Computing and Informatics, Vol. 27, 2008, 593–625

AGENT-BASED MEDICAL DIAGNOSIS SYSTEMS

Barna László Iantovics

Department of Computer Science
Petru Maior University
Iuliu Maniu 1, 540088 Tg. Mures, Romania
e-mail: ibarna@davos.science.upm.ro

Manuscript received 7 December 2006; revised 21 May 2007
Communicated by Patrick Brézillon and Jozef Kelemen

Abstract. Medical diagnostics elaboration many times is a distributed and cooperative work, which involves more medical human specialists and different medical systems. Recent results described in the literature prove that medical diagnosis problems can be solved efficiently by large-scale medical multiagent systems. Cooperative diagnosing of medical diagnosis problems by large-scale multiagent systems makes the diagnoses elaborations easier and may increase the accuracy of elaborated diagnostics. The purpose of the study described in this paper consists in the development of a novel large-scale hybrid medical diagnosis system called LMDS. The LMDS system is composed from physicians, medical expert system agents developed in our previous works and medical ICMA agents. Medical ICMA agents represent a novel class of agents with the ICMA architecture developed in our previous works, endowed with medical diagnosis capability. The main novelty of the LMDS system consists in the novel classes of agent members of the system and the manner in which the members of the system contribute to the problems solving. Each diagnostics can be elaborated cooperatively by more members of the system. The diagnosis system can solve difficult medical diagnosis problems whose solving must be discovered cooperatively by the members of the system. Many difficult medical problem solving requires medical knowledge that cannot be detained by a single physician or a medical computational system. Simulations prove the correctness in operation of the LMDS system.

Keywords: Medical diagnosis, medical diagnosis systems, complex systems, intelligent agents, multiagent systems, cooperative problem solving, medical applications, computational methods in medicine, applications to biology and medical sciences, medical expert system agents

1 INTRODUCTION

The *agents* represent cognitive systems with proprieties, like [2, 8]: increased autonomy in operation, communication and cooperation capability with other systems and learning capability. In the following, agents are humans and the artificial systems with agents' proprieties. An agent has [2, 8, 24] a problem solving *capability* and *capacity*. The capability of an agent consists in the *specializations* that the agent can use in the problems solving. A problem solving specialization represents a problem solving method. The capacity of an agent defines the resources that the agent can use in problem solving. The systems composed from more agents are called *multiagent systems* [25, 8, 41]. Many difficult problems solving is a naturally cooperative process, whose solving implies a cooperative multiagent system. Motivations of the use of cooperative multiagent systems in problem solving consists in the limited capabilities and capacities of individual agents.

Important applications of the agents and multiagent systems are in different problem solvings that appear in medicine [42, 43, 44, 36, 45, 54, 55, 58, 36, 62]. Many researches in the use of the agents in medicine represent the recent research direction, which intends to eliminate disadvantages of earlier developed medical computational systems (medical expert systems for example), that usually consist in limited autonomy, interaction capability with the environment and intelligence in the problem solving. As examples of medical problems that can be solved by artificial agents, medical diagnostics elaboration, medical data collections about patients, medical knowledge search, medical decisions support, pro-active assistance of the physicians during cooperative problem solving (for example, cooperative finding of the answer to the medical issue "the effectiveness of the latest treatment to cure an illness in a very advanced stage" by more physicians helped in their cooperation by their assistant agents) etc. can be mentioned. Medical diagnostics elaborations are often naturally distributed and cooperative processes, which involve human medical specialists and different medical systems [2, 14, 23, 36, 45]. Many medical diagnosis problems can be solved efficiently by large-scale medical multiagent systems [7, 9, 13]. In this paper, a novel hybrid large-scale medical diagnosis system that combines the advanteges of humans and agents in medical diagnostics elaborations is proposed.

The paper is structured as follows.

In Section 2, agent-based applications used for medical problems solving are presented. Medical problems are described, for whose solving agent-based approaches and large-scale medical diagnosis systems are used. A novel class of agents called *medical expert system agents* is presented, developed in our previous works, used as a member in the novel *BMDS* (*Blackboard-Based Medical Diagnosis System*) and *CMDS* (*Contract Net-Based Medical Diagnosis System*) medical diagnosis systems.

In Section 3, a novel medical diagnosis system called *LMDS* (*Large-Scale Medical Diagnosis System*) is proposed. A novel class of medical agents members of the developed system is presented, called *medical ICMA agents*. Previous works related with the medical ICMA agents are described; advantages of the LMDS system and the motivations of the system's development are presented.

In Section 4, the conclusions related with the proposed LMDS diagnosis system and the following research that will be made are presented.

2 AGENT-BASED MEDICAL PROBLEMS SOLVING

Agents has been applied for many problem solving that appear in the medical domains. As examples of applications of the agents and multiagent systems in medicine, we mention: *patients monitoring* [42], *patients management* [43, 44], *healthcare* [36, 45], *telehealth* [54], *healthcare emergency coordination and decision-support* [55], *web-enabled healthcare computing* [58], *ubiquitous healthcare* [36] and *simulation of spreads of infectious diseases* [62]. The paper [14] describes the state of the art of medical information systems and technologies at the beginning of the 21^{st} century. The complexity of construction of full-scaled clinical diagnoses is also analyzed.

A *medical diagnosis problem* consists in the description of an illness (combination of illnesses). The *solution of a medical diagnosis problem* represents the identified illness (illnesses) and the treatment (treatments) that must be applied to cure the illness (illnesses). The establishment of a medical diagnostic may have different difficulties [14, 51, 52, 53]. A patient may have combinations of illnesses; each of them may have different symptoms and syndromes, there may be dependencies between the patient's illnesses treatments (for example, a very effective medicine to cure an illness has negative effects to cure the other illness). The symptoms of more illnesses may have some similarities, which make their identification difficult. The symptoms of an illness can be different with different persons who suffer from that illness. In some situations, a patient does not exhibit the typical symptoms of a specific illness even if s/he suffers from it. In the case of some illnesses, the causes of the illnesses are not sufficiently known. A medicine to an illness may have different effects at different persons who suffer from that illness. A person may have allergy to a medicine (this information can be known in the person's medical history). An illness can be in a very advanced stage, that makes the diagnostic elaboration difficult (a usual treatment known to be effective to cure the illness in a less advanced stage cannot be applied). For example, we mention a huge tumor whose treatment requires more surgery interventions that must be realized in time. The treatment of such an illness must be carefully planned analyzing different situations (the loss of a huge quantity of blood during a surgery intervention) that can appear during the treatment application. Difficult medical cases are those, in which the patients' illnesses does not sufficiently match typical patterns known by physicians. An illness can be insufficiently known in medicine because it is either new or unusual. In such situations, the symptoms of an illness may or cannot be interpreted properly.

In the medical domains many medical diagnosis systems are used that operate in isolation or cooperate [14, 1, 13, 12, 15, 5, 6, 9, 10, 3, 23, 47]. One of the most recent research direction is represented by the *agent-based medical diagnosis* [5, 3, 13, 26]. Many difficult medical diagnosis problems must be solved cooperatively by more agents, members of a multiagent system, endowed with different medical knowledge [5, 3, 9, 10]. Some of the recently developed medical multiagent systems act as assistant of the physicians during medical problem solving. A motivation of cooperative solving of difficult medical diagnosis problems by multiagent systems consists in the limited knowledge and resources of the individual agents. Some diagnosis problems must be solved before a deadline. A subclass of the medical multiagent systems is represented by the *medical hybrid multiagent systems* [3, 5, 23]. A medical hybrid multiagent system is composed from different types of software agents and/or of robotic agents and/or human specialists.

Expert systems represent relatively classical applications that can solve problems like human specialists [50]. Expert systems can be endowed with medical diagnosis capability [12, 29, 32]. As examples of well known medical expert systems, we mention: *MYCIN* [28], *GIDEON* [29], *CARDIAG2* [30], *PUFF* [31] and *CASNET* [32]. In the paper [12], a general methodology based on Computer Algebra for constructing rule-based medical expert systems is proposed. For the implementation of the medical expert systems, the paper [12] proposes the *CoCoA language*.

Expert systems had success in specific, mainly quite narrow fields of medical expertise, but had problems to cover broader areas of expertise. Some of the problems related with the expert systems are their limited: flexibility, adaptability, extensibility and cooperation capability [33, 8]. The endowment of the expert systems with cooperation capability represents an important research direction [8, 33, 15]. In the paper [15], a system called *FELINE* composed of five autonomous agents (expert systems with some proprieties of the agents) endowed with medical knowledge is presented. These agents cooperate to identify the causes of anemia at cats. The paper [15] also presents a tentative development methodology for cooperating expert systems.

The *medical expert system agents* developed in our previous works represent expert systems specialized in medical diagnosis endowed with agents' capabilities [5, 6, 8, 10]. The medical expert system agents can perceive and interact with the environment. They can learn and execute different actions in the environment autonomously. They can communicate with other agents and humans that allow cooperative problem solving. A medical expert system agent can be endowed with specializations in more medical domains, for example with specializations in gastroenterology, endocrinology and rheumatology. Expert system agents can solve more flexibly and precisely a larger variety of problems than the expert systems [5, 8, 10]. The expert system agents increased intelligence versus the expert systems' intelligence is analyzed in [16]. Expert system agents can help physicians intelligently in different medical decisions elaboration [24].

In the papers [10, 3, 23], we have proposed a cooperative hybrid medical diagnosis system called *BMDS* (*Blackboard-Based Medical Diagnosis System*). The BMDS is composed from: physicians, *medical expert system agents* and different classes of assistant agents. The cooperative medical diagnosis problems solving by the diagnosis system is partially based on the *blackboard-based problem solving* [2, 46]. The problem solving by the BMDS system is similar with some cases in which more physicians with different medical specializations plan a difficult diagnostic establishment. The diagnosis system is proposed for difficult medical diagnosis problems solving (patients that suffer from combinations of illnesses) [3]. More agents may contribute to an overtaken medical diagnosis problem in the system, depending on the problem solving specializations. Each agent member of the system is specialized in different aspects of the medical problems solving.

In the papers [6, 5], we have proposed a cooperative hybrid medical diagnosis system called *CMDS* (*Contract Net Based Medical Diagnosis System*) that can solve a large variety of diagnosis problems. The CMDS system is composed from physicians and *medical expert system agents*. The problem solving specializations in the system are distributed between the agents' members. CMDS is an open system that can accept new agents as members. For the allocation of problems for solving, in the CMDS system a novel problem allocation protocol is used [11], which represents an adaptation of the *contract net problem allocation protocol* [25, 2, 63, 64]. The diagnosis system can solve randomly transmitted problems for solving to the agents. Agents may help each other during the problem solving processes by transmitting different useful medical information [5]. Medical information received during a diagnosis process may help an agent in the decisions elaboration, as well as in precise establishment of a problem solving and in the establishment of the best-fitted agent capable to solve a problem.

