ՀՀ ԳԱԱ ԻՆՖՈՐՄԱՏԻԿԱՅԻ ԵՎ ԱՎՏՈՄԱՏԱՑՄԱՆ ՊՐՈԲԼԵՄՆԵՐԻ ԻՆՍՏԻՏՈՒՏ

Կարեն Սնադայի Խաչատրյան

Հակադրության խնդիրներում փորձագիտական գիտելիքների բազաների ձևավորման և դրանց փնտրման մեթոդների հետազոտումը և ծրագրերի մշակումը

Ե13.05 – «Մաթեմատիկական մոդելավորում, թվային մեթոդներ և ծրագրերի համալիրներ» մասնագիտությամբ տեխնիկական գիտությունների թեկնածուի գիտական աստիճանի հայցման ատենախոսության

ՍԵՂՄԱԳԻՐ

Երևան – 2013

INSTITUTE OF INFORMATICS AND AUTOMATION PROBLEMS OF NAS RA

Karen Khachatryan

Developing Algorithms and Programs for Formation of and Search in Knowledge Bases of Combating Problems

Author’s abstract

For obtaining a candidate degree in technical sciences in specialty 05.13.05 “Mathematical modeling, numerical methods and program complexes”

Yerevan – 2013

Ատենախոսության թեման հաստատվել է Հայաստանի պետական ճարտարագիտական համալսարանում

Գիտական ղեկավար՝ ֆիզ.մաթ.գիտ. դոկտոր Է. Մ. Պողոսյան
ընդդիմախոսներ՝ ֆիզ.մաթ.գիտ. դոկտոր Ի. Դ. Զասլավսկի
տեխ.գիտ. թեկնածու Հ. Վ. Ասցատրյան

Առաջատար կազմակերպություն՝ Հայաստանի ամերիկյան համալսարան

Պաշտպանությունը կայանալու է 2013թ. օգոստոսի 28-ին, ժ. 16:00-ին ՀՀ ԳԱԱ Ինֆորմատիկայի և ավտոմատացման պրոբլեմների ինստիտուտում, թիվ 037 «Ինֆորմատիկա և հաշվողական համակարգեր» մասնագիտական խորհրդի նիստում (հասցեն՝ ք. Երևան, 0014, Պ. Սևակի փ. 1):

Ատենախոսությանը կարելի է ծանոթանալ ինստիտուտի գրադարանում:

Սեղմագիրն առաքված է 2013թ. հուլիսի 27-ին:

Մասնագիտական խորհրդի գիտական
քարտուղար, ֆիզ.մաթ.գիտ. դոկտոր Հ. Գ. Սարուխանյան

The subject of the dissertation has been approved in State Engineering University of Armenia.

Scientific adviser: Dr. of Phys. and Math. Sc. E. Pogossian
Official opponents: Dr. of Phys. and Math. Sc. I. Zaslavski
Cand. of Tech. Sc. H. Astsatryan

Leading organization: American University of Armenia

The defense will take place on 28th of August 2013, at 16:00 in the Institute of Informatics and Automation Problems of NAS RA, during the session of the 037 “Informatics and computer systems” special council (address: 1 P. Sevak str. 0014, Yerevan).

The dissertation is available at the library of the institute.

Author’s abstract is sent on 27th of July 2013

Scientific secretary of the specialized council:
Dr. of Phys. and Math. Sc. H. G. Sarukhanyan

INTRODUCTION

Actuality of the Subject

Current research is a sequel of the study of knowledge-based strategies where from the variety of problems, the class where *Space* of possible *Solutions* can be specified by *Reproducible* combinatorial *Game Trees* (SSRGT) is considered and ***unified*** algorithms and software, SSRGT Solver, for elaborating optimal strategies for any input specified problem of the class are developed.

The SSRGT is a spacious class of problems with only a few requirements as follows:

- there are (a) interacting actors, performing (b) identified types of actions in the (c) specified moments of time and (d) specified types of situations,
- there are identified benefits for each of the actors,
- the situations the actors act in and transformed after the actions can be specified by certain rules, regularities.

Many security and competition problems like network Intrusion Protection (IP), Management in oligopoly competitions, Chess-like combinatorial problems, Defense of military units from a variety types of attacks, Anomalies detection and correction in computations, etc., belong to SSRGT class.

Unified SSRGT specification of problems makes possible to design a unified Solver for the problems of the class.

Study of meaning processing for knowledge based strategies and translators in Armenia were started in the Computing Center of Academy of Sciences and State University of Armenia (now Institute for Informatics and Automation Problems) since its foundation in 1957 by Sergey Mergelyan.

In Pogossian's works it was shown that urgent and spacious SSRGT combating and competition problems are reducible to the standard kernel problem K of the class and we do focus chess as the K. The importance of the K- methodology is that it multiplies the achievements for particular problems of the SSRGT class. K- centric methodology enhances the effectiveness of SSRGT Solvers providing answers to the urgent SSRGT questions including the following ones: (a) measurement of the effectiveness of Solvers; (b) analysis and typifying combating knowledge; (c) construction of knowledge based SSRGT Solvers; (d) acquisition in a regular way SSRGT expert knowledge.

Solver of the SSRGT problems is a package aimed to acquire strategic expert knowledge to become comparable with a human in solving hard combinatorial competing and combating problems.

In fact, the following *three* tasks of expert knowledge acquisition can be identified in the process:

1. construction of the package of programs *sufficient* to acquire the ***meanings*** of the units of vocabulary (UV) of problems,
2. construction of procedures for *regular* acquisition of the meanings of UV by the package,
3. provision of means *measuring the effectiveness* of solutions of SSRGT problems.

The second task of *acquisition of complex expert knowledge* was planned to solve in the following two stages:

- *proving the sufficiency,* i.e. proving that Solver, in principle, can acquire the meanings of expert knowledge of an intensive SSRGT problem, e.g. for the kernel SSRGT chess game,
- *ensuring regularity,* i.e. to develop procedures for regular acquisition of SSRGT problems and meanings of UV of those problems.

The meanings are defined as certain assembles of regularities which have, at least, identifiers, *names*, and do combine with other meanings by *have, be, do* types of links, *relationships*, subjects to be varied in time, aspects, modality and certain other syntax categories.

The problem of modeling and processing the meanings of lexical units is especially singled out also in the recent researches of the semantic/meaning-based search in the web.

