Encyclopedia of Artificial Intelligence

Juan Ramón Rabuñal Dopico
University of A Coruña, Spain

Julián Dorado de la Calle
University of A Coruña, Spain

Alejandro Pazos Sierra
University of A Coruña, Spain

Information Science REFERENCE

INFORMATION SCI

Hershey · New York

Director of Editorial Content: Kristin Klinger
Managing Development Editor: Kristin Roth
Development Editorial Assistant: Julia Mosemann, Rebecca Beistline
Senior Managing Editor: Jennifer Neidig
Managing Editor: Jamie Snavely
Assistant Managing Editor: Carole Coulson
Typesetter: Jennifer Neidig, Amanda Appicello, Cindy Consonery
Cover Design: Lisa Tosheff
Printed at: Yurchak Printing Inc.

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue, Suite 200
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com/reference

and in the United Kingdom by
Information Science Reference (an imprint of IGI Global)
3 Henrietta Street
Covent Garden
London WC2E 8LU
Tel: 44 20 7240 0856
Fax: 44 20 7379 0609
Web site: http://www.eurospanbookstore.com

Library of Congress Cataloging-in-Publication Data

Encyclopedia of artificial intelligence / Juan Ramon Rabunal Dopico, Julian Dorado de la Calle, and Alejandro Pazos Sierra, editors.
p. cm.
Includes bibliographical references and index.
Summary: "This book is a comprehensive and in-depth reference to the most recent developments in the field covering theoretical developments, techniques, technologies, among others"--Provided by publisher.
ISBN 978-1-59904-849-9 (hardcover) -- ISBN 978-1-59904-850-5 (ebook)
1. Artificial intelligence--Encyclopedias. I. Rabunal, Juan Ramon, 1973- II. Dorado, Julian, 1970- III. Pazos Sierra, Alejandro.
Q334.2.E63 2008
006.303--dc22
2008027245

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

All work contributed to this encyclopedia set is new, previously-unpublished material. The views expressed in this encyclopedia set are those of the authors, but not necessarily of the publisher.

M

Modal Logics for Reasoning about Multiagent Systems

Nikolay V. Shilov
Russian Academy of Science, Institute of Informatics Systems, Russia

Natalia Garanina
Russian Academy of Science, Institute of Informatics Systems, Russia

INTRODUCTION

It becomes evident in recent years a surge of interest to applications of modal logics for specification and validation of complex systems. It holds in particular for combined logics of knowledge, time and actions for reasoning about multiagent systems (Dixon, Nalon & Fisher, 2004; Fagin, Halpern, Moses & Vardi, 1995; Halpern & Vardi, 1986; Halpern, van der Meyden & Vardi, 2004; van der Hoek & Wooldridge, 2002; Lomuscio, & Penczek, W., 2003; van der Meyden & Shilov, 1999; Shilov, Garanina & Choe, 2006; Wooldridge, 2002). In the next paragraph we explain what are logics of knowledge, time and actions from a viewpoint of mathematicians and philosophers. It provides us a historic perspective and a scientific context for these logics.

For mathematicians and philosophers logics of actions, time, and knowledge can be introduced in few sentences. A logic of actions (ex., Elementary Propositional Dynamic Logic (Harel, Kozen & Tiuryn, 2000)) is a polymodal variant of a basic modal logic **K** (Bull & Segerberg, 2001) to be interpreted over arbitrary Kripke models. A logic of time (ex., Linear Temporal Logic (Emerson, 1990)) is a modal logic with a number of modalities that correspond to "next time", "always", "sometimes", and "until" to be interpreted in Kripke models over partial orders (discrete linear orders for LTL in particular). Finally, a logic of knowledge or epistemic logic (ex., Propositional Logic of Knowledge (Fagin, Halpern, Moses & Vardi, 1995; Rescher, 2005)) is a polymodal variant of another basic modal logic **S5** (Bull & Segerberg, 2001) to be interpreted over Kripke models where all binary relations are equivalences.

BACKGROUND: MODAL LOGICS

All **modal logics** are languages that are characterized by syntax and semantics. Let us define below a very simple modal logic in this way. This logic is called **Elementary Propositional Dynamic Logic (EPDL)**.

