Medical Processes Agent-Based Critiquing System

Bošanský, Branislav
2009

Medical Processes Agent-Based Critiquing System

Post-Graduate Student:
MGR. BRANISLAV BOŠANSKÝ

Department of Medical Informatics
Instutite of Computer Science of the ASCR, v. v. i.
Pod Vodárenskou věží 2
182 07 Prague 8, CZ

bosansky@euromise.cz

Supervisor:
DOC. ING. LENKA LHOTSKÁ, CSC.

Department of Cybernetics
Faculty of Electrical Engineering
Czech Technical University in Prague
Technická 2
166 27 Prague 6, CZ

lhotska@labe.felk.cvut.cz

Field of Study:
Biomedical Informatics

This research was partially supported by the project of the Institute of Computer Science of Academy of Sciences AV0Z10300504, the project of the Ministry of Education of the Czech Republic No. 1M06014 and by the research program No. MSM 6840770012 "Transdisciplinary Research in Biomedical Engineering II" of the CTU in Prague.

Abstract

Processes and process modelling have proven themselves as a useful technique for capturing the work practice in business. We focus on their usage in the domain of healthcare and define two main types of processes in medicine – medical guidelines and organizational processes. Based on these types we present the architecture of a multi-agent system that is able to work with them and describe application of this multi-agent system as a critiquing decision support system for healthcare specialists.

1. Introduction

Developement of a system that will support the decision making of physicians and healthcare specialists is a long-term goal for researchers in artificial intelligence. Recently, there has been given an emphasis to monitoring systems that control and evaluate current situation (e.g. patient data, therapy, etc.) and alerts the medical staff in case of inconsistencies or possible danger. In order to recognize the occurrence of these situations such systems need to operate with appropriate knowledge. In the healthcare domain they can profit from medical guidelines which are sets of directions or principles that assist the physician [1] and are considered to be a good approach to standardize and improve health care [2]. When formalized, i.e. captured in a computer-interpretable form, they are being used in various decision support systems (e.g. in HeCaSe2 [3]).

The medical guidelines, however, can be seen as a specific way of process modelling. In our research we want to develop a system that would be able to work with knowledge captured in form of general processes – i.e. as with formalized medical guidelines, but also with organizational processes which are specific in each healthcare facility (e.g. activities necessary for transferring a patient from one department to a different one). Both of these types of processes were usually considered separately which resulted in different languages and different approaches (e.g. using Event-Driven Process Chains (EPC) to model organizational processes and GLIF for medical guidelines). In this paper we present the architecture of a multi-agent system that (1) is able to work with these general processes in healthcare domain, (2) can simulate them in given environment opening that way a possibility for future planning or process reengineering, and finally (3) can act as a critiquing and monitoring system that controls their adherence and can alert the medical staff.

The paper is organized as follows: in Section 2 we define the problem and theoretical foundations together with related approaches. Section 3 is focused on description of the architecture of the multi-agent system and behaviour of single agents. We describe the usage of the multi-agent system as a process-critiquing system in Section 4, following by an illustrative example and implementation issues in Section 5. We conclude and discuss the future work in Section 6.

2. Processes in Medicine and Related Work

The work practice (i.e. duties of employees, organizational procedures, specification of the order of activities, or necessary resources for each activity) can be captured using process modelling technique – i.e. as a sequence of actions, states, decision points, or steps splitting or joining the sequence. There are

various levels of processes in medical domain and with respect to terminology in [4] we can differentiate the *organizational processes* and the *medical treatment processes*.

2.1. Organizational Processes

The organizational processes in the healthcare domain are closely related to processes in other business areas, where the work practice has been captured for a long time using business process modelling languages. There are several studies [4, 5, 6] that analyze the problems of applying process modelling or usage of workflow management systems in medical care. They all agree that the implementation of this approach can improve current problems with organization, reduce the time of hospitalization and finally reduce the costs. However, they also point out, that till now, usage of processes is rather low and insufficient. The main reasons are more complex processes than in other fields of industry, or problems with interoperability resulting from inconsistencies of databases and used ontology or protocols. Finally, the captured work practice in healthcare is often very variable and closely depends on specific treatment of the patient. All these factors complicate successfull usage of classical business process management, or workflow management systems. Therefore, while working with organizational processes, we also need to take medical treatment of patients into consideration as well.

