre-processing step of eliminating the *L* predicate from Γ is eliminated.

6. The Relationship between Quantified Autoepistemic Logic and its Kernel

We now show how all occurrences of L including those within quotes as parts of structural descriptive names of sentences of Autoepistemic Logic may be eliminated from Γ: For example, if Γ consisted of the single default: $(\neg(L\ '(L\ '(\neg\pi))))\rightarrow\pi$ then the necessary equivalence is: $\kappa\equiv(\text{QAEL } \kappa((\neg(L\ '(L\ '(\neg\pi))))\rightarrow\pi))$

Unfolding AEL gives: $\kappa\equiv(((\neg(L'(L\ '(\neg\pi))))\rightarrow\pi)\wedge\forall i\forall\xi_i((L'\chi_i)\leftrightarrow([\kappa]\chi_i)))$. Since the quantified statement is connected to: $(\neg(L\ '(L\ '(\neg\pi))))\rightarrow\pi$ by a conjunction it may be assumed when simplifying that expression. Instantiating i so that χ_i is $(L\ '(\neg\pi))$ and using that instance gives the equivalent expression: $\kappa\equiv(((\neg([\kappa](L\ '(\neg\pi))))\rightarrow\pi)\wedge\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow([\kappa]\chi_i)))$. We would like to eliminate the remaining *L* in the first formulae but it is inside the scope of an entailment and therefore the (non-necessary) equivalence: $\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow([\kappa]\chi_i)))$ does not justify such a reduction merely by virtue of the two formulas being connected by conjunction. However, the entire formula allows the derivation of: $[\kappa](\forall i\forall\xi_i((L\ '\chi_i))\leftrightarrow([\kappa]\chi_i))$ which shows that $\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow([\kappa]\chi_i))$ may be assumed in any scope entailed by κ. Thus we can still reduce occurrences of *L* even embedded within an entailment. Thus, the above equation is equivalent to: $\kappa\equiv(((\neg([\kappa]([\kappa](\neg\pi))))\rightarrow\pi)\wedge\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow([\kappa]\chi_i))$ in which no occurrence of *L* nor quotation appears in the first formulae in the conjunction. Notating the above described process (i.e. sequence of deductions) as $(\Gamma\{L\ '/[\kappa]\})$ or rather the substitution of *L* ' by $[\kappa]$ gives the theorem:

QAEK1: $(\kappa\equiv(\text{QAEL } \kappa\ \Gamma)) \leftrightarrow (\kappa\equiv(\text{QAEL } \kappa\ (\Gamma\{L\ '/[\kappa]\})))$

The process which eliminated L from Γ can also be used to eliminate L from χ_i in the formulae: $\kappa\equiv((\Gamma\{L\ '/[\kappa]\})\wedge\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow([\kappa]\chi_i)))$. Since χ_i occurs within the modal scope of a κ entailment we are justified in replacing an instance of it by another formulae by assuming $\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow([\kappa]\chi_i)))$ for any other instance of χ_i since $[\kappa]\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow([\kappa]\chi_i)))$ follows from the overall equation. To replace each χ_i by a sentence which no longer contains L, we specify an ordering of all the sentences based on the maximum depth of Ls as they occur through the structural descriptive names of the sentences. A sentence χ_i with no L would be of depth 0, a sentence with L would be at depth 1, a sentence with 'L would be of depth 2, a sentence with ''L would be of depth 3 and so forth. The proof is by induction. The base case is always true since L is not in those sentences. The induction step proceeds by using $\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow([\kappa]\chi_i))))$ on sentences whose L depth is less than n to prove that relation for sentences whose depth is n. Notating the result of the above described process gives:

QAEK2: $(\kappa\equiv(\text{QAEL } \kappa\ \Gamma))\leftrightarrow (\kappa\equiv((\Gamma\{L\ '/[\kappa]\})\wedge\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow(([\kappa]\chi_i)\{L\ '/[\kappa]\}))))$

QAEK2 shows how all but one occurrence of L may be eliminated from the equivalence. Essentially κ is logically equivalent to a modal formula $\Gamma\{L\ '/[\kappa]\})$ not containing L conjoined to what is essentially a "definition" of L in terms of another modal formulae not containing L. This suggests that L is superfluous notation and that the essence of κ lies only in the first formulae. This intuition is easily proven:

QAEK3: $(\kappa\equiv(\text{QAEL } \kappa\ \Gamma))\leftrightarrow \exists p((\kappa\equiv(\text{L-EXT } p))\wedge(p\equiv(\text{QAEK } p\ \Gamma)))$

proof: By QAEK2 $(\kappa\equiv(\text{QAEL } \kappa\ \Gamma))$ is equivalent to: $(\kappa\equiv((\Gamma\{L\ '/[\kappa]\})\wedge\forall i\forall\xi_i((L\ '\chi_i)\leftrightarrow(([\kappa]\chi_i)\{L\ '/[\kappa]\}))$

Instantiating ZR2 with: $\Gamma:=\Gamma\{L\ '/[\kappa]\}$, $\xi:=i\xi_i$, $\pi:=L$, $\alpha:=([\kappa]\chi_i\{L\ '/[\kappa]\})))$ shows that the above expression is equivalent to: $\exists p((\kappa\equiv(p\wedge\forall i\forall\xi_i((L\ '\ \chi_i)\leftrightarrow(([p]\chi_i)\{L\ \ '/[p]\})))\ \wedge(p\equiv(\Gamma\{L\ '/[p]\})))$. Folding L-EXT and QAEK gives $\exists p((\kappa\equiv(\text{L-EXT } p))\wedge(p\equiv(\text{QAEK } p\ \Gamma)))$ QED.

QAEK3 divides the Autoepistemic equation into two distinct equivalences, one axiomatizing the kernel p and the other defining the stronger proposition κ which is the L-extension of p containing additional facts about the L predicate. LEXT1 and LEXT2 show that the L-extension κ is a conservative extension of the kernel and therefore it is not essential. For this reason it suffices to deal with just the necessary equivalence for the Quantified Autoepistemic Kernel in studying Quantified Autoepistemic Logic: $\varphi\equiv(\text{QAEK } \varphi\ \Gamma)$.

7. The Relationship between Quantified Autoepistemic Kernel and Quantified Reflective Logic

The modal representation of Reflective Logic [Brown 1989, 2003a, 2003c] may be generalized to a Quantified Reflective Logic as:

$$\kappa\equiv(\text{QRL } \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$$

where QRL is defined in Modal Logic as follows:

$(\text{QRL } \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =_{df}\ \Gamma\wedge\forall i\forall\xi_i((([\kappa]\alpha_i)\wedge\wedge_{j=1,m_i}\langle\kappa\rangle\beta_{ij})\rightarrow\chi_i)$

where Γ, α_i, β_{ij}, and χ_i are sentences of FOL which may contain free variables. The variables in ξ_i may occur in any of α_i, β_{ij}, and χ_i. When the context is obvious $\Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$ is omitted and instead just (QRL κ) is written. $\wedge_{j=1,m_i}$ stands for the conjunction of the formula which follows it as j ranges from 1 to m_i. If m_i=0 then it specifies #t. If i ranges over a finite number of defaults then $\forall_i$ may be replaced in this definition by a conjunction: $\wedge_i$. Interpreted as a doxastic logic, the necessary equivalence states:

that which is believed is logically equivalent to:

Γ and for each i, if α_i is believed and for each j, β_{ij} is believable then χ_i

Quantified Reflective Logic is an instance of the Quantified Autoepistemic Kernel. Specifically:

QAR1: $(\text{QRL } \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)\equiv(\text{QAEK } \kappa\ \Gamma\wedge\forall i\forall\xi_i(((L\ '\alpha_i)\wedge\wedge_{j=1,m_i}(\neg(L\ '(\neg\beta_{ij}))))\rightarrow\chi_i))$

proof: Unfolding QRL and QAEK gives identical formulas. QED.

We call the instance of the Quantified Autoepistemic Kernel in which no quantified variables in Γ cross a modal scope simply the Autoepistemic Kernel. (i.e. AEK). Likewise, we call the instance of Quantified Reflective Logic with no variables in the any sequence ξ_i simply Reflective Logic (i.e. RL).

$(\text{AEK}\ \varphi\ \Gamma)$ =df $\Gamma\{L/[\varphi]\}$

$(\text{RL}\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$ =df $\Gamma\wedge\forall_i((([\kappa]\alpha_i)\wedge\wedge_{j=1,mi}{<}\kappa{>}\beta_{ij})\rightarrow\chi_i)$

where Γ, α_i, β_{ij}, and χ_i are closed sentences of FOL. By closed it is meant that no sentence may contain a free variable.

By QAR1 Reflective Logic is clearly an instance of the Autoepistemic Kernel. However, in addition, it turns out that the Autoepistemic Kernel is also an instance of Reflective Logic:

AR2. The Autoepistemic Kernel is an instance of Reflective Logic. Specifically, for every FOL formulae Γ_i there exist FOL formulas: α_i, β_{ij}, and χ_i. such that: $(\text{AEK}\ \kappa\ (\forall_i\Gamma_i))\equiv(\text{RL}\ \kappa\ \#t\ \alpha_i{:}\beta_{ij}/\chi_i)$

proof: By QAR1 it suffices to prove that each $\Gamma_i\{L'/[\kappa]\}$, which we herebelow call Ψ is representable as

$\wedge_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,mi}({<}\kappa{>}\beta_{ij})))\rightarrow\chi_i)$

We choose a k-entailment: $([\kappa]\varphi)$ in Ψ of lowest scope that has not already been chosen. We use the laws of classical logic to place φ into conjunctive normal form (treating any embedded κ-entailment as another predicate). The following five theorem schemata of Z are then used to reduce the scope of $[\kappa]$[1] .

KU1: $([\kappa](\alpha_1\wedge...\wedge\alpha_n))\equiv(([\kappa]\alpha_1)\wedge...\wedge([\kappa]\alpha_n))$

KU2: $([\kappa](\alpha_1\vee...\vee\alpha_m\vee([\kappa]\varphi)\vee\beta_1\vee...\vee\beta_n))\equiv(([\kappa]\varphi)\vee([\kappa](\alpha_1\vee...\vee\alpha_m\vee\beta_1\vee...\vee\beta_n)))$

KU3: $([\kappa](\alpha_1\vee...\vee\alpha_m\vee(\neg([\kappa]\varphi))\vee\beta_1\vee...\vee\beta_n))\equiv((\neg[\kappa]\varphi)\vee([\kappa](\alpha_1\vee...\vee\alpha_m\vee\beta_1\vee...\vee\beta_n)))$

KU4: $([\kappa]([\kappa]\varphi))\equiv([\kappa]\varphi)$

KU5: $([\kappa](\neg[\kappa]\varphi))\equiv((\neg[\kappa]\varphi)\vee([\kappa]\#f))$

If the result begins with a conjunction, KU1 is applied. If the result begins with a disjunction with an embedded κ entailment or negation of a κ entailment then respectively KU2 or KU3 is applied. If the result is itself a k-entailment of the negation of a k-entailment then respectively KU4 or KU5 is applied. The over all process is repeated until no further KU rule is applicable. When the process finishes since none of the above rules is applicable if the overall formula is put into conjunctive normal form then every resulting disjunction must be of the following form when negations of entailments are ordered before entailments which are ordered before other expressions: $((\vee_{j=1,a}\neg([\kappa]\alpha_j))\vee(\vee_{j=1,b}([\kappa]\beta_j))\vee(\vee_{j=1,c}\chi_j))$ Pulling the first negation out and noting that $(\wedge_{j=1,a}([\kappa]\alpha_j))$ is equivalent to $([\kappa](\wedge_{j=1,a}\alpha_j))$ gives:

$((\neg([\kappa](\wedge_{j=1,a}\alpha_j)))\vee(\vee_{j=1,b}([\kappa]\beta_j))\vee(\vee_{j=1,c}\chi_j))$ or rather: $((\neg([\kappa](\wedge_{j=1,a}\alpha_j)))\vee(\vee_{j=1,b}([\kappa]\beta_j))\vee(\vee_{j=1,c}\chi_j))$

Letting α be $(\wedge_{j=1,a}\alpha_j)$ and χ be $(\vee_{j=1,c}\chi_j)$ gives: $((\neg([\kappa]\alpha))\vee(\vee_{j=1,b}([\kappa]\beta_j))\vee\chi)$

where α is #t if their are no α_j formulas (since that is the identity of conjunction) and where χ is #f if there are no χ_j formulas (since that is the identity of disjunction). Rewriting the above as an implication gives:

$(([\kappa]\alpha)\wedge(\wedge_{j=1,mi}({<}\kappa{>}\beta_j)))\rightarrow\chi_i$ where the resulting β_j are the negations of the previous ones. This formula is called a default. The conjunction of all the defaults is then written as:

$\wedge_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,mi}({<}\kappa{>}\beta_{ij})))\rightarrow\chi_i)$ where the defaults are not required to have any β subformulas. QED.

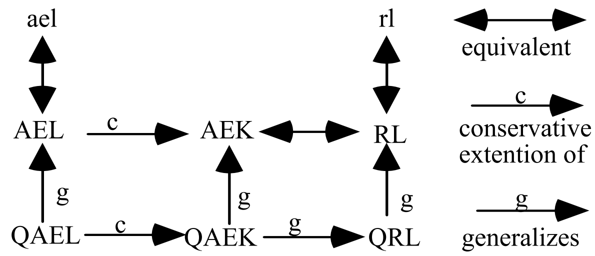
$(\text{RL}\ \kappa\ \#t\ \alpha_i{:}\beta_{ij}/\chi_i)$ is often written as: $(\text{RL}\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$ where Γ is all those defaults having no α (or where α is #t) nor β subformulas (and hence no modals) and i ranges over just the "real" defaults containing modals.

---

[1]When $([K]\psi)$ is viewed with K fixed as a unary symbol, it has the properties of a KU45 modal logic [Park] .

8. Conclusion

The nonmonotonic systems discussed herein are related as described in Figure 8.



ael = Autoepistemic Logic [Moore 1985]	rl = Reflective Logic [Brown 2003a, Brown 2003c]
AEL = Autoepistemic Logic in Modal Logic	QAEL = Quantified Autoepistemic Logic in Modal Logic
AEK = Autoepistemic Kernel in Modal Logic	QAEK = Quantified Autoepistemic Kernel in Modal Logic
RL = Reflective Logic in Modal Logic	QRL = Quantified Reflective Logic in Modal Logic

Figure 8: Relationships among NonMonotonic Systems

The original set theoretic description of Autoepistemic Logic (i.e. ael) is equivalent [Brown 2003b, 2003d] to the modal description AEL and the set theoretic description of Reflective Logic (i.e. rl) is equivalent [Brown 2003a, 2003c] to the modal description RL. Equivalence means that the meaning of the fixed-points of the set theoretic descriptions are identical to the solutions of the necessary equivalences of the modal systems whenever their inputs bear a similar relation. Since the modal systems (i.e. FOL+S5+A4) are much simpler than the set theoretic descriptions (i.e. FOL + Set Theory + FOL Syntax, + FOL Proof Theory) they provide a reduction in both conceptual and computational complexity. For this reason we focus on the modal systems: AEL and RL.

QAEL and QAEK, are respectively generalizations of AEL and AEK in which quantifiers are allowed to be inserted anywhere in the formulas and where such quantified variables may cross modal scopes. Since AEL and QAEL are proven by QAEK3 and LEXT2 to be conservative extensions (involving the superfluous *L* predicate) of AEK and QAEK respectively, these systems AEK and QAEK are said to be the kernels of AEL and QAEL respectively. Because the kernel systems eliminate all occurrences of *L*' and the biconditional relating *L*' to [k] they are more useful systems for both understanding and automatic theorem proving. For this reason we now focus on just the kernel systems: AEK and QAEK.

AEK is proven to be equivalent to RL by AR2. QRL is a generalization of RL where only universal quantifiers may be inserted and only inserted at the beginning of a default. By QAR1, QAEK is a generalization of QRL, But, in general, QRL is weaker than QAEK since for example it does not allow for existential quantifiers just before a default. Because QAEK and QRL differ while AEK and RL are equivalent, it follows that both QAEK and QRL can be said to be different quantificational generalizations of the Autoepistemic Kernel. Both are interesting systems with QAEK providing greater generality and QRL having deep relationships to nomonotonic logics with quantified default inference rules [Brown 2003e]. [Brown 2003f] describes an Automatic Deduction system for the propositional case of Autoepistemic Kernels (i.e. AEK) which reduces to the propositional case of Reflective Logic (i.e. RL)). Deduction Methods for the QAEL and QRL are discussed in [Brown 1987; Leasure 1993; Leasure & Brown 1995].

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant Numbers: MIP-9526532, EIA-9818341, and EIA-9972843.

Bibliography

[Bressan 1972] Bressan, Aldo 1972. *A General Interpreted Modal Calculus,* Yale University Press.

[Brown 1987] Brown, Frank M. 1987. "The Modal Logic Z", In *The Frame Problem in AI*"*; Proc. of the 1987 AAAI Workshop*, Morgan Kaufmann, Los Altos, CA .

[Brown 1989] Brown, Frank M., 1989. "The Modal Quantificational Logic Z Applied to the Frame Problem", advanced paper *First International Workshop on Human & Machine Cognition*, May 1989 Pensacola, Florida. Abbreviated version published in *International Journal of Expert Systems Research and Applications, Special Issue: The Frame Problem. Part A.* eds. Kenneth Ford and Patrick Hayes, vol. 3 number 3, pp169-206 JAI Press 1990. Reprinted in *Reasoning Agents in a Dynamic World: The Frame problem*, editors: Kenneth M. Ford, Patrick J. Hayes, JAI Press 1991.

[Brown 2003a] Brown, Frank M., "On the Relationship between Reflective Logic and Modal Logic", *Proceedings of the X-th International Conference Knowledge Dialogue Solution - Varna 2003*, pages 356 – 364, ISBN 954-16-0025-5, FOI-Commerce, Sofia, 2003.

[Brown 2003b] Brown, Frank M., "On the Relationship between Autoepistemic Logic and Modal Logic", *Proceedings of the X-th International Conference Knowledge Dialogue Solution - Varna 2003*, pages 382-390, ISBN 954-16-0025-5, FOI-Commerce, Sofia 2003.

[Brown 2003c] Brown, Frank M. "Representing Reflective Logic in Modal Logic", *Information Theories and Applications*, Vol. 10, no. 4, pages: 431-438, December 2003. (Revised version of "On the Relationship between Reflective Logic and Modal Logic")

[Brown 2003d] Brown, Frank M., "Representing Autoepistemic Logic in Modal Logic", *Information Theories and Applications*, Vol. 10, no. 4, pages 455-462, December 2003. Revised version of "On the Relationship between Autoepistemic Logic and Modal Logic"

[Brown 2003e., "On the Relationship between Quantified Reflective Logic and Quantified Default Logic" *Information Theories and Applications*, Vol. 10, no. 4, pages 447-454, December 2003. Revised version of KDS paper with same name.

[Brown 2003f] Brown, Frank M., "Decision Procedures for the Propositional Cases of 2nd Order Logic and Z Modal Logic Representations of a 1st Order L-Predicate Nonmonotonic Logic", *Automated Reasoning with Analytic Tableaux and Related Method: Tableaux 2003*, September 2003, Rome, *Lecture Notes in Artificial Intelligence 2796*, ISBN 3-540-40787-1, Springer-Verlag, Berlin, 2003.

[Carnap 1946] Carnap, Rudolf 1946. "Modalities and Quantification" *Journal of Symbolic Logic*, vol. 11, number 2, 1946.

[Carnap 1956] Carnap, Rudolf 1956. *Meaning and Necessity: A Study in the Semantics of Modal Logic*, The University of Chicago Press.

[Fine 1970] Fine, K. 1970. "Propositional Quantifiers in Modal Logic" *Theoria* 36, p336--346.

[Hendry & Pokriefka 1985] Hendry, Herbert E. and Pokriefka, M. L. 1985. "Carnapian Extensions of S5", *Journal of Phil. Logic 14*.

[Hughes & Cresswell 1968] Hughes, G. E. and Cresswell, M. J., 1968. *An Introduction to Modal Logic*, Methuen & Co. Ltd., London

[Konolige 1987] Konolige, Kurt 1987. "On the Relation Between Default Theories and Autoepistemic Logic", *IJCAI87 Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, 1987.

[Konolige 1987b] Konolige, Kurt 1987. "On the Relation Between Default Logic and Autoepistemic Theories", Artificial Intelligence 35(3):343-382.

[Konolige 1989] Konolige, Kurt 1989. "On the Relation between Autoepistemic Logic and Circumscription Preliminary Report", IJCAI89.

[Leasure 1993] Leasure, David E., The Modal Logic Z Applied to Lifschitz's Benchmark problems for Formal Nonmonotonic Reasoning, University of Kansas Ph.D Dissertation, University of Kansas Library , 1993.

[Leasure & Brown 1995] Leasure, David E. and Brown, Frank M. 1995. "AutoEpistemic Logic with Quantifiers", Proceedings Florida Artificial Intelligence Conference.

[Lewis 1936] Lewis, C. I. 1936. Strict Implication, *Journal of Symbolic Logic*, vol I.

[Moore 1985] Moore, R. C. 1985. "Semantical Considerations on Nonmonotonic Logic" *Artificial Intelligence*, 25.

[Park 1988], Park, S., On Formalizing Commonsense Reasoning Using the Modal Situational Logic and Reflective reasoning. Ph.D.dissertation, University of Texas, 1998.

[Parks 1976] Parks, Z. 1976. "An Investigation into Quantified Modal Logic", Studia Logica 35, p109-125.

Author Information

Frank M. Brown – University of Kansas, Lawrence, Kansas, 66045, e-mail: brown@ku.edu.

METHODS FOR SOLVING NECESSARY EQUIVALENCES

Frank Brown

Abstract: Nonmonotonic Logics such as Autoepistemic Logic, Reflective Logic, and Default Logic, are usually defined in terms of set-theoretic fixed-point equations defined over deductively closed sets of sentences of First Order Logic. Such systems may also be represented as necessary equivalences in a Modal Logic stronger than S5 with the added advantage that such representations may be generalized to allow quantified variables crossing modal scopes resulting in a Quantified Autoepistemic Logic, a Quantified Autoepistemic Kernel, a Quantified Reflective Logic, and a Quantified Default Logic. Quantifiers in all these generalizations obey all the normal laws of logic including both the Barcan formula and its converse. Herein, we address the problem of solving some necessary equivalences containing universal quantifiers over modal scopes. Solutions obtained by these methods are then compared to related results obtained in the literature by Circumscription in Second Order Logic since the disjunction of all the solutions of a necessary equivalence containing just normal defaults in these Quantified Logics, is equivalent to that system.

Keywords: Solving Necessary Equivalences, Modal Logic, Nonmonotonic Logic.

1. Introduction

Solving equations is an important aspect of Automated Deduction because the solutions to an equation can often be used in other parts of a theorem in order to prove the theorem or to derive consequences. Normally, we think of solving equations in some numeric valued algebra such as number theory, real algebra, complex algebra, or linear spaces; but there is no reason the process cannot be applied to radically different types of mathematical structures although the actual techniques for solving equations will depend on the nature of that structure's algebra. A mathematical structure of particular interest in Artificial Intelligence is the sets of sentences of First Order Logic (i.e. FOL) which are deductively complete by the laws of FOL because such sets form the basis of the fixed-point theories of nonmonotonic reasoning such as Autoepistemic Logic [Moore 1985], its kernel [Konolige 1987], Reflective Logic [Brown 1987], and Default Logic [Reiter 1980]. A set of sentences of FOL is said to be deductively complete if and only if all the theorems deducible from it by the laws of FOL are contained within it. Deriving properties of infinite sets of sentences would appear to involve a sophisticated automatic theorem prover for set theory and FOL syntax. However, we avoid this by noting that deductively complete sets of sentences may be represented by the proposition which is the meaning of all the sentences in that set. The modal sentence $[]((\wedge_i\Gamma_i)\rightarrow\alpha)$ where [] is the necessity symbol can then be used to represent the proof theoretic statement $`\alpha\in(\mathrm{fol}\{'\Gamma_i\})$ where $`\alpha$ is the name of a sentence α, for each i, $'\Gamma_i$ is the name of the sentence Γ_i of FOL, and fol is the set of FOL sentences derivable from $\{'\Gamma_i\}$ by the laws of FOL. A nonmonotonic fixed-point equation defined in set theory such as:

$$K=\mathrm{fol}(\{'\Gamma_i\}\cup\{`\alpha: (`(\neg\alpha)\notin K)\})$$

is then represented as the necessary equivalence:

$$k\equiv(\Gamma\wedge((\neg[]\neg(k\wedge\alpha))\rightarrow\alpha))$$

This algebra for solving for k in a necessary equivalence is just FOL supplemented with propositional variables (e.g. k above) and the necessity operator of a particular modal quantificational logic called Z. Translations of Autoepistemic Logic, Reflective Logic, and Default Logic, to Z are proven, respectively, in [Brown 2003c], [Brown 2003a], and [Brown 2003b]. An Automatic Deduction System, based on Z, for Autoepistemic Logic and Reflective Logic is given in [Brown 2003e]. Besides providing an algebra for representing such nonmonotonic systems (where quantified variables are not allowed to cross modal scopes), this modal representation has the additional advantage of providing an explication of what it means to have quantified variables crossing modal scopes:

$$k\equiv(\Gamma\wedge\forall\xi((\neg[]\neg(k\wedge\alpha))\rightarrow\alpha))$$

where ξ may occur free in α. Quantified Autoepistemic Logic, its Quantified Kernel, and Quantified Reflective Logic and their relationships are discussed in [Brown 2004]. Quantified Reflective Logic and Quantified Default

Logic and their relationship is discussed in [Brown 2003d]. Herein, we exemplify necessary equivalence solving by solving an example involving normal defaults, similar to the example given above, which is expressible in all of these quantified systems.