Let *true, false* be Boolean constants, *Prp* and *Rel* be disjoint sets of propositional and relational variable respectively. The syntax of the classical propositional logic consists of formulas which are constructed from propositional variables and Boolean connectives "$\neg$" (negation), "&" (conjunction), "$\vee$" (disjunction), "$\rightarrow$" (implication), and "$\leftrightarrow$" (equivalence) in accordance with the standard rules. EPDL has additional formula constructors, modalities, which are associated with relational variables: if r is a relational variable and φ is a formula of EPDL then

- $([r]\varphi)$ is a formula which is read as "box r-φ" or "after r always φ";
- $(\langle r\rangle\varphi)$ is a formula which is read as "diamond r-φ" or "after r sometimes φ".

The semantics of EPDL is defined in models, which are called **labeled transition systems** by computer scientists and **Kripke models**[1] by mathematicians and philosophers. A model M is a pair (D, I) where the domain (or the universe) $D\neq\emptyset$ is a set, while the interpretation I is a pair of mappings (P, R). Elements of the domain D are called states by computer scientists and worlds by mathematicians and philosophers. The interpretation maps propositional variables to sets of states $P: Prp\rightarrow 2^D$ and relational variables to binary relations on states $R: Rel\rightarrow 2^{D\times D}$. We write $I(p)$ and

I(r) instead of *P(p)* and *R(r)* whenever it is implicit that *p* and *r* are propositional and relational variables respectively.

Every model $M = (D, I)$ can be viewed as a directed graph with nodes and edges labeled by propositional and action variables respectively. Its nodes are states of *D*. A node $s \in D$ is marked by a propositional variable $p \in Prp$ iff $s \in I(p)$. A pair of nodes $(s_1,s_2) \in D \times D$ is an edge of the graph iff $(s_1,s_2) \in I(r)$ for some relational variable $r \in Rel$; in this case the edge (s_1,s_2) is marked by this relational variable *r*. Conversely, a graph with nodes and edges labeled by propositional and relational variables respectively can be considered as a model.

For every model $M = (D,I)$ the **entailment** (validity, satisfiability) **relation** $\models_M$ between states and formulas can be defined by induction on formula structure:

- for every state $s \models_M$ *true* and not $s \models_M$ *false*;
- for any state *s* and propositional variable *p*, $s \models_M p$ iff $s \in I(p)$;
- for any state *s* and formula φ, $s \models_M (\neg\varphi)$ iff it is not the case $s \models_M \varphi$;
- for any state *s* and formulas φ and ψ,
  $s \models_M (\varphi \& \psi)$ iff $s \models_M \varphi$ and $s \models_M \psi$;
  $s \models_M (\varphi \vee \psi)$ iff $s \models_M \varphi$ or $s \models_M \psi$;
- for any state *s*, relational variable *r*, and formula φ,
  $s \models_M ([r]\varphi)$ iff $(s,s') \in I(r)$ and $s' \models_M \varphi$ for every state s' ;
  $s \models_M (\langle r \rangle \varphi)$ iff $(s,s') \in I(r)$ and $s' \models_M \varphi$ for some state s' .

Semantics of the above kind is called **possible worlds semantics**.

Let us explain EPDL pragmatics by the following puzzle example.

Alice and Bob play the Number Game. Positions in the game are integers in [1..109]. An initial position is a random number. Alice and Bob make alternating moves: Alice, Bob, Alice, Bob, etc. Available moves are same for both: if a current position is n∈[1..99] then (n+1) and (n+10) are possible next positions. A player wins the game iff the opponent is the first to enter [100..109]. Problem: Find all initial positions where Alice has a winning strategy.

Kripke model for the game is quite obvious:

- States correspond to game positions, i.e. integers in *[1..109]*.
- Propositional variable *fail* is interpreted by *[100..109]*.
- Relational variable *move* is interpreted by possible moves.

Formula $\neg fail \,\&\, \langle move \rangle(\neg fail \,\&\, [move]fail)$ is valid in those states where the game is not lost, there exists a move after which the game is not lost, and then all possible moves always lead to a loss in the game. Hence this EPDL formula is valid in those states where Alice has a 1-round winning strategy against Bob.

