about the event-train correlation that were described in hisbooks I am using in my research and in doctoral study of my postgraduate students.

I'd like to stress very heavily. The reviewed book reveals the results of author's 50 years of experience. Each chapter in the book was previously published in Periodicum Biologorum from 1998-2010 years and received very positive feedback from different researchers. This book will be really useful for both professors and students, who are interested in the brain and computer science and its applications in Life and Business.

Anton Ivaschenko, Alfija Dijazitdinova

SCHEDULING BY CONDITIONS FOR TIME BASED REASONING

(Samara State Aerospace University, Povolzhskiy State University
of Telecommunications and Informatics, Samara, Russia)

Nowadays multi-agent technologies are being successfully applied in intelligent systems for resources scheduling and allocation. Especially good results are demonstrated in transportation logistics [1] where geographical constraints can be used for limitation of a number of iterations of agents' negotiation.

One of the main challenges in such systems is concerned with a compromise that needs to be found by a development team. From one side the agents should have freedom of interaction and no direct instructions can be given to them. From the other side the negotiation procedure should be open and comprehensible in order to make it possible for an operator to explain the results and manage the decision making process.

On of the perspective approaches of solving this contradiction is to introduce time management algorithms that are based on statistical analysis of event flows in the system. Events that happen in the system can characterize interaction with external environment (external events) or can be related to the messages that are sent between the agents (data exchange events). Both have irregular time intervals and can be described by non-equidistant time series.

Each agent as well as the operator (represented by the Center or the World agent) can analyze these time series, discover the strategies of negotiators and build its own strategy in the form of a schedule of limited data exchange. This approach intends a procedure of interaction, in which the agents don not exchange all the information that they have, but distribute it to portions, given at different time to different contractors.

In this paper we present a concept of such an approach captured in a technology of "scheduling by conditions", meaning time conditions primarily and illustrate how it can be implemented in peer-to-peer (P2P) outsourcing solution based on auctioning.

Modern integrated information space is a complex system with heterogeneous structure that includes various data bases, pieces of software, tools and systems for a support of decision making. The users of all these components form a certain virtual community that has access to all information items and change its behavior in real time to adaptively react to the events that happen in a process of enterprise functioning. The process of decision makers' interaction is based on information, tasks and reports exchange and results in appearance of a new knowledge, tasks, schedules, and reports that in turn initiate new waves of interaction.

As soon as an enterprise information space can be treated as a complex network of continuously running and co-evolving intelligent agents, the whole solution has been based on holons paradigm and bio-inspired approach [2], which requires development of new methods and tools for supporting fundamental mechanisms of self-organization and evolution similar to living organisms (colonies of ants, swarms of bees, etc).

As soon as in information space there should be several autonomous intelligent systems (agents) [3, 4] to provide decision making support at different stages of enterprise business, there should be organized an effective interaction between them. These components should compete and cooperate, coordinate and adapt their behaviors, aggregate their services to users and take various requirements individually. Each event that occurs here can influence the whole network and needs a collaborative reaction from all dependable components which take into account personal objectives and constraints of each decision making member.

Due to such group behavior and to be able to function in real time this network of intelligent systems and their users should be considered as a complex system with evolvable dynamics and investigated from statistical point of view. Another one requirement for a decision making process based on the agents' negotiation is that the final decision can require a complicated and time consuming process of data exchange between the agents. This process should be treated as self organization of software agents and human operators, which form a heterogeneous information space. That's why it should be managed to reduce the negative effects of human and time factor and assure functioning in real time.

To provide such ability the software solution should consider special functionality for statistical analysis based on recent developments in cross-correlation analysis of non-equidistant time series [5]. The models and methods of such analysis were successfully probated in social management and can be reused for management of multi-agent negotiations.

Also in this area there can be used event processing techniques [6, 7] for an effective continuous processing of time sensitive data in control centers. This technology deals with the analysis of streams of continuously arriving events with the goal of identifying instances of predefined meaningful patterns (complex events). Event processing offers a variety of special operations that are applied on events (e.g., event filtering, projecting, aggregating, splitting, transforming etc.), and enables a special (the event-driven) interaction model.

In many cases however, real-time awareness provided by event processing is not sufficient; real time actions need to be triggered not only by events, but also upon evaluation of additional background knowledge [8, 9]. This knowledge captures semantic metadata descriptions (the domain of interest), and the context related to critical actions and decisions. Its purpose is to be evaluated during detection of complex events in order to on-the-fly enrich events with relevant background information or to propose certain intelligent recommendations in real time.

Let us consider a generalized multi-agent world where tasks w_i (there can be orders, instructions or documents) a proceeded to decision makers represented by agents u_j. The objective of a task is to be proceeded by agents on time (particular KPI can be formulated as an early average absorption). The agents' objective is to receive the most corresponding tasks (with the highest relevance that can be calculated by a number of algorithms, e.g. see below).