The searching giants, like Google with its Knowledge Graph and Microsoft with its Satory are introducing semantic extensions to their search engines. However, the problem of understanding the natural language sentences as well as supporting semantic searches for languages *but* English still stand as research challenges in the web search.

The understanding of the meanings of natural language sentences is still not a solved problem also in machine translation. The ongoing UNL project (Universal Networking Language), with its projection to Armenian, is another example of the efforts to enhance computers to understand what is meant by natural languages.

The general overview of the problem of understanding the meaning of human vocabulary by computers and the methods to solve it are considered particularly in [Understanding Computers and Cognition, T. Winograd, F. Flores (Standford), 1987], publications of Marvin Minsky, Deb Roy (MIT), SOAR Project (Michigan University).

The objective of the work is to research and develop the algorithms for regular acquisition, presentation, evolvement and utilization of meanings of vocabulary of SSRGT Games in SSRGT Solver software framework. To achieve the objective following problems have been solved:

- Extending the graphical language interpreter and interface for formation and acquisition of expert strategic knowledge of arbitrary SSRGT problem.
- Designing and developing framework for presenting and manipulating acquired meanings.
- Developing algorithms for matching situations to the meanings.
- Developing combinative algorithms for constructing new meanings from existing ones or relations between them.

Objects of the research are the graphical language interpreter, the software framework for modeling the store of the acquired meanings and mechanism for evolving new meanings from the learned ones, as well as the algorithms of finding meanings active in given situations for the class of problems where space of solutions can be represented by reproducible game trees.

Methods of research are based on the theory of developing programming languages (UML, Java), mechanisms to solve constraint satisfaction problems as well as the principles of design of agent systems.

Scientific novelty

1. The modules for acquisition of communalized and personalized lexical units are developed, specified by multiple level of nested attributes and polymorphic properties in the framework of expert strategy knowledge acquisition by SSRGT Solver.
2. Algorithms for presentation and evolvement of meanings of SSRGT problems based on the *be-*, *have-*, *do* language syntax are implemented in extended Java language.
3. Algorithms for matching situations to the meanings are developed and realized based on techniques of iterative matching to sub-meanings and constraint satisfaction.
4. Algorithms for building new meanings from the existing ones based on combinative generalization approach are developed and realized.

Practical significance

1. Because SSRGT class includes a large list of practical problems (computer networks intrusion protection, optimal management and marketing strategy elaboration in competitive environments, defense of military units from a variety types of attacks, etc.) the software package can be applied for acquisition of communicable and personalized components of expert knowledge and for detection of relevant meanings in the situations of particular problems.
2. Language derived be-, have-, do- model of meaning presentation inherits the properties of object oriented (OO) paradigm in improving the visualization and understanding of structures of meanings, meanwhile, supports extra characteristics to overcome several limitations of modern compilers, interpreters which implement classical OO languages.
3. The instantiation of meanings from their sub-meanings enables the iterative activation of meanings in situations, while reusing the matched sub-meanings prevents the checking of same conditions more than once.
4. The techniques of combinative generalization of meanings enable extraction and insertion of new meanings (OO classes) into the existing hierarchies or building new relations between existing meanings in a dynamic way – without altering the definitions of existing classes and without recompilation and restarting the program.

Practical implementation

1. The results have been implemented and are used in "OMD" LLC, which provides market data management and analytical solutions that enable financial institutions to outmaneuver the competition by executing new trading strategies faster. The SSRGT Solver allows acquisition of financial news analyst's strategy knowledge via graphical interface and detects the relevant units within the vast sources of the financial news. The activated units can be used for building or modifying the trading strategies.
2. SSRGT Solver is installed in the American University of Armenia and in the Institute of Informatics and Automation Problems of NAS RA for educational and research purposes.

The following topics are presented to the defense

- Object oriented (OO) constructions and a model for presentation and manipulation of meanings of lexical units of SSRGT problems based on be-, have-, do language dimensions.
- The algorithm for matching situations to the meanings by iterative matching of their sub-meanings.
- The combinative generalization algorithms of meanings which allow extraction of new meanings from and building new relations between already acquired ones.

Approbation

The results of the work on dissertation have been presented at:

- International conference - Computer Science and Information Technologies (CSIT), Yerevan, Armenia, 2011;
- The Annual Conference of SEUA, Yerevan, 2013.
- Seminars at State Engineering University of Armenia, Institute of Informatics and Automation Problems of National Academy of Sciences of the Republic of Armenia, American University of Armenia.

Publications

The main topics of the dissertation are published in [1-4].

Structure of the dissertation

The dissertation contains an Introduction, five Chapters, a general summary with the conclusion and future works, the list of references with 124 entries and one Appendix. The total volume of the dissertation is 127 pages, with 3 tables and 44 figures.

CONTENT OF THE WORK

The **introductory** chapter describes the actuality, objectives, scientific novelty and practical significance of the work as well as gives the main information about the practical implementation of the results.

Chapter 1 is dedicated to an overview of the problem of the acquisition of meanings of the lexical units by computers, their presentation and detection in the relevant situations as well as the problems of building new meanings from already acquired ones. The aim is to develop models that will enable the presentation of meaning of acquired lexical units, support procedures of their evolvements and algorithms for matching the situations to the acquired meanings.

In ***Section 1.1*** and ***1.2*** the Optimal Strategy Provision (OSP) problems and their constructive SSRGT subclass of problems are considered. The latter describes problems which *space* of *solutions* can be represented as *reproducible game trees*. The problems of SSRGT class postulate existence of the following: (a) interacting actors (players, competitors, etc.) performing (b) identified types of actions in the (c) specified moments of time and in the (d) specified types of situations. There are identified benefits (goals) for each of the actors. The aggregation of situations represents the game tree, nodes of which are situations themselves and edges are the corresponding transformation actions of the interacting actors.

We say that SSRGT problem is solved for the actor **A** and for the given situation **x** if a GT(x, A) strategy is induced which represents all the possible performance trees of the strategies from **x** initial situation. The PPIT (Personalized Planning and Integrated Testing) Solver program was developed that implements PPIT algorithm. For inducing strategies PPIT uses an approach that relies on the experts' problem domain knowledge represented as plans, goals, and abstracts.