In the paper [26], a self-organizing medical diagnosis system, mirroring swarm intelligence to structure knowledge in holonic patterns is presented. The system sets up on an alliance of agents specialized in medical diagnosis that self-organize in holoarchy in order to provide viable medical diagnoses. Despite the difficulty of the problems that can be solved, the proposed agents exhibit a simple architecture built on reactive behavior. The main advantage of the diagnosis system consists in the fact that relatively simple agents can elaborate reliable diagnosis.

As more health-care providers invest on computerized medical records, more clinical data is made accessible. Diagnosis systems with built-in functions for knowledge discovery and data mining, concerning extracting and abstracting useful rules from such huge repositories of data, are becoming increasingly important for purposes such as of offering better service or care. In the paper [27], an intelligent medical diagnosis system with built-in functions for knowledge discovery and data mining is described. The implementation of machine learning technology in medical diagnosis systems seems to be well suited for medical diagnoses in specialized medical domains. Automatically generated diagnosis rules may be used in diagnostics elaborations.

In the paper [13], an *Internet-based holonic medical diagnosis system* for diseases is proposed. The proposed medical multiagent system combines the advantages of *holonic systems* and multiagent systems in order to implement an efficient and robust Internet-based diagnosis system for diseases. The proposed system consists of a tree-like structured cooperative alliance of agents specialized in medical diagnoses. Agents at higher levels of the holarchy are specialized in a broader field of diseases, while the leaves are experts in one specific disease. Higher level agents gather the results from all those immediate lower-level agents to which they had assigned a diagnosis request. They evaluate these results in order to come to a more comprehensive conclusion. If the results of such an analysis are not satisfying, the agents may decide to announce the request for a diagnosis to other agents spread over the Internet that may be able to contribute to the diagnosis generation.

Independent LifeStyle Assistant (*ILSA*) [56, 57] implemented by Honeywell Laboratories is an agent-based monitoring and supporting system to help elderly people to live more independently at home, by reducing caregivers load. It consists of a multi-agent system supporting continuous data monitoring via home-installed sensors. The collected data are processed to obtain response planning and machine learning. ILSA is implemented using the *JADE* agent platform.

3 LMDS DIAGNOSIS SYSTEM

3.1 Medical ICMA Agents

An *agent architecture* is essentially a map of the internals of an agent, its data structures, the operations that may be performed on these data structures, and the control flow between these data structures [25].

The software mobile agents can be considered a relatively new paradigm in the area of distributed programming and a useful supplement of traditional techniques like the Client/Server architecture. Mobile agent technology has been applied to develop the solutions for various kinds of parallel and distributed computing problems [48]. Many of the formal modeling of mobile agents is in terms of their mobility, they are not built upon a framework that explicitly supports the intelligent feature of the agents [48, 17, 18]. Many times, the multiagent systems formed by cooperative mobile agents are considered to be intelligent. The mobile agents' intelligence is considered at the level of multiagent system in which they operate. If the mobile agents cooperate they can solve difficult problems intelligently [48, 36]. Other disadvantages of recently developed mobile agents are in limitations related with the [49, 17, 18]: communication capability and protection possibility against different network sources and malicious hosts. The disadvantages mentioned before result from the mobile agents proprieties such as: mobility (the mobile agents migrate in the network during their operation), autonomy in migration, distributed and asynchronous operating manner.

In the papers [7, 4], we have proposed a novel mobile agent architecture called *ICMA* (*Intelligent Cooperative Mobile Agent Architecture*). Mobile agents endowed with the ICMA architecture are called *ICMA agents*. An ICMA agent denoted *MA* is composed from two parts (1): a static part denoted *Ss* and a mobile part denoted *Mp*.

$$MA = \langle Ss; Mp = \{M_1, M_2, \ldots\}\rangle \tag{1}$$

A *subagent* of an agent represents a component of the agent that has agent's proprieties [8, 2]. An agent may have more subagents. *Ss* is the static subagent of *MA* (*Ss* doesn't migrate in the network during its life cycle). The static subagent is responsible for the overtaking of the problems for solving. $Mp = \{M_1, M_2, \ldots\}$ contains a variable number of mobile subagents. *Ss* creates the mobile subagents. During its life cycle *Ss* can create new mobile subagents and eliminate the inefficient or useless mobile subagents. The mobile subagents have all the proprieties of the mobile agents described in the literature, they are responsible for the problems solving at the *hosts* distributed in the network [7, 4]. The hosts execute the problems solving descriptions from the mobile subagents' body. The papers [7, 4] analyze the ICMA agents' knowledge bases and operation.

ICMA agents can communicate efficiently [4]. The communication between different ICMA agents is realized via the subagents of the agents. Mobile subagents at different hosts can communicate using as interloper their creator static subagents (the static subagents addresses are not changing during their live cycle). A mobile subagent may transmit a message to a target mobile subagent, even if the target mobile subagent migrates in the network in the transmission time.

The ICMA architecture offers new security solutions in the protection of the mobile agents against the network sources and malicious hosts [19]. The increased security solutions are offered by the specific distributed operating manner of the ICMA agents (a network source or a host cannot access and/or modify all the information contained in the mobile part of an ICMA agent). During its life cycle, a created mobile subagent leaves the unnecessary knowledge (the mobile part of an ICMA agent deceases in size during a problem solving cycle). During a mobile subagent operation the static subagent can check the mobile subagent status, which hosts specified in the mobile subagent itinerary are not visited and which information contained in the mobile subagent body is modified without authorization.

An ICMA agent can solve problems intelligently [7]. The static subagent represents the intelligent part of an ICMA agent. A static subagent can be endowed with capabilities of the intelligent static agents, and can use resources of the computational system on which it operates. The limitations in the endowment of the mobile agents described in the literature consists in the limited resources and capabilities that they can use in the network and at the hosts. The endowment of a mobile agent with intelligence (for example, a component capable of autonomous learning) increases the agent's behavioral complexity and the body size. A large number of intelligent mobile agents transmitted in the network may overload the network with data transmission. A large number of intelligent mobile agents at a host may overload the host with data processing. ICMA agents can form intelligent cooperative multiagent systems [4]. The motivation consists in the capability of the ICMA agents to use efficiently the knowledge and resources detained by the static subagents for the efficient use of resources at the hosts.

A novel class of agents developed in our previous work is represented by the *ICMAE agents* (*Intelligent Cooperative Mobile Agents with Evolutionary Problem Solving Capability*) [20, 21, 22]. ICMAE agents are agents with the ICMA architecture, endowed with problem solving specializations based on genetic algorithms. The ICMAE agents operation is the same as ICMA agents operation, the only difference is in the specializations used at the hosts. In the case of the ICMAE agents, the hosts execute problem solving specializations based on genetic algorithms. During its life cycle an ICMAE agent can use the resources of its static subagent and resources of the hosts. The developed ICMAE agents prove that the novel ICMA architecture allows the creation of agents that can solve efficiently problems using genetic problem solving methods.

Simulations has been proposed as an efficient method for analyzing the complex nature and dynamic aspects of the mobile agents [59, 60, 61]. Simulations allow mobile agents to be analyzed and tested in a controllable environment before an actual implementation is built. The difference between simulation and the real implementation is that, in the simulation approach, the execution of a mobile agent is under the control of a simulator which can embody most major characteristics of the real operating environment such as the underlying network. Simulations of an ICMA [4, 7] and an ICMAE [21, 22] agent have been made. The simulated mobile agents have solved problems in problems solving cycles. A problem solving cycle begins, when a set of problems is overtaken for solving and is finished when all the overtaken problems are solved. The simulations were made for different sets of overtaken problems, with the purpose to analyze how the number of mobile subagents created by a static subagent influences the performance (the time when all the overtaken problems are solved within a problem solving cycle) of large numbers of problems. For each sets of problems and each number of created mobile subagents, simulations have been made for different overloading degrees of the hosts. In practical applications, more mobile agents may operate simultaneously at the same host that may overload the hosts from the environment in different degrees.

Simulations of the ICMA agent were realized in an environment composed from 10 hosts, transmitting between 40 and 80 problems to the agent for solving at each problem solving cycle. The ICMA agents were endowed with different sets of problem solving specializations (require different problem solving time). The static subagent has created between 1 and 11 mobile subagents at each problem solving cycle. Table 1 presents the improvement of the problem solving time of the simulated ICMA agent, in the solving of 50 or 80 problems, using a single mobile subagent versus the use of 2 to 11 mobile subagents. The first column in Table 1 shows the number of created mobile subagents, with which a single mobile subagent performance is compared in the solving of the same problems. The second and third columns present the improvement in the problem solving (how many times the use of more mobile subagents improves the problem solving time) in the case of a set of 50 and 80 problems averaged, realizing 50 simulations for different overloading degrees of the hosts.

Simulations of the ICMAE agent were realized for problems sets composed between 30 and 65 problems, using between 1 and 10 mobile subagents, in an environment composed from 3 hosts. The ICMAE agent was endowed with problem solving specializations based on genetic algorithms. Table 2 presents simulation results of

Mobile Subagents	*50 problems*	*80 problems*
2 *vs.* 1	1.49	1.64
3 *vs.* 1	1.95	2.48
4 *vs.* 1	2.69	3.35
5 *vs.* 1	3.44	4.53
6 *vs.* 1	3.99	5.50
7 *vs.* 1	4.33	6.24
8 *vs.* 1	7.00	5.54
9 *vs.* 1	8.56	5.86
10 *vs.* 1	10.74	7.12
11 *vs.* 1	11.85	7.67

Table 1. Simulation results of an ICMA agent

the ICMAE agent. The first column in Table 2 presents the number of created mobile subagents. The second, third and fourth columns show the average problem solving time (when the created mobile subagents have solved all the overtaken problems) in the solving of 30, 50 and 65 problems, realizing 50 simulations for different overloading degrees of the hosts.