Section 2 describes the Z Modal Quantificational Logic which is the algebra in which the necessary equivalence solving takes place. Section 3 discusses proving what is logically possible. Section 4 discusses necessary equivalence solving. Section 5 discusses deducing what is common to all solutions. Finally, some conclusions are drawn in Section 6.

2. The Modal Quantificational Logic Z

The syntax of Z Modal Quantificational Logic is an amalgamation of 3 parts:

(1) The first part is a First Order Logic (i.e. FOL) represented as the six tuple: ($\rightarrow$, #f, $\forall$, *vars*, *predicates*, *functions*) where $\rightarrow$, #f, $\forall$, are logical symbols, *vars* is a set of object variable symbols, *predicates* is a set of predicate symbols each of which has an implicit arity specifying the number of terms associated with it, and *functions* is a set of function symbols each of which has an implicit arity specifying the number of terms associated with it. Roman letters x, y, and z, possibly indexed with digits, are used as variables.

(2) The second part is an extension to allow Propositional Quantifiers. It consists of a set of propositional variables *propvars* and quantification over propositional variables (using $\forall$). Roman letters (other than x, y and z) possibly indexed with digits are used as propositional variables.

(3) The third part is Modal Logic [Lewis 1936] which adds the necessity symbol: [].

Greek letters are used as syntactic metavariables. $\pi,\pi_1...\pi_n,\rho,\rho_1...\rho_n$ range over the predicate symbols, ϕ, $\phi_1...\phi_n$ range over function symbols, $\delta,\delta_1...\delta_n$ range over terms, $\gamma,\gamma_1,...\gamma_n$, range over the object variables ξ, $\xi_1...\xi_n,\zeta,\zeta_1...\zeta_n$ range over a sequence of object variables of an appropriate arity, $f, f_1...f_n$ range over predicate variables, and $\alpha,\alpha_1...\alpha_n$, $\beta,\beta_1...\beta_n$, $\chi,\chi_1...\chi_n$, Γ, κ range over sentences. Thus, terms are of the forms: γ and (ϕ $\delta_1...\delta_n$), and sentences are of the forms: ($\alpha \rightarrow\beta$), #f, ($\forall\gamma\ \alpha$), ($\pi\ \delta_1...\delta_n$), ($f\ \delta_1...\delta_n$), ($\forall f\ \alpha$), and ([]α). A zero arity predicate π, propositional variable f, or function ϕ is written as a sentence or term without parentheses, i.e., π instead of (π), f instead of (f), and ϕ instead of (ϕ). $\alpha\{\pi/\lambda\xi\beta\}$ is the sentence obtained from α by replacing all unmodalized occurrences of π by $\lambda\xi\beta$ followed by lambda conversion. $\alpha\{\pi_i/\lambda\xi\beta_i\}_{i=1,n}$, abbreviated as $\alpha\{\pi_i/\lambda\xi\beta_i\}$, represents simultaneous substitutions. $\wedge_{i=1,n}\beta_i$ represents ($\beta_1\wedge...\wedge\beta_n$). The primitive symbols are listed in Figure 1. The defined symbols are listed in Figure 2.

Symbol	Meaning
$\alpha\rightarrow \beta$	if α then β.
#f	falsity
$\forall\gamma\ \alpha$	for all γ, α.
$\forall f\ \alpha$	for all f, α.
[] α	α is logically necessary

Figure 1: Primitive Symbols

Symbol	Definition	Meaning
$\neg\alpha$	$\alpha \rightarrow$ #f	not α
#t	$\neg$ #f	truth
$\alpha\vee\beta$	$(\neg\ \alpha)\rightarrow \beta$	α or β
$\alpha\wedge\beta$	$\neg(\alpha \rightarrow \neg\beta)$	α and β
$\alpha\leftrightarrow \beta$	$(\alpha\rightarrow \beta)\wedge(\beta\rightarrow \alpha)$	α if and only if β
$\exists\gamma\ \alpha$	$\neg\forall\gamma\ \neg\alpha$	some γ is α

$\exists f\alpha$	$\neg\forall f\ \neg\alpha$	some f is α
$<>\ \alpha$	$\neg[]\neg\alpha$	α is logically possible
$[\beta]\ \alpha$	$[](\beta\rightarrow\alpha)$	β entails α
$<\beta>\ \alpha$	$<>(\beta\wedge\alpha)$	α is possible with β
$\alpha\equiv\beta$	$[](\alpha\leftrightarrow\beta)$	α and β are synonymous
$\delta_1=\delta_2$	$(\pi\ \delta_1)\equiv(\pi\ \delta_2)$	δ_1 necessarily equals δ_2
$\delta_1\neq\delta_2$	$\neg(\delta_1=\delta_2)$	δ_1 does not necessarily equal δ_2
(det α)	$\forall f(([\alpha]f)\vee([\alpha]\neg f))$where f is not in α	α is deterministic
(world α)	$(<>\alpha)\wedge$(det α)	α is a world

Figure 2: Defined Symbols of Z

The laws of Z Modal Logic is an amalgamation of five parts:

(1) The first part, given in Figure 3, consists of the laws of a FOL [Mendelson 1964].

(2) The second part, which is given in Figure 4, consists of the additional laws needed for propositional quantification. The laws SOLR2, SOLA4 and SOLA5 are the analogues of FOLR2, FOLA4 and FOLA5 for propositional variables.

(3) The Third part, which is given in Figure 5, consists of the laws MR0, MA1, MA2 and MA3 which constitute an S5 modal logic [Hughes & Cresswell 1968] [Carnap 1956]. When added to parts 1 and 2, they form a fragment of a Second Order Modal Quantificational logic similar to a second order version of [Bressan 1972].

(4) The fourth part, which is given in Figure 6, consists of the Priorian World extension of S5 Modal Logic. The PRIOR law [Prior and Fine 1977] states that a proposition is logically true if it is entailed by every world. This law was implied in [Leibniz 1686]

(5) The fifth part, which is given in Figure 7, consists of laws axiomatizing what is logically possible. MA1a lets one derive theorems such as $<>\forall x((\pi\ x)\leftrightarrow(x=\phi))$ and therefore extends the work on propositional possibilities in [Hendry and Pokriefka 1985] to possibilities in FOL being simpler than the FOL approaches in [Brown 1987]. MA1b and MA4 allow one to derive $(\phi_1\ \xi_1)\neq(\phi_2\ \xi_2)$ when ϕ_1 and ϕ_2 are distinct function symbols. MA4 states that at least two things are not necessarily equal just as there are at least two propositions: #t and #f.

FOLA3: $\alpha\rightarrow(\beta\rightarrow\alpha)$

FOLA4: $(\alpha\rightarrow(\beta\rightarrow\rho))\rightarrow((\alpha\rightarrow\beta)\rightarrow(\alpha\rightarrow\rho))$

FOLA5: $((\neg\alpha)\rightarrow(\neg\beta))\rightarrow(((\neg\alpha)\rightarrow\beta)\rightarrow\alpha)$

FOLA6: $(\forall\gamma\ \alpha)\rightarrow\beta$ where β is the result of substituting an expression (which is free for the free positions of γ in α) for all the free occurrences of γ in α.

FOLA7: $(\forall\gamma\ (\alpha\rightarrow\beta))\rightarrow(\alpha\rightarrow(\forall\gamma\ \beta))$ where γ does not occur in α.

FOLR1: from α and $(\alpha\rightarrow\beta)$ infer β

FOLR2: from α infer $(\forall\gamma\ \alpha)$

Figure 3: The Laws of FOL

SOLR2: from α infer $(\forall f\ \alpha)$

SOLA6: $(\forall f\ \alpha)\rightarrow\beta$ where β is the result of substituting an expression (which is free for the free positions of f in α) for all the free occurrences of f in α.

SOLA7: $(\forall f(\alpha\rightarrow\beta))\rightarrow(\alpha\rightarrow(\forall f\ \beta))$ where f does not occur in α.

Figure 4: Additional Laws for Propositional Quantifiers

MA1: $([]\alpha)\rightarrow\alpha$

MA2: $([\alpha]\beta)\rightarrow(([]\alpha)\rightarrow([]\beta))$

MR0: from α infer $([]\ \alpha)$ provided α was derived only from logical laws

MA3: $([]\alpha)\vee([]\neg[]\alpha)$

Figure 5: Additional Laws of S5 Modal logic

PRIOR: $(\forall w((\text{world } w) \rightarrow ([w]\alpha)))\rightarrow []\alpha$ where w does not occur free in α.

Figure 6: Additional Law of a Priorian World Logic

MA1a: $([]\alpha)\rightarrow\alpha\{\pi/\lambda\zeta\beta\}$ if no unmodalized occurrence of a higher order var is in α.

MA1b: $([]\alpha) \rightarrow \alpha\{\phi/\lambda\zeta\delta\}$ where α may not contain an unmodalized occurrence of a higher order variable, nor an unmodalized free object variable.

MA4: $\exists x\exists y(x\neq y)$

Figure 7: Additional Laws of Z Modal Logic

3. Deriving what is Logically Possible

The main problem with using axiom scheme MA1a to prove that something is logically possible lies in finding the appropriate substitution for the parameter π. The theorem ZP1 given below is the basis of a heuristic for finding such instances. Intuitively, we know that the conjunction of instances of a predicate and the conjunction of instances of the negation of a predicate are logically possible whenever the two do not coincide on any instance. For example: where $a\neq c$ and $b\neq c$ then $((\pi\ a)\wedge(\pi\ b)\wedge(\neg(\pi\ c)))$ is logically possible but $((\pi\ a)\wedge(\pi\ b)\wedge(\neg(\pi\ a)))$ is not. Thus if a sentence can be written in the form: $(\Gamma\wedge(\forall x(\alpha\rightarrow(\pi\ x)))\wedge(\forall x(\beta\rightarrow(\neg(\pi\ x)))))$ where π does not occur in α, β, and Γ, then it is logically possible if and only if $\{x{:}\alpha\}\cap\{x{:}\beta\}$ is empty, which is to say that $\exists x(\alpha\wedge\beta)$ does not follow from Γ or to say that Γ and $\neg\exists x(\alpha\wedge\beta)$ is logically possible. In this manner determining whether a sentence with n predicates is logically possible can sometimes be reduced to determining whether a sentence with n-1 predicates is logically possible without having to guess any instances of π in MA1a.

<u>Theorem ZP1: The Possibility of a Disjoint Predicate Definition:</u> [Brown 1989]

If Γ, α, and β do not contain any unmodalized occurrences of π nor of any higher order variable then:

$$(<>(\Gamma\wedge(\forall\xi(\alpha\rightarrow(\pi\ \xi)))\wedge(\forall\xi(\beta\rightarrow\neg(\pi\ \xi))))) \leftrightarrow(<>(\Gamma\wedge(\neg\exists\xi(\alpha\wedge\beta))))$$

ZP1 is applicable to any theory which can be put into a prenix conjunctive normal form such that no disjunct contains more than one unmodalized occurrence of π, since by the laws of classical logic such theories are equivalent to an expression of the form: $(\Gamma\wedge(\forall\xi(\alpha\rightarrow(\pi\ \xi)))\ \wedge(\forall\xi(\beta\rightarrow\neg(\pi\ \xi))))$. It is also applicable to any theory which can be put into a disjunction of a prenix conjunctive normal forms whose disjunctions contains no more than one unmodalized occurrence of π. The reason for this is that disjunction (and existential quantifiers) associate through possibility: $(<>(\alpha\vee\beta))\leftrightarrow((<>\alpha)\vee(<>\beta))$. For this reason, ZP1 is decidable for many important cases of FOL including propositional logic and the case of a theory with a finite number of intensional objects.

<u>Example 1</u>: Deducing a Logical Possibility thrice using ZP1. Let k be any sentence.

$<>((P\ b)\wedge(P\ a)\wedge(\neg(Q\ b))\ \wedge\forall x((\neg(AB\ x))\rightarrow((P\ x)\rightarrow(Q\ x)))$
$\quad\wedge\forall x((x\neq b)\rightarrow\neg(AB\ x))\wedge\forall x(((x\neq a)\wedge(x\neq b))\rightarrow((P\ x)\leftrightarrow([k](P\ x))))\ )$

This is equivalent to:

$<>((P\ b)\wedge(P\ a)\wedge\forall x\ ((x=b)\rightarrow\neg(Q\ x))\wedge\forall x(((\neg(AB\ x))\wedge(P\ x))\rightarrow(Q\ x))$
$\quad\wedge\forall x((x\neq b)\rightarrow\neg(AB\ x))\wedge\forall x(((x\neq a)\wedge(x\neq b))\rightarrow((P\ x)\leftrightarrow([k](P\ x))))\ )$

Instantiating ZP1 by letting α be $(\neg(AB\ x))\wedge(P\ x))$, β be x=b, and Γ be the sentences not containing Q gives:

$<>((P\ b)\wedge(P\ a)\wedge\neg\exists x((\neg(AB\ x))\wedge(P\ x)\wedge(x=b))\wedge\forall x((x\neq b)\rightarrow\neg(AB\ x))\wedge\forall x(((x\neq a)\wedge(x\neq b))\rightarrow((P\ x)\leftrightarrow([k](P\ x))))\ )$

which is equivalent to: $<>((P\ b)\wedge(P\ a)\wedge(AB\ b))\wedge\forall x((x\neq b)\rightarrow\neg(AB\ x))\wedge\forall x(((x\neq a)\wedge(x\neq b))\rightarrow((P\ x)\leftrightarrow([k](P\ x))))\ )$

which is: $<>((P\ b)\wedge(P\ a)\wedge\forall x((x=b)\rightarrow(AB\ x))\wedge\forall x((x\neq b)\rightarrow\neg(AB\ x))\wedge\forall x(((x\neq a)\wedge(x\neq b))\rightarrow((P\ x)\leftrightarrow([k](P\ x))))\ )$

Instantiating ZP1 by letting α be x=b, β be $x\neq b$, and Γ be the sentences not containing AB, gives:

$<>((P\ b)\wedge(P\ a)\wedge\neg\exists x((x=b)\wedge(x\neq b))\wedge\forall x(((x\neq a)\wedge(x\neq b))\rightarrow((P\ x)\leftrightarrow([k](P\ x))))\ )$

which is equivalent to: $<>((P\ b)\wedge(P\ a)\wedge\forall x(((x\neq a)\wedge(x\neq b))\rightarrow((P\ x)\leftrightarrow([k](P\ x))))\ )$

which is: $<>(\#t\wedge\forall x(((x=a)\vee(x=b)\vee((x\neq a)\wedge(x\neq b)\wedge([k](P\ x))))\rightarrow(P\ x))\wedge\forall x(((x\neq a)\wedge(x\neq b)\wedge(\neg([k](P\ x))))\rightarrow\neg(P\ x))$

Instantiating ZP1 by letting α be $((x=a)\vee(x=b)\vee((x\neq a)\wedge(x\neq b)\wedge([k](P\ x))))$, β be $((x\neq a)\wedge(x\neq b)\wedge(\neg([k](P\ x))))$ and Γ be #t gives: $<>(\#t\wedge\neg\exists x(((x=a)\vee(x=b)\vee((x\neq a)\wedge(x\neq b)\wedge([k](P\ x))))\wedge((x\neq a)\wedge(x\neq b)\wedge(\neg([k](P\ x))))\ ))$

which $<>\#t$ which is #t.

4. Solving Necessary Equivalences when Variables Cross Modal Scopes

A necessary equivalence has the form $k\equiv(\Phi\ k)$ where k is a propositional variable. The goal is to transform the initial necessary equivalence into a (possibly infinite) disjunction, $\exists i(k\equiv\beta_i)$ with each β_i free of k. If $k\equiv\beta_i$ implies the original equation then β_i is a solution to the original equation. The following procedure solves necessary equivalences when no variable crosses modal scopes:

<u>Procedure for Solving Modal Equivalences [Brown 1986]:</u>

<u>Step 1</u>: First, one by one, each subformula α which contains k and is equivalent to $([]\alpha)$ is pulled out of the necessary equivalence causing it to be split into two cases. This is done by using the following theorem schema replacing any instance of the left side by the corresponding instance of the right side:

$(k\equiv(\phi\ \alpha))\leftrightarrow((\alpha\wedge(k\equiv(\phi\ \#t)))\vee((\neg\alpha)\wedge(k\equiv(\phi\ \#f))))$

<u>Step 2</u>: Second, the resulting equivalences are simplified by the laws of the modal logic.

<u>Step 3</u>: Third, on each disjunct the simplified value for k is back substituted into each such α or $(\neg\alpha)$ sentence thereby eliminating k from them.

<u>Step 4</u>: Fourth, the α and $(\neg\alpha)$ sentences are simplified using the modal laws giving a disjunction of necessary equivalences.

When no variables cross modal scopes, for any decidable case of First Order Logic, this method is an algorithm resulting in a finite disjunction of solutions as is illustrated in Example 2 below:

<u>Example 2: Solving a Modal Equation:</u> $k\equiv(((<k>\neg A)\rightarrow B)\wedge((\neg<k>\neg A)\rightarrow A))$

Step 1 gives: $((<k>\neg A)\wedge(k\equiv((\#t\rightarrow B)\wedge((\neg\#t)\rightarrow A))))\vee((\neg(<k>\neg A))\wedge(k\equiv((\#f\rightarrow B)\wedge((\neg\#f)\rightarrow A))))$

Step 2 gives: $((<k>\neg A)\wedge(k\equiv B))\vee((\neg(<k>\neg A))\wedge(k\equiv A))$

Step 3 gives: $((<B>\neg A)\wedge(k\equiv B))\vee((\neg(<A>\neg A))\wedge(k\equiv A))$

Step 4 gives: $(\#t\wedge(k\equiv B))\vee(\#t\wedge(k\equiv A))$ which is: $(k\equiv B)\vee(k\equiv A)$

The process described above provides an algorithm [Brown & Araya 1991] for solving necessary equivalences in the modal representations of Reflective Logic [Brown 2003a] and Autoepistemic Logic [Brown 2003c], and with some additional details of [Default Logic 2003b]. These modal representations can be generalized to allow for universally quantified variables crossing modal scopes (at the top level of the right side of the equation). We now address the problem of solving necessary equivalences when quantified variables cross modal scopes as in:

$$k\equiv(\Gamma\wedge\wedge_i\forall\xi_i((([k]\alpha_i)\wedge(\wedge_{j=1,m_i}(<k>\beta_{ij})))\rightarrow\chi_i))$$

We want to eliminate the modal expressions containing k from the right side of the necessary equivalence. The difficulty lies in the fact that in general we cannot apply Step 1 in the above algorithm because the modal expressions may contain the ξ_i variables which are captured by quantifiers inside the right side of the necessary equivalence. The solution to this dilemma is to allow quantified statements such as: $\forall\xi((([k]\alpha)\wedge(\wedge_{j=1,m}(<k>\beta_j)))\rightarrow\chi)$ to be divided into a finite number of instances for each particular formula, which may or may not hold in a particular solution, leaving all the remaining instances in the quantified statement:

<u>Step 0</u>: Divide a quantified expression over a modal scope into parts using the schema:

$$(k\equiv(\Psi\wedge\forall\xi((([k]\alpha)\wedge(\wedge_{j=1,m}(<k>\beta_j)))\rightarrow\chi_i)))$$

$$\leftrightarrow(k\equiv(\Psi\wedge\forall\xi((\phi\wedge([k]\alpha)\wedge(\wedge_{j=1,m}(<k>\beta_j)))\rightarrow\chi_i))\wedge\forall\xi((\neg\phi\wedge([k]\alpha)\wedge(\wedge_{j=1,m}(<k>\beta_j)))\rightarrow\chi)))$$

where ϕ specifies a finite number of instances thereby allowing the quantifier above it in the resulting expression to be eliminated. The parts remaining under the quantifier that are consistent with the solution may then sometimes be eliminated by theorems PR or NPR. We call this application Step 5:

Theorem PR: Reduction of Possible Reflections: If Γ, α, and β are sentences of Z, and if γ is not free in Γ then:

(∀γ <>(Γ∧(∀γ β)∧α)) →((k≡(Γ∧∀γ((<k>α)→β)))↔(k≡(Γ∧∀γ β)))

proof: (k≡(Γ∧∀γ ((<k>α)→β)))↔ (k≡(Γ∧∀γ β)) is true iff:

((k≡(Γ∧∀γ ((<k>α)→β))) → (k≡(Γ∧∀γ β))) ∧ ((k≡(Γ∧∀γ β)) → (k≡(Γ∧∀γ ((<k>α)→β))))

Using the hypothesis in each case we get:

(((k≡(Γ∧∀γ ((<k>α)→β)))→((Γ∧∀γ((<k>α)→β))≡(Γ∧∀γ β)))
∧((k≡(Γ∧∀γ β))→ ((Γ∧∀γ β)≡(Γ∧∀γ ((<k>α)→β)))))

which would be true if: (((k≡(Γ∧∀γ ((<k>α)→β)))→∀γ(<k>α)) ∧ ((k≡(Γ∧∀γ β))→∀γ (<k>α)))

which is implied by: (∀γ (<(Γ∧∀γ((<k>α)→β))>α))∧∀γ(<(Γ∧∀γ β)>α)

which is equivalent to just: ∀γ (<>(Γ∧(∀γ β)∧α)) which is the hypothesis of the theorem. QED.

Theorem NPR: Reduction of Normal Possible Reflections:

If Γ and α are sentences of Z, and if γ is not free in Γ then:

(∀γ <>(Γ∧(∀γ α))) →((k≡(Γ∧∀γ ((<k>α)→α)))↔(k≡(Γ∧∀γ α)))

proof: Letting β be α in PR gives: (∀γ <>(Γ∧(∀γ α)∧α)) → ((k≡(Γ∧∀γ((<k>α)→α)))↔(k≡(Γ∧∀γ α)))

Since (∀γ α) implies α this is proves the theorem. QED.

Example 3: Partially Solving a Modal Equation with Quantifiers over Modal Scopes

k≡((P a)∧(¬(Q b))∧∀x((¬(AB x))→((P x)→(Q x))) ∧∀x((<k>¬(AB x))→¬(AB x)) ∧∀x((P x)↔([k](P x))))

Step 0: Dividing ∀x((P x)↔([k](P x))) on a and b and simplifying by noting that [k]Pa gives :

(k≡((P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x))) ∧∀x((<k>¬(AB x))→¬(AB x))
∧((P b)↔([k](P b)))∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x))))))

Steps 1&2: Splitting on ([k](P b)) gives:

((([k](P b))∧(k≡((P b)∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x))) ∧∀x((<k>¬(AB x))→¬(AB x))
∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x)))))))
∨(¬([k](P b))∧(k≡((¬(P b))∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))
∧∀x((<k>¬(AB x))→¬(AB x)) ∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x))))))))

Step 0: In the first necessary equivalence (P b)∧(¬(Q b)) implies (AB b). Thus dividing the AB default ∀x((<k>¬(AB x))→¬(AB x)) on b gives: ∀x(((x≠b)∧(<k>¬(AB x)))→¬(AB x)) which is equivalent to: (<k>((x≠b)→¬(AB x)))→((x≠b)→¬(AB x)) resulting in:

((([k](P b))∧(k≡((P b)∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))
∧∀x((<k>((x≠b)→¬(AB x)))→((x≠b)→¬(AB x)))∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x)))))))
∨(¬([k](P b))∧(k≡((¬(P b))∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))
∧∀x((<k>¬(AB x))→¬(AB x)) ∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x))))))))

In the second necessary equivalence, if a=b then k≡#f which implies that ¬([k](P b)) is equivalent to ¬([#f](P b)) which is equivalent to #f. Thus a≠b on the second case giving:

((([k](P b))∧(k≡((P b)∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))
∧∀x((<k>((x≠b)→¬(AB x)))→((x≠b)→¬(AB x)))∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x)))))))

∨(¬([k](P b))∧(a≠b)∧(k≡((¬(P b))∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))

∧∀x((<k>¬(AB x))→¬(AB x)) ∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x))))))))

Step 5 twice: NPR is used to eliminate (<k>((x≠b)→¬(AB x))) from the first necessary equivalence and (<k>¬(AB x)) from the second necessary equivalence. On the first necessary equivalence the hypothesis to NPR is: <>((P b)∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))∧∀x((x≠b)→¬(AB x))∧∀x(((x≠a)∧(x≠b)) →((P x)↔([k](P x))))) and on the second necessary equivalence the hypothesis to NPR is: <>((¬(P b))∧(P a)∧(¬(Q b))∧∀x((¬(AB x))→((P x)→(Q x)))∧∀x(¬(AB x))∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x))))). The first possibility is deduced to be #t by three applications of ZP1 (See Example 1 herein). The second possibility is deduced to be a≠b by three applications of ZP1. Since a≠b is a hypothesis of this case it is true. Applying NPR in both cases then gives:

((([k](P b))∧(k≡((P b)∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))

∧∀x((x≠b)→¬(AB x))∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x)))))))

∨(¬([k](P b))∧(a≠b)∧(k≡((¬(P b))∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))

∧∀x(¬(AB x))∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x))))))))

Steps 3&4 twice: Since (P b) is in the first necessary equivalence, the entailment on the first case holds. Likewise since (¬(P b)) is in the second necessary equivalence (¬([k](P b))) is (¬([k]#f)) which is (<>k) which holds since #f is not a solution if a≠b. Thus we get:

(((k≡((P b)∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))

∧∀x((x≠b)→¬(AB x))∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x)))))))

∨((a≠b)∧(k≡((¬(P b))∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))

∧∀x(¬(AB x))∧∀x(((x≠a)∧(x≠b))→((P x)↔([k](P x))))))))

Since (AB b) is derivable in the first necessary equivalence, and since in either equation (P a) and (P b) hold if and only if each is entailed in k this is equivalent to:

((k≡((P b)∧(P a)∧(¬(Q b))∧∀x((¬(AB x))→((P x)→(Q x)))∧∀x((AB x)↔(x=b))∧∀x((P x)↔([k](P x)))))

∨((a≠b)∧(k≡((¬(P b))∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))∧∀x(¬(AB x)) ∧∀x((P x)↔([k](P x))))))

The result is essentially a disjunction of two necessary equivalences (which partially solve for k. Further derivation along these lines gives solutions making P hold for any subset of {x: #t}-{a b}. However, if all we want is what is common to all solutions then eliminating ([k](P x)) is not required as is shown in the next section.

