The advantages of fitting orthogonal polynomials to experimental data are well-known [2]. In our case, the situation is more complicated because of the nonlinear functional dependence of the target on the fitting parameters. However, the main advantage of this approach is sustained: due to the orthogonal property, each coefficient in the dispersion law representation can be determined independently from the others. If one has already obtained an evaluation of m^{-th} degree polynomial, an additional term in the dispersion law ((m+1) degree polynomial) requires only one new coefficient to be determined. The other coefficients remain the same, unlike in the Cauchy or Selmeier case. In the Cauchy case, high order polynomials may result in ill conditioned matrices. Besides, the joint confidence region in the parameter space, estimated by the covariance matrix, has minimum volume.

Conclusion

We have shown that the use of orthogonal polynomials in refractive index modeling is effective and highly productive. Although the involved coefficients have no physical meaning, this is also true for the Cauchy and Selmeier models. The OP principal feature is that the number of parameters to be fitted can be kept low at initial steps and then it can be increased, retaining the intermediate results. The orthogonal polynomials approach can be of use in many branches of material science, including photonic crystal design, optimization of elements for effective conversion of solar radiation, etc.

Acknowledgments

This work was partially supported by the National Science Foundation at the Ministry of Education of Bulgaria by grant D01-377/2006.

Bibliography

1. [Born, 1957] E. Born and P. Wolf, Principles of Optics, Ed. Princeton, New York, 1957.
2. [Himmelblau, 1970] D. Himmelblau, Process analysis by statistical methods, Ed. John Wiley & Sons, New York, 1970.

Authors' Information

Peter Sharlandjiev *– Central Laboratory of Optical Storage and Processing of Information, BAS, Acad.G.Bontchev St. bl. 101, Sofia-1113, Bulgaria; e-mail: pete@optics.bas.bg .*

Georgi Stoilov *– Institute of Mechanics, BAS, Acad.G.Bontchev St. bl. 4, Sofia-1113, Bulgaria; e-mail: gstoilov@imbm.bas.bg.*

FORMALIZATION OF INTERACTION EVENTS IN MULTI-AGENT SYSTEMS

Dmitry Cheremisinov, Liudmila Cheremisinova

Abstract: *The problem of the description of interaction between spatially divided agents in the form of dialogues is explored. The concept of processes synchronization is analyzed to formalize the specification of interaction at the level of events constituting the processes. The approach to formalization of the description of conditions of synchronization when both the independent behavior and the communications of agents can be presented at a logic level is offered. It is shown, that the collective behavior of agents can be specified by the synthetic temporal logic that unites linear and branching time temporal logics.*

Keywords*: multi-agent system, interaction protocol, time.*

ACM Classification Keywords*: I.2.11 [Computer Applications]; Distributed Artificial Intelligence, Multiagent systems; D.3.3 [Programming Languages]: Language Constructs and Features – Control structures, Concurrent programming structures*

Introduction

A multi-agent system can be considered as the organization of agents (by analogy to the human organization) or, in other words, as some artificial society. It is a computational system in which two or more agents interact or work together to perform a set of tasks or to achieve a set of goals [1]. One of the core concept of multi-agent systems is *interaction*, that is the foundation for cooperative behavior among several autonomous agents. Agent interactions are established through exchanging information in the form of messages that specify the desired performatives of interacting agents. Agent system can operate if the agents have a common understanding of the possible types of messages, then they must know which messages they can expect in a particular situation and what they may do when they got some message. So messages exchanged between agents in some multi-agent system need to follow some standard patterns which are described in agent interaction *protocol*.

Protocols play the central role in agent communication. An interaction protocol defines the rules the dialog among agents conforms to. It constrains the possible sequences of messages that may occur in agent interaction. Interacting agents should comply with an interaction protocol in order to engage permissible sequences of message exchange. When agent sends a message it can expect a response to be among a set of messages indicated by the accepted protocol. The interaction protocol can be assigned by the designer of the multi-agent system otherwise an agent needs to indicate the protocol that it wants to follow before it starts to interact with other members of the system.