Subagents	*30 problems*	*50 problems*	*65 problems*
No. 1	462 msec	558 msec	601 msec
No. 2	282 msec	372 msec	403 msec
No. 3	186 msec	258 msec	306 msec
No. 4	138 msec	198 msec	224 msec
No. 5	102 msec	150 msec	175 msec
No. 6	84 msec	138 msec	152 msec
No. 7	74 msec	115 msec	138 msec
No. 8	66 msec	104 msec	109 msec
No. 9	54 msec	95 msec	104 msec
No. 10	43 msec	72 msec	84 msec

Table 2. Simulation results of an ICMAE agent

The simulation results of the ICMA and ICMAE agents prove that the performance of a proposed mobile agent (ICMA and ICMAE) that uses more mobile subagents outperforms the performance when a single mobile subagent is used to solve the same set of problems composed from a large number of problems (sets consisting of between 40 and 80 problems in the case of the ICMA agent and sets consisting of between 30 and 65 problems in the case of the ICMAE agent).

OnkoNet mobile agents, described in the literature, have been used successfully for patient-centric medical problem solving [36]. The paper [36] introduces the notion *ubiquitous healthcare* (any-time/any-place access of health services via mobile computing devices), addressing the access of health services by individual consumers applying to mobile computing devices. This access requires different medical knowledge about the individual health status (relevant recent diseases or already available diagnostics). The OnkoNet mobile agent architecture involves architectures on the macrolevel and microlevel as well as cooperation protocols, and inference models for controlling the system's behavior. A developed medical ontology called *OntHoS* is presented, which consists in a collection of terms and definitions to represent organizational structures and processes in hospitals. The work presented in [36] emerged from a project covering all relevant issues from empirical process studies in cancer diagnosis/therapy down to system implementation and validation.

Mobile agents with the ICMA architecture can be endowed with medical diagnosing knowledge. This novel class of medical agents is called *medical ICMA agents*. Some introductory elements about the medical ICMA agents are presented in [9]. The medical ICMA agents represent a class of agents symilar with the OnkoNet mobile agents, capable to solve medical problems. A proposed medical ICMA agent, denoted MA has a set $Spec(MA) = \{S_1, S_2, \ldots, S_m\}$ of medical specializations detained by its static subagent denoted Ss. As examples of medical specializations with which MA can be endowed, we mention diagnosing knowledge in: general medicine, dermatology, oncology and cardiology. 1 presents a medical ICMA agent denoted MA which uses the $M_1, M_2, \ldots, M_n$ mobile subagents launched for problems solving in the network. $P_1, P_2, \ldots, P_v$ represent the medical problems transmitted for solving to MA. The problems are overtaken by the static subagent Ss of the agent MA.

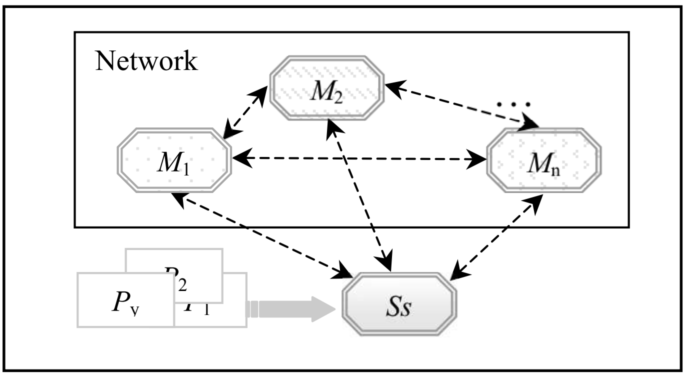

Fig. 1. A medical ICMA agent

The ICMA agents operation is described in [7, 4]. An ICMA agent can be specialized in problems allocation for solving to other agents that operate in the same environment, using mobile subagents created by the static subagent of the agent [4]. The medical ICMA agents operation (problem solving) and problem allocation for solving is the same as the ICMA agents operation, the only difference consists in the specializations used by the static subagent (the static subagent of a medical ICMA agent is endowed with specializations in medical diagnosis).

If the static subagent *Ss*, of a medical ICMA agent *MA*, cannot solve an overtaken problem (does not have the necessary capability and/or capacity), then the problem must be allocated for solving to another agent with medical knowledge. The problem can be allocated for solving by a mobile subagent of *MA* created by *Ss*. A mobile subagent may migrate in the network with an overtaken problem until the problem is solved. During its migration more agents may contribute to the problem solving, each of them making modifications on the statement of the problem solving. Each modification has as purpose to enclose the statement of the problem solving to the solution of the problem (the diagnosis that must be established). *SS* life cycle is the same with *MA* life cycle. A mobile subagent's life cycle is beginning, when the mobile subagent is launched for a problem solving, and is finished when the mobile subagent has returned the overtaken problem solution to its creator – static subagent. During its life cycle, *SS* creates different sets of mobile subagents that are launched for problems solving.

3.2 The LMDS Diagnosis System Description

In this paper, a cooperative hybrid medical diagnosis multiagent system called *LMDS* (*Large-Scale Medical Diagnosis System*) is proposed. Some introductory elements about the proposed diagnosis system are described in [9]. The LMDS system denoted *MDS* is composed from (2): a set *MD* of agents and a set *H* of hosts.

$$\begin{aligned} MDS &= MD \cup H, \\ MD &= D_1 \cup D_2 \cup \ldots \cup D_i, \\ H &= \{H_1, H_2, \ldots, H_i\}, \\ \forall j = \overline{1,i}, H_j &\prec (D_j = \{D_{j1}, D_{j2}, \ldots, D_{jq}\}). \end{aligned} \tag{2}$$

Each host H_j has a set $D_j = \{D_{j1}, D_{j2}, \ldots, D_{jq}\}$ of submitted agents: physicians, *medical expert system agents* and *medical ICMA agents*. Due to the cooperative solving of the medical problems by physicians and artificial agents the physicians are called "agents" (to a problem solving may contribute both artificial agents and humans). The medical ICMA agents have been described in the previous section. The medical expert system agents have been developed in our previous works. The *CMDS* [6, 5] and *BMDS* [10, 3, 23] medical multiagent systems, presented in the previous section, use medical expert system agents as members. Figure 2 presents the proposed LMDS system. $Mob_v, Mob_k, \ldots, Mob_h$ represent sets of ICMA mobile subagents, created by medical ICMA agents, launched to the hosts $H_1, H_2, \ldots, H_i$ for problem allocation for processing.

The *environment* in which the *LMDS* system operates is a heterogenous one. The artificial agents and the hosts operate in a network. The physicians operate in physical medical environments. The agents from the set *MD* can diagnose illnesses corresponding to their capability. The capability of an agent specialized in medicine consists in the medical specializations detained by the agent.

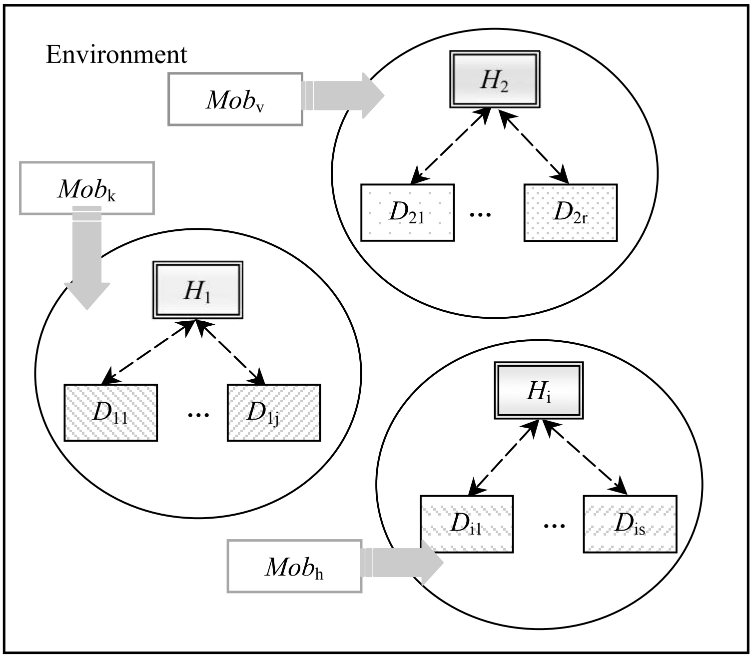

Fig. 2. The LMDS medical diagnosis system

For example, a physician specialized in cardiology can diagnose cardiology related illnesses. An expert system agent may have specializations in more medical domains.

The hosts represent artificial computational systems distributed in the network with capability to accept ICMA mobile subagents [4, 7]. Each host detains information about some of the agents' members of the diagnosis system. We call the agents about which a host detains information "submitted agents" to the host. Each agent is submitted to a single host. As examples of information detained by a host H_u ($H_u \in H$) about a submitted agent Ag_c ($Ag_c \in MD$), we mention: the specializations of Ag_c, the resources detained by Ag_c that can be used in the problem solving, etc. Each host detains different information about the other hosts from the diagnosis system. A host H_u ($H_u \in H$) may have e.g. the following information about another host H_v ($H_v \in H$): the number of agents submitted to H_v, the capabilities and capacities of the agents submitted to H_v, the type (human or artificial) of each agent submitted to H_v etc. Each host can communicate and cooperate with the submitted agents (human and artificial).

A host may assist the submitted physicians during problem solving processes. As examples of assistance that can be offered by a host H_r ($H_r \in H$) to a submitted physician denoted PH_k ($PH_k \in MD$), we mention:

- H_r translates the knowledge transmitted by a mobile subagent into an understandable form to PH_k. For example, H_r may translate an illness symptoms descriptions from English into Italian;
- H_r memorizes medical information transmitted to PH_k until PH_k will have time to overtake the information. PH_k receives the transmitted information when available;
- H_r searches for medical knowledge required by PH_k in distributed knowledge bases. For example, PH_k may require from H_r the description of an illness;
- H_r searches for medical data about a patient in distributed medical databases. For example, PH_k may require the patient's previous illnesses descriptions.

Figure 3 presents a physician denoted PH_k ($PH_k \in MD$) interaction with the *MDS* system. In Figure 3 the following notations are used: H_i represents the host with which PH_k interacts during its operation (PH_k is submitted to H_i); H_j, H_r,..., H_s represent the host about which H_i detains different information (submitted agents to the hosts, submitted agents specializations etc.); Ag_z, Ag_x,..., Ag_c represent the agents (human and artificial) submitted to H_i; M_f, M_g,..., M_h represents ICMA mobile subagents arrived for problems allocation for processing at the host H_i.