5. Aggregating the Solutions

Parallel Circumscription [McCarthy 1986] in Second Order Logic with circumscribed π_i, variable, and fixed predicates ρ_j is equivalent to the “infinite disjunction” of all the solutions of a necessary equivalence:

$$\exists k(k\wedge(k\equiv(\Gamma\wedge \wedge_i\forall\xi_i((\langle k\rangle\neg(\pi_i\ x))\rightarrow\neg(\pi_i\ x)) \wedge \wedge_j\forall\xi_j((\rho_j\ x)\leftrightarrow([k](\rho_j\ x))))))$$

For this reason, in some cases, such as in Example 3 above we can compare the disjunction of the solutions with results obtained by Circumscription. The key theorem for this comparison is the following theorem which allows fixed predicates to be ignored after the Circumscribed predicates are eliminated.

<u>The Fixed Predicate Lemma [Brown 1989]:</u> $\exists k(k\wedge(k\equiv(\Gamma\wedge \wedge_i\forall\xi((\rho_i\ x)\leftrightarrow([k](\rho_i\ x)))))) \equiv \Gamma$

The necessary equivalences related to Circumscription are a small subclass of the necessary equivalences that are expressible. The main goal herein is to solve necessary equivalences, rather than to compute the "infinite disjunction" of solutions which is all that Circumscription does. However, the solutions in Example 3 are aggregated and compared in Example 4 with Circumscription for the case where a≠b. This case encompasses the case where a and b are distinct 0-arity function symbols.[1]

[1] Herein, = is necessary equality as defined in Figure 2. It should not be confused with an extensional equality predicate which only provides substitution properties through nonmodal contexts.

Example 4: Aggregating the Solutions when a≠b to (P a)∧(¬(Q b))∧∀x((¬(AB x))→((P x)→(Q x))) by AB with P fixed and Q variable:

∃k(k∧(k≡((P a)∧(¬(Q b))∧∀x((¬(AB x))→((P x)→(Q x)))

∧∀x((<k>¬(AB x))→¬(AB x))∧∀x((P x)↔([k](P x))))))

From Example 3 and assuming a≠b we get:

∃k(k∧((k≡((P b)∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))∧∀x((AB x)↔(x=b))∧∀x((P x)↔([k](P x)))))

∨(k≡((¬(P b))∧(P a)∧(¬(Q b))∧∀x((¬(AB x))→((P x)→(Q x)))∧∀x(¬(AB x)) ∧∀x((P x)↔([k](P x)))))))

Pushing ∃k to lowest scope gives:

((∃k(k∧(k≡((P b)∧(P a)∧(¬(Q b))∧∀x((¬(AB x))→((P x)→(Q x)))∧∀x((AB x)↔(x=b))∧∀x((P x)↔([k](P x)))))))

∨(∃k(k∧(k≡((¬(P b))∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))∧∀x(¬(AB x))∧∀x((P x)↔([k](P x))))))))

By the Fixed Predicate Lemma (twice) this is equivalent to:

((P b)∧(P a)∧(¬(Q b)) ∧∀x((¬(AB x))→((P x)→(Q x)))∧∀x((AB x)↔(x=b)))

∨((¬(P b))∧(P a)∧(¬(Q b))∧∀x((¬(AB x))→((P x)→(Q x)))∧∀x(¬(AB x)))

which is equivalent to:

(P a)∧(¬(Q b))∧∀x((¬(AB x))→((P x)→(Q x)))∧(((P b)∧∀x((AB x)↔(x=b)))∨((¬(P b))∧∀x(¬(AB x))))

Since a≠b, It follows that ¬(AB a) and therefore that (Q a) holds. This is exactly what one would expect as is suggested by the following quote from [Konolige 1989]: "Consider the simple abnormality theory (see [McCarthy 1986]), with W ={∀x.Px ∧¬AB(x) → Qx, Pa, ¬Qb} (this is a variation of an example in [Perlis, 1986].) We would expect Qa to be a consequence of Circum(W;ab;Q), but it is not."[1] "The reason is that there are ab-minimal models of W in which b and a refer to the same individual and ¬Qa is true."[2]

6. Conclusion

The Z Modal Quantificational Logic provides an interesting algebra for deriving fixed-point solutions to necessary equivalences where universally quantified variables cross modal scope. Herein, some specific methods for solving some simple classes of problems have been described and exemplified. The presented methods do not solve all problems, but the Z logic provides a framework for developing more general solution methods generalizing the ones herein presented. Since many quantified nonmonotonic logics are representable in the Z Modal Logic, including Quantified Autoepistemic Logic, Quantified Autoepistemic Kernels, Quantified Reflective Logic, and Quantified Default Logic, such deduction techniques could be applicable to a wide range of quantified generalizations of most of the well known nonmonotonic logics.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant Numbers: MIP-9526532, EIA-9818341, and EIA-9972843.

Bibliography

[Bressan 1972] Bressan, Aldo 1972. *A General Interpreted Modal Calculus,* Yale University Press.

[Brown 1986] Brown, F. M. 1986. "A Commonsense Theory of NonMonotonic Reasoning", *Proceedings of the 8h International Conference on Automated Deduction, CADE8,* Lecture Notes in Computer Science, vol 230, Oxford, England, July 1986, Springer Verlag, New York.

---

[1] If a≠b, such as is the case where a and b are two distinct 0-arity function symbols, it is a consequence of Circumscription as defined in Second Order Logic. In Second Order Logic x=y is defined to be ∀ƒ((ƒ x)↔(ƒ y)).

[2] The obvious definition of minimal model does not give the desired properties. The definition of minimal model could be changed to being minimal not with respect to all models but with respect to those models giving function symbols the same interpretation.

[Brown 1987] Brown, Frank M. 1987. "The Modal Logic Z", In *The Frame Problem in AI"; Proceedings of the 1987 AAAI Workshop*, Morgan Kaufmann, Los Altos, CA .

[Brown 1989] Brown, Frank M. 1989. "The Modal Quantificational Logic Z Applied to the Frame Problem", advanced paper *First International Workshop on Human & Machine Cognition*, May 1989 Pensacola, Florida. Abbreviated version published in *International Journal of Expert Systems Research and Applications, Special Issue: The Frame Problem. Part A.* eds. Kenneth Ford and Patrick Hayes, vol. 3 number 3, pp169-206, JAI Press 1990. Reprinted in *Reasoning Agents in a Dynamic World: The Frame problem*, editors: Kenneth M. Ford, Patrick J. Hayes, JAI Press 1991.

[Brown 2003a] Brown, Frank M. 2003a "Representing Reflective Logic in Modal Logic", *Information Theories and Applications*, Vol. 10, no. 4, pages: 431-438, December 2003.

[Brown 2003b] Brown, Frank M. 2003b, "Representing Default Logic in Modal Logic", *Information Theories and Applications*, Vol. 10, no. 4, pages 439-446, December, 2003.

[Brown 2003c] Brown, Frank M. 2003c, "Representing Autoepistemic Logic in Modal Logic", *Information Theories and Applications*, Vol. 10, no. 4, pages 455-462, December 2003.

[Brown 2003d] Brown, Frank M. 2003d, "On the Relationship between Quantified Reflective Logic and Quantified Default Logic" *Information Theories and Applications*, Vol. 10, no. 4, pages 447-454, December 2003.

[Brown 2003e] Brown, Frank M. 2003e, "Decision Procedures for the Propositional Cases of 2^{nd} Order Logic and Z Modal Logic Representations of a 1^{st} Order L-Predicate Nonmonotonic Logic", *Automated Reasoning with Analytic Tableaux and Related Method: Tableaux 2003*, September 2003, Rome, *Lecture Notes in Artificial Intelligence 2796*, ISBN 3-540-40787-1, Springer-Verlag, Berlin, 2003.

[Brown 2004] Brown, Frank, 2004, "On the Relationships Among Quantified Autoepistemic Logic, its Kernel, and Quantified Reflective Logic", *Information Theories and Applications*, Vol. 11, 2004.

[Brown and Araya 1991] Brown,Frank M. and Carlos Araya 1991. "A Deductive System for Theories of NonMonotonic Reasoning," *Transactions of the Eighth Army Conference on Applied Mathematics and Computing, 1991*.

[Carnap 1956] Carnap, Rudolf 1956. *Meaning and Necessity: A Study in the Semantics of Modal Logic*, The University of Chicago Press.

[Hendry and Pokriefka 1985] Hendry, Herbert E. and Pokriefka, M. L. 1985. "Carnapian Extensions of S5", *Journal of Phil. Logic 14*.

[Hughes & Cresswell 1968] Hughes, G. E. and Cresswell, M. J., 1968. *An Introduction to Modal Logic*, Methuen & Co. Ltd., London.

[Konolige 1987] Konolige, Kurt 1987. "On the Relation Between Default Logic and Autoepistemic Theories", Artificial Intelligence 35(3):343-382.

[Konolige 1989] Konolige, Kurt 1989. "On the Relation between Autoepistemic Logic and Circumscription Preliminary Report", IJCAI89.

[Leibniz 1686] Leibniz, G. 1686. "Necessary and Contingent Truths", *Leibniz Philosophical Writings*, Dent&Sons. 1973.

[Lewis 1936] Lewis, C. I. 1936. Strict Implication, *Journal of Symbolic Logic*, vol I..

[McCarthy 1986] McCarthy, J. 1986, "Applications of Circumscription to Formalizing Common-Sense Reasoning", *Artificial Intelligence*, volume 28.

[Mendelson 1964] Mendelson, E. 1964. *Introduction to Mathematical Logic*, Van Norstrand, Reinhold Co., New York.

[Moore 1985] Moore, R. C. 1985. "Semantical Considerations on Nonmonotonic Logic" *Artificial Intelligence*, 25.

[Prior 1977] Prior, A. N. & Fine, k. 1977. *Worlds, Times, and Selves*, Gerald Duckworth.

[Reiter 1980] Reiter, R. 1980. "A Logic for Default Reasoning" *Artificial Intelligence*, 13.

Author Information

Frank M. Brown – University of Kansas, Lawrence, Kansas, 66045, e-mail: brown@ku.edu.

ABOUT NEW PATTERN RECOGNITION METHOD FOR THE UNIVERSAL PROGRAM SYSTEM "RECOGNITION"

Alexander Dokukin and Oleg Senko

Abstract: In this work the new pattern recognition method based on the unification of algebraic and statistical approaches is described. The main point of the method is the voting procedure upon the statistically weighted regularities, which are linear separators in two-dimensional projections of feature space. The report contains brief description of the theoretical foundations of the method, description of its software realization and the results of series of experiments proving its usefulness in practical tasks.

Keywords: pattern recognition, statistically weighted regularities, voting procedure.

Introduction

Nowadays there are a great number of effective pattern recognition methods based on voting procedure upon some kind of regularities in the data, as well as different approaches for searching these regularities. The term "regularity" is interpreted as some sub-region in space of prognostic variables where fraction of at least one of the classes differs significantly from its fraction in neighbor regions. For example, there are the method of voting upon the sets of irreducible tests [1] or representative tests [13], the method of voting upon statistically weighted syndromes (further in the text it is referred as SWS) [2], method of voting upon sets of logical regularities [3] and etc. Results of hands-on testing show the higher steadiness of voting procedure to the minor changes in training and testing samples, which leads to the significant increase of quality of voting-based methods. This advantage is especially important in relatively high-dimensional tasks with limited number of cases in data sets. The theoretical substantiation of this fact [4] exceeds the bounds of this report, but the detailed proof by the means of mathematical statistics is now being prepared for publication.

All these methods have one strong restriction and maybe even disadvantage. It is the fact that all regularities are some kind of hyper parallelepipeds in feature space with planes orthogonal to the datum lines. However in many tasks the essentially multidimensional regularities may arise which are separated from neighborhood by multivariate linear boundaries. So the method of two-dimensional linear separators (further referred as TLS) presents an attempt to complicate shape of elementary regularities preserving all advantages of voting procedure.

The Method of Two-Dimensional Linear Separators (TLS)

Further following notation will be used. Let's consider the set of permissible objects M , let's also consider that it presents Cartesian product of n sets of permissible values of features $M = M_1 \times ... \times M_n$. It is presumed that there is the unknown subdivision of the set M into l classes $K_1,...,K_l$. This subdivision is described by means of training sample $S_1,...,S_m$ of objects $S_i = a_{i1},...,a_{in}, i = 1,...,m$, for which the classification is known: $\alpha(S) = \alpha_1,...,\alpha_l$, where $\alpha_j =< S \in K_j >, j = \overline{1,l}$. It is necessary to restore the unknown classification of the testing sample $S^1,...,S^q$.

The main point of the method is the successive examination of different pairs of features and construction of the linear separator for every one of them and for every class. These separators must divide two-dimensional projections of objects of selected class and its additive inversion. For every class K_i and for every pair of features (u,v) the found line $L^i_{(u,v)}$ is called elementary regularity. Moreover the weight of the regularity $w^i_{(u,v)}$ is calculated due to the separating ability of the line.

The recognition is based on the weighted voting procedure by the set of elementary regularities. Let's consider the estimation for k-th class, the rest ones are calculated in much the same way. Each regularity $L^k_{(u,v)}$ refers the new object S to the k-th class or to its additive inversion, so the part of training objects of k-th class in the half plain there the object was referred to can be calculated for each pair of features $\upsilon^k_{(u,v)}(S)$. The final estimation is calculated according to the following formula:

$$\Gamma(S, K_1) = \frac{\sum\limits_{(u,v)} \upsilon^1_{(u,v)}(S) w^1_{(u,v)}}{\sum\limits_{(u,v)} w^1_{(u,v)}}.$$

The weights of regularities are calculated in much the same manner as in the method of statistically weighted syndromes (SWS) and depend on the quality of separating of training sample. If there are any errors in separation, i.e. some objects from the class are referred to its additive inversion or objects from inversion are referred to the class, the weight is set to be inversely proportional to the variance of the error

$$\frac{1}{p(1-p)},$$

there p is the part of errors. If this is not the case and all training objects are separated correctly than the variance of error is replaced with its Bayesian estimation

$$\frac{\int\limits_0^1 (1-p)^n \, dp}{\int\limits_0^1 (1-p)^n \, pdp}$$

In the TLS method the linear separators are sought by means of pattern recognition method called Linear Machine [5]. Its main point is that the task of finding separating line is replaced with the task of finding the maximal simultaneous subsystem of the system of linear inequalities and its subsequent solving by means of relaxation algorithm.

In conclusion of this paragraph let's consider the results of some hand-on testing. In the table 1 there are some tasks that clearly demonstrate the advantages of unification of voting procedure and complex elementary regularities. It contains the results of comparison of Linear Machine, SWS and TLS methods. The method with best performance is marked with gray color.

Task	LM	SWS	TLS
Breast	94.9	94.1	95.2
Ionosphere	85.2	90.1	90.1
Iris	97.5	95	97.5
Mel	50	65.6	68.8
Patomorphosis	76.5	85.3	91.2

Table 1. Comparison of LM, SWS ans TLS methods

The following tasks were considered during the test series:

- Breast – the breast cancer recognition, 9 features, 2 classes, 344 training examples, 355 testing ones (Breast cancer databases was obtained from Dr. William H. Wolberg from the University of Wisconsin Hospitals, Madison [6]);
- Ionosphere – the recognition of structural peculiarities in ionosphere, 34 features, 2 classes, 170 training examples, 181 testing ones (data from Johns Hopkins University Ionosphere database);
- Iris – Iris recognition, 4 features, 3 classes, 71 training examples, 81 testing ones (data from Michal Marshall's Iris Plants Database);

- Mel – Recognition of melanoma by the set of geometrical and radiological features, 33 features, 3 classes, 48 training examples, 32 testing ones [12];
- Patomorphosis – forecast of destruction level of malignant growth after chemotherapy by the set of parameters characterizing optical behavior of its cell nucleus, 7 features, 2 classes, 43 training examples and 31 testing ones (the data has been received from Dr. Matchak from Cancer Research Center of the Russian Academy of Medical Sciences).

In the table 2 the results of comparison of TLS with some other methods build-in to the Recognition software system are shown. The methods ate tested with two tasks which features are small number of objects in comparison with dimension of task. It is important that the suggested method has shown the significant increase of quality in this class of tasks.

Method	Mel	Patomorphosis
TLS	68.8	91.2
LM	50	76.5
SWS	65.6	85.3
LDF	59.4	76.5
AVO	62.5	76.5
IT	62.5	85.3
QNN	62.5	70.6
Perceptron	65.6	79.4
SVM	56.3	76.5

Table 2. Comparison with other methods on the tasks with short samples.

Following methods were used: TLS – Two-dimensional Linear Separators, LM – the mentioned above Linear Machine[5], SWS [2], LDF - Fisher's Linear Discriminant [7], AVO or ECA – Estimates Calculating Algorithm [8], IT – voting upon Irreducible tests [1], QNN – q Nearest Neighbors [7], Perceptron – Multilayer Perceptron [7,9], SVM – Support Vector Machine [10].

Software Realization

Software system "Recognition" has been developed in Dorodnicyn Computing Centre of Russian Academy of Sciences in cooperation with Solutions Ltd. The system's detailed description can be found, for example, in the proceedings of the Open German-Russian Workshop [11] or at the developer's Internet sight http://www.solutions-center.ru. In this article only the brief description of its basic principles will be given, because these principles have been being considered throughout the whole TLS' development process. They are universality, uniformity, modularity and intellectuality.

The universality of the system is understood as a wide coverage of different approaches to pattern recognition and classification including so-called classifier fusion, which are realized in the system's library of methods. The methods have been developed as separate interchangeable modules of uniform structure. On the software level each module is a single dynamic-link library with standardized interface.

While solving a wide variety of different practical task the initial assumption that each recognition method has its advantages and there is no single best one for every kind of tasks has been proven. So the main accent was made on using of different kinds of classifier fusions. And the uniformity of the methods allows combining the results of every subset of developed methods into one classifier, providing results more accurate than average and even close to maximal ones in automatic training mode. This fact allows claiming some kind of intellectuality of the developed system.

Passing on to software realization of Two-dimensional Linear Separators method itself, it is important to take note of two facts:

First of all, developing of TLS method's software realization was significantly simplified due to availability of software system "Recognition", since it was taking care of all the chores including preparation of methods environment, quality control and etc. So the developers in the person of the authors of this paper were able to get concentrated on the method itself.

Secondly, the fact that TLS method has significantly increased the quality of recognition for some kind of tasks has been already mentioned in section 2. Thus the addition of this method to classifier fusions allows increasing their quality greatly for these tasks. The experimental proof of this fact is shown in table 3. One of the simplest ways of constructing classifier fusion has been considered. The simple majority voting procedure has been applied firstly to the set of LM, SWS, LDF, AVO, IT, QNN, Perceptron and SVM, and secondly to the same set of algorithms in addition with TLS.

Task	Without TLS	With TLS
Mel	59.4	65.6
Patomorphosis	76.5	82.4

Table 3. Quality of majority voting

Conclusion

In conclusion we can claim that the developed method has justified our hopes. The combination of voting procedure and linear separators has increased recognition quality in some class of practically important tasks. Thus the developed software realization of TLS method can serve as a great support for researchers.

Acknowledgment

The research has been supported by RFBR grants NN 02-07-90137, 02-07-90134, 02-01-00558, 03-01-00580, 02-01-08007 and state program НШ-1721.2003.1 (scientific schools).

Also thanks goes to Dr. William H. Wolberg from the University of Wisconsin Hospitals, Madison; Vince Sigillito from Johns Hopkins University (vgs@aplcen.apl.jhu.edu); Michal Marshall (MARSHALL%PLU@io.arc.nasa.gov); Dr. Ganster from Linz University and Dr. Matchak from Cancer Research Center of Russian Academy of Medical Sciences for the granted data.

Bibliography

[1] А.Н.Дмитриев, Ю.И.Журавлев, Ф.П.Кренделев, О математических принципах классификации предметов и явлений. Сб. "Дискретный анализ". Вып. 7. Новосибирск, ИМ СО АН СССР. 1966. С. 3-11.

[2] Сенько О.В. Использование процедуры взвешенного голосования по системе базовых множеств в задачах прогнозирования// М. Наука, Ж. вычисл. матем. и матем. физ. 1995, т. 35, № 10, С. 1552-1563.

[3] V.V.Ryazanov, Recognition Algorithms Based on Local Optimality Criteria, Pattern Recognition and Image Analysis. 1994. Vol.4. no.2. pp. 98-109.

[4] Сенько О.В., Кузнецова А.В. Метод предварительной селекции признаков. // Докл. Всеросс. Конф. «Матем. методы распознавания образов - 11». М.: Регион-Холдинг, 2003, С. 171-173.

[5] Обухов А.С., Рязанов В.В. Применение релаксационных алгоритмов при оптимизации линейных решающих правил. Доклады 10-й Всероссийской конференции "Математические методы распознавания образов (ММРО-10)", Москва, 2001, 102-104.

[6] William H. Wolberg and O.L. Mangasarian: "Multisurface method of pattern separation for medical diagnosis applied to breast cytology", Proceedings of the National Academy of Sciences, U.S.A., Volume 87, December 1990, pp. 9193-9196.

[7] R.O. Duda and P.E. Hart. Pattern Recognition and Scene Analysis. A Wiley-Interscience Publication. John Wiley and Sons. New York, 1973.

[8] Ю.И.Журавлев, Об алгебраическом подходе к решению задач распознавания или классификации, Проблемы кибернетики. М.: Наука, 1978. Вып.33. С.5-68.

[9] Рязанов В.В., Челноков Ф.Б., О склеивании нейросетевых и комбинаторно-логических подходов в задачах распознавания, Доклады 10-й Всероссийской конференции "Математические методы распознавания образов (ММРО-10)", Москва, 2001,115-118.

[10] Christopher J.C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition, Appeared in: Data Mining and Knowledge Discovery 2, 1998, 121-167.

[11] Yu.I. Zhuravlev, V.V. Ryazanov, O.V. Sen'ko, A.S. Biryukov, D.P. Vetrov, A.A. Dokukin, N.N. Katerinochkina, A.S. Obukhov, M.Yu. Romanov, I.V. Ryazanov, F.B. Chelnokov. The Program System for Data Analysis "Recognition" (LOREG). The 6th German-Russian Workshop "Pattern Recognition and Image Understanding". Proceedings. Novosibirsk, Russia, 2003, pp. 255-258.

[12] Ganster H., Gelautz M., Pinz A., Binder M., Pehamberger H., Bammer M., Krocza J. Initial Results of Automated Melanoma Recognition //Proceedings of the 9th Scandinavian Conference on Image Analysis, Uppsala, Sweden, June 1995, Vol.1, pp. 209-218.

[13] Dyukova Ye.V., Ryazanov V.V., The Solution of Applied Problems by Recognition Algorithms Based on the Voting Principle. VTs Akad. Nauk S.S.S.R., Moscow, 1986.

Authors' Information

Alexander A. Dokukin – Dorodnicyn Computing Centre of the Russian Academy of Sciences, Vavilov st., 40, Moscow GSP-1, 119991, Russia; e-mail: dalex@ccas.ru

Oleg V. Senko – Dorodnicyn Computing Centre of the Russian Academy of Sciences, Vavilov st., 40, Moscow GSP-1, 119991, Russia; e-mail: senkoov@mail.ru

ANALYSIS OF SECURITY IN ARCHIVING

Dimitrina Polimirova–Nickolova

Abstract: Some basic types of archiving programs are described in the paper in addition to their advantages and disadvantages with respect to the analysis of security in archiving. Analysis and appraisal are performed on the results obtained during the described experiments.

Keywords: Web Security, Mail Security, Information Security, Archive Programs, Compressed Objects, Methods Of Encryption.

The Present Situation

In the development of the computer science the creation and the use of archived objects is a classical research problem, which has found different resolutions for decades past. Nowadays the availability of several dozens of methods and their varieties represent an excellent demonstration of the ambitions of the information systems' programmers and designers for a real high-speed and high-effective compression of information flows.

The following basic types of archiving programs could be defined with respect to the information security of compressed objects, obtained after examination of more than 320 archiving programs, known by now:

1) E-mail archiving programs – in this kind of archiving programs the relative homogeneity of the information flow (e-mail traffic) is used and the most suitable methods of compression are selected. There are some differences among the basic existing e-mail clients (MS Outlook, MS Outlook Express, Netscape Mail, Opera Mail, Eudora Mail, Pegasus Mail etc.), which make possible the applying of different realizations of the compressing process. The advantages consist in the multiple reduction of the saved e-mail folders' volume and in the high degree of security against unauthorized access (viruses, worms, spyware, malware etc.). The disadvantages above all are related to the consumption of computing resources to realize the right and the reverse transformation.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: DBX (Outlook Express Email Folder), IDX (Outlook Express Mailbox Index), PCE (Eudora Mailbox Name Map), MSG (Pegasus Mail Stored Messages to Be Sent), SNM (Netscape Mail Email Message File), BOE (Outlook Express Backup File).