It is necessary for any protocol itself to be correct and verifiable. If it is not correct then the agents that follow it may perform contradictory and unexpected actions leading to possible breakdown of the interaction. The central problem of the verification of interactions (dialogues of negotiations) that take place in open (not being cooperative) systems is the problem of conformance inspection between behavior of agents and interaction protocol. That is the protocol must be understandable by all agents of the system and the they behave according to this protocol. The implementation of conformance inspection confront with a problem of identification of dialogue steps between agents. Recognition of the dialogue step which is carried out by two spatially divided agents requires analyzing the concept of interaction of processes.

At the heart of the formal models of a protocol are cooperating sequential processes. Fundamental feature, the proposed protocol models differ, is the degree of synchronization of behaviors of participants of interaction. There is still a need for a proper formalism for the process of synchronization that is suitable for human understanding and automated implementation. In this paper we focus on logical analysis of synchronization of behaviors of interacting participants. The simple yet expressive class of interactions is considered, namely dialogues consisting of separate steps. The considered dialogues involve only two agents. This restriction allows concentrating on the kernel of the problem of synchronization in different formal models of interaction protocols. The agent interaction is considered as interaction between two (or more) processes. And a special case of such interaction is considered, when one of processes outputs at the same time as the other one inputs it. The actions of message exchanging have duration. The concept of a process and an event are analyzed to formalize the specification of interaction at the level of events.

Formalization of the concept of interaction event

Usually collective behaviour of multi-agent system is described as a dialogue of agents which communicate by means of sending and receiving messages. On each step of activity an agent carries out some action depending on its internal state and the received message. As a result of the action the agent changes its internal state and sends some messages to other agents. Speaking informally, the architecture of an agent includes 1) the internal structures of data defining internal states of the agent, 2) mail box containing messages from other agents, 3) integrity restrictions on the agent internal states, 4) actions which the agent can execute, and 5) the program that specifies the control of action execution. Execution of an action consists of 1) changing a current internal state of the agent and 2) sending a message to other agents. The current contents of a mail box consist of the messages received by the agent from other agents on the previous step. The global state of a multi-agent system consists of internal states and contents of mail boxes of all agents of the system [1].

To specify independent behaviors of agents, formalisms of high level abstractness are widely used, for example, such as temporal logic. At the same time the communications between agents are specified by means of the concepts of realization level, such as mail boxes and messages. One of the problems of such segregated approach to interaction lies in that it is extremely difficult to simulate interactions between agents though at the same time the independent behaviour of separate agents is described completely. This problem arises due to the absence of agent model unifying all aspects of both independent behaviours and the communication. The main reason of the absence of such a general model is that there exists no general conceptual basis unifying all abstractions, connected with collective behaviour of agents.

When analyzing the behaviour of multi-agent system agents are characterized by processes. The process is specified by exhaustive description of potential behaviour of the agent. The process consists of events. Thus, to be in position to analyze the concept of interaction of processes, a suitable axiomatization of the concept of an event is required.

The concept of an event allows abstracting from physical time when describing behaviour of a system. The widespread axiomatization of an event is connected with the assumption, that events have no duration [2]. The behaviour of a multi-agent system consists of some events – steps of dialogue between agents – and is sequential in this sense. For recognition of the step which is carried out jointly by two spatially divided agents, it is impossible to bypass the concept of parallelism.

The models of parallelism known in the literature could be roughly divided into two classes: 1) the models, in which concurrent execution of two processes is described by interleaving of (atomic) events of those processes; 2) models in which causal dependencies between events are set explicitly. Interleaving models are focused on systems with events considered as instantaneous and indivisible. In this case the act of interaction is a complete event which describes participation of all processes cooperating in this act [2]. This act as the step of a dialogue is carried out by two spatially divided agents and represents the event which should have duration and structure.

There is popular opinion the concept of an event having duration is reduced to the concept of an instantaneous event. The following formulation of this assumption is taken from Hoare [3, p. 24]; "The actual occurrence of each event in the life of an object should be regarded as an instantaneous or an atomic action without duration. Extended or time-consuming actions should be represented by a pair of events the first denoting its start and the second denoting its finish."