Section 1.3 considers the meaning presentation and processing approaches and discusses the existing knowledge representation systems. Table 1 summarizes the comparative analysis of that systems from the perspective to represent the required properties of the meanings of SSRGT problems.

From the analysis we conclude that, although, object oriented programming (OOP) systems cover most of the required properties there are following fundamental limitation in the functionalities of the compilers and interpretators of modern OOP languages:

- dynamic generation of new classses,
- matching classes by matching its attributes,
- dynamically change class hierarchies,
- identify inheritence hierarchies,
- dynamically build new classes from existing ones.

Table 1. Analysis of the knowledge representation models. Prod.Sys. stands for Production Systems - which is used in SOAR Project, Protégé - the ontology editor and knowledge acquisition system and OOP languages (C++, Java). The most important properties are marked in bold. '+' indicates the existence and '-' the absence of the property.

	Property	Prod. Sys.	Protégé	OOP
Acquisition of Meanings	Transparency	-	+	+
	Reuse	+	+	+
	Inheritance	-	-	+
	Polymorphism	-	+	+
	Describe static entities (for example goals)	-	+	+
	Define dynamic actions to build the algorithms	+	-	+
Development of Meanings	**Dynamically generate and integrate new meanings**	+	+	-
	Dynamically modify existing meanings	+	+	-
	Build new meanings from already existing ones	-	-	-
	Change hierarchies of existing meanings	-	-	-
	Build new relations (Inheritance etc.) between existing meanings	-	-	-
Recognition of Meanings	**Match situations to meanings**	+	-	-
	Support the nucleus level abstraction	+	+	+
	Build new meanings from t-prints	+	-	-

Finally the ***Section 1.4*** discusses the problem of inferring new meanings from the existing ones.

Summary of ***Chapter 1*** is given in ***Section 1.5***.

Chapter 2 describes the solutions to the shortcomings of the graphical language interpreter for expert strategy knowledge acquisition and develops the model for presentation of the meanings of acquired lexical units.

Section 2.1 discusses enhancements of the graphical language and interpreter that simplify and augment the descriptive functionalities of the interface by enriching it with the ability to describe more complex and flexible units.

In order to encapsulate the internal details of the meaning acquisition interface regarding the indexing and grouping the instances of nucleus abstracts, ***subsection 2.1.1*** introduces the new concept called **ar1** - *abstracts satisfying rule number 1*. The rule number 1 dictates the following restrictions:

- an ar1 must contain only nucleus attributes,
- there must be utmost one attribute of a given nucleus type,
- all attributes must belong to the same Id Group - i.e. have the same indexes.

According to these restrictions, ar1s embrace implied joins within the attributes using their indices, hence, removing the burden of the manual configuration of the indices from the interface.

Subsection 2.1.2 describes the achievements in modeling and working with abstracts having multilevel nested attributes. The other enhancement of the interpreter is the ability to define and work with abstracts having polymorphic properties introduced in ***subsection 2.1.3***. These are called **virtual abstracts** and play the role of *interfaces* or *abstract classes* of OO programming languages (Java, C++ etc.) with a difference that the virtuality here is achieved by the means of undefined values of attributes instead of methods. The concept of virtuality thus becomes pretty generalized and is available for all kind of abstracts, not just for procedural entities (Actions). A virtual unit is, therefore, a partial specification of a class or a procedural entity.

Syntactically, an abstract is virtual if it contains at least one attribute with undefined relation (the attribute condition is undefined if it is set to equal to '?' special value). To make the instantiations of such kind of abstracts possible one has to inherit a new abstract from them and specify undefined conditions/relations. This procedure is called virtual abstract **specification**. The use of a virtual abstract as an attribute in the other abstract is called a virtual abstract **usage**. What follows, is that the instance of the abstract can be built from various compositions of specific attribute instances.

To overcome the limitations underlined in the chapter one in ***Section 2.2*** a more flexible approach of meaning presentation based on the *be-*, *have-*, *do* dimensions of language grammar is developed. The refinement of meanings and the relevance of their representation using *be-*, *have-*, *do* dimensions of English grammar are argued in [On Modeling Cognition , Pogossian E., 2011][1].

Subsections 2.2.1 and ***2.2.2*** discuss the **graph of abstracts** (GA) as the sketch of the meaning representation model and provide the algorithm of a new meaning integration. The graph of abstract is a directed symmetric graph where edges represent one of be, have, do relations (with their

---

[1] Pogossian, E. On Modeling Cognition. Computer Science and Information Technologies (CSIT11). Yerevan, Sept.26-30, 2011. pp. 194-198.

reverse pairs) and nodes represent the abstracts. The latter like Markov algorithms is a kind of compositions of rules, regularities.

Meaning mR of the classifier cR (lexical units etc.) of realities R is the connectivity sub graph GmR of GA centered in the cR (Fig 1).

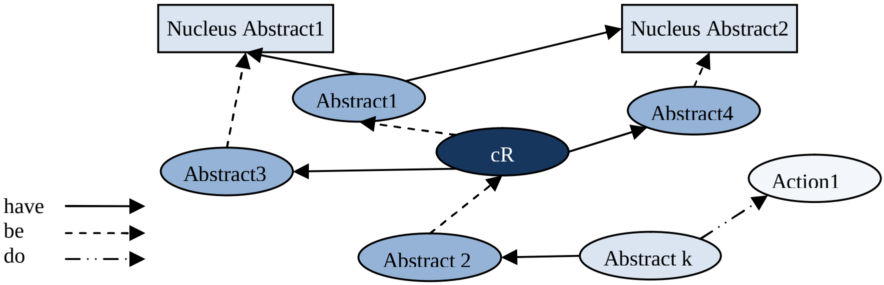

Fig 1.The meaning of cR classifier as a connectivity sub graph GmR centered in the cR.

The section is summarized by describing the **store of meanings** (Fig 2) which is built on the top of the graph of abstracts (abstracts, are the units of meanings) and is designed to serve as an interchange unit with User by covering the following functionalities:

- constructing abstracts,
- managing (modifying, removing) abstracts,
- storing and loading abstracts from permanent storage,
- finding, serializing and returning the abstracts.