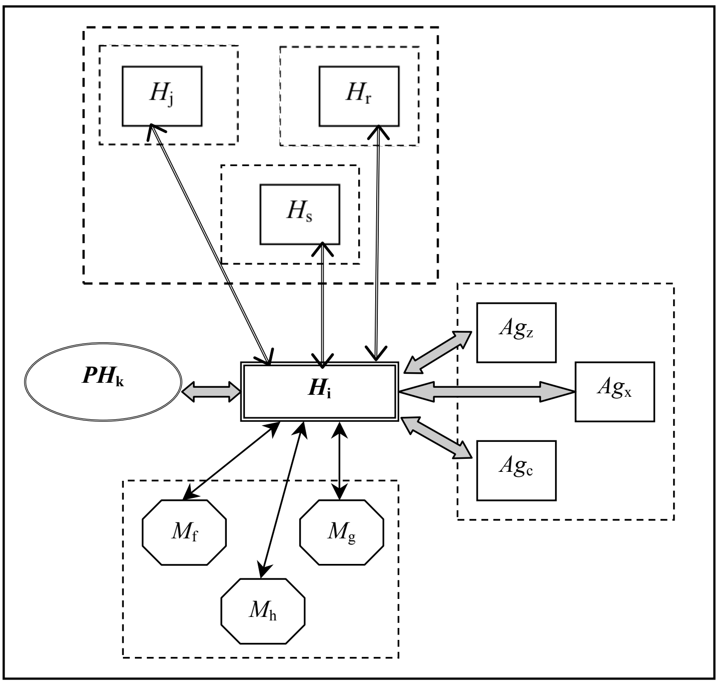

Fig. 3. A physician interaction with the LMDS system

In the following, operation of the LMDS system is described. The problems are transmitted randomly to the medical ICMA agents' members of the system. Each medical ICMA agent can receive problems transmitted for solving. A problem solving cycle is beginning, when the problem is received for solving, and is finished when the problem solution is obtained. The *Algorithm - Cooperative Medical Problem Solving* describes a diagnosis problem P_k solving cycle by the *MDS* system. The static subagent denoted *Ss* of a medical ICMA agent denoted *MA* overtakes the problem P_k description.

Algorithm – Cooperative Medical Problem Solving

{IN: P_k – the medical problem}

{OUT: SOL_k – the solution of P_k}

Step 1

@*Ss* overtakes the information that describes the problem P_k.

Step 2

If (*Ss* has the capability and capacity to processes the problem P_k) *then*

@*Ss* processes the problem P_k obtaining the result Q_k.

If (Q_k represents the problem P_k solution) *then*

$SOL_k = Q_k$

Goto Step 5.

else

@*Ss* creates a mobile subagent M_S.

@*Ss* endows M_S with the information and data known about P_k.

@*Ss* endows M_S with the information Q_k.

@*Ss* based on the problem solving statement and the information detained about the other hosts establishes M_S itinerary I_S.

@*Ss* launches M_S to the first host specified in the itinerary I_S.

EndIf

else

@*Ss* creates a mobile subagent M_S.

@*Ss* endows M_S with the knowledge known about the problem P_k.

@*Ss* establishes M_S itinerary I_S.

@*Ss* launches M_S to the first host specified in the itinerary I_S.

EndIf

Step 3

While (solution of the P_k problem is not obtained) *do*

@The current host H_c tries to find a submitted agent capable to process the knowledge contained in the mobile subagent M_S body.

If (exists at least one capable submitted agent) *then*

@H_c selects the best-fitted agent Ag_y capable to processes the problem.

@H_c transmits the knowledge transported by M_S to Ag_y.

@Ag_y processes the received knowledge obtaining the result Q_k.

@Ag_y transmits the obtained result Q_k to the host H_c.

@H_c endows the mobile subagent M_S with the new knowledge Q_k.

If (the problem P_k solution is obtained) *then*

@Let SOL_k be the solution of the problem P_k.

Goto Step 4.

EndIf

else

If (M_S itinerary I_S doesn't contain an unvisited host) *then*

@H_c establishes (based on the problem solving statement and the information known about the other hosts from the system) a new host H_q unvisited by M_S.

$I_S = I_S - \{H_c\}$.

$I_S = I_S \cup H_q$.

@H_c launches the mobile subagent M_S to the host H_q.

EndIf

@H_c transmits M_S to the next host specified in M_S itinerary I_S.

EndIf

EndWhile

Step 4

@M_S transports the problem P_k solution SOL_k to Ss (M_S is launched to Ss by the current host).

Step 5

@Ss transmits the solution SOL_k to the problem sender.

EndCooperativeMedicalProblemSolving.

The static subagent Ss of the agent MA will solve the diagnosis problem P_k, if it has the necessary specialization and capacity. If Ss cannot solve the problem P_k

(doesn't have the necessary capability and/or capacity), then it will create a mobile subagent M_S, that is endowed with the knowledge (information and data) known about the problem P_k (initial problem solving statement) and an itinerary I_S. As examples of knowledge that can be detained initially about a diagnosis problem (diagnosis of a patient's illness), we mention: the illness symptoms descriptions, the history of the symptoms, the patient's previous illnesses evolution and the patient's allergy to some medicines. The itinerary of an ICMA mobile subagent consists in the hosts that can be visited by the subagent. Each host included in the itinerary is estimated that have submitted agents "capable" (have the necessary capability and capacity) to processes the problem. The purpose of a host visiting consists in the processing of the knowledge detained about the problem (knowledge detained in the problem solving statement) by one or more agents submitted to the host. After a problem processing, a result is obtained (new knowledge) that can represent the problem solution (the identified illness and the established diagnosis to cure the illness) or may help the agents in following processing of the problem. A problem solving statement is changing during a problem solving process, in order to come closer to the problem solution. The itinerary of a mobile subagent is established based on the problem that must be solved and the knowledge detained about the hosts from the diagnosis system. A medical ICMA agent may collect information about the system from the host to which it is submitted. The hosts in the itinerary of a mobile subagent are ordered based on the problem processing capabilities of the agents submitted to the hosts. The first host specified in the itinerary is estimated that contains the best-fitted agent (agents) capable to process the problem. Initially the mobile subagent is transmitted to the first host specified in its itinerary. The mobile agent will visit different hosts until the transmitted problem is solved. The information about a problem solving statement (8), detained in a mobile subagent body, is understandable to the physicians and artificial agents (the information is grouped based on the specifics of the information) and the hosts may extract easily the knowledge that must be transmitted to submitted agents. Each group of information has an identifier that allows the identification of the group of information.

The knowledge detained in a mobile subagent's body contains different information and data obtained during a diagnosis problem solving process. As examples of knowledge contained in a mobile subagent body, we mention: the specification of the necessity to use a physician in the problem solving, the maximum allowed time for the problem solving, the necessary problem solving specialization, the illness symptoms descriptions, medical analyses results, different observations related to the illness, supposed illnesses etc. An agent who processes a problem transported by a mobile subagent agent may add, retract or modify the transported knowledge.

As examples of knowledge that can be added onto a mobile subagent M_S body, by an agent Ag_x ($Ag_x \in MD$) submitted to a host H_f ($H_f \in H$), we mention:

- a new supposed illness. Ag_x supposes that the patient has an illness;

- new questions that must be answered by other agents. Ag_x is limited in knowledge he is specialist in certain medical domains;
- the results of some medical analyses.

As examples of knowledge that can be eliminated from a mobile subagent M_S body by an agent Ag_x ($Ag_x \in MD$) submitted to a host H_f ($H_f \in H$), we mention:

- useless information. Some information from M_S body is not relevant in the diagnosis process;
- a supposed illness. Ag_x demonstrates that the patient does not have the supposed illness written onto M_S body.

As examples of knowledge that can be modified onto a mobile subagent M_S body by an agent Ag_x ($Ag_x \in MD$) submitted to a host H_f ($H_f \in H$), we mention:

- the knowledge that is changed in time. Some medical analysis results are changing in time (a diagnosis process may have a longer duration). Some patients do not describe correctly the symptoms of their illnesses.

In the case of a visitor mobile subagent, the host will verify if it has a submitted agent that can solve or process the problem carried by the mobile subagent. If at least one capable agent is found, then the best-fitted agent is selected, and the knowledge carried by the mobile subagent is transmitted to this agent. If the host does not find a capable agent, then the mobile subagent is transmitted to the next host specified in the mobile subagent itinerary. If the host does not find a capable agent, and the mobile subagent itinerary does not contain an unvisited host, then the host will introduce a new host in the mobile subagent itinerary.

The *Algorithm – Problem Allocation for Solving* describes the process of finding the best-fitted agent by the host H_c, capable to processes the problem P_k carried by an ICMA mobile subagent M_S. To establish the best-fitted agent capable to process the problem P_k, the host H_c announces the problem to a set SUB of submitted agents. In the establishment of the agents to which the announcement An should be sent, H_c uses its knowledge detained about the submitted agents and the knowledge detained in M_S body. As an example of information that can be used in the establishment of the agents to which a problem announcement should be sent we mention the specification in the body of the mobile subagent of the problem solving by a physician (the problem is considered to be difficult). Based on this information, H_c will send the problem announcement An to submitted physicians only. A problem announcement may be answered by more agents that have received the announcement. Based on the responses parameters values the host will choose the best-fitted agent.

Algorithm – Problem Allocation for Solving

{IN: P_k - the medical problem}

{OUT: Ag_x the agent selected for the problem P_k processing}

Step 1

$@H_c$ extracts the knowledge detained about the problem P_k from M_S body.

Step 2

$@H_c$ establishes the problem P_k announcement An.

$@H_c$ establishes the submitted agents SUB to which An should be sent.

Step 3

If $(SUB \neq \oslash)$ *then*

$@H_c$ transmits the announcement An to the selected agents SUB.

$While$ (the waiting time to the announcement An is not expired) *do*

$@H_c$ receives and evaluates the bids to the announcement An.

$EndWhile$

$@H_c$ selects the best-fitted agent Ag_x capable to process the problem P_k.

If $(\{Ag_x\} \neq \oslash)$ *then*

$@H_c$ transmits the knowledge about P_k carried by M_S to Ag_x.

else

@"There is no capable submitted agent to processes the problem P_k."

EndIf

else

@"There is no capable submitted agent to processes the problem P_k."

EndIf

EndProblemAllocationForSolving.