2.2. Medical Guidelines

Standardization of medical treatment processes is being done for a long time now known as medical guidelines. They contain recommended actions, directions, and principles for specific diseases, and they are all approved by appropriate expert committees helping that way physicians with clinical decisions. Several crucial positive factors have been identified when using guidelines [1]:

- they improve the quality of decisions as healthcare professionals can consult complicated situations in unknown areas and minimize the risk for a patient (e.g. to forget an examination that is important for this patient according to her/his condition)
- they are based on evidence-based medicine and help to reuse and disseminate the knowledge
- they help to standardize provided health care

However, the standard method of work with the guidelines (such as consulting, or using in practice) is solely based on a textual form. This, on one hand, helps the healthcare professionals to capture the knowledge in a straightforward way. On the other hand, such approach brings several complications. It is hard for physicians to do a quick consultation with the guideline during the examination of a patient, or to keep up with the relevant changes in new versions of the document.

Therefore a wide part of research in biomedical informatics is related to the formalization of medical guidelines into an electronic form. There are several workgroups and several languages (PRO*forma*, GLIF, Asbru, etc.) that captures the knowledge of a textual medical guideline into an electronic and structured form. All of them focus on specific parts – e.g. logic background in Asbru, or automatic execution and patient data retrievement in GLIF. They are all based on a process-oriented approach and specify the guideline as a sequence of actions, states, decision, or synchronization points. Research in decision support systems that work with formalized medical guidelines focuses mostly on acquisition, verification, or automatic execution of guidelines [1].

2.3. Related Work

The area of medical guidelines' execution is closest to our problem. There are several systems that can connect the guideline with the patient's health record, retrieve and store appropriate data and guide the physician by executing next steps and waiting for appropriate data to be entered. Within these systems, only a few ones profit from principles of multi-agent systems: Arezzo^{TM}[7], HeCaSe2 [3], or the work presented in [8].

Our approach differs from existing systems in several ways: firstly, as the guidelines as such are transformed into agents, which allows simultaneous work with a set of guidelines, not only with the selected one as in existing work. Secondly, our system is based on more general concept, therefore beside monitoring the proceeding of the guideline, it can also be used for simulation or general computing purposes. Finally, thanks to the distribution of knowledge, agents can focus on the specific activities.

3. Process-Based Multi-Agent System

In this section we present the architecture and the functioning of the multi-agent system (MAS) that realize the critiquing system. The architecture is based on the one presented in [9] later enhanced in [10]. As discussed in Section 2, the architecture is more general and it can be used on simulating other process-based systems as well.

The architecture and different types of agents are depicted in Figure 1. Let us now describe these agents and their purpose more in detail.

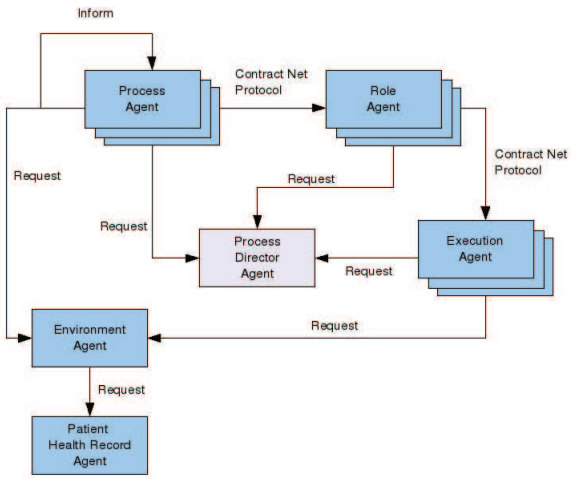

Figure 1: The architecture of a multi-agent system that is able to work (e.g. simulate, critique) with processes.

3.1. Environment Agent

Every agent-based simulation is situated in some environment which is represented by the Environment Agent in this architecture. With respect to the level of detail that we want model using this system the environment could represent the virtual world (e.g. a department of a hospital, etc.) with existing objects (e.g. RTG or EEG machines, wheel-chairs, beds, etc.).