2) Converting archiving programs – these are archiving programs, that have the possibility to transform objects compressed by a given method in objects compressed by another method. Two variants exist with regard to this transformation: a) without a restoration of the object in its initial appearance; b) with a restoration of the object in its initial appearance. Their advantages consist in the use of a compression method which is optimal for a given type of information (e.g.: .jpg, .gif, .doc, .xls, .ppt etc.). Their disadvantages reside in the high complexity of operating environment.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: ACE (WinAce Compressed File), RAR (WinRAR Compressed Archive (RarLab)), ZIP (Compressed Archive File), AIN (AIN Compressed Archive), GZIP (GNU Zip Compressed Archive), UC2 (Compressed File).

3) Multiple archiving programs – these are programs which perform some successive kinds of archiving processing on the object for compression by using several methods of compression differing by their characteristics. In this manner a different (fully optimized) method of compression could be applied for the different parts of the object. The advantages lay in the very high flexibility, functionality and adaptivity to the different parts of the compressed objects which differ by their internal structure. The disadvantages are connected to the high initial expenditure needed for the creation of library of modules for similar methods of compression and the realization of a relevant environment, suitable for analysis of the separate parts of the objects.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: ARJ (Robert Jung ARJ Compressed Archive), JAR (JAR Archive (ARJ Software, Inc.)), TAR (Tape Archive File), AI (Ai Archiver Archive), LHA (Compressed Archive File), ZOO (ZOO Compressed Archive File).

4) Image archiving programs – this is an extremely live problem in the present-day real-time processing of video and image web-objects. The predominating trend in this processing is the obligatory compressing of the object immediately after its creation. The transmission and the processing of the object is fully realized in a compressed state to the last moment of its reproduction on the relevant media. The advantages consist in the significant reduction of the objects' dimension and the time needed for transmission, retransmission and processing. The disadvantages are connected to the high expenditure for the hardware components, which realize the compression partly or fully. A reasonable compromise in this respect are the combined (software-hardware) methods of compression.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: AIS (ACDSee Image Sequence File), B&W (Image Lab), BIL (ArcView Image File (ESRI)), BIN (Micrografx Designer 7 Project Image), CPT (Corel Photo-Paint Image (Corel)), PDB (PhotoDeluxe Image (Adobe)).

5) Data archiving programs – these are programs specialized in the creation, the processing and the use of compressed objects which result from information flows owning "data" characteristics. In the different platforms and operating systems the notion "data" has a different sense. In this instance we are concerned only by the fact, that the data in the different phases of their existence pass in a compressed form, exist for a fixed time in this form and a little time before to be "processed" the compressed objects are decompressed. The advantages lay in the reasonable degree of the optimal use of the resources. The disadvantages consist in the "superfluous" operations for compression and decompression.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: DOC (Word Document (Microsoft)), PDF (Acrobat Portable Document Format (Adobe)), TXT (Text File), XLS (Excel Worksheet (Microsoft)), XML (Extensible Markup Language File), PPT (Power Point Presentation (Microsoft)).

6) Executable archiving programs – the aim of these programs is to accomplish some specificity of the compression, connected with the possibilities for running the compressed objects. These are active objects which

own the capability for algorithmic branching of events depending on the used scenario. The advantages are connected to the extremely precise use of computing resources and the very high degree of protection against “reverse engineering”. The disadvantages consist in the dependence from the platform, the operating system, the applications on use and the human factor.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: EXE (Executable File (Microsoft)), PE (Portable Executable File), PL (Linux Shell Executable Binary), FOX (FoxBase/FoxProt Executable File), FMX (Oracle Executable Form (FRM)), XXY (SPARC Executable Script File).

The Problem

Protection of information is accomplished by data encryption. Data encryption is a process in which the contents of a message or a file is tangled to such extent that it becomes unintelligible to anybody. To enable the message decoding or the file reversal to its initial state, it is necessary to own some key or access code. This concept is similar to the one for data compression. Thus, two different goals could actually be achieved by using the same approach:

1) Size reduction, which is accomplished by data compression via encoding.
2) Making data unreadable, which encoding performs in the case of encryption.

The results of the experiments which were carried out will be shortly revealed to facilitate better achievements in enhancing security of objects, and especially for compressed objects. The goal of these experiments was to examine and analyze the combination of data compression and data encryption [1].

The first study analyzes the SPEED of the encoding process. In this regard, the following four tasks were defined:

1) Evaluation of the resulting files, compressed with popular compressing programs. Particular experiments were made for all 18 extensions. Their file size was 1 Mb, and all of them were compressed with the most popular compressing programs. The results for all 18 extensions used during the study can be seen in Figures 1a,b,c,d,e,f.

Fig. 1a

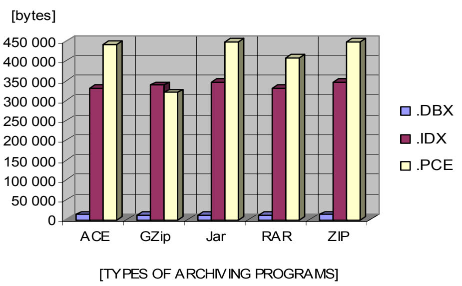


Fig. 1b

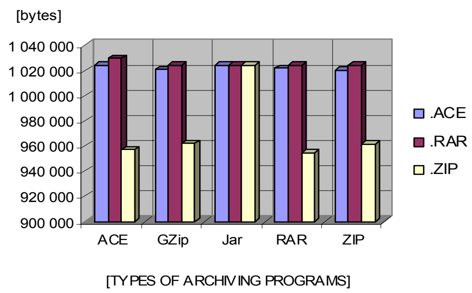


Fig. 1c

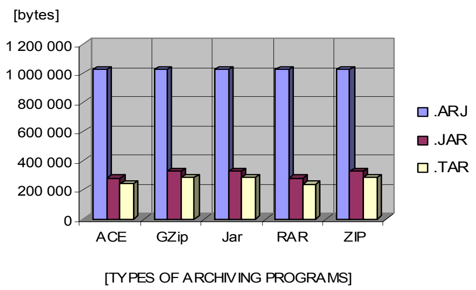


Fig. 1d

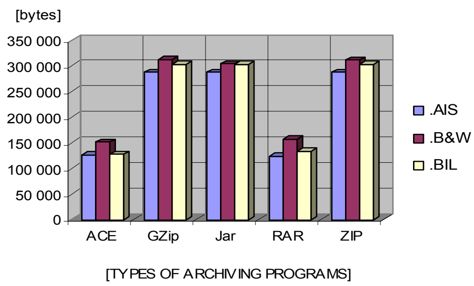


Fig. 1e

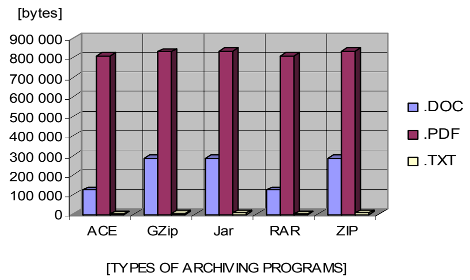
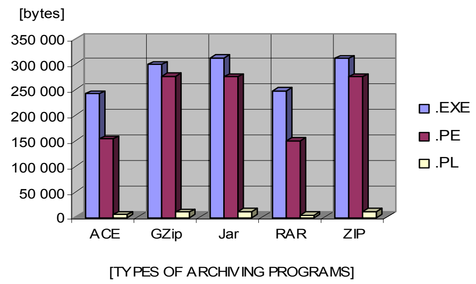


Fig. 1f



2) Encryption of the objects before compression. The same 18 extensions (3 for each type of archiving program) with original file size 1 Mb were used. The experiments which were made examined the file size of the encrypted 18 original extensions (Figure 2) and the time period needed for encryption of those 18 file extensions (Figure 3).

Fig. 2. File sizes of encrypted extensions

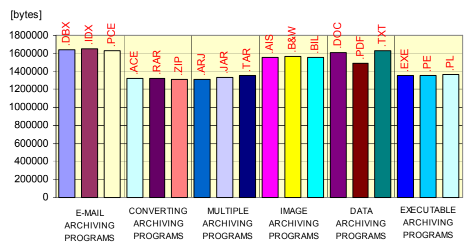
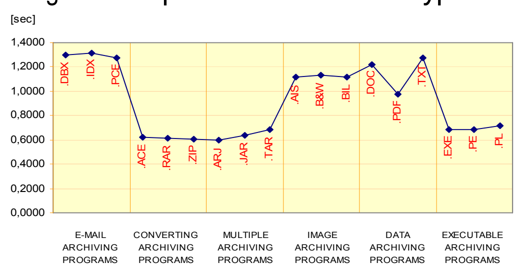


Fig. 3. Time period needed for encryption



3) Encryption of the objects after compression. The results used were the same as the ones obtained in the first task during this study. All compressed 18 extensions were encrypted, and their file sizes (Figure 4) and time periods (Figure 5) needed for encryption after compression were examined.

Fig. 4. File sizes of the encrypted after the compression extensions

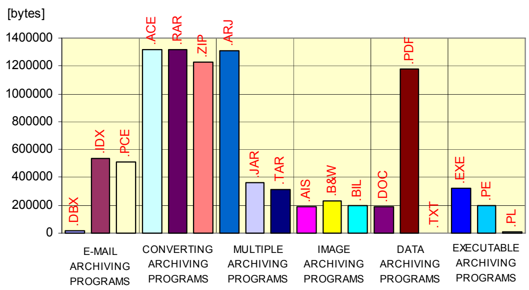
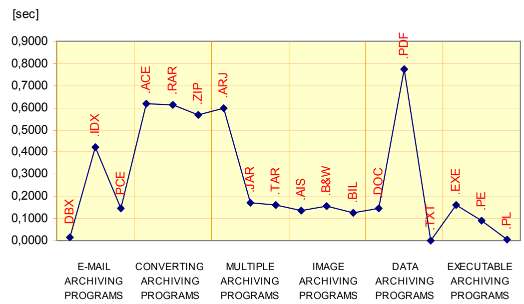
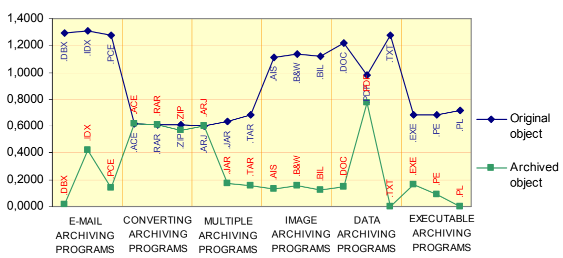


Fig. 5. Time periods needed for encryption after the compression



4) Comparison of the time periods needed for encryption of the objects before and after compression (Figure 6).

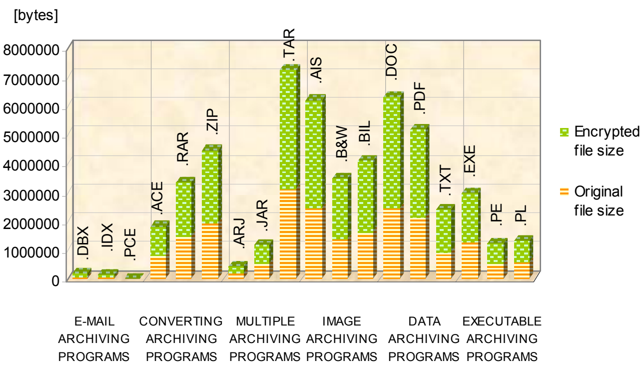
Fig. 6.



The second study analyzes the SIZE of the object created after the encryption. In this regard, the following three tasks were defined:

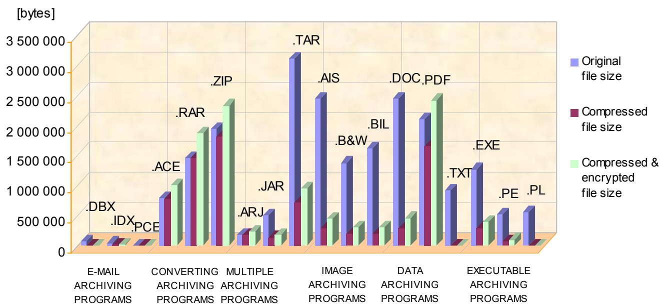
1) Encryption of objects with different file formats. Displayed in Figure 7 are the original file sizes, which are different for the 18 used extensions, and the file sizes obtained after the encryption of the original files.

Fig. 7.



2) Encryption of the files compressed with different compressing programs. Displayed in Figure 8 is the resulting comparison of the sizes of the original files and the sizes of the files encrypted after compression.

Fig. 8.



3) Comparison of the sizes of the original files and the sizes of the encrypted files. Encrypted original file sizes and encrypted-after-compression file sizes were examined, and the results for all 18 extensions are displayed in Figures 9a, b, c, d, e, f.

Fig 9a.

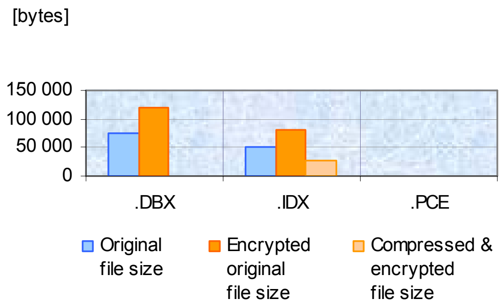


Fig. 9b.

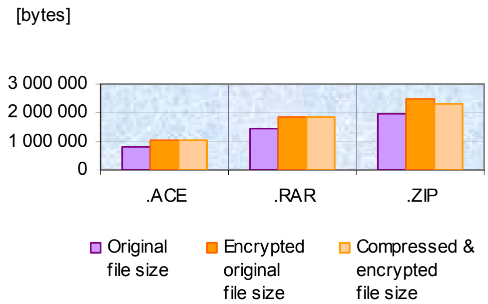


Fig 9c.

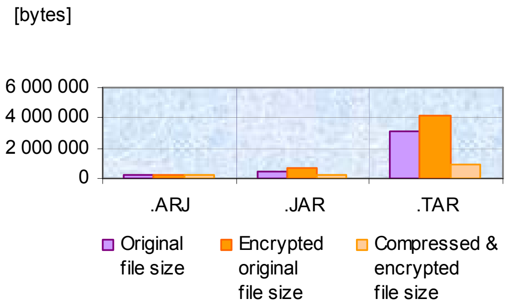


Fig. 9d.

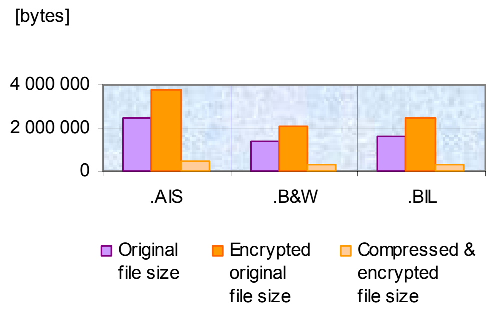


Fig 9e.

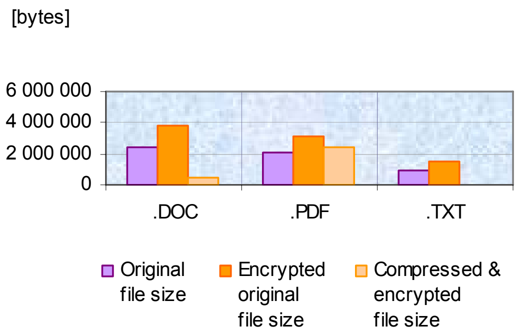
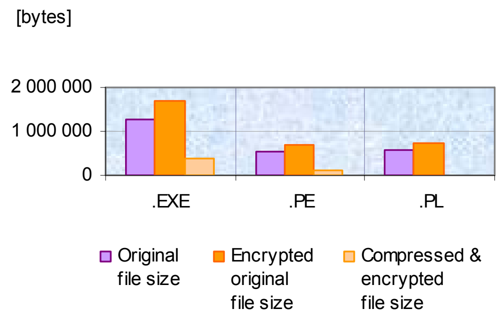


Fig. 9f.



The following assessments could be made from the experiments which were carried out:
1) The speed of the encoding process is higher if the object has been compressed before the encoding. This is due to the decrease of the amount of information for encoding after the compression.
2) The size of the resulting file decreases, if it is first compressed and then encrypted. In many cases, if the object is encoded without compression, its size is increased.
3) Some future investigations could be made in connection with the size of the password used in the encryption process and the effect of passwords on the compression process [2, 3, 4].

Conclusions

A thorough examination of the influence of some chosen parameters of information security on the methods of compression of objects is required.

It is also necessary to create a set of criteria for appraisal of the various commercial compressing and archiving programs in connection with the information security.

Bibliography

[1] Alistair Moffat, Andrew Turpin, Compression and Coding Algorithms, Kluwer Academic Publishers, 2002

[2] Ed Skoudis, Counter Hack: A Step-by-Step Guide to Computer Attacks and Effective Defenses, Prentice-Hall PTR, 2002.

[3] K. Jallad, J. Katz, and B. Schneier, Implementation of Chosen-Ciphertext Attacks against PGP and GnuPG, Information Security Conference 2002 Proceedings.

[4] Rob Shimonski, Introduction to Password Cracking, IBM Developer Works Hacking Techniques article, July 2002.

Author Information

Dimitrina Polimirova–Nickolova – National Laboratory of Computer Virology – BAS
1113 Sofia, Acad. G. Bonchev Str., Block 8, Office 104; poly@nlcv.bas.bg

REALIZATION OF OPEN ADDRESSING HASH TABLE IN THE CHAINED ALLOCATED MEMORY

Valentina Dyankova and Rositza Hristova

Abstract: In this article, we examine a realization of an open addressing hash table in the chained allocated memory, giving us the opportunity to decrease the number of linear probing when a given element has not been inserted in the table

Keywords: open addressing hash table, collision, search for an element, deleting of an element

Introduction

The extraction of the particular piece or pieces of data from a previously stored huge volume of data is a fundamental operation called searching. It is an undividable part of a set of real tasks. Usually, the goal of searching is to gain access to the data, which is contained in an element, and to move on the next process or task. Applications of the searching method are widely distributed and they include different sets of operations. Hash table is a fundamental and widely used data structure implementing fast searching algorithms. At the realization point of the structure, different approaches could be used to adapt in a better way the requirements of the speed efficiency, used memory, etc.

In the current article, we examine a realization of an open addressing hash table in the chained allocated memory. It gives us the opportunity to reduce significantly the number of linear probing when we determine the fact that a particular element is not included in the hash table as well as speed up the resulting in success process of searching.

Open Addressing Hash Table Concepts

A hash table represents an aggregation of elements, each of which has a key (identification part) and a body (data part). The value of the key identically differ it from the rest of the elements. From an organizational and processing view, the values of the elements' data part in the hash table are not of primary importance. That is the reason why we do not examine a concrete defined type of the proposed hash table. The access to an element in the hash table is implemented by transforming the element's key in its address. Consequently, searching could be presented as an image $hash: K \to A$, called hash table. Because of the representation of the hash table in the memory, the transformation of the key is taken down to transforming the index of the array. In this way, if N is the total number of elements in the array, then $hash: K \to$ *{0, 1, 2, …,* N *-1}*

Choosing a well-functioning and efficient hash table is a guarantee of uniformly distributed elements of the hash table in the array, but that is not a purpose of this article. The classic hash function -- $hash(k) = k\%N$, where $k \in K$ and k is the element key – will be used without considering the fact that elements' keys are natural numbers.

In this way, the power of set K is greater than the possible number of elements in the array and this might cause the situation where two elements with different keys, $k_1 \neq k_2$, pretend to occupy the same location in the array, $hash(k_1) = hash(k_2)$. Such elements are called synonyms, and the phenomenon – collision.

The problem of solving collisions has different solutions, but from all of them, we will examine the method of linear open addressing. In this method, if an element pretends to be placed at a position that has been already occupied by another element then the array is scanned in order for an open position. Sequential search would have been realized with the function $rehash(i) = (i+1)\%N$. The effect that takes place when solving the collisions in the open addressing table is called primary clustering. The elements that cause the collision (with the same value of the hash function) are located sequentially with respect to the order of their entries. Then the elements that are about to be placed in the array also cause collisions since their original place in the array has been already taken.

The case in which elements with different assigned values from the hash function pretend to the same index of the array is called secondary clustering. The gathered roles of elements in the hash table are called clusters.

The basic operations with hash tables and standard algorithms for their implementation are:

Searching for an element with key k – the array index a_1 under which an element must be found is calculated with the help of the hash function $hash(k) = a_1$; but if there is another element with the same index, the calculation for a position is done by the statement $a_2 = rehash(a_1)$. Positioning of a_2 is similar. This process of linear probing continues till: an element with key k is found (successful end); an element with a key equal to null is found (unsuccessful end), or an element with a particular key does not exist (unsuccessful end) when the hash table has been scanned.

Inserting an element with a key k – if it has been determined that an element with such key is not in the hash table, then it is inserted.

Deleting of an element with key k – the element's key is given a null value at position i after a success is returned from executing the operation, searching of an element with key k, and determining its location at position i.

Problem Solving

In the classic literature, the question about deleting an element from an open addressing table is either not mentioned [Амерал, 2001], [Рейнголд, 1980], [Мейер, 1982], or mentioned in one of the following ways:

- The implementation of the operation is possible, but it is very complex [Амерал, 2001].
- Marking of an element by the obvious method breaks the chain of synonyms [Наков, 2002], [Шишков, 1995].
- Entering a specification with tree conditions: the element is filled; the element is empty, and it has never been filled; the element is empty, but has been previously filled [Шишков,1995], [Смит,2001], [Sedgewick,1998].
- The chained synonyms in the array are rearranged, so that the elimination of an element has no effect on the searching or inserting algorithms [Шишков, 1995].
- Secondary hashing all elements between the deleted element and the next available position [Sedgewick,1998].

Two implementations of deleting an element are examined -- the obvious method, [Азълов, 1995] and secondary hashing method, [Sedgewick, 1998].

So, the way the searching operation has been given to us and our previous comments arise the following questions:

- How does deleting of an element from the chained linear probes reflect on element, which is a member of this chain?
- Does the case that a null key in consecutive linear probes is being reached give us the opportunity to state that the element in interest is not found?
- Is it necessary to search the table until a never filled element is reached in order to be determined that the element in interest does not exist?

Example

An illustration of the raised question is the following example: in the hash table of size 13 are inserted the elements with keys

- *14 (hash(14)=14%13=1);*
- *16 (hash(16)=16%13=3);*
- *29 (hash(29)=29%13=3, rehash(3)=4%13=4);*
- *55 (hash(55)=55%13=3, rehash(3)=4%13=4, rehash(4)=5%13=5);*
- *21 (hash(21)=21%13=8);*
- *35 (hash(35)=35%13=9);*
- *49 (hash(49)=49%13=10);*
- *50 (hash(50)=50%13=11).*

0	1	2	3	4	5	6	7	8	9	10	11	12
0	14	0	16	29	55	0	0	21	35	49	50	0

Applying the widely used in the literature algorithm for searching elements with keys *14, 20, 55, 42, 48*, the following results and questions are obtained:

- About *14:* index 1 is received by using the hash function. It stores the key that we are looking for;
- About *20:* index 7 is received by using the hash function. It stores the null key. This shows that the place is empty and the result of searching is unsuccessful.
- About *55:* Applying the hash function, we receive index 3, where is stored key $16 \neq 55$.. Following the classical algorithm (linear probing is executed until an element with a given key is reached or en empty space in the array), we apply the method of linear probing twice to receive the key *55*.
- About *42:* Applying the hash function, we receive index 3, where is stored key $16 \neq 42$. Exercising the method of linear probing three times, we reach key 0. Searching ends up unsuccessful.
- About *48:* Applying the hash function, index 9 is received. It stores a key $35 \neq 48$. Using the method of linear probing three times we receive key 0. It is an unoccupied space and searching ends up unsuccessful. This case raises the following question: Is it possible at the time of receiving index 9 to conclude that searching exits with failure. The latter follows from the fact that collision did not occur at position 9 as we continue to insert the elements. So there is no doubt that the element we are searching for will not appear in the consecutive linear probing for this position.

An illustration of the questions 2.1 and 2.2 is the situation when we search for element with key *55* after we have previously deleted element with key *29.* Locations of the elements after the deleting are:

0	1	2	3	4	5	6	7	8	9	10	11	12
0	14	0	16	0	55	0	0	21	35	49	50	0

Then, applying the searching algorithm for element with key *55* and going through the elements, index 4 is reached. It stores key 0 i.e. the location is empty or according to the classical algorithm (searching until a given key is found or an empty location is reached), searching would have finished unsuccessful. This contradicts the fact that collision occurred at position 9 and if it has not contained the key of our interest, then we continue with the linear probing until we reach the key of interest (successful end) or to position at which no collision has occurred (unsuccessful end)

Solution to the Problem

The basic problem to which we offer a solution in this article is the implementation of a searching algorithm for an element inserted in open addressing hash table. The algorithm reduces the number of linear probing as long as a particular element has not been inserted in the table. Excluding an element from the table has no harmful effects on the algorithms for searching and inserting an element in the inner chains of synonyms when collisions are being solved. On the other hand, the use of the inefficient "garbage collector" is not necessary.