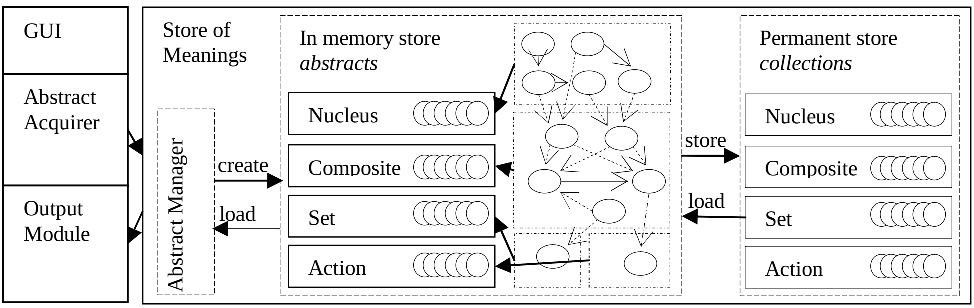

Fig 2. Store of meanings.

Section 2.3 gives the Summary of **Chapter 2.**

Chapter 3 describes and signifies the problem of developing a procedure to find the meanings active in the given situations. It argues, that the matching procedure which determines the active meanings often dominates all other computations and determines the speed of execution of the entire system. Hence, the representation model of the meanings significantly affects the design and efficiency of the matching algorithm.

The following properties of the *be-, have-, do* meaning presentation model as well as the properties of the Solver problem domain itself, make it possible to implement the matching algorithm using the effective approaches of the Rete algorithm developed by Forgy:

- the *be-, have-, do* model allows matching meanings by iterative matching of the sub-meanings,
- there are many meanings which are used repeatedly as sub-meanings in other meanings,
- the changes in sequential situations are usually very small.

Section 3.1 gives the detailed description of the implemented matching algorithm. It operates on the top of the **graph of abstracts** (representing so called **test** nodes, because each abstract in the graph defines some set of regulations) and maintains another graph - **graph of instances** (so called **memory** nodes). **Filtering** and **conjunction** types of test nodes are distinguished in the graph of abstracts according to their functional characteristics. Similarly, **alpha** and **beta** memory nodes are introduced as containers of complete and partial matches attached to the filtering and conjunction nodes respectively as shown in Fig 3.

The algorithm traverses the constituents of situations - nucleus instances, on the graph of abstracts resulting partial and complete matches. It exploits the first two properties of the meaning presentation model by sharing common test and memory nodes. It takes advantages from the third property by storing match information in the graph of instances between cycles, so that it only matches a unit of a situation - a nucleus instance, against each condition once, even if (as is likely) the nucleus instances remains in situations for many cycles.

When a new instance is added to the situation, the matching procedure "filters" it down the graph of abstracts, causing new partial matches to be formed and possibly causing one or more abstracts to be instantiated and passed to further processing. Similarly, the nucleus instance removal procedure "filters" down the removed nucleus instance causing the partial matches including the connected instances to be removed from the memory nodes.

Although, Rete is considered as one of the most effective pattern matching algorithms still has certain shortcomings which are underlined in Table 2.

Subection 3.1.2 represents the specific algorithms developed for the activation of different type of abstracts. Here the sequence of nucleus abstracts is modeled as a branch of filtering nodes which ends with a reference to an alpha memory node. Correspondingly, composite (including ar1s) abstracts are represented as nodes with both filtering and conjunction characteristics. Straightforward algorithms for their activation from the activation of their attributes is represented with the discussion of partial instance matching and instance queuing mechanisms. The matchings of Set abstracts and Actions are considered as analogical procedures of composite abstract

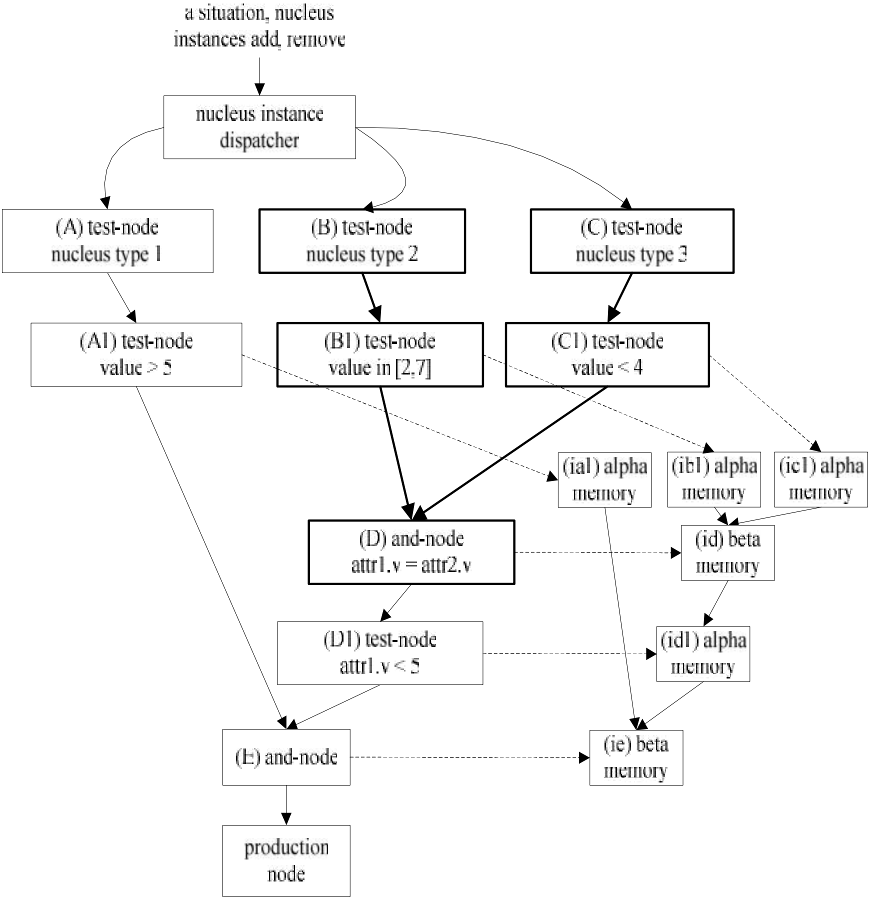

Fig 3. The structure of a small graph of abstrtacts and graph of instances. The pairs of nodes labled (A) - (A1), (B) - (B1), (C) - (C1) as well as (D) - (D1) are representing the filtering branches. The nodes labeled (D) and (E) are representing conjunction or and nodes. It can be observed that each filtering branch has an attached alpha memory node (labeled (ia1), (ib1), (ic1) and (id1)). Similarly, each conjuntion node has a beta memory node attached (nodes, with (id) and (ie) labels). The composition of nodes with thick borders reresent the rooted tree of the abstract labeled with node (D).

activation. Finally, activations of *virtual* and *usage* abstracts are discussed as special types of filtering nodes.