3.2. Execution Agents

Execution Agents (EA) are representing concrete physicians, nurses, patients, or other employees of the facility that are involved in the processes. These agents are based on a reactive architecture in the form of hierarchical rules, which can be automatically generated based on possible activities that the specific EA can participate in. Each EA has several pre-defined rules, that for basic behaviour in the environment (i.e. responding to messages sent by other agents, sending appropriate messages to the Environment Agent during the execution of the activity, etc.). Then, for each activity that the agent (hence the represented person) can participate in, one additional rule is generated. These rules can be activated (when the condition for the process execution are met, and the EA can perform this action) or deactivated (execution of this process is no longer possible) by a message sent by appropriate Role Agent (see below). Finally, the Execution Agent autonomously chooses which of the activated processes it will execute based on the priority in which the rules are ordered.

3.3. Role Agents

Role Agents (RA) represent the roles in the environment (i.e. general roles for patient, nurse, physician, etc.). RA receives the proposal from a Process Agent (see below) and finds appropriate Execution Agent(s) (EA). The reason of using special agents for roles is in a typical usage of hierarchical structure of roles at workplace (e.g. a secretary, a nurse, or a doctor are all also employees, etc.). Therefore, when a RA receives a proposal from a Process Agent, it starts to find the appropriate EA between agents that posses this role (using contract-net protocol (CNP)), but also roles, that are more general in the hierarchy.

If multiple EAs are able to execute given activity and only one is needed, RA will choose the most suitable of them according to its internal rules, which are always domain or role dependent (e.g. in a simulation that occurs in some virtual world, the EA that is closest to the place of execution can be notified, or in another case the EA that is currently idle).

3.4. Process Agent

For every step in the process notation (i.e. activity, event, decision point, etc.) there is one Process Agent (PA) created in the system. The PA is responsible for a proper execution of the activity. Firstly it controls whether the initial conditions for the process are met: if the predecessing PA has successfully finished its execution, if all input objects have the needed values (using simple request protocol to Environment Agent), and if there exist appropriate agents that will execute this action (using CNP to those RAs that are connected with this activity). When all mandatory conditions hold, the PA starts the execution of the process (e.g. the simulation, calculation or a decision process, etc.) and after successfull finish, the PA is responsible for notifying the Environment Agent about the results of the activity (using simple request protocol) and the next succeeding Process Agent about the successfull finish (using simple inform protocol). Our approach takes into account the possibility of temporal suspension of the activity and reflecting the partial results in the environment, replacing the EA with another, coordination of several EAs participating on a single activity, or optional input objects.

Note, that each step of the process has its Process Agent – i.e. not only active steps (steps that represent activities as such) by also so called passive steps (usually related to the events (in EPC) or patient state (in GLIF)), flow-splitting (i.e. decision points), and flow-joining elements have appropriate Process Agent as well.

4. Critiquing System

We have described in detail the architecture of the multi-agent system that is able to work (e.g. simulate them) with the processes. One of possible application of such multi-agent system can be in critiquing – monitoring the correct execution of processes such as formalized medical guidelines or organizational processes in healthcare facilities.

We accentuate the Process Agents (PAs) and description of their behaviour, while other agents behave exactly in the way described in previous section. The main idea is that each PA is responsible for one step in the guideline, it monitors data fields in patient's health record related to the given step, and tries to estimate the outcome of the step simulating that way future developement of diagnostics or therapy. Whenever appropriate input data changes PAs update predicted output values and simulate the process further. Therefore, whenever the output data fields are changed by the physician in the way which PA has not expected an alert occurs.

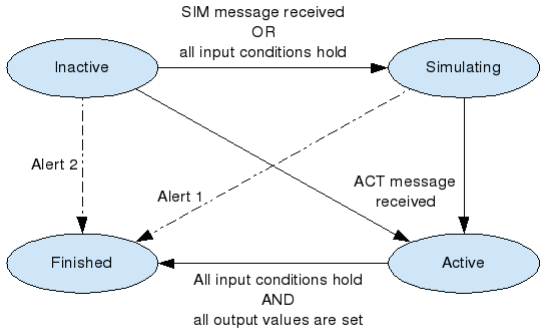

Figure 2: The states of Process Agents during critiquing. The solid arrow indicates valid transition, the dashed arrows indicate possible inconsistencies.