The current article gives a solution to this problem, as data (recording occurring of a collision at a particular place when we inserted the elements in the hash table) is stored for each position. This leads to the idea of using alternative array *ph* from Boolean values:

$$ph[i] = \begin{cases} true \\ false \end{cases}$$

Then, the location of the examined elements above will modify both arrays:

- *14 (hash(14)=14%13=1):*

	0	1	2	3	4	5	6	7	8	9	10	11	12
t	0	14	0	0	0	0	0	0	0	0	0	0	0
ph	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*

- *16 (hash(16)=16%13=3):*

	0	1	2	3	4	5	6	7	8	9	10	11	12
t	0	14	0	16	0	0	0	0	0	0	0	0	0
ph	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*

- *29 (hash(29)=29%13=3; rehash(3)=4%13=4);*

	0	1	2	3	4	5	6	7	8	9	10	11	12
t	0	14	0	16	29	0	0	0	0	0	0	0	0
ph	*false*	*false*	*false*	*true*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*

- *55 (hash(55)=55%13=3, rehash(3)=4%13=4, rehash(4)=5%13=5):*

	0	1	2	3	4	5	6	7	8	9	10	11	12
t	0	14	0	16	29	55	0	0	0	0	0	0	0
ph	*false*	*false*	*false*	*true*	*true*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*

- *21 (hash(21)=21%13=8):*

	0	1	2	3	4	5	6	7	8	9	10	11	12
t	0	14	0	16	29	55	0	0	21	0	0	0	0
ph	*false*	*false*	*false*	*true*	*true*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*

- *35 (hash(35)=35%13=9):*

	0	1	2	3	4	5	6	7	8	9	10	11	12
t	0	14	0	16	29	55	0	0	21	35	0	0	0
ph	*false*	*false*	*false*	*true*	*true*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*

- *49 (hash(49)=49%13=10):*

	0	1	2	3	4	5	6	7	8	9	10	11	12
t	0	14	0	16	29	55	0	0	21	35	49	0	0
ph	*false*	*false*	*false*	*true*	*true*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*

- *50 (hash(50)=50%13=11):*

	0	1	2	3	4	5	6	7	8	9	10	11	12
t	0	14	0	16	29	55	0	0	21	35	49	50	0
ph	*false*	*false*	*false*	*true*	*true*	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*false*

At these current states of both arrays, searching of an element with a key *48* will lead to application of the hash function before key *48 - hash(48)=48%13=9.* This location is not occupied by the element with the key of interest and $ph[i] = false$, so that applying the method of linear probing is not necessary and the conclusion for an unsuccessful searching end could be drawn.

Program Implementation

Implementation of the proposed solution (language C++|):

```
const int nilkey=0;
template<class T>
struct element {int key; T info;};
template<class T>
class hashtable
{ private:
      int tabsize;
      int free;
      element* t;
      bool* ph;
  public:
      hashtable();
      hashtable (int n);
      bool is_full();
      int search (int k);
      void insert (element e);
      void del (int k);
};
template<class T>
hashtable<T>::hashtable()
{ tabsize = 0; free=0; }
template<class T>
hashtable<T>::hashtable(int n)
{ tabsize = n; free=n;
  t = new element[tabsize];
  ph = new bool[tabsize];
  for (int i=0; i < tabsize; i++)
      { t[i].key=nilkey; ph[i]=false; }
}
int h(int k)  { return k%tabsize; }
int r(int i)  { return (i+1)%tabsize; }
template<class T>
bool hashtable<T>::is_full()
{ return free==0; }
template<class T>
int hashtable<T>::search( int k)
{ bool b=false; int i=h(k); int j=i;
  while ( t[i].key!=k && ph[i] && !b )
        { i=r(i); b = i==j; }
  if (t[i].key==k) return i;
  else return –1;
}
template<class T>
void hashtable<T>::insert( element E)
{ int i=search(E.key);
  if ( i<0 && !is_full() )
       { i=h( E.key);
         while    (    t[i]!=nilkey    )    {
ph[i]=true; i=r(i); }
         t[i]=E; free--;
}      }
template<class T>
void hashtable<T>::del( int k)
{ int n=search(k);
  if (n>=0) { t[n].key=nilkey; free++; }
}
```

Result Analysis

The number of linear probing in the open addressing table at the time the search is done depends on:

1. The ratio $\alpha = M/N$, where M is the number of the stored elements in the table vs. N , the total number of elements in the table. In the incomplete table (small α), it is expected most of the searches to end up in several probing. In contrast, when the table is almost complete (α has a value close to *1*) , searching could require a big number of linear probes.
2. The way of generating clusters in the hash table. The observation shows that the average number of linear probing resulted in unsuccessful searching is proportional to squares of clusters' length. The successful searches are always cheaper (less in number probes) than the unsuccessful ones.

The grades on number of linear probing resulted in successful search (1) and unsuccessful one (2) are given by D. Knut [Кнут, 1978] when the following stages are performed:

- Grades are pessimistic and based on the fact that an element's key k can appear at any moment.
- Grades loose their precision when α is close to *1*.

$$S(\alpha) = \frac{1}{2}(1 + \frac{1}{1-\alpha}) \quad (1) \qquad\qquad U(\alpha) = \frac{1}{2}(1 + \frac{1}{(1-\alpha)^2}) \quad (2)$$

Table 1 gives us different values of α in order: average number of probes necessary to successfully find an element using the formula (1) that calculates $S(\alpha)$; the average value received by implementing the classic algorithm in the program and the average value received by the given algorithm in this article.

N	*α*	0.25	0.50	0.66	0.75	0.83	0.90	0.95
	S(α)	1.167	1.5	1.971	2.5	3.441	5.5	10.5
1009	Classic algorithm	1.132	1.325	1.566	2.217	3.153	5.31	8.739
1009	Proposed implementation	1.132	1.325	1.566	2.217	3.153	5.31	8.739
10007	Classic algorithm	1.126	1.419	1.682	2.095	2.751	4.168	7.047
10007	Proposed implementation	1.126	1.419	1.682	2.095	2.751	4.168	7.047
100003	Classic algorithm	1.121	1.367	1.725	2.123	2.751	4.318	8.257
100003	Proposed implementation	1.121	1.367	1.725	2.123	2.751	4.318	8.257

Table 1.

We see that when the keys are equally distributed, the results are expected to be better than the average possible. We have proved that both methods' efficiency is the same.

Using formula (2), the grade we receive is more expensive (average number of probing is greater) than the grade received by using formula (1). The suggested algorithm improves the expected number of probes when the element with a particular value for us is not in the hash table. Table 2 represents the different values of α in order: average number of probes necessary for unsuccessful searching in the hash table and calculated using formula (2); the received result is interpreted as a mean value of the program implementation using the classic approach and the mean value generated by the examined algorithm in this article.

N	*α*	0.25	0.50	0.66	0.75	0.83	0.90	0.95
	U(α)	1.389	2.5	4.825	8.5	17.80	50.5	200.5
1009	Classic algorithm	1.368	2.324	4.169	7.655	19.858	39.836	177.844
1009	Proposed implementation	1.045	1.289	2.231	5.236	16.123	34.688	163.653
10007	Classic algorithm	1.362	2.40	4.071	7.335	16.929	34.466	119.056
10007	Proposed implementation	1.033	1.417	2.13	4.409	12.368	26.784	97.504
100003	Classic algorithm	1.381	2.354	4.322	7.127	14.018	39.283	175.473
100003	Proposed implementation	1.042	1.367	2.403	4.247	9.207	30.851	151.717

Table 2.

We see from Table 2 that the suggested implementation influences the number of comparisons when searching is unsuccessful. When α is closed to $1/2$ (that is preferable loading of the open addressing hash table), the obtained result is comparable with $S(\alpha)$.

Following tables 3 and 4 indicate the statistics resulted by counting $U(\alpha)$. Counting is done in three ways:

N	α	0.25	0.50	0.66	0.75	0.83	0.90	0.95
1009	Classic algorithm	1.366	2.008	2.834	3.968	5.668	9.854	21.72
1009	Proposed implementation	1	1	1	1	1	1	1
10007	Classic algorithm	1.322	2.043	2.923	3.963	5.72	10.622	20.245
10007	Proposed implementation	1	1	1	1	1	1	1
100003	Classic algorithm	1.347	2.003	3.009	4.159	5.783	10.355	20.889
100003	Proposed implementation	1	1	1	1	1	1	1

Table.3

a) Table 3 shows the statistics obtained when there are no collisions in the hash table. In contrast, determining of a missing key with the suggested algorithm is performed with one comparison. This much is necessary for a successful searching of an element. Look, that the values of α are closed to *1*, which is acceptable to this kind of hash table, is important to the improvement of the classic algorithm.

b) Table 4 contains statistics showing that the set of keys K exceeds 10 times the size of the hash table N and the keys are normally distributed (the average length of the chains of elements is less than 10 elements). In this way, finding of a missing key with the suggested algorithm of loading the table $\alpha < 0.85$ (preferable for open addressing tables) is accomplished by less than $1/3$ linear probing.

N	α	0.25	0.50	0.66	0.75	0.83	0.90	0.95
1009	Classic algorithm	1.388	2.466	4.396	6.206	13.624	32.008	158.326
1009	Proposed implementation	1.034	1.366	2.492	3.778	9.004	22.446	151.369
10007	Classic algorithm	1.377	2.472	4.407	8.255	15.096	46.891	174.926
10007	Proposed implementation	1.052	1.461	2.48	5.55	10.206	36.688	143.522
100003	Classic algorithm	1.390	2.413	4.606	7.88	15.509	45.809	196.747
100003	Proposed implementation	1.049	1.427	2.702	4.929	10.819	36.568	182.90

Table.4

Conclusion

Suggested implementation of hash table in the consecutive allocated memory can be used with any hash functions and any way of processing collusions. In any case, the number of probing is reduced drastically when the result of searching an element is unsuccessful. The operations, adding or deleting of an element, directly or indirectly perform searching of an element and their speed is also improved. Additional advantage is that the use of “garbage collector” is not necessary in deleting an element. The proposed application of the algorithm for linear searching is preferable to applications using frequent execution of the operation unsuccessful search and capricious of used memory.

Literature

[Азълов, 1995] П.Азълов. Програмиране. Основен курс. Изд. АСИО, София, 1995.
[Амерал, 2001] Л.Амерал. Алгоритми и структури от данни в С++. ИК СОФТЕХ, 2001.
[Careson] D.Careson. Software Design Using C++. An Online Book, http://cis.stvincent.edu/swd
[Кнут, 1978] Д.Кнут. Искусство програмирования для ЭВМ, т.3. Сортировка и поиск. Изд. Мир, Москва, 1978.
[Мейер, 1982] Б.Мейер, К.Бодуэн. Методы программирования. т.2. Изд. Мир, Москва, 1982.
[Наков, 2002] П.Наков, П.Добриков. Програмиране=++Алгоритми; Изд.TopTeam Co., София,2002
[Sedgewick, 1998] R.Sedgewick. Algorithms in C, Part 4: Searching, Adddison Wesley, 1998
[Смит, 2001] Т.М.Смит. Принципи и методи на програмирането с PASCAL. Изд. Техника, София, 2001
[Рейнголд, 1980] Э.Рейнголд,Ю.Нивергельт,Н.Део. Комбинаторные алгоритмы. Теория и практика. Изд.Мир, М., 1980
[Шишков, 1995] Д.Шишков и др. Структури от данни. Изд.Интеграл, Добрич, 1995

Authors' Information

Valentina S. Dyankova – e-mail: vspasova@yahoo.com
Rositza P. Christova – e-mail: r.hristova@fmi.shu.bg.net
Shumen University “Ep. Konstantin Preslavsky”, Faculty of Mathematics and Informatics; Str. “Universitetska” 115, Shumen, 9712; Bulgaria

MATHEMATICAL PACKAGES FOR TEACHING AND RESEARCH IN INTERNET – APPLICATION AND INFORMATION SUPPORT

Tsvetanka Kovacheva

Abstract. The paper considers the use and the information support of the most important mathematical Application Packages (AP), such as Maple, Matlab, Mathcad, Mathematica, Statistica and SPSS – mostly used during Calculus tuition in Universities. The main features of the packages and the information support in the sites of the producers are outlined, as well as their capacity for work in Internet, together with educational sites and literature related to them. The most important resources of the TeX system for preparation of mathematical articles and documents are presented.

Key words: Internet, mathematical Application Packages, mathematical programs, Maple, Matlab, Mathcad, Mathematica, Statistica, SPSS.

1. Introduction

The use of mathematical program systems grows in various spheres of decision making, at symbolic (analytical) mathematical calculations and in application of numerical methods. Their capacity, wide spread and mutual integrity determine the development of the Computer Mathematics. The packages Mathematica, Maple, Matlab are oriented to the solution of scientific tasks, while Mathcad is mainly for engineering problems. Statistica is directed to statistical tasks. These packages are used for research during computer experiments as well as for teaching in Universities.

This allows high efficiency of the work of the lecturers, who apply them in the training process in Mathematics and in their research activities. Many textbooks and monographs are prepared with their use. The packages are useful for students, post graduate students, engineers and experts in their everyday work.

2. Main Mathematical Packages in Internet, used for Teaching in Mathematics in the Universities

2.1. Mathematica. This is one of the best mathematical packages which has high calculation capacity and is simple in use. It was developed in 1988 by Wolfram Research [3]. Its advent gave a boost for development of contemporary scientific computations. It allows various calculations needed to solve problems in the technical and the social spheres. The version Mathematica 5 is one of the most powerful and developed in detail systems. It includes important extensions for many numerical and symbol operations on the basis of new generation algorithms. The requirement for work with the package for Windows 98/Me/NT 4.0/2000/XP is for availability of 345 MB free memory on the hard disk.

The main features of the package are:

- working environments: Windows, Macintosh, Unix, Linux.
- possibility to be used as a calculator.
- ability to operate easily within seconds with symbol equations as well as with numbers of different orders.
- by speed of calculations and volume of the processed information the system is on the top fist place.
- it has embedded programming language.
- large amount of embedded commands and special mathematical functions.
- possesses a multitude computational and own algorithms.
- operation with data with arbitrary formats (more than 50 types) and following processing by various functions.
- convenient interface with embedded text editor.
- 2D and 3D graphics, used for visualization of curves and surfaces in 3D space and possibilities for animation.
- presentation of formulae and graphs in polyfraphic lay-out.
- very good documentation with large number of applications.

- MathLink protocol – templates control the system interface, providing links with external programs written in на C, Microsoft Excel, Microsoft Word, as well as connection among the kernels of the systems of one or several computers.
- provides export of electronic (HTML) and publishing documents.

The site of the company [3] contains products data, services and resources as well as topical news and events. There is a lot of information about the last edition of the package. Students can be trained on-line by specialized mini-courses. New products of the company are outlined: web Mathematica2 – adding dynamic calculations, fast and easy visualization to the web site; gridMathematica – allows complex calculations by use of clusters, multiprocessor machines and computer networks; Mathematica Kit – allows users to develop own automated experiments, systematic study of bound systems, graphic generation for faster analysis; CalculationCenter 2 – calculation software, combining computer capacity with a simple intuitive user interface, which is a perfect tool for professionals needing fast solution of their technical problems. Best pages of the site are: Documentation Center, Mathematica Information Center, Mathematica Training , Student Center, webMathematica Examples, Wolfram Graphics Gallery. The company Newsletters are sent to the users e-mails.

2.2. Maple. The product represents a powerful computer system with expanded capabilities where the mathematical calculations are automated with an arbitrary order of complexity. The package was developed by Waterloo Maple Software in 1988 [4]. It gives opportunities for efficient solutions of Algebraic and Geometrical problems, tasks of Mathematical Analysis, Discrete Mathematics, Probability Theory and Statistics, Combinatorial Calculus, integral transformations, numerical calculations, financial mathematics, etc. The version Maple 9.5 includes many new algorithms and computational methods. It offers more possibilities for the lecturers during preparation of course materials and better understanding of mathematical and engineering notions. The well-developed programming language allows user to create independently commands and applications to solve specific tasks. The large library of embedded functions and operations accelerates the development of mathematical programs. The package maintains MathML2.0. standard. This makes the version a basic tool for Internet Mathematics and sets a new level of compatibility of the multi-user environment. TCP/IP protocol provides dynamic access to information from other Internet sites. For distant training the company has developed MapleNet version. The set of tools allows creation of interactive training applications and their distribution through Internet. In order to work in Windows® XP (Pro and Home), Intel® Pentium® II 233MHz or fully compatible environments the program needs memory 128MB, 150MB, 230MB respectively.

The main features of the package are:

- work in Windows, Macintosh, Unix, Linux environment.
- powerful programming language of 4th generation (4GL).
- symbol and numerical algorithms for solution of mathematical problems, including the numerical algorithms of NAG company.
- more than 3000 embedded functions.
- clear and convenient interface with embedded text editor.
- use of the accepted mathematical calculations.
- possibility to be used as a calculator.
- entering of spreadsheets with numbers and symbols (characters).
- avoiding rounding errors – in case of operations with fractions and roots the latter are not reduced to decimal form.
- allows conversion into LaTeX format.
- reference books for physical constants and units with automated formulae calculation.
- 2D and 3D graphics and animation which allows rotation of 3D surfaces in real time.
- presentation of formulae and graphs in polygraphic format.
- wide range of tools applications (Maple PowerTools™) and packages.
- reference system – context-dependent help, browser, topical and full text search.
- provides export in electronic (HTML) and publishing document formats.

The site of the company [4] gives information about the company and how to establish contacts, contains references to various resources of the package in Internet, allows free receipt of each edition of the electronic journal "The Maple reporter". The company maintains News and Discussion Forum. The new products are: Maple 9.5; MapleNet; Maple T.A – offering Web-based system for generation of texts, problems, automated evaluation of replies and participation of students; Precalculus Study Guide – an interactive training guide assisting lecturers in mathematics for non-standard calculations. Maple Professional Toolboxes – expands the results and functionality of Maplesoft products in specialized application areas; Third Party Products – a back-up program designed to support creativity, knowledge and ideas of the users. Information about the capabilities for interactive registration of purchased products, installation instruction of a current revised version, a jump to the needed installation file, applications, information for students and lecturers about Maple tools can be found on the user resources pages - Web Store , Application Center, MaplePrimes, Student Center, Maple for High Schools, MapleConnect, Training, Technical Support, Publications, Register Product.

2.3. MATLAB. The package MATLAB is one of the most functional and well-developed systems for science and engineering. It allows to reduce the time for analysis and development of the projects and hence their costs in the process of finding of efficient solutions. The package is developed by the company Mathworks [2] in 1980, which provides informational maintenance of the product. The version Matlab 7 includes a multitude of new functions in the programming area and possesses an efficient program code for building graphs, visualization, mathematics, data acquisition, and provides high throughput. The hardware requirements for the system are the following: Operating System – XP; Processors - Pentium III, IV, Xeon, Pentium M, AMD Athlon, Athlon XP, Athlon M; Disk Space * 400MB (MATLAB ONLY with Help)*; RAM 256MB 512MB.

The main features of the package are:

- allows procedural, object oriented and visual programming.
- works in Windows, Macintosh, Unix, Linux environments.
- allows creation of professionally complex highly-productive applications, which operate with large data arrays.
- contains tools for 2D and 3D graphics
- contains a large amount of embedded algorithms for mathematical calculations and graphical visualization.
- possesses more than 600 mathematical, statistical and engineering functions.
- the developed programs are disseminated as readable M-files.
- it has very good documentation.
- there is a possibility for inclusion to widespread office and construction programs as well as to Internet.
- contains more than 50 applications for: mathematics, analysis and design of control systems, signal processing, image processing, finances.

The site of the company [2], besides advertising and description of Application Packages launched be the company, contains references of files for updating, news with important and constantly updated information, announcements for future seminars, jobs, etc. Very useful are the archives of various programs for MATLAB, rendered to the package users. The new products include- Distributed Computing Toolbox, SimDriveline , Video and Image Processing Blockset, Filter Design HDL Coder, Fixed-Point Toolbox , RF Blockset , RF Toolbox, Simulink Control Design, Simulink Parameter Estimation .The company gives opportunity for trial of its products. The selected product is checked in a given inquiry. A password is sent to an indicated electronic address, which allows jump to the necessary installation files, which can be downloaded and used for a month. Each user can obtain free product in pdf-files, the full documentation of MATLAB and Toolboxes.

2.4. Mathcad The package MathCad was developed by MathSoft Inc. [1]. It is an interactive tool for mathematical and scientific-technical computations. Due to the embedded algorithms, many mathematical problems can be solved without programming. The new version of the package MathCad 12 is for networks. It has a more perfect mathematical core and additional options for storing and publishing of documents in various formats [9]. The improved productivity in problem solution allows simultaneous implementation of the task and the documentation. The integration with the new portal server Microsoft SharePoint allows archiving, control of the versions and publishing of package worksheets in the local and global networks. The requirements to a computer working with MathCad 12 are: Windows 98 SE, ME, NT 4.0 SP6, 2000 SP2, XP or better, processor Pentium 233 MHz or more, a minimum of 96 MB RAM, 256 MB or more (for improvement of productivity); CD-ROM; SVGA video card; Internet Explorer 5.5 or higher version.

The main features of the package are the following:

- works in Windows environment.
- a large number of embedded user commands and operators, functions and algorithms to solve mathematical problems, a complex of numerical and symbol mathematics (SmartMath-mode).
- own programming language Connex Script.
- natural record of the formulae.
- operation with physical value.
- 2D and 3D graphics, animation of images and possibility to create virtual physical experiments by mathematical modeling of physical experiments.
- specialized OLE objects, which allow good interaction with other engineering, graphical and business applications and data sources.
- opportunity for preparation of documents and electronic books with consequent launching in real time.

The site [1] contains company information, data about partners, jobs, company contact, news – events, latest news for work with the company, etc.; solutions - calculations management suite, designate, link with application server, support – consultations, training, technical support, software guarantees, product registration. There are brief descriptions of Application Packages and useful hints for their modernization, files with applications, extension library files with new functions, as well as service packages for system improvement. Mathcad Application Server allows implementation, control and dissemination of calculations and data by Internet, work within Internet on problems, distribution of Mathcad work documents in Internet; transition from WorkSheet to WebSheet; on-line tasks, Smart Interface, pseudo-animation and organization of knowledge control. The site has a lot of electronic books, graphs and animations developed by users.

2.5. Package integration

A trend of juxtaposition and integration of the different program packages is noted. For example Mathematica и Maple have good capacity for visual programming. Matlab includes a library for analytical calculations. Maple и Mathcad allow use of Matlab functions and practical operations with them. The system for analytical calculations Maple and the computational environment MATLAB give good possibility to conduct laboratory exercises on Mathematics, to develop course projects as well as to carry out faster various research, during solution of scientific or engineering problems. Maple is used in MATLAB for analytical transformations, while Maple addresses MATLAB for numeric calculations. Maple documents are automatically converted to LaTeX documents or HTML pages. Figures obtained by Maple and MATLAB are stored practically in all available formats. The packages are constantly updated, the apparatus is developed and the resources are increased. The advantage of the packages is the invariability of the set of main commands and of the language construction at advent of new versions. Mathcad is upgraded by a symbol processor for symbolic calculations, similar to the processor used in Maple. The package contains and activates Matlab component – the block of the mathematical system

2.6. Specialized program packages

2.6.1. Statistica

Statistica was developed in 1993 by StatSoft Inc. [6]. This package has the newest computer and mathematical methods for data analysis and visualization, data bases management and development of user applications with analysis procedures for research, education, technology, business, etc. It allows descriptive statistics, data grouping, correlations, interactive probabilistic calculator, variance and covariance analysis, discriminant and factor analysis, time series analysis, etc. Statistica 5.5 is the newest edition. The minimal computer configuration is IBM PC AT-386 SX20, 4MB RAM, VGA, Mouse, Windows 95, 20MB HDD free.

The main features of the package are:

- working environment - DOS, Windows, Macintosh
- link with other Windows applications – data is entered in the package by macros in Excel and special SQL-commands

- a powerful program language
- realizes the classical statistical methods and special data mining
- large set of data analysis procedures
- specialized modules for sociological or biomedical studies, solution of technical and industrial problems: quality control cards, process analysis, experiment design
- 2-D or 3-D (4-D type graphs), matrices and icons, animation
- realizes graphic oriented approach to data analysis and competes Mathematika in graphic capacity
- own command language SCL
- three volume documentation (3000 pages)
- supports all standards of contemporary office applications: import of spreadsheets, publication of results in Internet
- very well developed Help

The site [6] gives information about the company, news, products, projects, training – courses, seminars, consultations, etc., link with SPC-consulting portal, link to the Knowledge Portal – information about the package and books for data analysis, visualization, classification, prognoses and development of applications in various areas. The section Hot News contains information about the newest textbooks for statistics, realized by the package, presentations, projects, etc.

2.6.2. SPSS

The SPSS package of SPSS Inc. [5] became known to the scientific and business world by applications on large computers. Due to the sufficiently powerful statistical analysis it is used by statisticians-professionals as well as by scientists and lecturers in institutions and universities. It can be used also in various applications of the mathematical statistics– quality control, for example. Realizes factor, discriminant and cluster analysis, etc. The 7.0 edition has a large capacity for data management, data processing and operations with spreadsheets. This version has higher productivity, calculation speed and expanded functionality. The minimum requirements for operation of the package are: 486DX-2 or higher, 16 MB RAM.

Main features of the package:

- working environment – Windows.
- almost full set of statistical procedures (more than 60).
- high accuracy of calculation.
- simple and convenient interface.
- convenient graphic (more than 50 types of diagrams).
- well developed tools for preparation of reports.
- possibilities for interchange with other Windows applications and connection with large formats of data bases.
- this is one of the packages with largest value of the parameter power.
- export of tables and text in ASCII format
- extended number of output data exchange applications
- easy interpretation by Internet technologies
- perfection of easy training by introduction of the training facility Navigator

The site [5] contains company information, software solutions, tutorials, support, seminars, popular white papers and demos; news; vertical markets - education, financial services, survey and market research, government; technologies – analytical applications, data mining and text mining, statistical analysis and others.