Let us set the following task lifecycle events: w_i:

$\varepsilon_i = \varepsilon(w_i, t_i^0)$ – appearance of w_i;

$e_{i,j} = e(w_i, u_j, t_{i,j})$ – incoming of w_i to u_j;

$e'_{i,j} = e'(w_i, u_j, t'_{i,j})$ – excluding of w_i from u_j view;

$e''_{i,j} = e''(w_i, u_j, t''_{i,j})$ – execution of w_i start by u_j;

$\varepsilon'_{i,j} = \varepsilon'(w_i, u_j, t*_{i,j})$ – execution of w_i done by u_j.

A semantic description of every task can be represented by tag clowd $T(w_i) = \{(\tau_{i,k_w}, \omega_{i,k_w})\}$, where τ_{i,k_w} is a tag with a weight ω_{i,k_w}.

Relevance of a task to an agent can be described by an index:

$$R(w_i, u_j) = \sum_{k_w=1}^{M(w_i)} \sum_{k_u=1}^{M(u_j)} f(\omega_{i,k_w}, \omega_{j,k_u}) \cdot [\tau_{i,k_w} = \tau_{j,k_u}], \quad (1)$$

where a statement in [ ] returns 1 if true and 0 if false, and

$$f(\omega_{i,k_w}, \omega_{j,k_u}) = \min(\omega_{i,k_w}, \omega_{j,k_u}) - |\omega_{i,k_w} - \omega_{j,k_u}|. \quad (2)$$

Each agent can have an own behavior model, e.g. for a taxi driver the following objective can be reasonable:

$$\sum_{i=1}^{N_w} e''(w_i, u_j, t''_{i,j}) \cdot (t*_{i,j} - t''_{i,j}) \to \max. \quad (3)$$

The scheduling problem can be formalized as:

$$\sum_{i=1}^{N_w} \sum_{j=1}^{N_u} \varepsilon(w_i, t_i^0) \cdot e''(w_i, u_j, t''_{i,j}) \cdot (t''_{i,j} - t_i^0) \to \min, \quad (4)$$

$$\sum_{i=1}^{N_w} \sum_{j=1}^{N_u} \varepsilon(w_i, t_i^0) \cdot e''(w_i, u_j, t''_{i,j}) \cdot (t''_{i,j} - t_i^0) \to \min, \quad (5)$$

$$\sum_{i=1}^{N_w} \sum_{j=1}^{N_u} e''(w_i, u_j, t''_{i,j}) \cdot R(w_i, u_j) \to \max, \quad (6)$$

$$\forall w_i : \sum_{k=1}^{N_u} \varepsilon(w_i, u_k, t^0{}_{i,k}) = 1, \ \sum_{j=1}^{N_u} e''(w_i, u_j, t''_{i,j}) = 1. \quad (7)$$

This problem cannot be solved by an optimization algorithm because of the specifics of problem domain: agents interact according to their own behavior model and tasks are received in real time.

To solve this problem we propose an approach based on scheduling by conditions. Let us consider a control system with input of $\varepsilon_{i,j}$ and output that contains $e''_{i,j}, \varepsilon'_{i,j}$. Feedback can be formalized as a chain of events $e_{i,j}$ and $e'_{i,j}$. Note that $e''_{i,j}$ should always have a corresponding $e'_{i,j}$. To create these events a scheduling procedure should be implemented that prescribes a distribution of tasks between the agents across time.

As soon as the described events are represented by non equidistant time series, one can define a discretization interval Δt and determine a rhythmicity function:

$$\rho(J\Delta t) = \sum_{i=1}^{N_w}\sum_{j=1}^{N_u}\Big(\varepsilon(w_i, t_i^0)\cdot e(w_i, u_j, t_{i,j})\cdot \theta(J\Delta t - t_{i,j})\cdot \cdot\big(1 - e'(w_i, u_j, t'_{i,j})\big)\theta\big(t'_{i,j} - (J+1)\Delta t\big)\Big), \quad (8)$$

where $\theta(x)$ is a step function.

In case of regular event flow: $t^*_{i,j} = t''_{i,j} + \Delta t^*_{i,j}$, $t''_{i,j} = t_{i,j} + \Delta t_j^s + \zeta_{i,j}$:

$$\rho(J\Delta t) = \text{constanta}. \quad (9)$$

An architecture that implements the described approach is described on Fig. 1.