P_k problem announcement An emitted by a host H_v $(H_v \in H)$ has the form (3).

$$\langle Id_g; Knowledge_g; Eligibil_g; Bid_g; Time_g \rangle. \tag{3}$$

Id_g represents the P_k problem announcement identifier. $Knowledge_g$ represents the knowledge detained about the transmitted P_k problem (contains the problem solving statement – information obtained during the problem processing). $Eligibil_g$ specifies the criteria of bid acceptance. As an example of eligibility criteria, we mention the specification to use a physician in the problem processing (P_k is considered to be difficult). Bid_g tells to the contacted agents what information must be provided with the bid. Returned bid specifications gives to the announcement sender host a basis for comparing bids received from more agents. As an example of information that can be provided with the bid specification, we mention the estimated problem solving time. $Time_g$ is the deadline for receiving bids.

A response of an agent Ag_x ($Ag_x \in MD$) to the P_k problem announcement An has the parameters (4).

$$\langle Address_f; Id_f; Resp_f; Capab_f; Capac_f; Relev_f \rangle. \tag{4}$$

$Address_f$ represents Ag_x address. Id_f represents the announcement An identifier. $Resp_f$ represents the bid to the P_k problem solving (acceptance or rejection). $Capab_f$ represents the capability of Ag_x (the specialization that can use Ag_x in the P_k problem processing). $Capac_f$ represents the processing capacity of Ag_x. $Relev_f$ specify the importance of the P_k problem processing by Ag_x (the measure in which the P_k problem processing by Ag_x approaches the solution).

In the following, a cooperative problem solving process by more agents is described formally. The general case is presented, when a static subagent cannot solve an overtaken problem; from this reason he must cooperate with other agents in order to solve the problem.

A Cooperative Problem Solving Process

@*The problem transmission for solving*

$P_z \Rightarrow Ss.$

@*The creation of an ICMA mobile subagent*

$Ss \rightarrow M_j(P_z; I_j = \{H_i, H_r, \ldots, H_k\}).$

@*The cooperative solving of the problem*

$M_j(P_z; I_j) \Longrightarrow \{H_i[A_i], H_r[A_r], \ldots, H_k[A_k]\}.$

EndCooperativeProblemSolvingProcess.

P_z represents the problem that must be solved. Ss represents the static subagent of an ICMA agent who has overtaken the problem P_z for solving. M_j represents an ICMA mobile subagent created by Ss. I_j represents M_j itinerary (specifies the hosts H_i, H_r,..., H_k, estimated that must be visited in order to solve P_z). A_i, A_r,..., A_k represent the sets of agents submitted to the hosts H_i, H_r,..., H_k, that have contributed to P_z processing. The set A_v of agents at a host H_v that contribute to a problem processing can be the empty set. $|A_v|$ denotes the number of agents in the set A_v. $|A_v| = \emptyset$ (if during the problem processing any agent did not contribute at the host H_v) or $|A_v| \neq \emptyset$ (if at least one agent has contributed at the host H_v to the problem processing).

In the *MDS* system each agent has a *role*. The notion *role* is defined in [2, 8]. A role in a multiagent system defines the manner in which the agents that take over the role contribute to the problem solving. An agent who takes over a role must have a set of specializations, which allows the agent to fulfil its role in the multiagent system. In the *MDS* system there are two roles (Figure 4): by diagnosis problems processing denoted *process a*nd decision making about the diagnosis problems processing denoted *decision*. An agent who takes over the *process* role will contribute to the problems processing during its live cycle. The medical ICMA agents, physicians and medical expert system agents have the *process* role. An agent who takes over the *decision* role will contribute to the problems solving during its live cycle, by deciding what to do with the overtaken problems (in the case of an overtaken problem they will decide to which agent it should be sent for processing). The medical ICMA agents and the hosts have the *decision* role. The medical ICMA agents have both roles (*decision* and *process*).

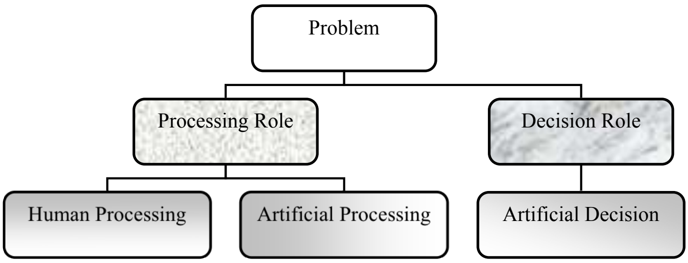

Fig. 4. The roles in the LMDS system

The knowledge used by an agent must be represented using a *knowledge representation language* understandable by the agent (the agent can solve problems using the knowledge). In a multiagent system *ontologies* (dictionaries of used terms) must be used necessary in specifying the used knowledge meanings. The papers [2, 8] define the notions *knowledge representation language* and *ontology*. A medical ontology represents a dictionary of medical terms [40]. As examples of developed medical otologies described in the literature, we mention: *GALEN* [34], *UMLS* [35], *OntHoS* [36], *LinkBase* [37], *TAMBIS* [38] and *GENE* [39].

Figure 5 presents the knowledge and rationality distribution in the *MDS* system between the system's members. $PH_1, PH_2, \dots, PH_c$ represent the physician members of the system. $Ar_1, Ar_2, \dots, Ar_b$ represent the artificial agent members of the system. Each physician can solve problems based on the medical knowledge s/he detains. Each artificial agent can solve problems based on a knowledge base that contains the specializations of the agent. A medical specialization detained by an artificial agent is represented as a set of rules by the form specified in (7) and (8) presented in Section 3.3.

The knowledge of the artificial agents and the knowledge (information and data) detained about the diagnosis problems solving statements are represented symbolically (symptoms of the illnesses, syndromes of the illnesses etc. – are represented using words in a natural human language) and numerically (medical analysis results – some parts of them may have numerical representation). The knowledge detained about a diagnosis problem solving statement has the form (9). If a diagnosis problem is transmitted for solving by an ICMA mobile subagent, then the knowledge about the problem solving statement is detained in the subagent body. The knowledge detained in a problem solving statement (9) is "understandable"

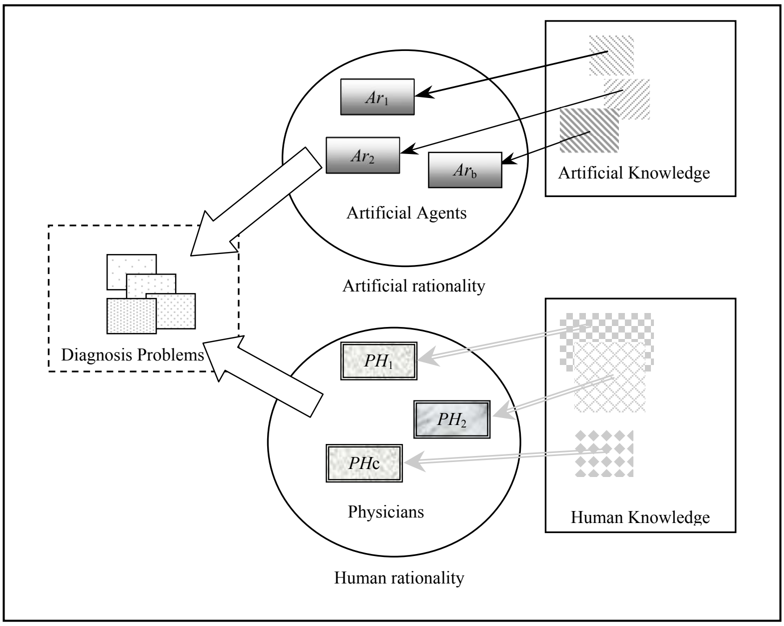

Fig. 5. The knowledge and rationality distribution in the LMDS system

(problems can be solved using the knowledge) to the physicians and to the artificial agents. The artificial agents can use the detained rules in the processings in the form specified in (7) and (8). The precondition of a rule may fit some of the knowledge detained in a problem solving statement. The postcondition of a rule specifies the processing that can be realized to the knowledge detained in a problem solving statement. In an LMDS system composed from a large number of agents, where a large quantity of medical knowledge is used, ontologies can be defined that may help the physicians and/or artificial agents during their operation. For example, we mention an ontology that describes illnesses in a language (English for example). Such an ontology may help the physicians during the diagnosis establishment.

3.3 Simulations and Experiments

We have realized simulations of an LMDS system denoted $DIAG$ (5). Ag_m represents a medical ICMA agent specialized in general medicine. The specialization of Ag_m, $Spec(Ag_m) = S_m$ is detained by its static subagent Ss. Ag_c represents an expert system agent with a specialization $Spec(Ag_c) = S_c$ that represents diagnosing knowledge in cardiology. Ag_u represents an expert system agent with a specialization $Spec(Ag_u) = S_u$ that represents diagnosing knowledge in urology. H_1 and H_2 represent the hosts from the $DIAG$ system. Ag_m and Ag_u are submitted to H_1. Ag_c is submitted to H_2.

$$\begin{aligned} \mathit{DIAG} &= \mathit{MD} \cup H, \\ H &= \{H_1, H_2\}, \\ \mathit{MD} &= \{Ag_m, Ag_c, Ag_u\}, \\ \mathit{Spec}(\mathit{DIAG}) &= \langle \mathit{Spec}(Ag_m); \mathit{Spec}(Ag_c); \mathit{Spec}(Ag_u)\rangle. \end{aligned} \tag{5}$$

Ag_m, Ag_c and Ag_u may processes (add, retract or modify) the knowledge detained in an ICMA mobile subagent body. For doing such operations, each of the agents Ag_m, Ag_c and Ag_u uses a specialization that contains a set of rules.

Let R (6) be a set of rules detained by an agent member of the DIAG system. $|R|$ denote the number of rules in the set R of rules; $|R| = w$ (w is a natural number).

$$R = \{R_1, R_2, \ldots, R_w\}. \tag{6}$$

A rule R_k ($R_k \in R$) has the form (7).

$$\begin{gathered} \langle \mathit{No}_k; \mathit{Domain}_k; \mathit{Type}_k \rangle \\ \mathit{Prec}_k \to \mathit{Post}_k. \end{gathered} \tag{7}$$

No_k represents the R_k rule identifier (each rule detained by an agent has a unique number as identifier). Domain_k represents the medical domain in which the R_k rule allows medical knowledge processing. Type_k represents the type of the R_k rule. Each rule has a single type (one of the following): *add* (for adding knowledge), *retract* (for retracting knowledge) or *modify* (for modifying knowledge).

Prec_k represents the R_k rule precondition. Prec_k specifies the conditions that must be verified in order to apply the R_k rule. Post_k represents the R_k rule postcondition. Post_k specify the changes (adding, retracting or modifying) that must be made on the knowledge specified in the R_k rule precondition.