Let us now describe the critiquing more in detail. We distinguish four basic states of a PA (see Figure 2) – *inactive*, *simulating*, *active*, and *finished*. At the beginning, each PA is in the *inactive* state. PA in this state behaves the same way as in simulation before the execution of the activity – it periodically checks the objects in input condition whether they hold. In the critiquing phase therefore is PA periodically checking the associated fields in patient's data model (such as blood pressure, height, etc.) together with the message from predecessing PA (whether it has finished the activity or not).

The agent can get to the *simulating* state when at least one of two following conditions holds: (1) the agent receives the SIM message (i.e. predecessing agent has finished the simulation of the process), or (2) all input conditions for the process execution are met, and the agent has not received ACT message from its predecessor. When the PA is in *simulating* state, it checks again all of its input conditions and in case that some of them are not evaluable (i.e. data in the patient's data record are missing), they are estimated using k-means technique with respect to other patients' data. Such an estimation is necessary for proper running the correspondent action (e.g. setting the concrete diagnose, measuring the blood pressure, etc.) that would yield the simulation output of the process that can be temporarily stored in the simulation environment (but not the patient's data record) and other PAs can work with them. After finishing the simulation of the process, the PA sends a SIM message to the appropriate successor meaning the simulation of its activity has finished.

The agent gets to the *active* state when it receives the ACT message from its predecessor. In this state the PA behaves very similarly to *simulating* state with one difference: in case that all input conditions are met and the output value has been updated in the patient's data record (by the doctor), the agent moves to *finished* state and sends the ACT message to the appropriate successor.

The alert for the doctor occurs in the case when the output values of the process are updated but the agent is not in the *active* state. This can happen because of (1) the step was executed before its predecessors were successfully finished, or (2) the step was not expected to be executed. We can recognize these cases based on the current state the PA is in, when the output values are updated. For the first case the PA would be in *simulating* state, for the second case the PA would be in *inactive* state.

5. Experiments and Implementation

In this section we present an illustrative example, which is the basis for our preliminary experiments of presented process agent-based critiquing system. We demonstrate a possible application using a simplified version of the guideline for a hypertension treatment following by the biref description of the implementation details.

5.1. Guideline Critiquing

In Figure 3 we depicted a very simplified version of a hypertension guideline for demonstrating exemplary situations that can arise during the critiquing of medical processes. Note, that the guideline is simplified for explanatory reasons and in the system full medical processes representing the real diagnosis and

therapy processes (corresponding to formalized medical guidelines used in practice) would be used. Moreover, the description for two decision steps are shortened: (*) under the term "patient with high pressure" we understand a patient with blood pressure value at least 180/110 (values for systolic pressure/diastolic pressure), or at least one blood pressure value of at least 140/90 from three different sessions; (**) there are several possible complications for hypertension therapy such as SCORE value [11] over 5%, patient diagnosed with diabetes mellitus, and many others.