In ***Section 3.2*** the representation of the constituents of a **t-print** is considered. *t-prints* are defined as bundles of output objects or the indices of the active at a time t *perceptors*, *utilities* and *abstracts*. The mentioned components are modeled as follows:

- the output of perception is modeled as a composition of instances of certain abstracts satisfying several basic conditions (ar1s),
- the active abstracts are modeled as a collection of names (indices) of the abstracts which got activated during the processing cycle of the input of a perception,
- and, utilities, as values assigned to the prints by certain *evaluators* of the Solver.

The section is summarized by describing the store of t-prints as a sequence of t-prints - representing some kind of snapshots, and delta-prints - representing the change between two successive prints.

Table 2. The shortcomings of the Rete algorithm and the developed solutions within the model.

Shortcomings	Solutions
The size of the used memory may be combinatorially explosive.	N/A
Sharing network structure is not advantageous for parallel matching.	Introducing new concept - *graph of instances*, separate from graph of abstracts, and achieving parallelism by processing different branches of the game tree.
To maintain consistent state in the network Rete must perform extensive computation for abstracts that are inactive.	Partial active abstracts are cached and reused in the successive situations.

Section 3.3 summarizes the **Chapter 3**.

Chapter 4 is devoted to the development of techniques for the evolvement of meanings. It discusses and underlines the following shortcoming of using OO paradigms for meaning representation:

- it is necessary to start the descriptions from the **root** of the hierarchy, while *categorization starts from the middle.*

This, implies that experts know **everything** about the universe to be modeled. However, as argued in the literature, it is not. Therefore, to handle this implication the chapter develops certain techniques for meaning acquisition, which permit the engineers/experts to start descriptions intuitively from the basic level and find the super levels in due course.

To restrict the "space" of the meaning in the graph of abstracts ***Section 4.1*** introduces the idea of the **skeleton of a meaning**, which, basically, defines the key units of the meaning by covering its most important components - the components which are directly used in the activation of the meaning.

Section 4.2 develops the basic techniques for the meaning evolvement in the form of elementary generalization and specialization operations within the *be-*, *have-*, *do-* meaning presentation model. It outlines **Increase** and **Substract** elementary generalization as well as

Restrict and **Disjoint Sum** elementary specialization operations. In their turn, the generalization and specialization of a meaning are defined by the sequence of elementary generalization and, respectively, specialization operations.

Section 4.3 considers the generalization of two meanings. To define the degree of generalization (how close is it to the (or a) least generalization) the concept of **strongly compatible** abstracts is introduced based on their proximity in the *be* connection chain. In the subsections, the detailed descriptions of the algorithms developed for the generalization of various types of abstracts are discussed. It is claimed that the generalization of other abstracts is either a special case or can be brought to the generalization of composite abstracts. Consequently, the efficiency of generalization of two meanings is argued to be assessed by the efficiency of the generalization of composite abstracts.

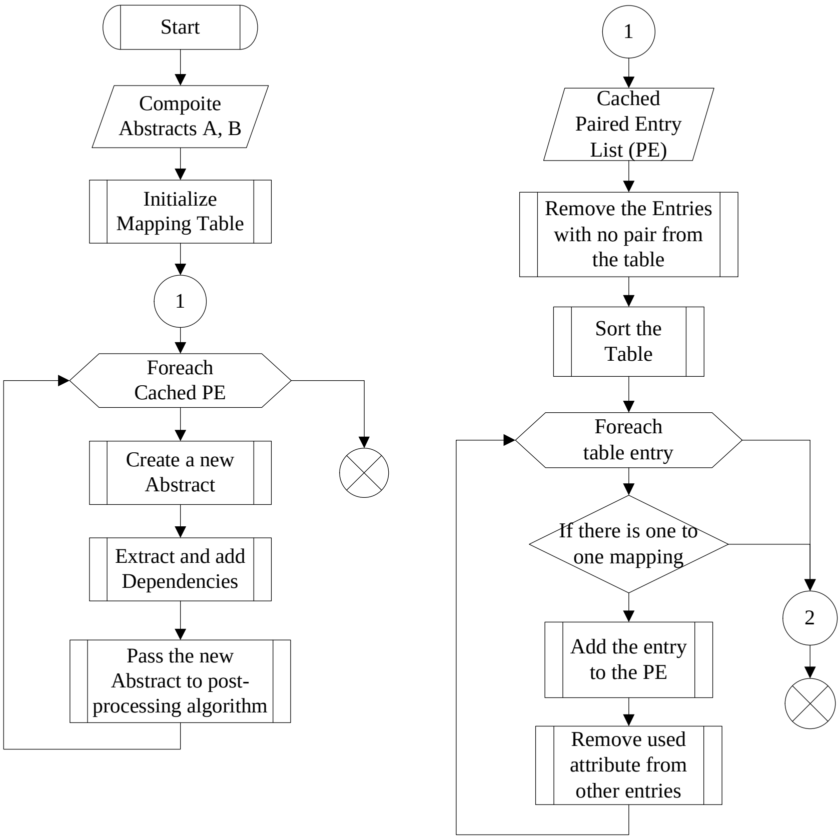

Fig 4. The flow chart of the algorithm of two composite abstract generalization (continued in Fig 5).

Subsection 4.3.4 describes the generalization algorithm of composite abstracts. Fig 4 and Fig 5 represent the flow chart of the algorithm. It uses the semantics of the *be* connection to find the meanings which are *strongly compatible*. In other words, the closest pairs of meanings in the *be* connection chain. Further, to deal with the extraction of relations/dependencies between attributes the algorithm of equivalence checking of arithmetic expressions developed by Ghodrat is adopted.