The precondition and postcondition of each rule may have some of the parameters (8).

$$\langle \mathit{Syp}_k[1]; \mathit{Syd}_k[2]; \mathit{Anly}_k[3]; \mathit{Past}_k[4]; \mathit{Ils}_k(\mathit{Pi}_k)[5]; \mathit{Clas}_k[6] > . \tag{8}$$

Ils_k represents an illness. Pi_k represents the probability of occurrence of the Ils_k illness. Syp_k represents the symptoms of the Ils_k illness. Syd_k represents the syndromes of the Ils_k illness. Anly_k represents medical analysis results necessary in the Ils_k illness identification. Past_k represents illnesses that have appeared in the past (illnesses that may influence the occurence of Ils_k illness). Clas_k contains the specification of different classes of illnesses in which it is assumed that the Ils_k illness is included.

In each rule some parameters of the postcondition and/or precondition may be missing. Each parameter in (8) has a unique identifier (a natural number between 1 and 6) in the parameter list. For example, the parameter Syd_k that has associated the number 2, specifies that the parameter value (values) contain the description of one (more) syndrome (syndromes) of the Ils_k illness.

A diagnosis problem P_k solving statement may have some of the parameters (9).

$$\langle |Symp_k|[1]; |Synd_k|[2]; |Anal_k|[3]; |Past_k|[4];$$

$$|Sup_k(Ps_k)|[5]; |Clas_k|[6]; |Demons_k(Pd_k)|[7]\rangle. \tag{9}$$

If a problem P_k is transported by an ICMA mobile subagent denoted M_S, then the problem solving statement is detained in the M_S body. Symp_k represents the patient's illness symptoms. Synd_k represents the patient's illness syndromes. Anal_k represents different analysis results realized in order to identify the patient's illness. $Past_k$ represents patient's illnesses from the past. Sup_k represents supposed illnesses of the patient. Ps_k represents the probabilities of occurrence of the supposed illnesses. Demons_k represents demonstrated illnesses of the patient. Pd_k represents the probabilities of occurrence of the patient's demonstrated illnesses. A supposed illness is a demonstrated illness with a low probability of occurrence. During a diagnosis process the agents may change the probability of occurrence of an illness specified in a parameter list. For example, a supposed illness may become demonstrated illness after performing some medical analyses (the probability with which the illness occurrence is supposed has been increased). Clas_k contains the specification of different classes of illnesses in which it is assumed that the patient's illness is included. Each parameter in (9) has a unique identifier (a natural number between 1 and 7), that help the agents establish the rules that can be used in the processing (makes easier the rules precondition matching, when some parameters in a mobile subagent body have no value).

If in a problem solving statement more demonstrated and/or supposed illnesses are specific, to each illness a unique natural number is attached as identifier (each illness identifier is different in a problem solving statement). In this situation, to each information in the parameter list the illness identifier is attached, that helps the agents identify to what illness the information is attached. As an example, we consider that in the case of a patient, two illnesses denoted Il_1 and Il_2 are demostrated. To the illness Il_1 the identifier 1 is attached. To the illness Il_2 the identifier 2 is attached. For example, $|Sa\ :1, Sb\ :1:2, Sc\ :2|[1]$ specifies the first parameter (specified by [1]) in a problem solving statement (9) (the first parameter specifies illness symptoms); Sa specifies a symptom that is associated with the illness Il_1 (identifier :1 is attached to Sa), Sb specifies a symptom that is associated with both illnesses Il_1 and Il_2 (identifiers :1 and :2 are attached to Sb) and Sc specifies a symptom that is associated with the illness Il_2 (identifier :2 is attached to Sc).

In the following, solving of a cardiology related illness problem P_{card} by the $DIAG$ system is described (the problem P_{card} solution SOL_{card} that must be obtained represents the identified illness). Initially P_{card} contains the problem initial description (different information and data detained initially about the patient's illness).

Problem Solving Description

Step 1

- The problem P_{card} is overtaken by the static subagent Ss of Ag_m (Ag_m is the only ICMA agent in the $DIAG$ system).

Step 2

- Ss processes the problem P_{card} using the specialization S_m. SS makes some observations Q_{gen} related with the patient's illness (Q_{gen} does not represent solution of the problem P_{card}. Ss cannot solve the problem P_{card} because it does not have the necessary specialization).
- $P_{card} = P_{card} \cup Q_{gen}$.
- Ss creates a mobile subagent M_S which is endowed with the knowledge P_{card} and the itinerary $I_S = \{H_1, H_2\}$.
- Ss launches M_S to the first host H_1 ($H_1 \in I_S$) specified in the M_S itinerary I_S.

Step 3

- H_1 cannot find any agent capable of processing the problem P_{card}.
- $I_S = I_S - \{H_1\}$.
- H_1 launches M_S to the next host H_2 ($H_2 \in I_S$) specified in the M_S itinerary I_S.

Step 4

- H_2 establishes the agent Ag_c which can process P_{card}.
- H_2 transmits the knowledge carried by M_S to Ag_c.
- Ag_c solves the problem P_{card}, using S_c, obtaining the solution SOL_{card}.
- Ag_c transmits the problem solution SOL_{card} to the host H_2.
- H_2 endows M_S with the problem solution SOL_{card}.
- $I_S = I_S - \{H_1\}$.
- H_2 launches M_S to Ss.

Step 5

- Ss transmits the problem solution SOL_{card} to the problem sender.

EndProblemSolvingDescription.

$$P_{card} = \{ \text{ a cardiology related illness description } \}.$$

SOL_{card} represents the solution of the problem P_{card}.

$$SOL_{card} = \{ \text{ the identified cardiology related illness } \}.$$

The problem P_{card} solving process can be described as follows (10):

$$\begin{aligned} &\mathrm{Ag}_m(P_{card}; S_m) \Rightarrow \mathrm{Ag}_c(P_{card} \cup Q_{gen}; S_c) \Rightarrow \mathrm{SOL}_{card}, \\ &\mathrm{DIAG}(P_{card}; S_m, S_c) \Longrightarrow \mathrm{SOL}_{card}. \end{aligned} \tag{10}$$

Ss (*Ss* is the static subagent of Ag_m) processes P_{card} using the specialization S_m obtaining the knowledge Q_{gen}. Ag_c processes $P_{card} \cup Q_{gen}$ using the specialization S_c obtaining the solution SOL_{card}. Q_{gen} represents different general observations related to the patient's illness elaborated by *Ss*. In the P_{card} problem solving the *DIAG* system has used the specializations S_m and S_c.

Simulations show the correctness of the problem solving by the *DIAG* diagnosis system. The simulations were realized for the diagnosis of usual illnesses (identified by physicians specialized in general medicine), cardiology and urology related illnesses. The diagnosis system can solve a diagnosis problem, if it has the necessary medical problem solving knowledge distributed between the member agents. The accuracy of the diagnostics elaborated by the *DIAG* system depends on the accuracy of information and data specified in the rules preconditions and postconditions (the rules are established by human specialists). In cooperation with physicians, a *knowledge engineer* establishes the medical diagnosing knowledge that is retained as rules. The form of the rules specified by (7), (8) and the problem solving statement description specified by (9), can be adapted to the specific features of the medical problems (what medical information and data must be processed during the diagnostic processes) that must be solved by the agents. For example, during a diagnosis process the history of the symptoms of an illness can be used. In this case, there must exist rules detained by agents, whose precondition and/or postcondition contain history of symptoms. The problem solving statement will contain such information. To increase the elaborated diagnostics accuracy the artificial agents must be endowed with learning capability. They must adapt or retract the rules that have some uncertainties (some erroneous and/or missing data).

3.4 Motivations and Advantages of the LMDS System

One of the most important directions of research related with the medical agents consists in the development of large-scale medical diagnosis systems [7, 9, 13]. Many of the existent medical knowledge, medical information and data detained about patients (medical history for example) are distributed. Medical decisions elaborations may involve solving of different problems. Distributed medical information and data must be collected, analyzed and processed. For many of these problem solvings, the agent-based approaches are the best-fitted solutions. The agents can solve, based on their proprieties (autonomy, capability to perceive the environment, capability to execute actions in the environment, capability to learn autonomously, capability to assist pro-actively humans in the decisions elaboration, capability to communicate and cooperate in the problems solving) problems that cannot be solved by traditional medical systems (medical expert systems for example) [3, 5, 23].

The agents can solve problems using combinations of problem solving methods [25, 8]. A motivation of combination of more problem solving methods consists in the maximization of the methods advantages and minimization of their disadvantages. For example, an agent can be endowed with a component that uses neural networks and fuzzy inferences. As an example of a problem that can be solved using such a hybrid method we mention measuring of the cardiac function quantitatively and evaluating the motions of continuous cardiac muscle for detecting the asynergy in the left ventricle, using X-ray photograms of the left ventricle [47]. Agents may integrate and extend different existing problem solving technologies [8, 23, 2]. The use of cooperating agents in problem solving has as advantage, namely the combination of the agents capabilities and capacities.

The medical diagnosing knowledge in the LMDS system is distributed between the agents (physicians and artificial medical agents) members of the system. Each host from the system detains information about a set of submitted agents, this helps in the establishment to which agents the overtaken problems should be sent for processing. A host also detains information about some other hosts members of the system. The problems that cannot be solved by the agents submitted to a host are transported for solving to other agents by ICMA mobile subagents. A mobile subagent migrates with an overtaken problem from host to host, until the problem is solved. ICMA mobile subagents can be used successfully in the solving of the problems, whose solving requires knowledge and/or resources distributed in the system.

In the LMDS system, the medical diagnosis problems are transmitted randomly for solving to the medical ICMA agents. The agents members of the system can solve problems simultaneously, each agent has specializations and resources that can be used in the problems processing. An agent may overtake more problems for processing depending on its specializations and capacity. Each problem overtaken by the system can be solved cooperatively by more agents members of the system.

The LMDS system can solve medical problems that can be broken into subproblems. A static subagent of an ICMA agent can solve some subproblems of an overtaken problem. The rest of the subproblems can be transmitted for solving by mobile subagents of the agent. Based on the obtained subproblems solutions the static subagent will form the solution of the problem. As an example of subproblem of a medical diagnosis problem, we mention a medical analysis necessary in increasing the accuracy of an illness identification. The medical analysis can be realized by a human medical specialist. As example of another subproblem we mention the recognition of the disorder of an internal human organ, based on different data detained about the organ functioning. This subproblem can be solved by a specialized agent during the subproblem solving.