2.7 TeX in Internet (type-setting of texts in Latex system and mathematical publications in Internet)

TeX is a widely spread system for editing of mathematical articles (documents) [20]. They are set up by a text editor, which produces ASCII file, by use of special keywords (commands) for text formatting (fonts, paragraphs, formulae, etc.). The initial versions of TeX were designed for free distribution and for this reason there is a lot of information for it in Internet. Most complete and constantly updated is the so-called archive CTAN (Comprehensive TeX Archive Network) [38]. Copies of this archive and site are located in web pages od various countries in English [33] and in German [34]. Most complete and useful resources of TeX in Internet in Russian are the pages of the Association of the Users of Cyrillic TeX [13], of the Cyrillic TeX server of MGU [15] and others. One of the mostly spread micro-packages is LaTeX [22] It offers facilitation to write formulae, their placement on the page, automated numbering of sections, formulae, citing, etc.

3. Internet Information Support of the Considered Packages in Internet

3.1. Internet sites linked to the program packages

Some of the relevant Internet sites are:

- *site of the company Softline* [11] – the company was established in 1993 and distributes software for research, engineering and education activities to leading companies in Russia. The software section contains information about software categories, producers and names of the packages. A catalog of courses and free seminars for work with application packages, conducted by the Training Center of the company, is included. Supports an Internet-shop for software. News related to various packages, to the company Training Center are reported. The application packages are divided into sections – Mathematics, Data analysis, Statistics, etc. Price lists of the companies are included as well.
- *educational Mathematical Site Exponenta* [7] – supports the application of the mathematical packages in education and research. The objective is the creation of unified information space for all users of scientific software and the site is oriented to students and lecturers. The site contains electronic versions of User Manual and demo-version of Maple 5 [28], two books for initial work with Matlab [28]; Electronic books on Mathcad [28]; User Manual [35] and demo-version of Statistica 5.1 [28], description of Statistica 5.5 capacity [17], archive of an electronic text book on statistics [28]. There is a large number of solved problems and examples, realized by the considered application packages. The site contains also information and announcements for conferences and seminars on respective topics, annotations of books, articles, abstracts of doctor theses, etc. There is an archive of the published scientific-practical journal "Exponenta Pro. Mathematics Applications" [18].
- *Mathematics Resources* [8] (Dr. Carol Lawrence Assistant Professor of Mathematics от North Carolina Wesleyan College) – contains Quick Reference, Computer Tools, Database Information, Electronic Journals, Lessons, Mathematics Departments, Mathematician Organisations, Pre-print Archives, Math Search.
- *Math Software* [9] - 586 math Software Science & Technology sites.
- *Research Statistical and Statistical Analysis Directory* [10] – online courses, free software and demonstration, statistical methods, a forum for the discussion of statistical methods and analysis, books.

3.2 Books and Electronic Articles

Internet contains a lot of papers and books dedicated to the considered program packages and their application for solution of mathematical problems, such as [12, 14, 16, 19, 20, 21, 23, 24, 25, 26, 27, 29, 30, 31, 32, 36, 37].

4. Conclusion

Various problems can be solved with the help of the considered mathematical packages, such as: mathematical investigations requiring calculations and analytical deriving, development and analysis of algorithms, mathematical modeling, computer experiment, data processing and analysis, visualization, scientific and engineering graphics, development of graphical and computational applications.

The choice of a product is defined by the specifics of the problem and application area, while the volume and the price of the program packages plays a significant role. The requirement to cover a larger part of the syllabus of mathematics is of significant importance for Universities. The application of the packages contributes the students to acquire habits to operate with ready application mathematical packages and programs. Their qualification is improved, which allows selection of optimal solution method requiring minimal time to solve the problem, to interpret it correctly and to visualize the results.

References

Sites of producers of mathematics software:

1. Company Mathsoft (Mathcad) www.mathsoft.com
2. Company Mathworks (Matlab) http://www.mathworks.com
3. Company Wolfram Research (Mathematica) http://www.wolfram.com/
4. Company Waterloo Maple Software (Maple)http://www.maplesoft.com
5. Company SPSS Inc. www.spss.com
6. Company StatSoft Inc.(Statistica) www.statsoft.ru

Mathematical Education Sites – dedicated to application of mathematical package in education and science:

7. Educational Mathematical Site Exponenta http://www.exponenta.ru/
8. Mathematics Resources http://faculty.ncwc.edu/clawrence/resources.html
9. Math Software http://www.cbel.com/math_software/
10. Research Statistical and Statistical Analysis Directory – http://www.statistics.com
11. Softline http://www.softline.ru

Articles and books in Internet:

12. Applied program support http://kek.ksu.ru/eos/ppp/ychebnik.html
13. Association of the Users of Cyrillic TeX http://www.cemi.rssi.ru/cyrtug/
14. Aivazjn A., V. Stepanov "Program support of statistic data analysis" http://www.cemi.rssi.ru/rus/publicat/e-pubs/ep97001t.htm
15. Cyrillic TeX server of MGU http://tex.msu.ru/
16. Common characteristic on the market of applied program support http://kek.ksu.ru/eos/ppp/Vvedenie.html
17. Description of the package Statistica 5.5 http://www.exponenta.ru/soft/Statist/statistica5_5/index.asp
18. "Exponenta Pro. Mathematics in applications" http://www.exponenta.ru/journal/).
19. German Scientific Computing http://www.scicomp.uni-erlangen.de
20. Govoruhin V., Tsibulin V "Computer in Mathematical Studies: Maple, MATLAB, LaTeX. Training Course", Peter, 2001. (in Russian) www.math.ru/mexmath/kym/book2/
21. Introduction http://old.sgu.ru/users/matmodel/tutorial/tutor7/introduction.htm
22. LATEX http://www.latex-project.org/
23. List of the literary resources and Internet – links of statistics http://www.tolcom.ru/kiril/Library/Book1/content999/content_L.htm
24. Maple on the Web http://andrey-ts.narod.ru/Maple/wmaple3.html
25. Mathcad http://twt.mpei.ac.ru/ochkov/VPU_Book_New/mas/
26. Mathcad-manual on higher mathematics http://www.exponenta.ru/soft/mathcad/learn/learn.asp
27. Mathcad & Maple - Units http://twt.mpei.ac.ru/ochkov/Unit_MC_MP/Unit_MC_MP.htm
28. "Mathematical package Maple V Release 4: User Manual", Demo-version of Maple V Release 4, Demo-version of Statistica 5.1, Electronic text book on statistics, Electronic books on Mathcad, "Begining with work on Matlab", "Matlab: Simulink&Toolboxes" http://www.exponenta.ru/educat/free/free.asp#statistica1
29. Online courses http://www.statistics.com/
30. Program support for research (packages Maple and Matlab) www.math.rsu.ru/metmat/kym/MME/courses/prog/
31. Steinhaus S. Comparison of mathematical programs for data analysis http://www.exponenta.ru/educat/free/compare/ncrunch.pdf
32. Technologies for Education http://www.insead.fr/Encyclopedia/Education/Advances/Technologies/
33. TEX ftp://ftp.chg.ru/pub/TeX/CTAN/, ftp://ctan.math.utah.edu/, ftp://cis.uniroma2.it/tex
34. TEX http://www.dante.de/software/ctan/
35. User Manual of Statistica 5.1 http://www.exponenta.ru/soft/Statist/stat5_1/1/1.asp
36. User Manual of Mathcad 6.0 and Mathcad PLUS 6.0 http://www.exponenta.ru/soft/Mathcad/UsersGuide/0.asp
37. Web site - system Mathematica http://integrals.wolfram.com/
38. Welcome to CTAN: the Comprehensive TeX Archive Network http://www.ctan.org

Author Information

d-r Tsvetanka Kovacheva – Department Mathematics of Technical University, 9010 Varna, 1 Studentska str., e-mail: Tsetska.Kovacheva@tu-varna.acad.bg

EXTENDED EXECUTIVE INFORMATION SYSTEM (EEIS)

Todorka Kovacheva

Abstract: In the following paper a new class of executive information system is suggested. It is based on a selforganization in management and on a module modeling. The system is multifunctional and multidisciplinary. The structure elements of the system and the common features of the modules are discussed.

Keywords: Extended executive information system, evolution management, module modeling, conflict resolution, data warehouse, selforganization, NLP, reality games, agent tree, multiagent system.

Introduction – the Need of a New Class of Information System

The enterprise management is performed by people, taking positions at different levels of management hierarchy. At the top level are the so-called top-managers (See figure 1). They bear the whole responsibility for realizing the management process from setting the goals to their realization. They are forced continually to take decisions, some of them in situations, critical for the enterprise.

The top-managers deal with large volume of information, caused by the constantly circulating internal and external information streams. They control the realization of a number of activities such as:

- goal definition and planning;
- firm resource integration;
- selection and motivation of the staff;
- execution of the planned tasks;
- management of financial funds and other activities, related to the process of management.

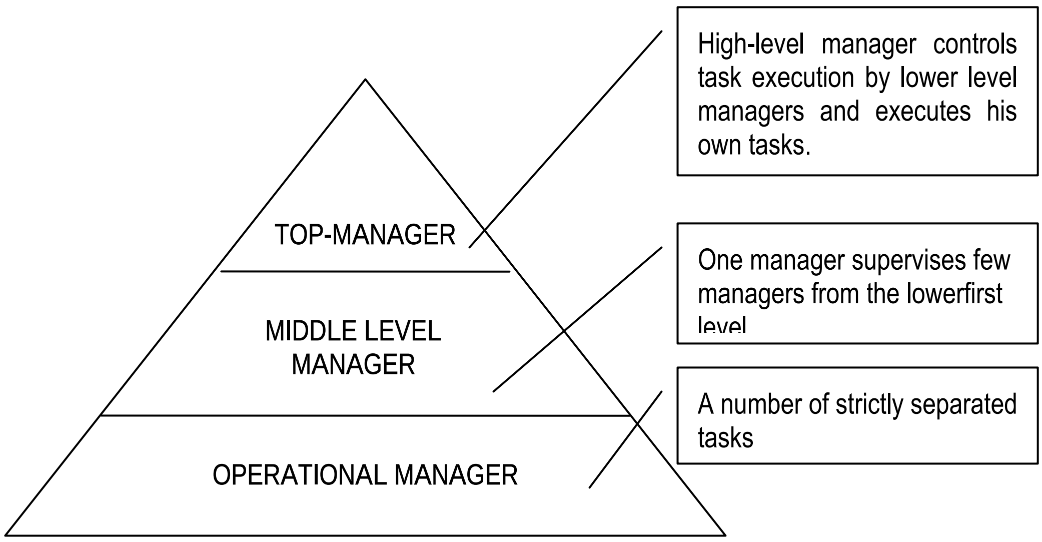


Figure 1: Hierarchy of Management

High-level managers need to be informed for the actions and the intentions of the competitors and for the market trends, as well as to follow the technological evolution of the society, in order to implement at the right time the new technologies in business. Everyone involved in such activities needs to be a narrow specialist in his own field. Top-managers should possess all these abilities, otherwise they will not be able to execute an effective control and management.

Managers do not have free time. They deal with huge amount of information but the human mind does not have the capability to operate with it at this point of evolution. There will always be a risk of missing an important for the enterprise information and some possibilities.

All this increases the tension in managenr's job and leads to a constantly high stress level. The stress decreases the quality of the enterprise management. The process of management depends on people, which are biochemical systems where flow specific biochemical processes. The number of infarcts, apoplexies and other disorders connected with the stress grows up. Top-managers have no time for personal life and they could not have an adequate rest. Every absence off or disturbance in their work will force them not only to handle the actual information but also the accumulated one too.

The main problems of the top-managers can be defined as follows:

- the managers does not have free time;
- huge volumes of information to be operated;
- a risk to miss an important information;
- high stress level;
- decrease of the quality of the enterprise management;
- health problems.

Considering all mentioned above we can assume that every inovation in the field of management must be directed to supporting manager's activities, increase the free time of the managers, improve the management quality and decreasing the stress levels in daily round. This leads to the development of a new class of systems for top-managers based on selforganization and new models of management. They must be adapted to management levels and imitate the thinking process of humans mind. Such system, called extended executive information system, is suggested below.

Common Features of the Extended Executive Information System (EEIS)

Doing their activities managers make operational, tactical and strategical decisions every day. These decisions are better when they exactly correspond to the goals and ambitions of the management subjects and help to realize those ambitions for a minimum time and with small investment and resources. This way the enterprise development is better controled and thus the corporative goals will be achieved.

The choice of the manager, represented as a management decision, is a process, which directs the object's behaviour to the chosen direction and changes its present condition to a new, much perfect, future one, improving it and as a result the object evoluates. The managers take decisions through the whole management process. The last one represents a combination of common management functions, such as: goal definition, planning and organization, motivation, control and regulation [Hristov, 1997]. No one from these functions can exist without the others and no one is more significant than the others. They need to be integrated in a united management system. This will contribute for choosing the adequate tools, methods and technologies for achieving the corporate goals.

The extended executive information system has to cover all of the management stages and assists managers in the process of making their decisions. It includes goal definition, knowledge of the object of management, modeling, creation of an ideal model, evaluation of the model, development and implementation of the strategy, elimination of the errors, implementation of the renovated strategy, evaluation of the results and repeating the circle until the goals are achieved. The next step is stabilization of the system and decisions for its future development.

1.Goal definition.

It is the first and main stage of the management process. It determines the direction of the enterprise development. The goal must be specified and clearly defined. It must not contain any contradictions. But the goal definition is not enough for its realization. Many initiatives are needed. The most important one is the evaluation of the goals which includes the definition of the degree for its realization, the time needed, etc. This is quite important since the chosen goal might not be achieved.

The EEIS in this stage has to support a *module for the evaluation of the goal*. It includes some criteria and is connected with the other modules. Some of this criteria can be taken from the neuro linguistic programming

(NLP). NLP is a new direction in psychology, differentiate as an independent science. It helps to answer to the question such as:"Could the goals be reached?", "Does its realization depend only on its own resources or on external factors?", "What are the benefits and the disadvantages?", "Do the choosed goal contradict to other corporate goals?" etc. The careful and prompt goal formulation increases the opportunity of diverting the intentions into adequate action needed for their realization. [Older, 2000]. NLP is not only useful as far as the goal definition is concerned. It could be implemented to the whole process of management. That is why it is to be used more frequently in economical theory as well as practice. Its main components are:

- *Neuro* – based on investigation of the way of work of neuro processes;
- *Linguistic* – verbal and noverbal presentation;
- *Programming* – thinking and behavior models [Older, 2000].

2.Knowledge of the object of management.

Every object can be represented as a system. "A system is defined as an integrated set of components, or entities, that interact to achieve a particular function or goal…Systems are composed of interrelated and interdependent subsystems." [Schultheis, 1992, p.31]

At this stage it is important to gather as much information about the management object as possible and to define the elements, subelements and interactions between them. This is the basic system reasearch. It helps to determine some definite dependences, behavioural trends, principles actions etc. This is the moment most errors are made. The reason could be either important factor ignorance or extra factors taken into consideration. The result at the end than is unrealistic. The level of detailization must be defined very carefully. The software realization of this stage can be represented by the *information module*.

3.Modelling.

Managing the object requires modeling. Through the model the information about its elements and subelements is viewed as an easy to understand form. Some alternatives of development and change are outlined. At this stage the software must help to construct the management model. It must be universal and adapted for modeling of different kinds of systems – economical, social, biomedical etc.

The module "Modeling" is based on a model scheme which can be used in different situations and stages of time and for the research and management of heterogeneous objects. It functions as a template for building models describing them in details and in an easy to manipulate form (See figure 2).

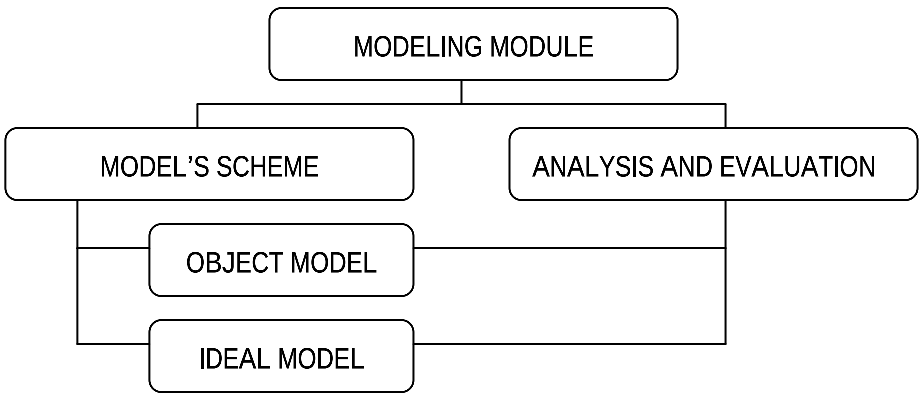


Figure 2: The elements of the modeling module

4.Creation of the ideal model..

Defining the values of the parameters which describe the required final state of the system.

5.Evaluation of the model.

At this stage the strong and weak aspects of the model are analyzed. The current is compared with the wished one. The elements and subelements which must be changed are defined as well as those which could not be modified. The corporate opportunities are clarified and the managers can see is there any potential, not completely used.

The activities in points 4 and 5 are included in module “Modeling”.

6.Development of the activity strategy.

A system of activities is created, so that the corporate goals are achieved. It is based on a model analysis by comparison of two stages (current and ideal) as well as on different analytical techniques. They help to see the results of the managers' decision and the number of possible alternatives.

This stage covers the management function – planning. It is represented by the ***module “Planning”*** which includes techniques for developing a business plan for the enterprise activities as well as many other plans with different levels of detailization.

7.Implementation of the strategy.

A step by step realization of the developed plan in point 6. This is the time for recruitment of workers and their work motivation. At this stage the model is checked in real conditions and the errors can be determined. The result is also evaluated. It covers the management functions: organization, motivation and control.

The modules needed are as follows:

- The *module “Organization and control”*. It controls the step by step realization of the plan, and includes control points and gives signals if an error occures.
- The *module “Staff motivation”* is also very important. It is based on a system of activities, comprising sanctions and rewards, as well as personnel stimulation. It contains techniques for evaluation of the personnel work, and definition of the level of individual development, the need for a change in the salary or performed position, additional education, etc. The presense of this module as well as the module for conflict resolution (which is described bellow) is a precondition for effective human resource management, and can also be applied in large enterprises with thousans of workers.

8.Elimination of the errors.

Changes in the model and in the strategy could be made if nessesary. It covers the management function regulation. Overcome the difference between the planned and actual indexes.

At this stage the *module “Regulation”* is used. Its purpose is to offer a variety of alternate decisions for eliminating the errors with all the arguments leading to the choice.

9.Implementation of the renovated strategy.

It covers the management functions: organization, motivation, control and regulation.

10.Evaluating the results and repeating the steps from 1 to 9 until the requirements are fulfilled.

The transition to the new state of the management objects is made.

11.Status stabilization and making decision for the future enterprise development.

Now it must be checked if there is a stable structure. On this basis the new decisions about the enterprise development are made.

Conflict Resolution

Another very important module should be included in the system - *the module “Conflict resolution”*. Its function is to define in advance the possible conflict zones and points in the enterprise and to make them known. On this basis the conflicts are resolved before realy to appear. The module contains identification tools of warning signs for conflicts and also instruments for their non-admission.

A *module for simulations (Reality games)* can be build. With its help the possible situations as a rezult of a management decision can be foreseen. Through the same principle an interactive education for conflicts management could be made.

Every enterprise consists of people who manage it and people who are managed. With their united efforts they reach the corporate goals, working in harmony. There are different kinds of relationship and interconnections between them are. The managed individuals can also influence the managers and their decision.

As much as the purposes and interests of the managed and managers coinsides as less unconflicts there will be in the enterprise activity. The contradictory goals can provoke a conflict. The zone of correspondences must be defined in the goals, interests and abilities of the both sides of the management process (See figure 3).

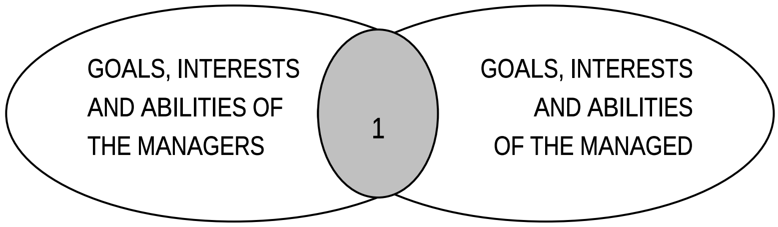


Figure 3: The zone of correspondences

If the corporate goals are in zone 1 (See figure 3), their realization will be maximum effective. The coordination between these two groups may be achieved with the help of many psychological and financial activities as well as with the proper selection of the staff. This way, by means of some manipulation techniques the conflicts can be decreased to the required minimum. It is nesessary because of the fact that if they are not too many, they could provoke the constant aspiration of self-perfection. This can be the reason for corporate growth and selforganization. The good knowledge of the conflict zones is important.

The software application developed for this purpose must include a large volume of information, which must be structured, compared and analysed. As a result the software has to determine the potential conflict zones, the zones of correspondences, as well as to make suggestions for prevention and overcome of the potential conflicts. The data which could be included in the module are the following:

- Individual characteristics of the managers at the different levels – their personal interests, motivation for work, success and failures, personal experience, family, communication and organizational skills, leadership, relationships with other people, a state of health etc. as well as the individual problem zones – where and under what conditions he/she is inclined to conflicts. The same information must be provided also for the staff and for the enterprise as a whole.
- Goals of the managers, the staff and the enterprise.
- Abilities and experience of the managers and the managed. This is their personal potential. On it depends the trust of the managed to the managers and their ability to do the assigned tasks. It is very important the managers to have more abilities and to be more competent and qualified than the staff. The opposite would be a prerequisite of a frustration from workers and can provoke a variety of problems.
- The desire of personal development and self-perfection of the participants in the management process. The future development of the entire enterprise depends on it.

An individual model for each participant in the management process must be developped. In the process of work these models are updated, operated and compared. This makes possible the conflict zones to be outlined and topical information aboit the current state of the management objects and subjects to be kept. The managament decisions made this way will satisfy the needs not only of the managers but also of the managed. This will guarantee harmony and perfect functioning of the enterprise.

System Development

From the point of view of cybernetics the enterprise can be defined as a selforganizing system, which constists of managing and managed subsystems. Having this in mind the EEIS must contain these subsystems, including their functions and way of working, based on the enterpruse structure. The structure of the enterprise must contain all system elements grouped by categories (See table 1).

MAIN CATEGORIES
Human Resources
Activities
Material Resources
Departments
Communication chanels
Software
Other

Table 1: Main categories of the system – enterprise

From these structure elements can be differentiated separate modules. Subsequently they can unite and grow into an EEIS. So the first category of the table could be described from the module “Human resources” (See table 2).

Module “Numan resources”	
Subcategories	*Describtion of the subcategories*
Personal data	Address, phone number etc.
CV data	Education, working experience, interests etc. before enrolling.
Positions in the enterprise	Personal evolution.
Motivation	Why one would like to work? Why one would like to be a part of the enterprise?
Family status	A full describtion.
Health status	A detail information about past and present disorders as well as the trends and genetic predispositions
Personal abilities	Communication and organizational skills, abbility to work in a team, aggression and so on.
Potential	Is he using all his potential? How can he deal with specific tasks in such fields in which he has no experience before and has no information? Has he an analytical mind? etc.

Table 2: Module “Human resources”

In the enterprise must be created a system of criteria for an evaluation of the staff, based on a system of indexes which can be used in the EEIS. It must include also the information from the tradition software for human resource management, accounting etc.

Such level of detailization must be present in the all other mudules which build the extended executive information system. The EEIS uses the information from all available sources at the enterprise. The inplementation of the technology of data warehoising is needed to unite data from different sources and in different format. Used as a base of the EEIS, the data warehouse makes possible to keep all the available software applications at the enterprise and use them by the implementing the new technologies.

The structure of EEIS is presented at figure 4.

To build these system elements we can use the agent technology. Every structure element can be developed as an agent. The system will turn into a multiagent system where the functions of each agent are adequate to the functions of the manager which position is covered. So we build the agent tree based on the enterprise model and the system of busines rules.

The managing subsystem includes the main business rules as well as an analyzing and control tools. The whole activity of the enterprise subordinates these rules. The managing subsystem controls for the execution of the planned tasks, that are put into it. The planned indexes are compared with the existing through the channels of feed-back relationships from and to the managing subsystem. It is possible to set levels of freedom for the managing subsystem which allow it to make changes in the primary choosed goal using the data from the managed subsystem. This will turn it into a selforganizing system with its own intellect, analogous to this of the managers.