Now it is known that this opinion is erroneous, and often splitting, i. e. the use of pairs of instantaneous events to model events having duration, is unnatural. Mutual irreducibility of the concept of an event having duration and the concept of instantaneous event is proved formally and constructively [4]. The formal proof is based on the incomparability of these formalisms describing event systems [5]. Systems of the durational events are described by the causal relation (branching-time temporal logic), systems of instantaneous events – by relation of consequence and parallelism (linear-time temporal logic).

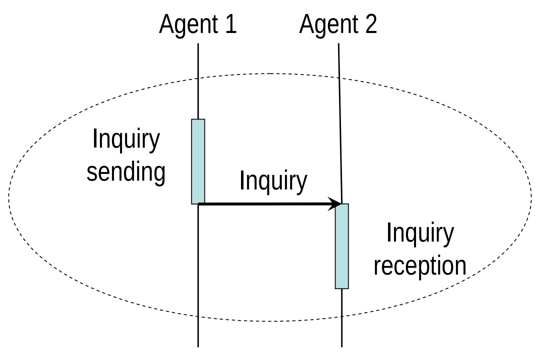

Fig. 1. Structure of the first model of a step of dialogue

The problem under discussion is how processes and events can be assembled together into a system in which the components interact with each other and with their external environment. The elementary structure of the dialogue step is a pair of durational events which constitute the step densely without a time interval between. First event of the pair can be interpreted as "pronouncing" of the message by one of the agents; the second event can be interpreted as "perception" of this message by the other participant of the dialogue. The basic feature of this structure is the assumption of density of the event composition and that members constituting the event belong to behaviours of different agents (fig. 1). Absence of a time interval between of pair of durational events designates that the event of synchronization of corresponding processes is instantaneous.

The first model of an interaction event

Ignoring the functionality of agents, it is possible to consider synchronization of their behaviour as the only goal of interaction. Thus, the dialogue step is the composition of three events, two events are durational ones, and the other is instantaneous event. However the events constituting interaction still remain occurring simultaneously in different processes.

On the one hand, synchronization of agent behaviours occur during the rare moments, in the rest of the time communicating agents behave independently from each other. On the other hand, processes should interchange information about current states to ensure synchronization. Formally it can be reached by splitting of all events constituting agent behaviour on internal and external ones. Only external events of the agent behaviour can be "visible" to the other agents. In this case the specification of the agent behaviour is the cause-effect relation on a set of possible events. In particular, this relation describes the reasons of occurrence of external and internal events.

Let a composition of a durational internal event E and instantaneous external event y is an operation. A composition $y \rightarrow E$ of durational and instantaneous events is called as a waiting operation that waits the external event y, and a composition $E \rightarrow y$ is called as an acting operation which effect is the realization of external event y. It is necessary to note, that the event sequence in both operations is the same: the first one is durational event, the second is instantaneous event. The mentioned compositions allow considering dependences between events in a composition as cause and effect because from physical reasons, event-consequence occurs behind event-reason without overlapping on time. Waiting operation is a durational event and the reason of its termination is the occurrence of y. Acting operation is a durational event too and it is the reason of occurrence of y. The symbol "$\rightarrow$" can be interpreted as cause and effect dependence between events.

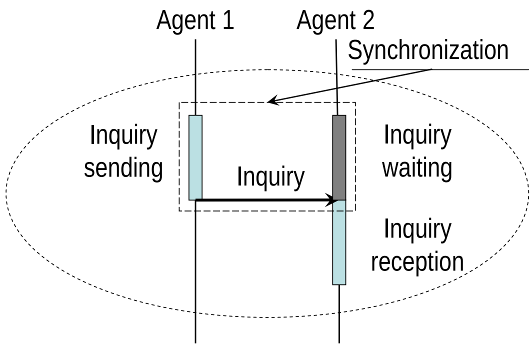

Fig. 2. Synchronization of behaviour of agents

Such treatment of waiting and acting operations is a basis of the formal semantics [6, 7] of PRALU language [8] in which conjunctions of Boolean variables describe external events of operations. PRALU language in this interpretation represents the synthetic temporal logic uniting linear and branching time temporal logics supplied by the assumption of density of time [9]. Temporal formulas of this logic are interpreted as the statements concerning event sequences of two sorts: instantaneous and durational.