The artificial agents can be endowed with new medical specializations. Inefficient specializations can be eliminated or adapted. LMDS is an open system, each host may submit new agents. The hosts may assist physicians in their interaction with the system by translating the information transmitted to the physicians into a form understandable to the physicians. A physician may require knowledge necessary in the problems solving from the host to which it is submitted. For example, the physician may require an illness description and a patient's previous illnesses.

A physician and an artificial agent submitted to the same host can diagnose the same illness. The obtained solutions can be compared by the physician. The same solution obtained by the physician and the artificial agent increases the certitude in the correctness of the obtained solution. If the obtained solutions differ, the physician and the artificial agent must reanalyze the problem solving. The problem may be transmitted for solving to other agents. A problem solution will be established by the agents (human and artificial) who have participated in the diagnosis process. An obtained solution must be validated by a physician (physicians) specialized in the medical domain in which the identified illness (illnesses) is included.

4 CONCLUSIONS

Medical diagnostics elaborations often represent a naturally distributed and cooperative processes, which involves human medical specialists and different medical systems [2, 14, 23, 36, 45]. The results described in the literature prove that many medical diagnosis problems can be solved efficiently by large-scale medical multiagent systems. The development of large-scale medical diagnosis systems represents an important recent research direction [7, 9, 13].

In this paper, we have proposed a cooperative hybrid large-scale medical diagnosis system, called *LMDS* (*Large-Scale Medical Diagnosis System*) with physicians and artificial agents (*medical expert system agents* and *medical ICMA agents*) as members. The proposed hybrid medical diagnosis system is a complex system. It is composed from physicians and artificial agents that cooperate in order to discover solutions of difficult medical diagnosis problems. Difficult medical cases are those in which identification of the illnesses and establishment the corresponding efficient treatments is difficult. The necessary knowledge for the problem solving in the LMDS system is distributed between the humans and agents members of the system. Each member of the system contributes to the problem solving depending on its detained knowledge. A contribution to a problem solving by a member of the system may make the problem following processing easier for other members. The system's members cooperate in order to handle the complexity of the diagnosis establishment.

The agents called *medical expert system agents* have been developed in our previous works and applied for different medical problems solving [5, 6, 8, 10]. In previous works, a novel mobile agent architecture called *ICMA* (*Intelligent Cooperative Mobile Agent Architecture*) was developed. The ICMA architecture allows the creation of mobile agents; this partially eliminates disadvantages of recently developed mobile agents described in the literature [4, 7, 19]. Applications of the *ICMA* agents for problem solving based on genetic problem solving methods are presented in [20, 21, 22]. Medical ICMA agents represent agents with the ICMA architecture endowed with medical diagnosing knowledge. Some introductory elements about the medical ICMA agents were described in [9].

The LMDS system is not intended to substitute the physicians. In a diagnostic establishment, the system combines the physicians and artificial agents' advantages related with their *capabilities* and *capacities* to elaborate medical diagnostics. Physicians may solve problems using their medical knowledge and intuition (the intuition is a specific property of human intelligence). However, they can solve difficult medical diagnosis problems that cannot be solved by artificial agents. The artificial agents cannot solve problems that are too different from known problems solving, and problems where difficult to handle uncertainties (some erroneous and/or unknown information and data) appear during solving. The artificial agents might analyze details that can be ignored by physicians. However, the artificial agents may help the physicians increase the elaborated diagnostics accuracy. As examples of information that can be ignored by physicians in a diagnostic establishment, and that can be verified by an artificial agent if the information exists in a medical data-base, we mention: the patient's allergy to a medicine, the contraindications of a medicine, etc.

The next research includes the endowment of the ICMA agents with autonomous learning capability. ICMA mobile subagents may transmit useful information to their creator's static subagents. Static subagents can learn from the received information, modifying the detained medical knowledge. The motivations of the possibility to endow an ICMA agent with autonomous learning capability consist in the ICMA agents' increased communication capability [4], capability of the ICMA agents to transport knowledge to other agents [4], increased protection possibility of the ICMA agents against network sources and malicious hosts [19] and increased intelligence in operation [7].

REFERENCES

[1] Abdelaziz, T.—Elammari, M.—Unland, R.: Visualizing a Multiagent Based Medical Diagnosis System Using a Methodology Based on Use Case Maps. In: G. Lindemann (Ed.): MATES 2004, Springer-Verlag, Berlin, Heidelberg, LNAI, Vol. 3187, 2004, pp. 198–212.

[2] Ferber, J.: Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence. Addison Wesley, 1999.

[3] Iantovics, B. L.: A Novel Diagnosis System Specialized in Difficult Medical Diagnosis Problems Solving. In: M. A. Aziz-Alaoui, C. Bertelle (Eds.): Proceedings of the Emergent Proprieties in Natural and Artificial Dynamical Systems (EPNADS '05) a workshop within European Conference on Complex Systems, Le Havre University Press, Paris, 2005, pp. 107–112.

[4] Iantovics, B. L.: A New Intelligent Mobile Multiagent System. Prceedings of the IEEE International Workshop on Soft Computing Applications (IEEE SOFA 2005), Szeged-Hungary and Arad-Romania, IEEE Press, 2005, pp. 153–159.

[5] Iantovics, B. L.: A Novel Medical Diagnosis System. Proceedings of the 4th International Conference on Theory and Applications in Mathematics and Informatics, Albac county, Acta Universitatis Apulensis, Vol. 10, 2005, pp. 315–330.

[6] Iantovics, B. L.: A Novel Medical Diagnosis System. In: P. Bourgine, F. Kepes, M. Schoenauer (Eds.): Proceedings of the European Conference on Complex Systems (ECCS 2005), Paris, 2005.

[7] Iantovics, B. L.: A Novel Mobile Agent Architecture. Proceedings of the 4th International Conference on Theory and Application of Mathematics and Informatics, Albac county, Acta Universitatis Apulensis, Vol. 11, 2006, pp. 295–306.

[8] Iantovics, B. L.—Chira, C.—Dumitrescu, D.: Principles of the Intelligent Agents. Casa Cartii de Stiinta Press, Cluj-Napoca, 2007. (book in Romanian).

[9] Iantovics, B. L.: Cooperative Medical Diagnosis Systems. Proceedings of the International Conference Interdisciplinarity in Engineering, Tg. Mures, 2005, pp. 669–674.

[10] Iantovics, B. L.: Medical Diagnosis Systems. Proceedings of the Symposium "Colocviul Academic Clujean de Informatica", Babes-Bolyai University, Cluj-Napoca, 2005, pp. 33–39.

[11] Iantovics, B. L.: A Novel Task Allocation Protocol in Distributed Multiagent Systems. In: C. Enachescu, D. Radoiu (Eds.): Proceedings of the 4th International Conference in Education/Training and Information/Communication Technologies, Sovata, 2005, pp. 1–6.

[12] Laita, L. M.—Gonzlez-Paez, G.—Roanes-Lozano, E.—Maojo, V.—Laita, L.: A Methodology for Constructing Expert Systems for Medical Diagnosis. In: J. Crespo, V. Maojo, F. Martin (Eds.): ISMDA 2001, Springer-Verlag, LNCS, Vol. 2199, 2001, pp. 146–152.

[13] Unland, R.: A Holonic Multi-Agent System for Robust, Flexible, and Reliable Medical Diagnosis. In: R. Meersman, Z. Tari (Eds.): OTM Workshops 2003, Springer-Verlag, LNCS, Vol. 2889, 2003, pp. 1017–1030.

[14] Vesnenko, A. I.—Popov, A. A.—Pronenko, M. I.: Topo-Typology of the Structure of Full-Scaled Clinical Diagnoses in Modern Medical Information Systems and Technologies. Plenum Publishing Corporation Cybernetics and Systems Analysis, Vol. 38, 2002, No. 6.

[15] Wooldridge, M.—O'Hare, G. M. P.—Elks, R.: FELINE – A Case Study in the Design and Implementation of a Co-Operating Expert System. Avignon 1991, France, 1991.

[16] Iantovics, B. L.: The Intelligence of the Expert System Agents. Proceedings of the 2nd International Conference on Economics, Law and Management, Petru Maior University Press, Tg. Mures, 2006, pp. 91–101.

[17] Iantovics, B. L.: Mobile Agents. Profesor Gheorghe Farcas la varsta de 70 de ani, Petru Maior University Press, Tg. Mures, 2004, pp. 171–185.

[18] Iantovics, B. L.: Intelligent Mobile Agents. Proceedings of the Symposium "Zilele Academice Clujene", Babes-Bolyai University, Babes-Bolyai University Press, Cluj-Napoca, 2004, pp. 67–74.

[19] Iantovics, B. L.: Security Issues of the Mobile Multiagent Systems. Proceedings of the International Conference European Integration between Tradition and Modernity, Petru Maior University Press, Tg. Mures, 2005, pp. 770–779.

[20] Iantovics, B. L.: Evolutionary Mobile Agents. Proceedings of the International Conference European Integration between Tradition and Modernity, Petru Maior University Press, Tg. Mures, 2005, pp. 761–769.

[21] Iantovics, B. L.: Evolutionary Problems Solving in Networks. Proceedings of the Symposium "Colocviul Academic Clujean de Informatica", Babes-Bolyai University, Babes-Bolyai University Press, Cluj-Napoca, 2005, pp. 39–45.

[22] Iantovics, B. L.: Problem Solving Using Evolutionary Mobile Agents. In: M. Megan (Ed.): Proceedings of 9th National Conference of the Romanian Mathematical Society, Vest University Press, Timisoara, 2005, pp. 408–420.

[23] Iantovics, B. L.: A Novel Diagnosis System Specialized in Difficult Medical Diagnosis Problems Solving. Emergent Proprieties in Natural and Artificial Systems, Understanding Complex Systems, Springer-Verlag, Heidelberg, 2006, pp. 187–197.

[24] Iantovics, B. L.: Problems Solving Using Assistant Agents. Scientific Bulletin of the Petru Maior University, Tg. Mures, XVIII, 2006, pp. 173–179.

[25] Weiss, G. ed.: Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. MIT Press Cambridge Massachusetts, London, 2000.

[26] Unland, R.—Ulieru, M.: Swarm Intelligence and the Holonic Paradigm: A Promising Symbiosis for Medical Diagnostic Systems Design. Proceedings of the 9th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems, Melbourne, Australia, 2005.

[27] Alves, V.—Neves, J.—Maia, M.—Nelas, L.: A Computational Environment for Medical Diagnosis Support Systems. In: J. Crespo, V. Maojo, F. Martin (Eds.): ISMDA 2001, Springer-Verlag, Berlin, Heidelberg, LNCS, Vol. 2199, 2001, pp. 42–47.