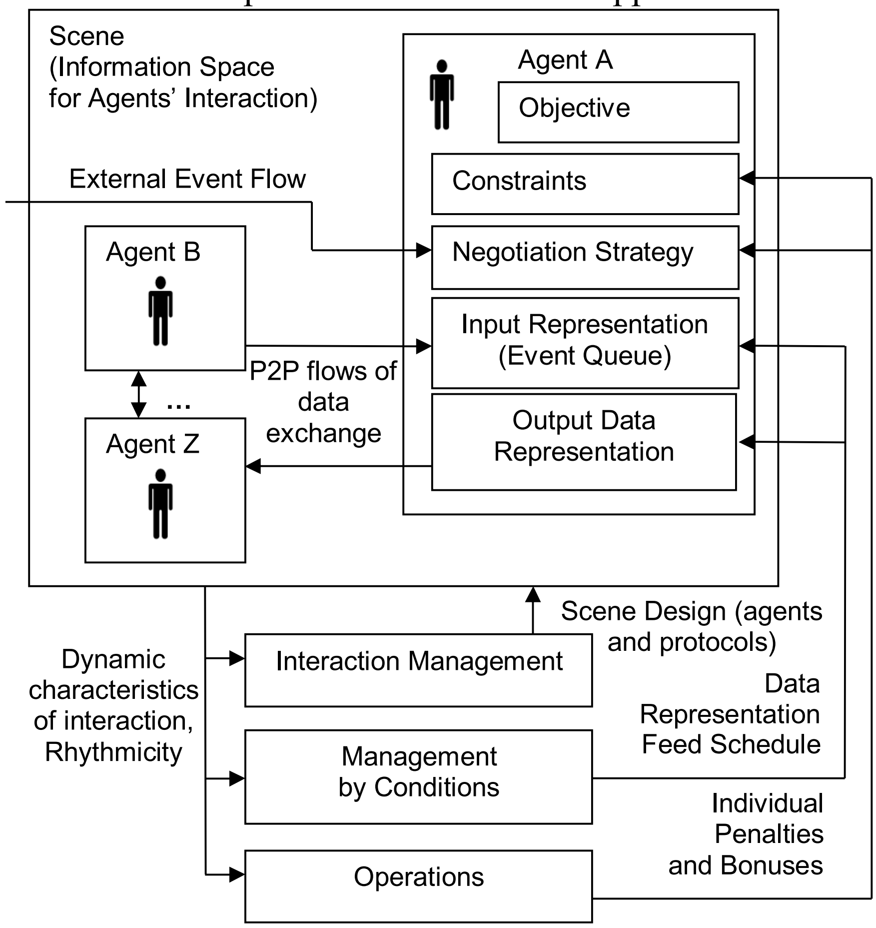

Fig. 1. An agents' world architecture for scheduling by conditions

In this paper we describe a solution for time based reasoning based on a technology of "scheduling by conditions". The example is given for P2P outsourcing problem using iteration auctions with time-based regulation. This approach might be interesting for developers of multi-agent technologies for intelligent resources scheduling.

References

1. Glaschenko, A., Ivaschenko, A., Rzevski, G., Skobelev, P.: Multi-agent real time scheduling system for taxi companies. Proceedings of the 8-th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009), Budapest, Hungary, pp. 29-36 (2009)

2. Leitao, P.: Holonic rationale and self-organization on design of complex evolvable systems", HoloMAS 2009, LNAI 5696, Springer-Verlag Berlin Heidelberg, p. 1-12 (2009)

3. Gorodetskii, V.I.: Self-organization and multiagent systems: I. Models of multiagent self-organization. Journal of Computer and Systems Sciences International, Volume 51, Issue 2, pp. 256-281 (2012)

4. Gorodetskii, V.I.: Self-organization and multiagent systems: II. Applications and the development technology. Journal of Computer and Systems Sciences International, Volume 51, Issue 3, pp. 391-409 (2012)

5. Prokhorov, S. et al: Applied analysis of random processes, Samara Scientific Center of RAS, 582 p. (2007)

6. Anicic, D., Fodor, P., Rudolph, S., Stuhmer, R., Stojanovic, N., Studer, R.: A rule-based language for complex event processing and reasoning, P. Hitzler, T. Lukasiewicz (eds.), RR 2010, LNCS, vol. 6333, Springer, Heidelberg, pp. 42-57 (2010)

7. Anicic, D., Rudolph, S., Fodor, P., Stojanovic, N.: Stream Reasoning and Complex Event Processing in ETALIS, Semantic Web Journal, doi>10.3233/SW-2011-0053, Special Issue: Semantic Web Tools and Systems (2012)

8. Krämer, J., Seeger, B. Semantics and implementation of continuous sliding window queries over data streams. ACM Transactions on Database Systems, 34(1), pp. 1-49 (2009)

9. Artikis, A., Paliouras, G., Portet, F., Skarlatidis, A.: Logic-based representation, reasoning and machine learning for event recognition. In J. Bacon, P. R. Pietzuch, J. Sventek, & U. C¸ etintemel (Eds.), Proceedings of the 4th ACM International Conference on Distributed Event-Based Systems. New York, NY, USA: ACM, DEBS'10, pp.282-293 (2010)