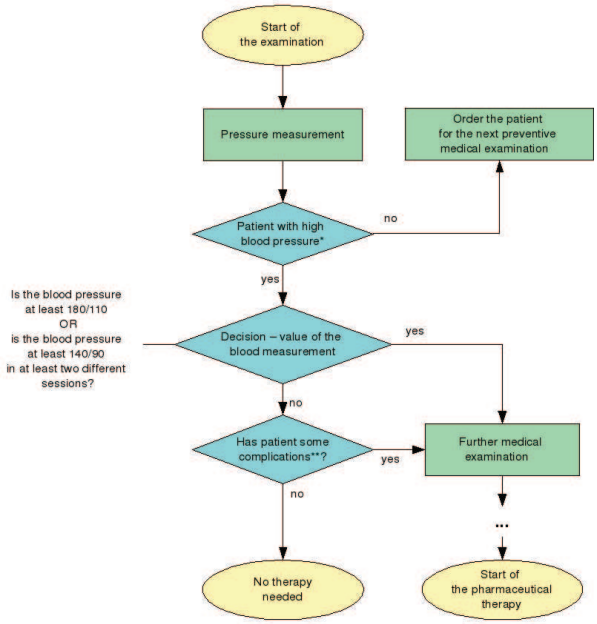

Figure 3: Simplified guideline for hypertension in GLIF

When a patient comes to a preventive examination (or he/she is examined during a longer stay in a hospital) his/her blood pressure is measured and then several decision steps (with possibly further necessary examination) is performed in order to decide whether to begin a pharmaceutical therapy or not. Let us now consider a patient that has a high value of blood pressure (over 180/110). After setting these values into patient's health record, Process Agents (PAs) in the right branch of the guideline would change their state to *simulating* as it is expected that this patient would be treated pharmaceutically[1]. However, in case the physician enters the data for a pharmaceutical treatment without performing further necessary examination, the PA associated with the "Start of pharmaceutical therapy" state would alert the system, as it would change its state in an unexpected way (from the *simulating* state into the *finished* state). In the other case, if the physician enters the data indicating no pharmaceutical treatment at all, the PA associated with the "No therapy needed" state would alert as it would unexpectedly change its state from *inactive* to *finished*.

The second type of alert can be more useful when a set of multiple medical processes is considered concurrently. Let us assume there also is a process describing a diagnosis and a therapy for diabetes mellitus present in the critiquing system. Now let us have a patient that has only one value of blood pressure over 140/90 and other values are from the interval 130-139/85-89. For such a patient no pharmaceutical therapy is needed in case he/she does not have any complications stated above. However, the patient could have results from previous laboratory examinations in his/her data record and in the process related to the diabetes mellitus diagnosis could diagnose this patient with a second type of diabetes mellitus. This diagnosis, as it is being only estimated by PAs in *simulating* state, is set in the environment using only Environment Agent, not storing this prediction in the patient's health record. Therefore the PA related to "Has patient some complications" would send the SIM message to the right branch of the guideline (hence the PA related to 'Start of pharmaceutical therapy" would be in *simulating* state) and the physician can be alerted when he/she indicates that there is no therapy needed.

5.2. Implementation

We implement described multi-agent system using the JADE framework[2], with the JADEX [12] as the extending reasoning engine for the agents. The implementation follows the architecture presented in previous section and depicted in Figure 1. Thanks to using JADE, the communication between agents is designed with respect to FIPA communication standards and as such can be extended with appropriate ontologies and communication standards in healthcare (e.g. designing the communication between the Environment Agent and Patient Health Record Agent with respect to HL7 version 3 standard).

During the implementation we decided not to follow the principles of offline transformation of the process knowledge into the rules for agents as described in [10]. In the approach presented in this paper each agent, that participates in the execution (i.e. Process Agents, Role Agents, and Execution Agents), requests the necessary information (e.g. predecessors of the Process Agent, necessary inputs, etc.) from the Process Director

[1]Note, that if further medical examination is needed, but has not been done yet, the PA connented to "Further medcial examination" would esimate the appropriate output values based on existing data from other patients and passes forward the SIM message.

[2]http://jade.tilab.com/

Agent (PDA). PDA reads the formalized processes in a relevant formalism (medical guidelines, organizational processes) and answers agents to their requests. This approach is equivalent to the offline transformation (by means of usage of processes), but more adaptive in case a change in processes occurs.

6. Discussion and Conclusion

In this paper we have presented the novel way of using the multi-agent system (MAS) as a technological framework for medical processes critiquing decision support system. The approach has several crucial advantages that differentiate it from existing approaches. Firstly, it uses the architecture of the MAS that can work with organizational processes and medical guidelines together. This creates a possibility to develop appropriate monitoring system that is able to control the work practice in a healthcare center jointly on several levels – the procedures for examination reservation or transportation of a patient on one hand, but also the treatment of specific diseases on the other one.