The composition of events considered above allows describing independent behaviours of agents. Parallel execution of waiting operation $y \rightarrow E$ by one agent and acting operation $E \rightarrow y$ by the other one results in synchronization of behaviour of agents during the moment of occurrence of instantaneous event y (fig. 2). By the definition the effect of a waiting operation is its termination at a moment of occurrence of instantaneous external event y.

The line of "life" of the agent consists of pairs waiting and acting operations. The boundary between waiting and acting operations serves as the synchronization event. Here the action consists from "perception" of the accepted message and "pronouncing" new one. Obviously, the occurrence of synchronization depends on duration of acting operations.

The second model of an interaction event

A dialogue step can be considered also as the other composition of some three events. One of them is durational, and the others are instantaneous (fig. 3). In this model of a dialogue step the interaction itself is a durational event. This event, having duration, should have a physical basis. Without loss of a generality it is possible to consider that event of interaction occurs in an environment of agents. In this model interaction event becomes not distributed, but a local one in the external environment.

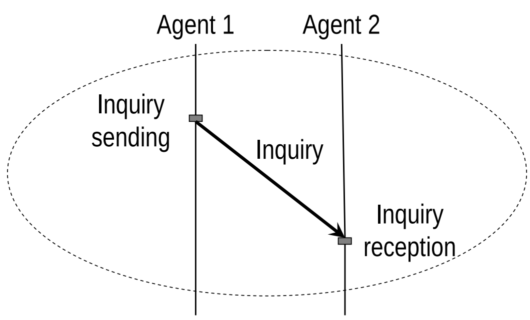

Fig. 3. Structure of the second model of an interaction event

Concept of an environment

From analysis of physical realizations of distributed systems it follows that the synchronization requires a special organization of a system of cooperating agents. Two basic types of the system organization aimed achieving synchronization are known: synchronous and asynchronous systems. The standard definition of the distinction of these types of systems declares that synchronous systems have the same shared "clock", and in asynchronous systems each agent has its own independent clock. It is obvious, that the shared "clock" belong to an external environment of all agents of the synchronous system.

In traditional interaction theories CCS [3] or CSP [10] the concept of an environment is used implicitly, hence it is not formalized. CCS and CSP rely on the concept of an environment having the following distinguishing features. The environment is considered simply as the other agent. In other words, the environment for the given agent includes all other agents of the system that operate in parallel with this agent. In this case an agent and an environment are objects of the same nature. It is obvious, that this assumption of properties of an environment is not good enough from the point of view of specifying an agent interaction directed to achievement of synchronization. Such an approach is justified only by the following reasoning. It is considered that the concept of an environment concerns with the system realization and it is not represented at the level of agent behaviour.

Our purpose is to offer a formal model of interaction which is not concerned with a system realization. We consider an environment is essentially distinct from agents. The basis of this approach is that the interaction is considered as the communication act consisting of sending and receiving of messages. This formalization of interaction originates from Shannon's paper about the theory of communication [11] in which interaction is considered as a way to transfer the message from a sender to a receiver through a medium, also called as transfer environment. Physical realization of an environment can be a computer program, a device or a physical environment.

Obviously, synchronization of agent behaviours is impossible without fixing data which are transferred by an environment during agent interaction [12]. Thus, environment serves as a model of transport system to deliver messages. In other words the environment can be considered as the memory that is shared with all agents. This memory is known as a global state of multi-agent system. In its most simple form, the communication can be based on the fixed set of differing signals. In the case of binary signals the representation of a global state is a set of the Boolean variables which values are possible signals. In the case of structural signals an agent environment usually refers as message passing system. The concept of an environment is closely concerned with a notion of autonomy of agents. Autonomy of agents has its focus on freely choosing between actions and on acting independently. Autonomy means also that the agents receive all information only through an environment.

System, in which the behaviour of an environment is deterministic, refers to closed system. In the case of a closed system it is supposed, that the reasons of all events are inside of the system and its behaviour is completely self controlled. If the behaviour of an environment is nondeterministic, the system refers to an opened system. Unlike the memory considered in the theory of finite state machines, the behaviour of the memory of an environment of the open multi-agent system can depend on uncontrollable conditions.