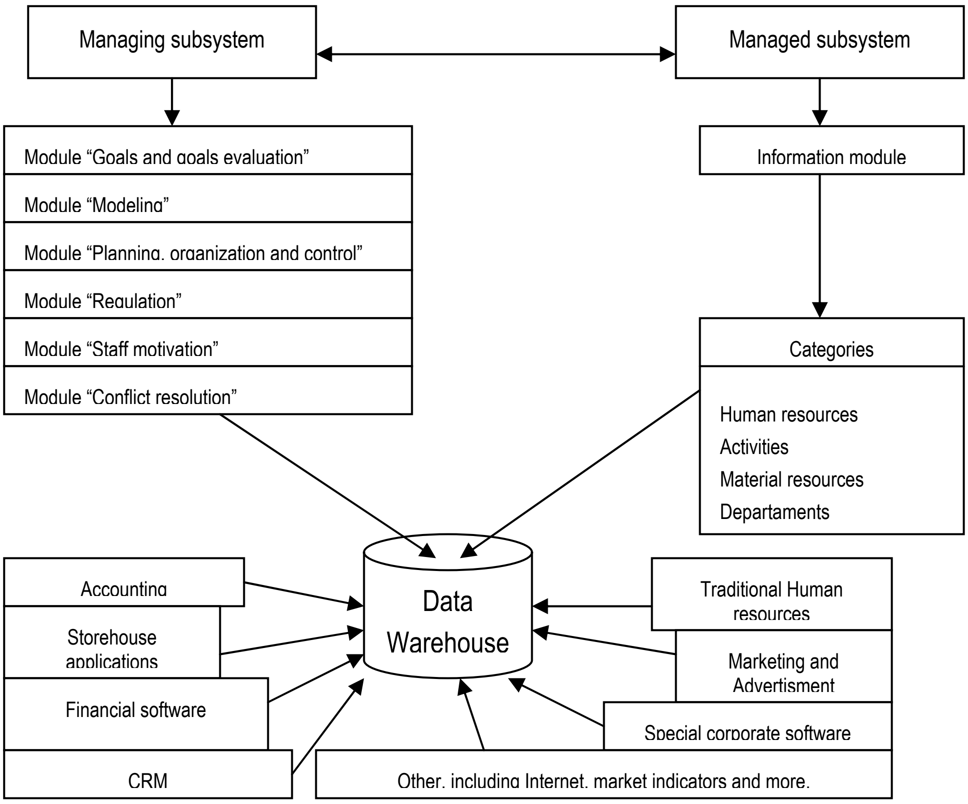


Figure 4: The structure of the extended executive information system

Conclusion

Having in consideration all mentioned above, we can make the conclusion that the Extended Executive Information System has the abbility to supersede the managers in many of their functions and to increase the quality of their work and the quantity of their free time – a resource which the managers do not have at this stage of evolution of the civization. The system is multifunctional and multidisciplinare. It is an information system of the future, based on the human thinking activity.

Bibliography

[Hristov, 1997] Христов С., Основи на управлението, ИУ – Варна, 1997, стр.7 Management basics
[Older, 2000] Олдер Г., Хэзер Б., NLP – полное практическое руководство, София, 2000, стр.37
[Schultheis, 1992] Schutheis R., Sumner M., Management Information systems (The Manager's View), IRWIN, USA, 1992

Author Information

Todorka Kovacheva – Departament of informatics, University of Economics – Varna, Bulgaria,
Corresponding address: "Feliks Kanitc" Str. 3, et.1, app.1, Varna 9000, Bulgaria,
e-mail: todorka_kovacheva@yahoo.com

INFORMATION MODELLING OF TWO-DIMENSIONAL OPTICAL PARAMETERS MEASUREMENT

Georgi Stoilov, Nikola Mechkarov, and Peter Sharlandjiev

Abstract: A method for measurement and visualization of the complex transmission coefficient of 2-D micro-objects is proposed. The method is based on calculation of the transmission coefficient from the diffraction pattern and the illumination aperture function for monochromatic light. A phase-stepping method was used for diffracted light phase determination.

Keywords: microscopy, phase-stepping method, interferometry, inverse problem in optics

Introduction

The optical characteristics of different objects are an indicator of their state (cells), production quality (optical elements), homogeneity (solutions and gels), etc. For measurement of these characteristics various laboratory and industrial measuring systems are exploited. The most common building elements are light source, imaging optical system (mirrors, objectives, beam splitters, etc.) and photosensitive (recording) device. Sometimes the utilization of an objective for the measuring system is undesirable, more expensive or even impossible. For that reason, new methods for micro-objects optical parameters calculation on the basis of information, derived in the Fraunchofer zone, are under development.

Method of Measurement

The method of measurement is based on the inverse problem in optics [1] - the object transmission function is derived from the known intensity distribution function of the illuminating beam aperture and the measured intensity distribution on a screen behind the object. It is known that the phase-stepping method [2] allows measurement of the light complex amplitude. The simultaneous use of the phase-stepping method and diffracted light intensity measurement makes possible the calculation of the transmission coefficient in a complex form. To eliminate the unknown parameters of the illumination system, information from additional reference measurements is used.

Set-up

An exemplary interferometric set-up is shown in Fig.1. The light beam from the laser source (L) is split by the beam splitter (S1) into an object beam (OB) and a reference beam (RB). The object beam passes through the investigated object (O) and the transmitted light is detected by the CCD camera (CCD). The reference beam after being reflected by a phase-stepping mirror (pH), passes through a lens (LN) and illuminates the CCD camera.

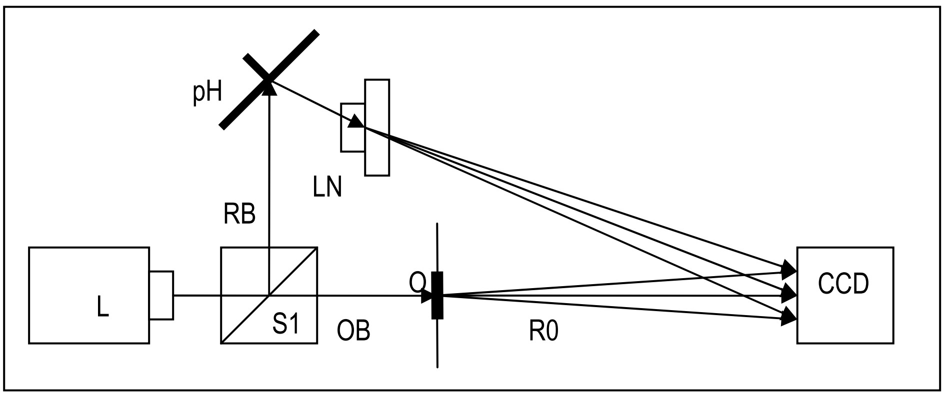


Fig. 1.

Approximations and Applicability

In this investigation the 3-D objects are approximated with 2-D objects and we suggest that the objects are thin. With this approximation we can find a single solution. For real 3-D measurements, a tomographic method could be used, based on recording a set of images while the CCD is moving towards the object.

Scattering and reflection, as optical parameters of the object, are not taken into account. These parameters can also be ignored when measuring a large class of objects..

Utilizing light with a wavelength of 0.5-1.0 μm is limited by the CCD sensor characteristics. According to Abbe's theory, using shorter wavelengths results in higher measurement accuracy.

Normally, the size of the CCD sensor is 5-10 mm. This limits the range of the usable spatial frequencies. When the illuminating beam has an aperture of about 5 μm and the amplitude and the phase are constant over the whole cross-section, the spatial frequencies are localized within an angle of about 5 deg, but if the aperture is a periodical structure (grating) with a spacing of 1 μm, this angle increases to 45 deg. In order to make use of all the spatial frequencies, the CCD camera must be placed close to the object, which means that the object may fall outside the far-field (Fraunhofer) zone.

The theoretical investigations of the errors, resulting from the spatial localization of the CCD sensor, ADC precision, pixelization, object thickness and its optical characteristics are an important part of the experimental set-up preparation. The difficulties in ensuring some of the conditions can prove a barrier, given the present state of the hardware.

Mathematical Model

When the distance between the object and the recording plane is much greater than the object size, the recorded intensity distribution can be approximated with a Fourier transformation of the object image [3,4]. In this case the total light intensity resulting from the combination of the object and the reference beams and measured by the CCD camera is given by:

$$A = \left[F(O.R0) + R1\right]^2, \tag{1}$$

where *A* is the intensity of each pixel, *O* is the object transmission function, *R*0 is the amplitude distribution function over the object beam aperture, *R*1 is the amplitude distribution function over the reference beam aperture and *F* is the Fourier transformation.

Taking the second power of the expression, we get:

$$A = \left[F(O.R0)\right]^2 + R1^2 + 2F(O.R0).R1\,. \tag{2}$$

From (2) it is seen that the first and the second additives represent the intensity from the object in the absence of reference and from the reference in the absence of object. In order to eliminate these terms, two separate measurements (two frames) must be conducted:

$$O2 = \left[F(O.R0)\right]^2 \tag{3}$$

and

$$R2 = R1^2\,. \tag{4}$$

For each frame *A* the corresponding corrected *B* is calculated, where:

$$B = (A - O2 - R2)/2 = F(O.R0).R1 \tag{5}$$

Using phase shifting (4-steps method) of *RB* the image (*B*4) can be calculated, which is the Fourier transformation of the aperture of the corresponding illuminating system in the frequency domain.

$$O.R0 = F^{-1}(B4) \tag{6}$$

Finally, a reference measurement must be provided to eliminate *R*0. It represents a measurement without an object and its physical essence is measurement of the aperture function of the measurement system. In this way, the influence of many inaccuracies in setting the measurement system parameters, such, as illuminating beam intensity distribution, measurement aperture geometry, reference beam and its parameters are compensated.

$$R0 = F^{-1}(Br4) \tag{7}$$

$$O = F^{-1}(B4)/R0 \tag{8}$$

O is a matrix of complex numbers. The modulus describes the attenuation and the argument – the phase shift for each point of the measured object.

Information Model

A program is written for simulation of the set-up on Fig.1 and processing of the information, derived from the CCD camera. The program is used to evaluate the requirements to the set-up and the constraints to the method applicability. The program is written in C++ for a PC platform. In some cases an FFT algorithm is used, but a direct integration version is also developed, due to the fact that the requirements for applying FFT cannot be always satisfied.

The object is chosen to be an amplitude (and/or phase) plate with a transmission function O = k0 + k1 SIN (k2 X). A photograph of an amplitude plate is shown in Fig. 2. The distribution function of the reference beam is chosen to be R0 = n0 + n1 SIN (n2 Y). The amplitude distribution is shown in Fig. 3.

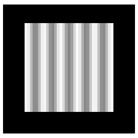


Fig.2 Measured object (simulation formula)
O (x, y) = 1+0.3 SIN (kX))

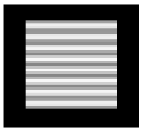


Fig.3 Measuring system aperture
(simulation formula)

This choice is made, in order to observe the interference pattern and the mutual influence of the aperture function and the object beam. The values of the pixels in each calculated image is rounded to 0.5 % of the maximum. The idea of this choice is to simulate the most popular 8-bit ADC used in frame grabbers.
Fig.4 shows amplitude and phase maps of the object function calculated by Fourier transformation.

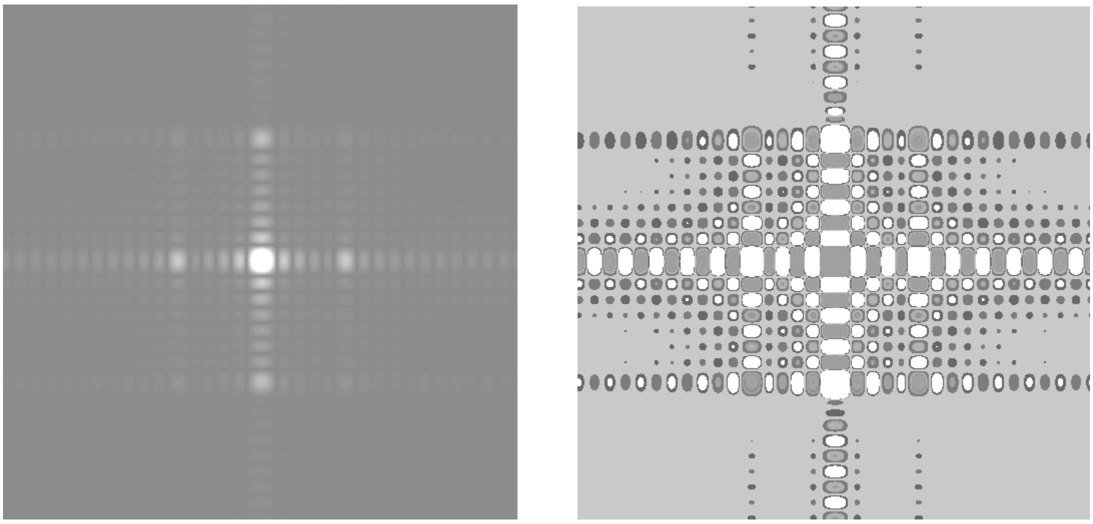


Fig.4 Final information before the reverse Fourier transform: amplitude and phase maps

The images in Gig.5, 6 and 7, obtained after application of the inverse Fourier transform, are close to the input.

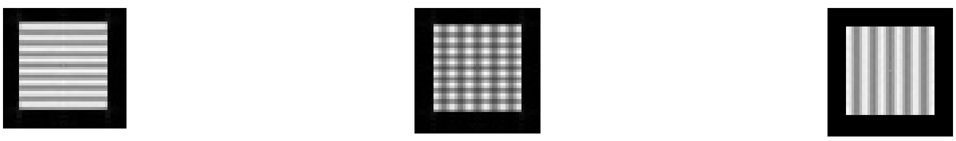


Fig.5 Calculated system transmission function distribution

Fig.6 Calculated object + system distribution

Fig.7 Calculated object

In the inverse calculation, due to rounding of the values of some pixels, the object function takes very high values – more than 50 times higher than the measured value maximum. This requires the use of a “window“-function to mask the undesired and unexpected results. The masking function (Fig. 8) is calculated from the distribution and replaces the calculated values. To eliminate small single spots, a smoothing filter with a suitable aperture (2-10 pixels) is used.

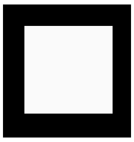


Fig.8 Calculated window of valid information

Fig.9 shows the transmission coefficient distribution in the middle of the image (in the cross-section), which, in that case, is simulated with values even greater than unity.

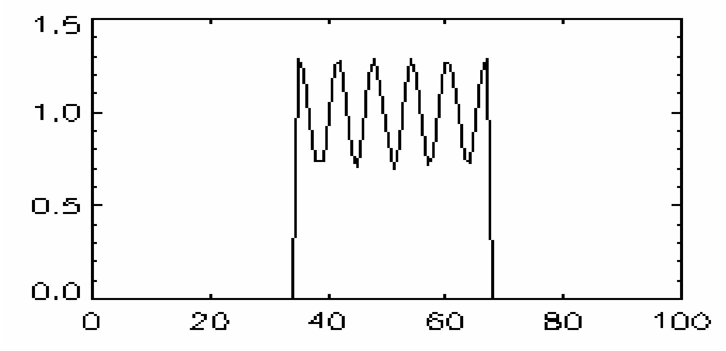


Fig.9 Cross-section of the intensity in the middle of a reconstructed image

Discussion and Conclusion

The proposed approach for measuring the optical characteristics of 2-D objects is very suitable for in-situ observation of living cells. The results of the modeling give us grounds to claim, that it is possible to use 8-bit ADC for visualization of phase objects. If this technique is to be used for measurement, an ADC with higher resolution is needed. The laser power and the CCD sensor sensitivity are essential, because the object can be damaged by the higher energy density. The investigation of living cells in liquid media can be realized by the use of pulsed laser.

Measurement of objects larger than 5-10 μm requires positioning of the CCD camera at a distance of more than 0.1-0.2 m from the object. In this case, this method is unsuitable and even impossible. On the other hand, the observation of small objects leads to an increase in the high-frequency components of the spatial frequencies. A CCD sensor with large sensitive area is required for the registration of these spectral components.

References

1. Baltes H. P., Inverse Source Problems in Optics, Springer-Verlag Berlin-New York, 1978.
2. Tiziani H.J., Optical methods for precision measurements, Optical and Quantum Electronics 21 (1989) 253-282.
3. Cowley J. M., Diffraction Physics, North-Holland Publishing Company, New York, 1975.
4. Rastogi P., Sharma A., Systematic approach to image formation in digital holography. Opt.Eng.42(5) 1208-1214 (May 2003).

Authors' Information

Georgi Stoilov – CLOSPI, Research Fellow, CLOSPI - BAS, 1113 Sofia, Acad. G.Bonchev Str. 101, PO Box 95, gstoilov@optics.bas.bg

Nikola Mechkarov – CLOSPI, Research Fellow, CLOSPI - BAS, 1113 Sofia, Acad. G.Bonchev Str. 101, PO Box 95, nikola@optics.bas.bg

Peter Sharlandjiev – CLOSPI, Assossiated Professor, CLOSPI - BAS, 1113 Sofia, Acad. G.Bonchev Str. 101, PO Box 95, pete@optics.bas.bg

SPREAD SPECTRUM WATERMARKING TECHNIQUE FOR INFORMATION SYSTEM SECURING

Todor Todorov

Abstract: In this paper we consider a computer information system and a way to realize the security of the data in it with digital watermarking. A technique for spread spectrum watermarking is presented and its realization with MathLAB 6.5 is shown.

Keywords: Web-based information systems, Spread spectrum watermarking, Images.

Introduction

Web-based information systems and their security with digital watermark are the main subjects of this paper.

On-line information automobile system Auto-World for examination and study of the different kinds of automobiles is developed. The application, which is based on the client/server module, has Application Space and Space DataBase, which are connected by PHP code. The Application Space operates with web browser, which associates with Apache Server by HTTP protocol. The application uses MySQL databases due to the necessity of convenient and rapid work with large amounts of data.

A digital watermark is intended to complement cryptographic processes. It is a visible, or preferably invisible, identification code that is permanently embedded in the data and remains present within the data after any decryption process. Here we present an algorithm realized with MathLab 6.5 functions and used for digital signing of images included in the presented information system.

Characteristics of Web-based Information Systems

Information systems are based on well-arranged data systems for parts of our real surroundings. It is supported by computing systems. An example of such system's data content is the information, supported by the external memory of a computing system and concerns the firms, their owners, the stocks they produce, preserve and sell/buy, the price, material and labour expenses for the particular activity. This data can show typical relations in the particular sphere – for example transport (air, railway, bus, marine), medical and other types of information insurance of the population.

The information system is a system, which processes the basic information activities - adding, preserving, processing and spreading information [2].

An information system can be considered as a model of a real process, the realization of which is an important aim for the work of an institution. Besides the organization of the basic actions (movement of objects such as stocks, services, capitals, people etc.) in it, another process must be organized - documenting the basic elementary actions, which change the process (adding, preserving and processing data for it). Instead of object, documents are moved in an information system. They show the corresponding actions and submit them in the model.

The Organization of an information system involves the organization of a particular information process, and that is why different circumstances must be clarified.

- Which data for the objects will be put in the documents.
- How will the documents be organized (at the entrance and at the exit).
- For which other processes additional information will be needed. (consistency) and which are their sources.
- How will the documents be processed, i. e. what are the problems and how will they be solved.
- What will be the purpose, the range of the information system and prospective, restrictions needed for the environment of its application.
- What results are expected - technical, economical, social etc.
- What expenses and inculcating are acceptable for the information system.

The development of local computer networks and Internet leads to fast augmentation in the appliance of software products in the organization of business objects' management.

An important part of this process is the development of web based information systems in different spheres, based on the client/server technology and data base usage.

Dynamic technologies are also very important for web-based systems.

Web-based information systems for automobiles are: http://www.peugeot.bg, http://www.bmw.com, http://www.theautochannel.com, http://www.auto.dir.bg, http://fiat.com.

Web based Information-reference system Auto-World is developed for examination and study of different automobiles. The project, which is based on the client/server module, has Application Space and Space DataBase, which are connected by PHP code [1] [10]. The Application Space operates with web browser, which associates with Apache Server by HTTP protocol [7]. The application uses MySQL databases [9] due to the necessity of convenient and rapid work with large amounts of data.

Auto-World aims to help people in examination and study of different kinds of automobiles, to present the most famous marks and models automobiles, their constructive and technical characteristics. The system offers text, video and audio information with references for them and opportunities for studying and testing in the sphere of automobiles. Due to the necessity of good results, an appropriate data base with information for the models, lessons and tests is built.

Securing Images with Digital Watermark

The proliferation of digitized media (audio, image, and video) is creating a pressing need for copyright enforcement schemes that protect copyright ownership. Conventional cryptographic systems permit only valid keyholders access to encrypted data, but once such data is decrypted there is no way to track its reproduction or retransmission. Therefore, conventional cryptography provides little protection against data piracy, in which a publisher is confronted with unauthorized reproduction of information. A digital watermark is intended to complement cryptographic processes. It is a visible, or preferably invisible, identification code that is permanently embedded in the data and remains present within the data after any decryption process.

Referring to Fig. 1 for a block diagram of a general watermarking system, the secret key is used to generate the random sequence W in this case [6].

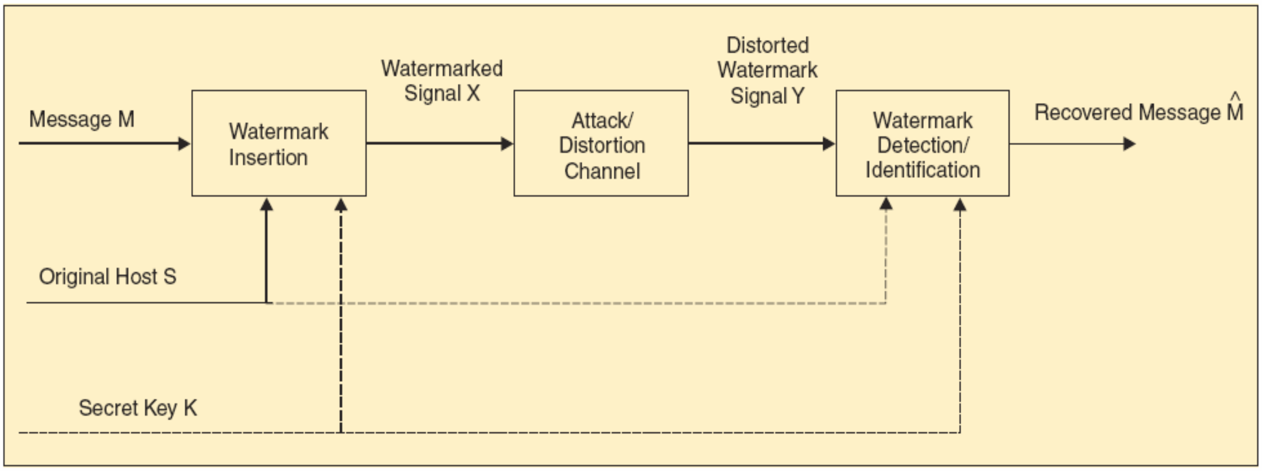


Fig. 1.

In order to be effective, a watermark should have the characteristics outlined below.

Unobtrusiveness: The watermark should be perceptually invisible, or its presence should not interfere with the work being protected.

Robustness: The watermark must be difficult (hopefully impossible) to remove.

In particular, the watermark should be robust in the following areas:

– Common signal processing: The watermark should still be retrievable even if common signal processing operations are applied to the data.

- Common geometric distortions: rotation, translation, cropping and scaling.
- Subterfuge attacks (collusion and forgery): In addition, the watermark should be robust to collusion by multiple individuals who each possess a watermarked copy of the data.
- Universality: The same digital watermarking algorithm should apply to all three media under consideration.
- Unambiguousness: Retrieval of the watermark should unambiguously identify the owner.

Spread Spectrum Watermarking

The watermark should not be placed in perceptually insignificant regions of the image (or its spectrum), since many common signal and geometric processes affect these components [4].

The problem then becomes how to insert a watermark into the most perceptually significant regions of the spectrum in a fidelity preserving fashion. Clearly, any spectral coefficient may be altered, provided such modification is small. However, very small changes are very susceptible to noise. To solve this problem, the frequency domain of the image or sound at hand is viewed as a communication channel, and correspondingly, the watermark is viewed as a signal that is transmitted through it. Attacks and unintentional signal distortions are thus treated as noise that the immersed signal must be immune to. We originally conceived our approach by analogy to spread spectrum communications [5]. In spread spectrum communications, one transmits a narrowband signal over a much larger bandwidth such that the signal energy present in any single frequency is undetectable. Similarly, the watermark is spread over very many frequency bins so that the energy in any one bin is very small and certainly undetectable. To insert a watermark in the frequency domain of an image we should first apply DCT(Discrete Cosine Transformation). This is a standard way to represent an image in frequency domain.

We compute NXN DCT coefficient matrix of an NXN image using next equation where A(i,j) is the intensity of the pixel in row i and column j.

$$B(k_1,k_2)=\sum_{i=1}^{N-1}\sum_{j=1}^{N-1}4.A(i,j).\cos\left[\frac{\pi.k_1}{2.N}(2.i+1)\right].\cos\left[\frac{\pi.k_2}{2.N}(2.j+1)\right]$$

Then we determine perceptually significant regions – with highest magnitude coefficients of the transform matrix and insert there the watermark. At the end inverse DCT are made.

Structure of Watermark

A watermark consists of a sequence of real numbers $X=x_1 \ldots\ldots x_n$.

In practice, we create a watermark where each value xi is chosen independently according to N(0,1) (normal distribution) using such distributions leaves one particularly vulnerable to attacks using multiple watermarked documents.

We extract from each document D a sequence of values $V=v_1 \ldots. v_n$, into which we insert a watermark $X=x_1 \ldots x_n$ to obtain an adjusted sequence of values $V'=v'_1 \ldots v'_n$. V' is then inserted back into the document in place of V to obtain a watermarked document D'. One or more attackers may then alter D', producing a new document D*. Given D and D*, a possibly corrupted watermark X* is extracted and is compared to X for statistical significance.

When we insert X into V to obtain V' we specify a scaling parameter α, which determines the extent to which X alters V ($v'_i = v_i\,(1+a_i x_i)$).

We measure the similarity of X and X* by sim(X,X*)= X*X/SQRT(X*,X*).