Secondly, it offers several possible ways how to alert healthcare personnel. In the Section 4 we described only the basic one regarding to correct sequence of the performed actions (i.e. whether executed action was executed before its predecessors or the action was not expected to be executed at all). However, thanks to the distributed nature of the system, it can be further improved and specific Process Agents can be enhanced with machine learning techniques that would also alert the doctor about the quality of the entered data.

Finally, such a system can also be used as a simulation tool for processes analysis during organizational process reengineering in healthcare environment, as it also can work with the appropriate medical knowledge, that is necessary to gaining proper results.

In future work, we intend to practically test the presented architecture as a critiquing system in a hospital department, to practically evaluate the approach and identify further improvements. Our critiquing system would focus on hypertension together with related diseases (such as diabetes mellitus and dyslipidemia).

References

[1] D. Isern and A. Moreno, "Computer-based execution of clinical guidelines: A review," *International Journal of Medical Informatics*, vol. 77, no. 12, pp. 787 – 808, 2008.

[2] R. Lenz, R. Blaser, M. Beyer, O. Heger, C. Biber, M. Baumlein, and M. Schnabel, "It support for clinical pathways–lessons learned," *International Journal of Medical Informatics*, vol. 76, no. Supplement 3, pp. S397 – S402, 2007. Ubiquity: Technologies for Better Health in Aging Societies - MIE 2006.

[3] D. Isern, D. Sánchez, and A. Moreno, "Hecase2: A multi-agent ontology-driven guideline enactment engine," in *CEEMAS '07: Proceedings of the 5th international Central and Eastern European conference on Multi-Agent Systems and Applications V*, (Berlin, Heidelberg), pp. 322–324, Springer-Verlag, 2007.

[4] R. Lenz and M. Reichert, "IT support for healthcare processes - premises, challenges, perspectives," *Data Knowl. Eng.*, vol. 61, no. 1, pp. 39–58, 2007.

[5] X. Song, B. Hwong, G. Matos, A. Rudorfer, C. Nelson, M. Han, and A. Girenkov, "Understanding requirements for computer-aided healthcare workflows: experiences and challenges," in *ICSE '06: Proceedings of the 28th international conference on Software engineering*, (New York, NY, USA), pp. 930–934, ACM, 2006.

[6] A. Kumar, B. Smith, M. Pisanelli, A. Gangemi, and M. Stefanelli, "Clinical guidelines as plans: An ontological theory," *Methods of Information in Medicine*, vol. 2, 2006.

[7] J. Fox, A. Alabassi, V. Patkar, T. Rose, and E. Black, "An ontological approach to modelling tasks and goals," *Computers in Biology and Medicine*, vol. 36, no. 7-8, pp. 837 – 856, 2006. Special Issue on Medical Ontologies.

[8] T. Alsinet, C. Ansótegui, R. Béjar, C. Fernández, and F. Manya, "Automated monitoring of medical protocols: a secure and distributed architecture," *Artificial Intelligence in Medicine*, vol. 27, no. 3, pp. 367 – 392, 2003. Software Agents in Health Care.

[9] B. Bosansky and C. Brom, "Agent-based simulation of business processes in a virtual world," in *HAIS '08: Proceedings of the 3rd international workshop on Hybrid Artificial Intelligence Systems*, pp. 86–94, Springer-Verlag Berlin, Heidelberg, 2008.

[10] B. Bosansky and L. Lhotska, "Agent-based simulation of processes in medicine," in *Procceding of PhD. Conference*, pp. 19–27, Institute of Computer Science/MatfyzPress, 2008.

[11] R. Conroy, K. Pyorala, A. Fitzgerald, S. Sans, A. Menotti, G. De Backer, D. De Bacquer, P. Ducimetiere, P. Jousilahti, U. Keil, I. Njolstad, R. Oganov, T. Thomsen, H. Tunstall-Pedoe, A. Tverdal, H. Wedel, P. Whincup, L. Wilhelmsen,

and I. Graham, “Estimation of ten-year risk of fatal cardiovascular disease in europe: the score project.,” *European Heart Journal*, vol. 24, no. 11, pp. 987–1003, 2003.

[12] B. Lars, P. Alexander, and L. Winfried, “Jadex: A bdi-agent system combining middleware and reasoning,” 2005.