The specification of interaction in the form of description of a message passing system does the description of autonomous behaviour of the separate agent not closed because this description is not enough for understanding of the complete behaviour of the agent. Obviously, most important property of the message passing system is restriction on length of durational events, imposed by this system.

Time as a logic concept

One of the possible semantics of ts and their interaction is that of differential equations. The agents represent relations between continuous-time functions, and the interactions are the continuous-time functions by their nature. In the systems of differential equations a continuous time variable t is used. The task of an execution environment is to find a solution of the system of differential equations, i.e., a set of continuous-time functions that satisfy all the relations. This solution is seeked by "integrating" or "numerically solving" differential equations. The problem with this approach is that it is in general case a hard mathematical problem requiring much effort in searching a solution.

There are several meanings in which computation of "solving differential equations" may be understood. If by "solving" we mean manipulating the mathematical expressions of the differential equations to get a mathematical formula for the closed-form solution, then this (to the extent it is algorithmic) is computation of the familiar discrete, symbol-manipulating variety. Alternately, we may take “solving” in the extended sense, in which a multy-agent system is “solving” a system of differential equations when the multy-agent system behavior can be described (perhaps approximately) by those differential equations. Here, the description of the multy-agent system by differential equations has nothing to do with whether it is can be computated or not. Multy-agent systems may be described by differential equations, but it may be not computable.

Differential equations can be discretized to get difference equations. In this case, a global clock defines the discrete points at which signals have values (at the ticks). Difference equations are considerably easier to implement in software than differential equations. Their key weaknesses are the global synchronization implied by the clock, and the awkwardness of specifying irregularly timed events and control logic.

In the previous part of the paper time was considered as the logic concept expressed by relations between events through their sequence and order. Time is discrete, because there is an observable time quantization by the events that is fixed in behaviour of an environment.

Time cannot be measured, if we do not impose some restrictions on the duration of events in all components of a multi-agent system. Time is measured if each event in a history of the system behaviour is accompanied with a number that expresses either duration of the event or specifying the moment of time when it occurs. Synchronization of behaviour of agents means that time is measured. But measured time model is not difference equations model.

Measured time can be realized, if we assume, that the duration of all simultaneously executed acting operations in a multi-agent system is identical. It is natural to accept this duration as the unit of time. In this case in the closed systems the duration of waiting operations is expressed by an integer $i \geq 1$. The assumption that duration of all simultaneously executed acting operations is identical holds in synchronous systems (global synchronization implied by the clock). Obviously, this assumption specifies a pairs of interacting waiting and acting operations by the counter number of the appropriate step of time. Synchronous system keeps the assumption that the time is discrete and measured.

Other assumption that allows realizing measured time is that any operation, carried out in parallel to itself, is illegal. In this case realization of any operation in a history of the agent functioning can be accompanied with some counter number of this realization. The formal proof of this statement is in [7]. The function which calculates a counter number of the operation realization (from the start of the system) when this operation starts can be used for measurement of time. Interaction occurs only in pairs of waiting and acting operations which have the same counter number. This is known as a rendezvous condition.

Rendezvous models is a part of Hoare's communicating sequential processes (CSP) [3] and Milner's calculus of communicating systems (CCS) [10]. In these frameworks, rendezvous is atomic, instantaneous action of communication. If two processes are to communicate, and one of them reaches first the point at which it is ready to communicate, then it stalls until the other process will be ready to communicate. Here “atomic” (rendezvous) means that the two processes are simultaneously involved in the exchange, and that the exchange is initiated and completed in a single uninterruptible step. Rendezvous in our framework is an event with duration and structure. A key weakness of rendezvous of CSP and CCS is that maintaining determinacy can be difficult. The proposed framework could break down the problem of specification determinacy.

An asynchronous system keeps the assumption that time is discrete and measured, but rejects the assumption that duration of all simultaneously executed acting operations is identical. The last principle of measuring time differs from that for synchronous system.

Conclusion

The independent behaviour of agents in the majority of models of multi-agent systems is described by means of formalisms of high level abstractness, but the communication is specified by the concepts close to realization. The difference of levels of the description does not allow simulating communications between agents at the level in which their independent autonomous behaviour is described. This problem arises because of absence of agent models that unify all aspects of local behaviour and the communications.