To decide whether X and X* match, one determines whether sim(X,X*)>T, where T is some threshold. Setting the detection threshold is a classical decision estimation problem in which we wish to minimize both the rate of false negatives (missed detections) and false positives (false alarms).

Such a watermark is very robust to the most of common signal processing and geometric distortions.

Presented algorithm have been realized with MathLab 6.5 functions [8] and used for digital signing of images included in the Auto-World information system.

Original Image

Watermarked image

Conclusions and Future Work

The proliferation of digitized media (audio, image, and video) is creating a pressing need for copyright enforcement schemes that protect copyright ownership. Presented information system is only one of the multimedia applications that could be secured with watermarks. We also explored application of other watermarking algorithms[3] and in the future we will continue to research this area.

Bibliography

[1] Кастането, Х., Х. Роат, С. Шуман, К.Сколо, Д. Вилайът. Професионално програмиране с PHP. СофтПрес, 2001.
[2] Лазаров, Л., М. Лазарова, Информационни системи. Славянска култура и образование, 2001.
[3] Bogdanova, G., T. Todorov, Application of digital watermarking, Journal of International Research Publications [serial online]. Bulgaria, Science Invest Ltd - branch Bourgas 2004.
[4] Cox, I., J. Kilian, T. Leighton, G. Shamoon. Secure spread spectrum watermarking for multimedia. Proceedings of the IEEE International Conference on Image Processing. Vol. 6. , 1997.
[5] Pickholtz, R., D. Schilling, L. Millstein. Theory of spread spectrum communications. IEEE Trans. Commun., vol. COMM-30, 1982.
[6] Podilchuk, C., E. Delp. Digital Watermarking Algorithms and Applications. IEEE Signal Processing Magazine, 2001.
[7] http://www.apache.org
[8] http://www.mathworks.com/
[9] http://www.mysql.com
[10] http://www.php.net

Author Information

Todor Todorov – PhD Student, Institute of Mathematics and Informatics, BAS, V.Tarnovo: +359 62 630132, e-mail: todor@moi.math.bas.bg

THE B-TERMINAL BUSY PROBABILITY PREDICTION

Stoyan Poryazov

Abstract: In the teletraffic engineering of all the telecommunication networks, parameters characterizing the terminal traffic are used. One of the most important of them is the probability of finding the called (B-terminal) busy. This parameter is studied in some of the first and last papers in Teletraffic Theory. We propose a solution in this topic in the case of (virtual) channel systems, such as PSTN and GSM. We propose a detailed conceptual traffic model and, based on it, an analytical macro-state model of the system in stationary state, with: Bernoulli–Poisson–Pascal input flow; repeated calls; limited number of homogeneous terminals; losses due to abandoned and interrupted dialling, blocked and interrupted switching, not available intent terminal, blocked and abandoned ringing and abandoned conversation. Proposed in this paper approach may help in determination of many network traffic characteristics at session level, in performance evaluation of the next generation mobile networks.

Keywords: terminal teletraffic, call blocking, human behavior, nonlinear system of equations.

AMS Subject Classification: 68N01, 65H10, 94C99, 60K30

1. Introduction

In the teletraffic engineering of all the telecommunication networks, parameters characterizing the terminal traffic are used. One of the most important of them is the probability of finding the called (B-terminal) busy. This parameter is studied in some of the first [Johannesen 1908] and last [Zeng et al 2002] papers in Teletraffic Theory. We propose a solution of this topic in the case of (virtual) channel systems, such as PSTN and GSM.

We propose a detailed conceptual traffic model of a (virtual) circuit switching telecommunication network and, based on it, an analytical macro-state model of the system in stationary state, with: BPP (Bernoulli–Poisson–Pascal) input flow; repeated calls; limited number of homogeneous terminals; losses due to abandoned and interrupted dialing, blocked and interrupted switching, not available intent terminal, blocked and abandoned ringing and abandoned conversation.

Proposed in this paper approach may help in determination of many network traffic characteristics at session level, in performance evaluation of the next generation mobile networks ("...dealing with traffic modelling in NG All-IP networks we have to consider three dependent components: mobility behavior of the user, session level teletraffic demands and packet level teletraffic demands."[Koucheryavy et al 2004]).

2. The Conceptual Model

In this paper two types of virtual devices are used: base and comprising base devices.

2.1. Base Virtual Devices and Their Parameters

We will use base virtual device types with names and graphic notation shown on Fig.1. For every device we propose the next notation for its parameters: Letter F stands for intensity of the flow [calls/sec.], P = probability for direction of the external flow to the device considered, T = mean service time in the device of one served call [sec.], Y = intensity of the device traffic [Erl], N = number of service places (lines, servers) in the virtual device (capacity of the device). In the normalized models [Poryazov 2001], used in this paper, every base virtual device, except the switch, has no more of one entrance and/or one exit. Switches have one entrance and two exits.

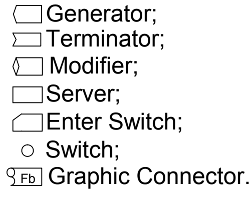


Fig.1. Graphic and text notations of the virtual device types, used in the conceptual model.

Characterizing the intensity of the flow, we are using the next notation: *inc.F* for incoming flow, *dem.F*, *ofd.F* and *rep.F* for demand, offered and repeated flows respectively [ITU E.600]. The same characterization is used for traffic intensity (Y).

2.2. The Virtual Base Device Names

In the conceptual model each virtual device has its own name. The names of the devices are constructed on their position in the model basis.

The model is divided on service stages (**d**ialing, **s**witching, **r**inging and **c**onversation).

Every service stage has branches (**e**nter, **a**bandoned, **b**locked, **i**nterrupted, **n**ot available, **c**arried), correspondingly to the modelled possible cases of ends of the calls' service in the branch considered.

Every branch has two exits (**r**epeated, **t**erminated) which show what happens with the calls after they leave the telecommunication system. Users may make a new bid (repeated call), or to stop attempts (terminated call).

In virtual device name construction the corresponding bold letters from the names of stages, branches end exits are used in the order shown below.

Virtual Device Name = <BRANCH EXIT><BRANCH><STAGE>

A parameter's name of one virtual device is a concatenation of parameters name letter and virtual device name. For example, "*Yid*" means "traffic intensity in interrupted dialing case"; "*Fid*" means "flow (calls) intensity in interrupted dialing case"; "*Pid*" means "probability for interrupted dialing"; *Tid* = "mean duration of the interrupted dialing"; "*Frid*" = "intensity of repeated flow calls, caused from (after) interrupted dialing".

2.3. The Paths of the Calls

Figure 2 shows the paths of the calls, generated from the A-terminals in the proposed network traffic model and its environment. *Fo* is the intent intensity of calls of one

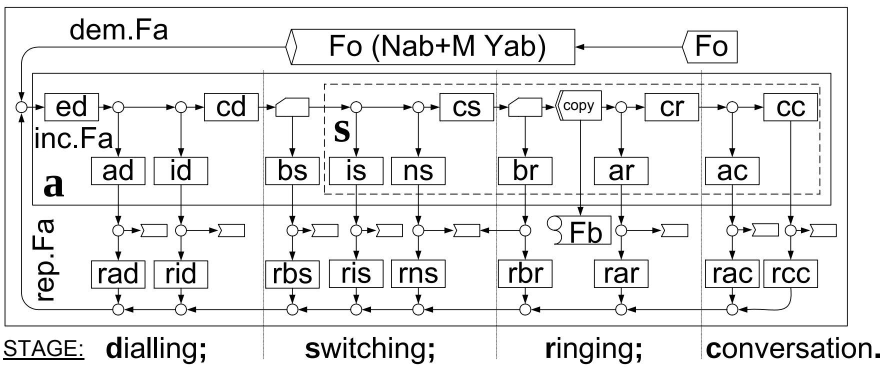


Figure 2. The paths of the calls, generated from the A-terminals in the proposed model.

idle terminal; M is a constant, characterizing the BPP flow of demand calls (*dem.Fa*). If M = -1, the intensity of demand flow corresponds to Bernoulli (Engset) distribution, if M = 0 - to the Poisson (Erlang), and if M = +1 - to the Pascal (Negative Binomial) distribution. In our analytical model every value of M in the interval [-1, +1] is allowed. The BPP-traffic model is very applicable [Iversen 2003].

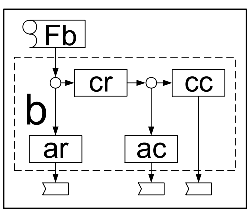


Figure 3. The paths of the calls, occupying B-terminals and corresponding virtual devices.

2.4. The Comprising Virtual Devices and Their Names

The next important virtual devices, comprising several base virtual devices, are considered:

a = a virtual device comprises all the A-terminals (calling) in the system. It is shown with continuous line box in Fig. 2. The devices outside the a-device belong to the network environment. The calls in the environment do not occupy network devices, but they form the incoming to the network flows;

b = a virtual device comprises all the B-terminals (called) in the system. The paths of the calls occupying B-terminals and corresponding virtual devices, included in the box with dashed line, are shown in Fig. 3;

ab = a virtual device comprises all the terminals (calling and called) in the system.

s = a virtual device corresponding to the switching system. It is shown with dashed line box in the Fig. 2.

The flow of calls (B-calls), with intensity Fb, occupying the B-terminals (Fig. 3), is coming from the Copy device (Fig. 2). This corresponds to the fact that at the beginning of the ringing a second (B) terminal in the system becomes busy. The second reason for this conceptual modelling trick is that the paths of the A and B-calls are different in the telecommunication system's environment, after releasing the terminals (compare Figures 2 and 3).

There are two virtual devices from type Enter Switch (Fig. 2) - before Blocked Switching (bs) and Blocked Ringing (br) devices. These devices deflect calls if there is no free line in the switching system and the B-terminal is busy. The correspondent transitions probabilities depend on the macrostate of the system (Yab).

The macrostate of a (virtual) device is defined as the mean number of simultaneously served calls in this device, in the observed time interval (similar to “mean traffic intensity” in [ITU E.600]).

3. The Analytical Model

3.1. Main Assumptions

For creating a simple analytical model, we make the next system of twelve (A1 - A12) assumptions:

A1. We consider a closed telecommunication system with functional structure shown in Fig. 2 and Fig. 3;

A2. All the terminals are homogeneous, e.g. all relevant characteristics are equal for every terminal;

A3. Every terminal directs the all its calls only to other terminals, not to itself;

A4. All virtual devices in the model (Fig.2 and Fig. 3) have unlimited capacity, with exceptions of the ab-device comprising all the $Nab \in [2, \infty)$ terminals and switching system (s) which has capacity of *Ns* internal switching lines. Every terminal has capacity 1, common for both incoming and outgoing calls;

A5. Every call, from the incoming in the telecommunication system flow (*inc.Fa*), falls only on a free terminal. This terminal becomes busy A-terminal. One call may occupy only one terminal and one terminal may serve only one call;

A6. Every call may occupy one internal switching line, if it find free one, independently from the state of the intent B-terminal (busy or free);

A7. Probabilities of direction of calls to, and duration of occupation of devices ar, cr, ac and cc are the same for A and B-calls (Fig.2 and Fig. 3);

A8. We consider probabilities for direction of calls to, and holding times in the base virtual devices as independent from each other and from intensity $Fa = inc.Fa$ of incoming flow of calls. Values of these parameters are determined from users' behaviour and technical characteristics of the communication system. (This is not applicable to Pbs and Pbr only - see 2.4.);

A9. The system is in stationary state. This means that in every virtual device in the model (including comprising devices like switching system), the intensity of input flow $F(0, t)$, calls holding time $T(0, t)$ and traffic intensity *Y(0, t)* in the observed interval *(0,t)* converge to the correspondent finite numbers F, T and Y, when $t \to \infty$. In this case we may apply the theorem of Little [Little 1961] and for every device: $Y = FT$;

A10. The flows directed to A-terminals (*Fa*) and to B-terminals (Fb) are ordinary. For Fa this is usual premise, but for Fb A5 may be acquitted from results like in [Burk 1956] and [Vere-Jones 1968];

A11. The mean probability of a call to find B-terminal busy at the first and at the all next repeated attempts is one and the same. This is the only assumption in this paper, causing systematic error. In [Poryazov 1992] is shown, on the basis of comparison of analytic and simulation [Todorov, Poryazov 1985] results, that this error is stable and don't exceed 5% of the Pbr in the reasonable traffic load interval;

A12. All variables in the analytical model may be random and we are working with their mean values.

3.2. Equations

From definitions of *a* and *b* comprising devices and assumptions A1 and A4, obviously the sum of traffic intensities of A and B-terminals gives the traffic intensity of the all occupied terminals in the system:

$$Ya + Yb = Yab \leq Nab \, . \quad (1)$$

Theorem 1. Traffic intensity of B-terminals may be calculated from the equation

$$Yb = FbTb \, , \quad (2)$$

where:

$$Fb = Fa(1 - Pad)(1 - Pid)(1 - Pbs)(1 - Pis)(1 - Pns)(1 - Pbr) \, , \quad (3)$$

$$Tb = ParTar + (1 - Par)[Tcr + PacTac + (1 - Pac)Tcc] \, . \quad (4)$$

Proof: Equation (2) is the formula of Little for device *b* (A9). $Fb = inc.Fb$ is intensity of calls occupying B-terminals and (3) is direct corollary from A1, Fig. 2, A8, and A9.

We may receive the expression for B-terminals holding time (4) from the next considerations. From A1, Fig.3, A8 and A9 follow that:

$$Yb = Fb\ Tb = Yar + Ycr + Yac + Ycc; \quad (5)$$

$$Yar = Far\ Tar = Fb\ Par\ Tar; \quad (6)$$

$$Ycr = Fcr\ Tcr = Fb\ (1\text{-}Par)\ Tcr; \quad (7)$$

$$Yac = Fac\ Tac = Fb(1\text{-}Par)Pac\ Tac; \quad (8)$$

$$Ycc = Fcc\ Tcc = Fb(1\text{-}Par)(1\text{-}Pac)\ Tcc. \quad (9)$$

After replacing (6), (7), (8) and (9) in (5), we receive (4).

Theorem 2. A-terminals' traffic intensity (Ya) is determining from the expression:

$$\begin{aligned} Ya = FaTa = Fa\{(&Ted+PadTad+(1\text{-}Pad)(PidTid+(1\text{-}Pid) \\ &(Tcd+PbsTbs+(1\text{-}Pbs)(PisTis+(1\text{-}Pis) \\ &(PnsTns+(1\text{-}Pns)(Tcs+PbrTbr+(1\text{-}Pbr)Tb))))))\}. \end{aligned} \quad (10)$$

The proof of Theorem 2 is very similar to the proof of Theorem 1, but assumption A11 is used in addition to A1, A8 and A9.

Following the same technique one may easy receive equation (11.1) of the system (11):

$$\begin{aligned} Yab \ = \ Fa\ \{&Ted + PadTad + (1\text{-}Pad)[PidTid + (1\text{-}Pid)[Tcd + PbsTbs + \\ &(1\text{-}Pbs)[PisTis + (1\text{-}Pis)\ [PnsTns + \ (1\text{-}Pns)[Tcs + PbrTbr + 2(1\text{-}Pbr)Tb]]]]]. \end{aligned} \quad (11.1)$$

$$Fa \ = \ dem.Fa \ + \ rep.Fa. \quad (11.2)$$

$$dem.Fa \ = \ Fo\ (Nab \ + \ M\ Yab). \quad (11.3)$$

$$\begin{aligned} rep.Fa \ = \ Fa\ \{&PadPrad + (1\text{-}Pad)[PidPrid + \ (1\text{-}Pid)[PbsPrbs + (1\text{-}Pbs) \\ &[PisPris + \ (1\text{-}Pis)[PnsPrns + (1\text{-}Pns)[PbrPrbr + (1\text{-}Pbr) \\ &[ParPrar + \ (1\text{-}Par)[PacPrac + (1\text{-}Pac)Prcc]]]]]]]\}. \end{aligned} \quad (11.4)$$

$$Pbr = \frac{Yab\text{-}1}{Nab\text{-}1} \quad \text{if} \quad 1 \le Yab \le Nab,$$
$$Pbr = 0 \qquad \text{if} \quad 0 \le Yab < 1. \tag{11.5}$$

$$ofd.Fs = Fa\ (1\text{-}Pad)(1\text{-}Pid)\ . \tag{11.6}$$

$$Ts = [PisTis + (1\text{-}Pis)[PnsTns + (1\text{-}Pns)[Tcs + PbrTbr + (1\text{-}Pbr)Tb]]]. \tag{11.7}$$

$$ofd.Ys = ofd.Fs\ \ Ts. \tag{11.8}$$

$$Pbs = Erl\_b\ (Ns,\ ofd.Ys). \tag{11.9}$$

We may remark that the equations:

(11.2)	simply expresses that intensity of the flow of calls occupying A-terminal is sum of primary (demand) and repeated calls [ITU E.600] (see Fig. 2);
(11.3)	shows intensity of demand calls as a function of the intensity of generated calls from one idle terminal (*Fo*) and the macrostate (*Yab*) of the system (BPP - flow, see Fig. 2);
(11.4)	determinates the intensity of repeated calls (*rep.Fa*), as a function of transitions probabilities in the model. It is received in the same way as (11.1) (see Fig. 2);
(11.5)	is discussed in 4;
(11.6)	expresses intensity of the offered to the switching system flow of calls;
(11.7)	*Ts* is the holding time of calls in the switching system, received in the same way as (11.1) (see Fig. 2);
(11.8)	defines offered traffic (*ofd.Ys*) to the switching system [ITU E.600];
(11.9)	expresses usage of the Erlang-B formula for determination of the blocking probability in the switching system, on the basis of the number of internal switching lines (*Ns*) and offered traffic.

In general, the system of equations (11) has:

- 9 equations;
- 9 output parameters with unknown values: *Yab, Fa, dem.Fa, rep.Fa, Pbs, Pbr, ofd.Fs, Ts, ofd.Ys*;
- 32 input parameters with known (given) values: *Nab, Ns, M, Fo, Ted, Pad, Tad, Prad, Tcd, Pid, Tid, Prid, Tbs, Prbs, Pis, Tis, Pris, Pns, Tns, Prns, Tcs, Tbr, Prbr, Par, Tar, Prar, Tcr, Pac, Tac, Prac, Tcc, Prcc*.

4. The B-Terminal Busy Probability Prediction

Theorem 3. The probability of finding the B-terminal busy (Pbr) if $1 \le Yab \le Nab$, is:

$$Pbr = \frac{Yab - 1}{Nab - 1} \tag{12}$$

Proof: According assumptions A1 all the calls are directed to the terminals inside the investigated system. Follow A2, all the terminals have equal probability to be called, but A3 excludes the calling A-terminal. Similar natural assumption is made from [Ionin 1978]. Consequently every call is directed with equal probability to $Nab\text{-}1$ terminals, with $Yab\text{-}1$ of them busy. From A10 it follows that two calls can't come simultaneously and hence their probabilities to find B-terminal busy are independent from each other (they depend from the system state only).

Let call number i finds $Yab(i)$ busy terminals. Than its probability to find B-terminal busy ($Pbr(i)$), under the assumptions made, is:

$$Pbr(i) = \frac{Yab(i) - 1}{Nab - 1} \tag{13}$$

If we consider n calls reaching B-terminals, then their mean probability Pbr to find B-terminals busy is:

$$Pbr = \frac{1}{n}\sum_{i=1}^{n} Pbr(i) = \frac{1}{n}\sum_{i=1}^{n} \frac{Yab(i)-1}{Nab-1} = \frac{1}{(Nab-1)} \frac{\sum_{i=1}^{n}(Yab(i)-1)}{n}. \quad (14)$$

From the A9 follows that the system is in a stationary state. Therefore, the mean value of the intensity of the terminal traffic exists and equals to the Yab. In the other words:

$$Yab = \lim_{n\to\infty} \frac{\sum_{i=1}^{n} Yab(i)}{n}. \quad (15)$$

From (15) when $n \to \infty$ and (14), we receive (12).

Theorem 4. A threshold ($thr.Fa > 0$) of the intensity of the input flow Fa exists, so that in the interval $Fa \in [0, thr.Fa]$ busy terminals exist, but there are not losses due the finding B-terminal busy. In this case:

$$\text{thr}.Fa = \frac{1}{S_1}, \quad (16)$$

where:

$S_1 = Ted + Pad\ Tad + (1 - Pad)(Pid\ Tid + (1 - Pid)(Tcd + Pbs\ Tbs + (1-Pbs)\ (Pis\ Tis + (1-Pis)(Pns\ Tns + (1 - Pns)\ (Tcs + 2\ Tb)))))$.

Proof: We may present (11.1) in the form:

$$Yab = Fa(S_1 - S_2 Pbr), \quad (17)$$

where:

$S_2 = (1 - Pad)(1 - Pid)(1 - Pbs)\ (1 - Pis)(1 - Pns)(2\ Tb - Tbr)$.

If we change Pbr with 0 in (17), we receive:

$$Yab = FaS_1. \quad (18)$$

Equation (18) is received without assumptions of any dependence between Fa and Pbr and therefore is true in any cases, when $Pbr = 0$. From (18) it is obviously that the value of $thr.Fa$ from (16) ($S_1 > 0$ in the all working systems) is the only when $Yab = 1$ and $Pbr = 0$.

Comment: The fact, that in the interval $Fa \in [0, thr.Fa]$ we have not losses due finding B-terminal busy, must be understand in the asymptotic case, when $t \to \infty$. In the other words, losses may exist, but:

$$Pbr = \lim_{t\to\infty} \frac{Zbr.a(0,t)}{Zbr.a(0,t) + Z.b(0,t)} = 0, \quad (19)$$

where $Zbr.a(0,t)$ notes the number of the all calls finding the B-terminal busy in the interval of observation $(0,t)$ and $Z.b(0,t)$ is the number of calls successfully sizing B-terminals in the same time interval.

Based on Theorem 3 and Theorem 4, we may define (11.5). This definition is used in a very simple teletraffic model [Poryazov 1991] without detailed proof.

5. Conclusions

- Detailed conceptual and analytical models of a telecommunication system are created.
- A mathematical model, which may be used for prediction of the probability of finding B-terminal busy, is proved.

Bibliography

[Burk 1956] Burk, P. J. The output of a queueing system. J. Op. Res. Soc. Amer. 4, 699-704, 1956.

[Ionin 1978] Ионин Г. Л. Исследование необратимых состояний системы массового обслуживания с повторными вызовами и конечным числом источников. В кн.: Информационные сети и их анализ. Москва, «Наука», 1978. с. 13-31.

[ITU E.600] ITU-T Recommendation E.600: Terms and Definitions of Traffic Engineering. (Melbourne, 1988; revised at Helsinki, 1993).

[Iversen 2003] Villy B. Iversen TELETRAFFIC ENGINEERING HANDBOOK. ITU-D SG 2/16 & ITC. Draft, December 2003, pp. 328. http://www.tele.dtu.dk/teletraffic/ (11.12.2004)

[Johannesen 1908] Fr. Johannesen. "Busy" The frequency of reporting "busy" and the cost caused thereby. The Copenhagen Telephone Company, Copenhagen, October 1908, 4 pp.

[Koucheryavy et al 2004] Y. Koucheryavy, D. Moltchanov, J. Harju, G. Giambene. Cross-layer black-box approach to performance evaluation of next generation mobile networks. NEW2AN 2004, 7 pp.

[Little 1961] J. D. C. Little. A Proof of the Queueing Formula L=λW. Operations Research, 9, 1961, 383-387.

[Poryazov 1991] S. Poryazov. Determination of the Probability of Finding B-Subscriber Busy in Telephone Systems with Decentralized Control. Comptes Rendus de l'Academie Bulgare des Sciences – Sofia, 1991, Tome 44, No.3, pp. 37-39.

[Poryazov 1992] S. Poryazov. Determining of the Basic Traffic and Time Characteristics of Telephone Systems with Nonhomogeneous Subscribers. 4th International Seminar on Teletreffic Theory and Computer Modelling – Moscow October 20-25, 1992 – pp. 130-138.

[Poryazov 2001] S. Poryazov. On the Two Basic Structures of the Teletraffic Models. Conference "Telecom'2001" - Varna, Bulgaria, 10-12 October 2001 – pp. 435-450).

[Todorov, Poryazov 1985] Todorov, S. Poryazov. Basic Dependences Characterizing a Model of an Idealised (Standard) Subscriber Automatic Telephone Exchange. Proc. of the 11-th ITC, September 4-11, 1985 Kyoto, Japan, pp 4.3B-3-1- 4.3B-3-8.

[Vere-Jones 1968] Vere-Jones D. Some applications of probability generating functionals to the study of input-output streams. J. Roy. Statist. Soc., 1968, B30, No 2, 321-333.

[Zeng et al 2002] H. Zeng, Y. Fang, I. Chlamtac. Call Blocking Performance Study for PCS Networks under More Realistic Mobility Assunptions. Telecommunication Systems 19:2, 125–146, 2002, Kluwer Academic Publishers.

Author Information

Stoyan A. Poryazov - Institute of Mathematics and Informatics, Bulgarian Academy of Sciences,
Acad. G. Bonchev Str., Block 8, 1113 Sofia, Bulgaria, Tel: (+359 2) 979 28 46,
Fax: (+359 2) 971 36 49, e-mail: stoyan@cc.bas.bg

Table of Contents of IJ ITA Volume 11, Number 1

Table of Contents of IJ ITA Volume 11, Number 2

Table of Contents of IJ ITA Volume 11, Number 3

Table of Contents of IJ ITA Volume 11, Number 4

AUTHORS' INDEX

TABLE OF CONTENTS