In ***subsection 4.3.5*** the algorithm of extraction of a virtual abstract by the generalization procedures is discussed. The importance of the latter one is argued as a procedure of extraction new

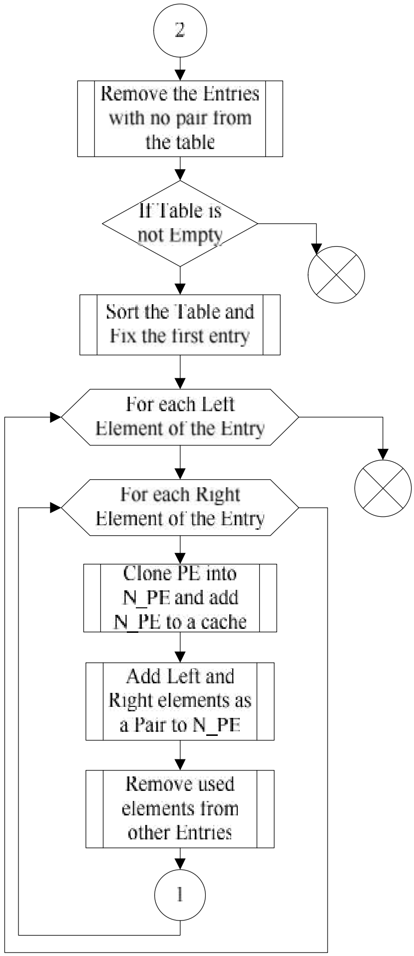

Fig 5. The flow chart of the algorithm of two composite abstract generalization (continuation of Fig 4).

classes analogical to the *interfaces* of OO languages. The section is summaries by underling that the proposed techniques can be used to:

- find common parts of two meanings,
- dynamically generate and integrate a new meaning between the *be* connection chains,
- extract a common Interface.

The above achievements solve the issues like *dynamically change the class hierarchies* and *automatically identify the inheritance relations*. The solution to the first problem can be modeled as a generalization of two abstracts which have a common parent. In this case, by checking the homomorphism from the base abstract to the generated one, it is possible to alter the inheritance relation by injecting the new abstract into the existing hierarchy. Similarly, the second issue can be solved by checking isomorphism of the generalized abstract with one of the source abstracts. On the grounds that both homomorphism and isomorphism checking algorithms are aware about the attributes pairs which shall be compared, the complexity of these procedures is reduced to polynomial time.

Section 4.4 talks about the problem of the evaluation of the usefulness of the generalized meanings in the automated generalization procedures. It defines the utility of the meaning as its relevance to the "winningness". ***Section 4.5*** discusses the approaches of generalizing multiple meanings.

In the ***Section 4.6*** is exposed the summary of ***Chapter 4.***

Chapter 5 is devoted to the experiments of the regular acquisition of expert knowledge of SSRGT problems as well as to the prospects of integrating the SSRGT Solver framework into the OMD LLC for financial news analysis.

Section 5.1 describes the experiments of chess problem. Particularly it presents the detailed acquisition procedures of certain important chess meanings ("Mate" etc.). In **Section 5.2** the expert knowledge forms of Intrusion Protection problem are presented. It is shown that they are either goals or rules and are represented as production rules. The latter ones can be obviously described by the abstracts' description Interface of the SSRGT Solver.

Section 5.3 is devoted to the experiments of acquisition Supply Chain Management (SCM) problem knowledge into the Solver. Various nucleus abstracts describing the smallest components of the TAC SCM (Trading Agent Competition) domain and their compositions as TAC object types are developed. In the section we argue that generic terms using to form strategies like "action over an object" can be specified by defining virtual actions with corresponding specifications for virtual objects.

Section 5.4 describes the study of using the SSRGT package for regular acquisition of financial news analyst expert's strategic knowledge via graphical interface. The research is aimed to develop a software to detect the units of relevant news from the vast sources of the financial news (Fig 6). The solution is composed from three stages:

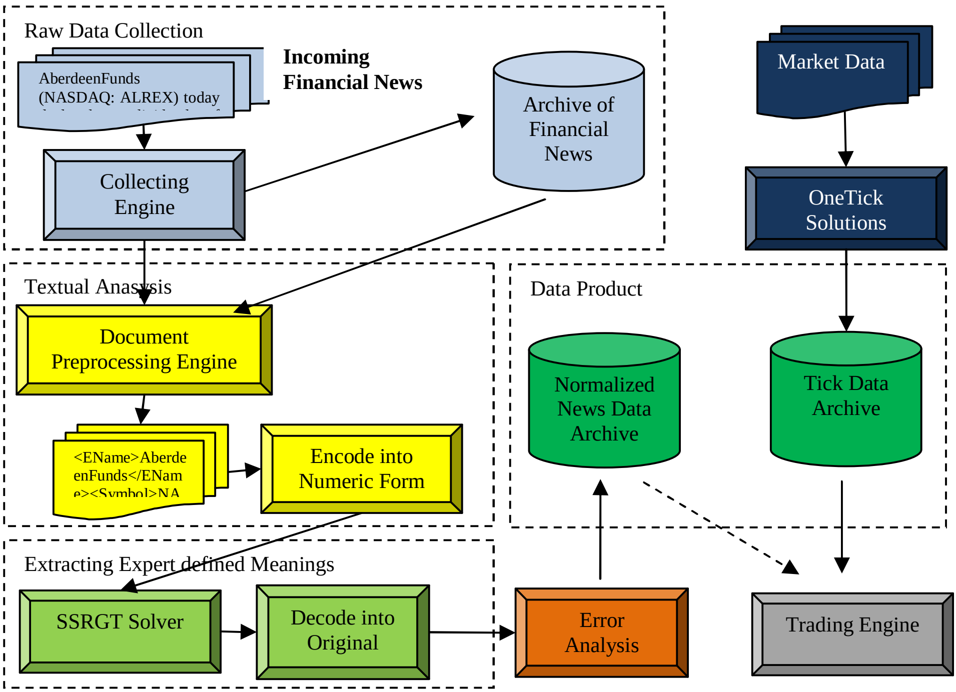

Fig 6. The design of the system analyzing financial news using SSRGT Solver and integrating them into the News Data archive, which along with Tick Data collected by the OneTick solutions would be used by the Trading strategy developers.

- Expert knowledge acquisition,
- Matching news to the acquired meanings,
- Error analysis.

In the first phase of the solution we distinguish and define the nucleus abstracts of the financial news. These are represented as various abstracts describing different classes ("Company", "Person", "Money", "Date" etc.).