In the paper we suggest to describe the synchronization conditions by specification of event properties which have been not concerned with the realization of these events. Our approach allows specifying both the independent behaviour and the communication at a level of logic. It is shown, that the collective behaviour of agents can be described by the synthetic temporal logic that unites the linear and branching time temporal logics. Such synthetic logic is one of interpretations of the existing PRALU language.

The transition from the analysis to design in development of the software is always based on mapping or transformation of conceptual models. The use of models during the analysis is inevitable. Analysis models target to describe system of the real world as mapping into some problem area. The concepts used in analysis model concern directly with concepts of system of the real world. On the other hand, the models used for design use an additional level of abstraction and pursue other purpose. Models for designing describe such concepts of the software, as objects, structures and processes which only are indirectly connected with concepts of problem area. The purpose of model for designing consists in masking details of realization and to create a formal basis for its subsequent transformation in the program.

Our contribution in this paper is both a new behavior model of agent interaction in multiaget systems, and the interactinon event abstraction which seems to be a good abstraction for distributed and parallel programming. Distributed systems design is unnecessarily complex because our current conceptual models do not provide the right kinds of abstractions. By adding appropriate abstraction to our models, we can also reduce the conceptual distance between analysis and design.

Acknowledgements

The research was partially supported by the Fond of Fundamental Researches of Belarus (Project F07-125).

Bibliography

1. Subrahmanian V.S., Bonatti P., Dix J. et al. “Heterogeneous Agent Systems”, MIT Press, 2000.
2. Brookes S.D., Hoare C. A.R., and Roscoe A.D. “A Theory of Communicating Sequential Processes”, Journal of the ACM, no 31(3), pp. 560–599, 1984.
3. C.A.R. Hoare “Communicating Sequential Processes”, Prentice Hall International Series in Computer Science, 1985.
4. Van Glabbeek R., Vaandrager F. “The Difference between Splitting in n and n+1”, Report CS-R9553, Centre for Mathematics and Computer Science, Amsterdam 1995; Abstract in: Proceedings 3rd Workshop on Concurrency and Compositionality, Goslar, March 5-8, 1991 (E. Best & G. Rozenberg, eds.), GMD-Studien Nr. 191, Sankt Augustin, Germany 1991.
5. Cheremisinov D.I. “The Real Difference between Linear and Branching Temporal Logics”, Workshop on Discrete-Event System Design DESDes ’ 04, University of Zielona Gora Press, Poland, 2004, pp. 103–108, 2004.
6. Cheremisinov D.I. “Formal description of behaviour of the distributed systems”, Minsk: Belarus, 1991, preprints no 38. – 44 p. (in Russian).
7. Cheremisinov D.I. “The morphisms of reactive system formal languages”, Informatics, no 1 (5), pp. 76–88, 2005 (in Russian).
8. A.D. Zakrevskij, “Parallel algorithms for logical control”, Minsk, Institute of Engineering Cybernetics of NAS of Belarus, 202 p., 1999 (in Russian).
9. Cheremisinov D.I. “About interpretation of temporal logic at symbolical verification”, Informatics, no 1, pp. 131–138, 2004 (in Russian).
10. Milner R. “A calculus of communication systems”, LNCS 92, Springer Verlag, 1980.
11. Shannon C.E. “A mathematical theory of communication”, Bell System Technical Journal, vol. 27, pp. 379–423 and pp. 623–656, 1948.
12. Odell J., Parunak H. V. D., Fleischer M., Brueckner S. “Modeling Agents and their Environment”, Proceedings of the Third International Workshop on AgentOriented Software Engineering, Lecture Notes in Computer Science, Springer Verlag (Berlin, D), vol. 2585, pp. 16–31, 2003.

Authors' Information

Dmitry Cheremisinov, Liudmila Cheremisinova *– The United Institute of Informatics Problems of National Academy of Sciences of Belarus, Surganov str., 6, Minsk, 220012, Belarus, Tel.: (10-375-17) 284-20-76, e-mail: cher@newman.bas-net.by, cld@newman.bas-net.by*