[28] Shortliffe, E. H.: Computer-Based Medical Consultations:MYCIN. Elsevier, New York, 1976.

[29] Bravata, D. M.—Sundaram, V.—McDonald, K. M.—Smith, W. M.—Szeto, H.—Schleinitz, M. D.—Owens, D. K.: Evaluating Detection and Diagnostic Decision Support Systems for Bioterrorism Response Emerging Infectious Diseases. Emerging Infectious Diseases, Vol. 10, 2004, No. 1, pp. 100–108.

[30] Adlassing, K. P.: Cardiag 2 Expert System. IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-16, 1986, No. 2.

[31] Aikins, J. S.—Kunz, J. C.—Shortliffe, E. H.—Fallat, R. J.: PUFF: An Expert System for Interpretation of Pulmonary Function Data. Comput. Biomed. Res., Vol. 16, 1983, No. 3, pp. 199–208.

[32] Kulikowski, C. A.—Weiss, S. M.: Representation of Expert Knowledge for Consultation: the CASNET and EXPERT Projects. In: P. Szolovits (Ed.): Artificial Intelligence in medicine. Boulder:Westview Press, 1982, pp. 21–56.

[33] Kuhl, E. J.—Graham, J. H.: ESAgent: Expert System Control of Simulated Agent-Based Mobile Robots. Intelligent Systems Research Laboratory, Technical Report TR-ISRL-04-02, Department of Computer Engineering and Computer Science University of Louisville, Louisville, 2004.

[34] Zhang, S.—Bodenreider, O.: Comparing Associative Relationships Among Equivalent Concepts across Ontologies. Medinfo, 2004, pp. 459–463.

[35] Bodenreider, O.: The Unified Medical Language System (UMLS): Integrating Biomedical Terminology. Nucleic Acids Res., Vol. 32, 2004, No. 1, pp. 267–270.

[36] Kirn, St.: Ubiquitous Healthcare: The OnkoNet Mobile Agents Architecture. In: M. Aksit, M. Mezini, R. Unland (Eds.): Proceedings of the 3rd International Conference Netobjectdays. Objects, Components, Architectures, Services, and Applications for a Networked World (NODe 2002), Springer-Verlag, Germany, LNCS, Vol. 2591, 2003.

[37] Montyne, F.: The Importance of Formal Ontologies: A Case Study in Occupational Health. In: A. d'Atri, M. Missikoff (Eds.): Proceedings of the OES-SEO 2001 Rome Workshop, Luiss Publications, 2001. Available on: `http://cersi.luiss.it/oesseo2001/papers/28.pdf`.

[38] Stevens, R.—Baker, P.—Bechhofer, S. G.—Jacoby, A.—Paton, N. W.—Goble, C. A.—Brass, A.: TAMBIS: Transparent Access to Multiple Bioinformatics Information Sources. Bioinformatics, Vol. 16, 2002, No. 2, pp. 184–186.

[39] Ashburner, M.—Ball, C. A.—Blake, J. A.—Botstein, D.—Butler, H.—Cherry, J. M.—Davis, A. P.—Dolinski, K.—Dwight, S. S.—Eppig, J. T. et al.: Gene Ontology: Tool for the Unification of Biology. The Gene Ontology Consortium. Nat. Genet., Vol. 25, 2002, pp. 25–29.

[40] Nealon, J. L.—Moreno, A.: The Application of Agent Technology to Healthcare. Proceedings of the First Agentcities Workshop: Challenges in Implementing Open Environments (AAMAS 2002), Bologna, Italy, 2002.

[41] Pfeifer, R.—Scheier, C.: Understanding Intelligence. MIT Press, 1999.

[42] Hayes-Roth, B.—Hewett, M.—Washington, R.—Hewett, R.—Seiver, A.: Distributing Intelligence within an Individual. In: L. Gasser, M. Huhns (Eds.): Distributed Artificial Intelligence II. Pitman Publishing: London and Morgan Kaufmann: San Mateo, CA, 1989, pp. 385–412.

[43] Lanzola, G.—Falasconi, S.—Stefanelli, M.: Cooperative Agents Implementing Distributed Patient Management. Proceedings of the MAAMAW-96 Conference, Berlin, Springer-Verlag, 1996.

[44] Lanzola, G.—Falasconi, S.—Stefanelli, M.: Cooperative Software Agents for Patient Management. Proceedings of the AIME-95 Conference, 1995, pp. 173–184.

[45] Huang, J.—Jennings, N. R.—Fox, J.: An Agent-Based Approach to Health Care Management. International Journal of Applied Artificial Intelligence, Vol. 9, 1995, No. 4, pp. 401–420.

[46] Jagannathan, V.—Dodhiawala, R.—Baum, L. S. (Eds.): Blackboard Architectures and Applications. Academic Press, San Diego, 1989.

[47] Mitsukura, Y.—Mitsukura, K.—Fukumi, M.—Akamatsu, N.—Witold-Pedrycz, W.: Medical Diagnosis System Using the Intelligent Fuzzy Systems. In: M. Gh. Negoita (Ed.): KES 2004, Springer-Verlag, LNAI, Vol. 3213, 2004, pp. 807–826.

[48] Kowalczyk, R.—Braun, P.—Mueller, I.—Rossak, W.—Franczyk, B.—Speck, A.: Deploying Mobile and Intelligent Agents in Interconnected

E-Marketplaces. Journal of Integrated Design and Process Science, Transactions of the SDPS, Vol. 7, 2003, No. 3, pp. 109–123.

[49] Xu, H.: A Model-Based Approach for Development of Multi-Agent Software Systems. Ph. D. thesis, the Graduate College of the University of Illinois at Chicago, 2003.

[50] Carstoiu, D. I.: Expert Systems. All Press, Bucharest, 1994 (in Romanian).

[51] Varachiu, N.—Karanicolas, C.—Ulieru, M.: Computational Intelligence for Medical Knowledge Acquisition with Application to Glaucoma. Proceedings of the First IEEE Conference on Cognitive Informatics (ICCI02), Calgary, Canada, 2002, pp. 233–238.

[52] Ulieru, M.—Pogrzeba, G.: Integrated Soft Computing Methodology for Diagnosis and Prediction with Application to Glaucoma Risk Evaluation. Proceedings of 6th IASTED International Conference on Artificial Intelligence and Soft Computing, Banff, Canada, 2002, pp. 275–280.

[53] Nagarajappa, N.: Uploading Data from the Local to the Centralized Database for Glaucoma Diagnosis and Prediction. Internal Report, Electrical and Computer Engineering, University of Calgary, 2002.

[54] Ulieru, M.—Grabelkovsky, A.: Telehealth Approach for Glaucoma Progression Monitoring. International Journal: Information Theories and Applications, Vol.10, 2005, pp. 326–329.

[55] Cabri, G.—De Mola, F.—Muratori, N.—Quitadamo, R.—Zambonelli, F.: UbiMedic: a Ubiquitous Computing Solution for Medical Emergencies. Proceedings of the 2nd International Workshop on Multi-Agent Systems for Medicine, Computational Biology, and Bioinformatics, Hakodate, Japan, 2006, pp. 42–46.

[56] Bellavista, P.—Bottazzi, D.—Corradi, A.—Montanari, R. Challenges, Opportunities and Solutions for Ubiquitous Eldercare. DEIS University of Bologna Technical Report, October 2005.

[57] Haigh, K. Z.—Kiff, L. M.—Kiff, L. M.—Myers, J.—Guralnik, V.—Geib, C. W.—Phelps, J.—Wagner, T.: The Independent LifeStyle Assistant (I. L. S. A.): AI Lessons Learned. The Sixteenth Innovative Applications of Artificial Intelligence Conference (IAAI-04), San Jose, CA., 2004, pp. 852–857.

[58] Eder, L. ed.: Managing Healthcare Information Systems with Web-Enabled Technologies. Idea group Publishing, 2000.

[59] Li, X.—Cao, J.—He, Y.: A Direct Execution Approach to Simulating Mobile Agent Algorithms. Journal of Supercomputing, Kluwer Academic Publishers, Vol.29, 2004, pp. 171–184.

[60] Dikaiakos, M. D.—Samaras, G.: Quantitative Performance Analysis of Mobile Agent Systems: A Hierarchical Approach. Technical Report TR-00-02, Department of Computer Science, University of Cyprus, 2000.

[61] Dikaiakos, M. D.—Samaras, G.: Performance Evaluation of Mobile Agents: Issues and Approaches. Performance Engineering, Springer-Verlag, LNCS, Vol. 2047, 2001, pp. 148–166.

[62] Yergens, D.—Hiner, J.—Denzinger, J.—Noseworthy, T.: Multi Agent Simulation System for Rapidly Developing Infectious Disease Models in Developing Countries. Proceedings of the 2nd International Workshop on Multi-Agent Systems

for Medicine, Computational Biology, and Bioinformatics, Hakodate, Japan, 2006, pp. 104–116.

[63] Davis, R.—Smith, R. G.: Negotiation As a Metaphor for Distributed Problem Solving. Artificial Intelligence, Vol. 20, 1983, No. 1, pp. 63–109.

[64] Smith, R.: The Contract Net Protocol: High Level Communication and Control in a Distributed Problem Solver. IEEE Transactions on Computers, Vol. C-29, 1980, No. 12, pp. 1104–1113.

Barna László Iantovics received B. Sc. in mathematics and informatics from Transilvania University of Brasov, Romania; M. Sc. in mathematics and informatics from Transilvania University, and Ph. D. in artificial intelligence from Babes-Bolyai University of Cluj-Napoca, Romania. Currently he is Senior Lecturer at Petru Maior University of Tirgu Mures, and researcher at the research centre "Center for the Study of Complexity" at Babes-Bolyai University. In 2007–2008 he is invited professor at FH Kufstein Tirol University of Applied Sciences, Austria. Scientific activities include: more than 45 scientific papers, 2 books about multi-agent systems and 2 books about computer programming. He organized the RoEduNet '2005 International Conference in Romania. He reviewed: 1 paper as external reviewer at ICNAAM 2007 International Conference in Corfu, Greece; 11 papers as external reviewer BIC International Conference in Wuhan, China and 1 paper for the Journal of Numerical Analysis, Industrial and Applied Mathematics. He was/is scientific committee member in the following conferences: ESM 2006 International Conference in Toulouse, MAICS 2007 International Conference in Chicago, EPNACS 2007 International Conference in Dresden and ESM 2007 International Conference in Malta.