Next by combination of different nucleus abstracts more complex abstracts like: (a) Initial Public Offering - a type of public offering where shares of stock in a company are sold to the general public; (b) Mergers - corporate finance and management dealing with the buying, selling, dividing and combining of different companies and similar entities; (c) Stock Splits - increase the number of shares in a public company; (f) Spin-offs - type of corporate action where a company "splits off" sections of itself as a separate business and others are constructed.

The second phase of the solution includes building instances of different nucleus abstracts from the news and matching them to the meanings defined by the SSRT Solver. To solve this problem we used a two level classification of the news text. As a result defined type of classes are assigned to different words and word phrases. Corresponding nucleus instances are built from the resulting classified sentence (for each class attached to the given word or phrase a separate instances is built) and the output is passed to the Solver in a form of numeric sentence (a group of instances). The latter triggers the matching algorithm and activates the instances of the defined meanings. In the next phase of the procedure the numeric sentences are translated into the texts and passed to the error checking block.

Although, the current implementation of the error checking assumes the intervention of the expert it significantly reduces the amount of time spent to find and analyze the interesting news.

Section 5.5 presents the summary of the **Chapter 5**.

Main results of the work

1. The modules of SSRGT Solver for acquisition of communalized and personalized lexical units are developed, specified by multiple level of nested attributes and polymorphic properties in the framework of expert strategy knowledge acquisition [1].
2. A model and algorithms for presentation of SSRGT meanings by *be-*, *have-*, *do* language categories and their evolvement are developed and implemented in an extended Java language [2, 4].
3. The techniques of matching situations to the meanings by iterative matching to sub-meanings of meanings are developed and realized based on the pattern matching algorithms [3].
4. The algorithms based on combinative generalization approach for building new meanings and new relations between them from the existing ones are developed and its effectiveness is experimentally proved [4].
5. The problem of analysis of financial news in the "OMD" LLC is studied in frame of SSRGT class for application of SSRGT Solver.

List of Publications

[1] Khachatryan, K., Vahradyan, V., "Graphical Language Interpreter Unified for SSRGT Problems and Relevant Complex Knowledge", CSIT11: Proceedings of International Conference in Computer Science and Information Technologies, Yerevan 2011, pp. 178-182.

[2] Khachatryan, K., Grigoryan, S., "Java Programs for Presentation and Acquisition of Meanings in SSRGT Games", Proceedings of SEUA Annual conference, Yerevan 2013, pp 135-141.

[3] Khachatryan, K., Grigoryan, S., "Java Programs for Matching Situations to the Meanings of SSRGT Games", Proceedings of SEUA Annual conference, Yerevan 2013, pp 127-135.

[4] Khachatryan, K., "A Combinative Approach to Generalization of Meanings", Transactions of IIAP of the NAS of RA, Mathematical Problems of Computer Sciences, Yerevan 2013, pp. 54-65.

Կ. Ս. Խաչատրյան

Հակադրության խնդիրներում փորձագիտական գիտելիքների բազաների ձևավորում և դրանց փնտրման ալգորիթմերի մշակում

ԱՄՓՈՓԱԳԻՐ

Հետազոտության նպատակն է մշակել ծրագրային համակարգ և ալգորիթմներ շարահյուսական միավորների իմաստների ներկայացման, իրավիճակներին՝ դրանց համապատասխանեցման, ինչպես նաև՝ առկա իմաստներից նոր իմաստների կամ առկա իմաստների միջև նոր հարաբերությունների կառուցման համար:

Շարահյուսական միավորների իմաստների մոդելավորման և մշակման խնդիրը հատկապես կարևորվում է համացանցում իմաստային որոնման ժամանակակից հետազոտություններում:

Փնտրման ոլորտում առաջատար ընկերությունները, ինչպիսիք են Google-ը և Microsoft-ը, առաջարկում են իմաստային որոնման ընդլայնումներ՝ "Knowledge Graph" և "Satory" համակարգերի տեսքով:

Այնուհանդերձ, բնական լեզվական նախադասությունների իմաստների ըմբռնման, ինչպես նաև՝ անգլերենից բացի այլ լեզուներով իմաստային փնտրման խնդիրները դեռևս մնում են իմաստային որոնման համակարգերում՝ իբրև հետազոտական մարտահրավերներ:

Բնական լեզվական նախադասությունների իմաստային ըմբռնման խնդիրը լուծված չէ նաև մեքենայական թարգմանությունների ոլորտում: UNL (Universal Networking Language) նախագիծը, ներառյալ մշակումը հայերեն լեզվի համար, SOAR նախագիծը (Միչիգանի համալսարան) համակարգիչների կողմից իմաստային ըմբռնման հետազոտությունների այլ օրինակներ են:

Է. Պողոսյանի աշխատություններում ցույց է տրվել, որ կարևոր և ընդարձակ մրցակցային դասի խնդիրները՝ ներառյալ համակարգչային ցանցերի պաշտպանությունը ներխուժումներից, մրցակցային միջավայրում արդյունավետ մենեջմենթի և մարքեթինգային ռազմավարությունների մշակման խնդիրը,

շախմատանման կոմբինատոր խնդիրները, տարատեսակ հարձակումներից ռազմական միավորների պաշտպանությունը, հաշվողական գործընթացներում շեղումների հայտնաբերումն ու այլ խնդիրներ բերվում են լուծումների բազմությունը վերարտադրելի խաղային ծառի միջոցով ներկայացվող դասի (SSRGT) ստանդարտ միջուկային **K** խնդրին: K - մեթոդաբանության կարևորությունն այն է, որ SSRGT դասի առանձին խնդիրներում ձեռքբերումները բազմապատկվում են:

Իմաստավարության բարդության պատճառով և առավելագույն արդյունավետության հասնելու նպատակով, մենք կենտրոնացնում ենք մեր հետազոտությունները SSRGT դասում և ընտրում շախմատը՝ որպես միջուկային խնդիր:

Աշխատանքի նպատակները

- Ընդլայնել SSRGT դասի կամայական խնդրի փորձագիտական գիտելիքների ձևավորման և ընկալման գրաֆիկական լեզվի ինտերպրետատորը և ինտերֆեյսը:
- Ձեռք բերված իմաստների ներկայացման և գործածման համար նախագծել և մշակել ծրագրային համակարգ:
- Մշակել իրավիճակներն իմաստներին համապատասխանեցման ալգորիթմներ:
- Մշակել համակցական ալգորիթմներ առկա իմաստներից նոր իմաստների կամ առկա իմաստների միջև նոր կապերի կառուցման համար:

Գիտական նորույթը

1. Փորձագիտական գիտելիքների ձեռքբերման SSRGT Solver ծրագրային համակարգում մշակվել են բազմամակարդակ ներդրված ատրիբուտներ և բազմաձև (polymorphic) հատկություններ ունեցող ծրագրային միավորներ՝ հաղորդակցվելի և անձնավորված շարահյուսական միավորների ներկայացման համար [1]:
2. Իրականացվել են ալգորիթմներ և ծրագրեր SSRGT խնդիրների իմաստների ներկայացման և զարգացման համար՝ հիմնված լեզվի շարահյուսության "լինել-, ունենալ-, անել" (be-, have-, do) մոդելի վրա [2, 4]:
3. Մշակվել և իրագործվել են իրավիճակներն իմաստներին համապատասխանեցման ալգորիթմներ՝ հիմնված ենթաիմաստների իտերատիվ համապատասխանեցման և սահմանափակումների բավարարման մեթոդների վրա [2, 3]:
4. Մշակվել և իրագործվել են առկա իմաստներից նոր իմաստների կառուցման ալգորիթմներ՝ հիմնված համակցական ընդհանրացման մոտեցման վրա [4]:

Կիրառական նշանակությունը

1. Ծրագրային փաթեթը կարող է կիրառվել SSRGT խնդիրների փորձագիտական հաղորդակցվելի և անձնավորված գիտելիքների բաղադրիչների ձևավորման, ընկալման և համապատասխան իրավիճակներում դրանց հայտնաբերման նպատակով:
2. SSRGT Solver ծրագրային համակարգը տեղադրվել է Հայաստանի ամերիկյան համալսարանում և ՀՀ ԳԱԱ ԻԱՊԻ -ում՝ կրթական և հետազոտական

նպատակներով, ինչպես նաև "Օ Էմ Դի" ՍՊԸ-ում ֆինանսական նորությունների վերլուծության համար:

3. Իմաստների ներկայացման "լինել-, ունենալ-, անել" մոդելը ժառանգում է օբյեկտ կողմնորոշված (ՕԿ) ներկայացման հատկությունները և միաժամանակ ապահովում ժամանակակից ՕԿ լեզուների կոմպիլյատորների և ինտերպրետատորների անկատարությունների հաղթահարման լրացուցիչ հատկություններ:

К. С. Хачатрян

Разработка алгоритмов и программ для формирования и поиска в базах знаний проблем противодействия и конкуренции

РЕЗЮМЕ

Целю исследования является разработка программного обеспечения и алгоритмов для представления смысла синтаксических единиц, сопоставление ситуаций этим смыслам, а так же выявление смысла из уже доступных значений синтаксических единиц и построение новых отношений между уже доступными.

Проблеме моделирования и обработки значений лексических единиц придается особое значение при семантическом поиске в Интернете.

Поисковые гиганты, как Google и Microsoft предлагают семантические расширения для поисковых систем в виде Knowledge Graph и Satory.

Однако, проблемы понимания смыслов предложений естественного языка, а также поддержка семантического поиска в языках отличных от английского, до сих пор являются задачами исследований в сфере веб-поиска.

Проблема понимания смысла предложений естественного языка до сих пор не решена и для машинного перевода. Другими примерами таких исследований являются проект UNL (Universal Networking Language) с ее проекцией для армянского языка и проект SOAR в Мичиганском Университете.

В работах Погосяна показано, что важный и широкий класс задач противодействия и конкуренции, включая защиту компьютерных сетей от вторжений, оптимальный менеджмент в конкурентной среде, защита обьектов от атак, проблема обнаружения аномалий в вычислительных процессах и.т.д., сводятся друг к другу. Множество решений этого класса задач SSRGT можно представить в виде репродуцируемого игрового дерева, а исследования для класса можно сосредоточить для одной из них распространив затем успешные результаты на весь класс.

С целью получения конструктивных результатов наши исследования проблемы оперирования смыслами проводятся для проекции проблемы в класс SSRGT, в то время

как эксперименты по подтверждению утверждений проведены для представителя класса SSRGT – шахмат.

Цель работы

- Расширение интерпретатора и интерфейса графического языка, для поддержки формирования, приобретения и использования стратегических знаний произвольных проблем класса SSRGT.
- Проектирование и разработка пакета программного обеспечения для усвоения новых смыслов и управления приобретёнными.
- Разработка алгоритмов для классификации ситуаций по их смыслам.
- Разработка алгоритмов построения новых смыслов и выявления отношений между уже существующими.

Научная новизна

1. Разработаны модули приобретения коммуницированных и персонализированных стратегий пакета SSRGT Solver с возможностями полиморфизма и множественности уровней вложенных атрибут [1].
2. Разработаны структуры представления смыслов SSRGT проблем, основанные на интерпретации и расширении структур абстрактных классов ОО языков синтактическими категориями глаголов Быть-, Иметь-, Делать (Be-Have-Do) [2, 4].
3. Разработаны алгоритмы и программы смысловой классификации ситуаций, основанные на итеративном сопоставлении ситуаций элементам декомпозиции смыслов и удовлетворении заданных ограничений [3].
4. Разработаны алгоритмы и программы обобщения смыслов посредством анализа и композиции из уже существующих [4].

Практическая значимость

5. Пакет Solver находит примененение в приобретении знаний и выявления смыслов ситуаций SSRGT проблем.
6. Пакет Solver установлен в АУА и в Институте проблем информатики и автоматизации НАН РА в образовательных и исследовательских целях, а так же в "OMD" ООО для анализа финансовых новостей.
7. Модель Быть-, Иметь-, Делать представления смыслов наследует свойства ОО языков одновременно преодолевая некоторые несовершенства компиляторов и интерпретаторов современных этих языков.

Ծավալը՝ 22 էջ: Տպաքանակը՝ 100:
ՀՀ ԳԱԱ ԻԱՊԻ կոմպյուտերային պոլիգրաֆիայի լաբորատորիա:
Երևան, Պ. Սևակի 1