COMBINING KNOWLEDGE, ACTIONS AND TIME

Logic of Knowledge

Logics of knowledge are also known as **epistemic logics**. One of the simplest epistemic logic is **Propositional Logic of Knowledge for $n>0$ agents (PLK_n)** (Fagin, Halpern, Moses & Vardi, 1995). A special terminology, notation and **Kripke models** are used in this framework. A set of relational symbols *Rel* in PLK_n consists of natural numbers *[1..n]* representing names of agents. Notation for modalities is: if $i \in [1..n]$ and φ is a formula, then $(Ki\ \varphi)$ and $(Si\ \varphi)$ are used instead of $([i]\ \varphi)$ and $(\langle i \rangle\ \varphi)$. These formulas are read as "(an agent) *i* knows φ" and "(an agent) *i* can suppose φ". For every agent $i \in [1..n]$ in every model $M = (D, I)$, interpretation *I(i)* is an "**indistinguishability relation**", i.e. an equivalence relation[2] between states that the agent *i* can not distinguish. Every model *M*, where all agents are interpreted in this way, is denoted as $(D, \sim_1, \dots \sim_n, I)$ with explicit $I(1) = \sim_1, \dots I(n) = \sim_n$ instead of brief standard notation *(D,I)*. An agent knows some "fact" φ in a state *s* of a model *M*, if the fact is valid in every state *s′* of this model that the agent can not distinguish from *s*:

- $s \models_M (K_i\ \varphi)$ *iff* $s' \models_M \varphi$ for every state $s' \sim_i s$.

Similarly, an agent can suppose a "fact" φ in a state *s* of a model *M*, if the fact is valid in some state *s′* of this model that the agent can not distinguish from *s*:

- $s \models_M (S_i\ \varphi)$ *iff* $s' \models_M \varphi$ for some state $s' \sim_i s$.

The above possible worlds semantics of knowledge is due to pioneering research (Hintikka, 1962).

Temporal Logic with Actions

Another propositional polymodal logic is **Computational Tree Logic with actions (*Act-CTL*)**. *Act-CTL* is a variant of a basic propositional branching time temporal logic Computational Tree Logic (CTL) (Emerson, 1990; Clarke, Grumberg & Peled, 1999). In *Act-CTL* the set of relational symbols consists of action symbols *Act*. Each action symbol can be interpreted by an "instant action" that is executable in one undividable moment of time.

Act-CTL notation for basic modalities is: if $b \in Act$ and φ is a formula, then , $(A_bX\ \varphi)$ and $(E_bX\ \varphi)$ are used instead of $([b]\ \varphi)$ and $(\langle b\rangle\ \varphi)$. But syntax of *Act-CTL* has also some other special constructs associated with action symbols: if $b \in Act$ and φ and ψ are formulas, then $(A_bG\ \varphi)$, $(A_bF\ \varphi)$, $(E_bG\ \varphi)$, $(E_bF\ \varphi)$, $A_b(\varphi\ U\ \psi)$ and $E_b(\varphi\ U\ \psi)$ are also formulas of *Act-CTL*. In formulas of *Act-CTL* prefix "*A*" is read as "for every future", "*E*" – "for some future", suffix "*X*" – "next state", "*G*" – "always" or "globally", "*F*" – "sometimes" or "future", the infix "*U*" – "until", and a sub-index "*b*" is read as "in *b*-run(s)".

We have already explained semantics of $(A_bX\varphi)$ and $(E_bX\varphi)$ by referencing to $([b]\ \varphi)$ and $(\langle b\rangle\ \varphi)$. Constructs "$A_bG$", "$A_bF$", "$E_bG$", and "$E_bF$" can be expressed in terms of "$A_b(\ldots U\ldots)$" and "$E_b(\ldots U\ldots)$" , for example: $(E_bF\varphi) \leftrightarrow E_b(true\ U\ \varphi)$. Thus let us define below semantics of "$A_b(\ldots U\ldots)$" and "$E_b(\ldots U\ldots)$" only. Let $M = (D, I)$ be a model. If $b \in Act$ is an action symbol, then a partial *b*-run is a sequence of states $s_0, \ldots s_k, s_{(k+1)}, \ldots \in D$ (maybe infinite) such that $(s_k, s_{(k+1)}) \in I(b)$ for every consecutive pair of states within this sequence. If $b \in Act$ is an action symbol, then a *b*-run is an infinite partial *b*-run or finite *b*-run that can not be continued[3]. Then semantics of constructs "$A_b(\ldots U\ldots)$" and "$E_b(\ldots U\ldots)$" can be defined as follows:

- $s \models_M A_b(\varphi\ U\ \psi)$ iff for every *b*-run $s_0, \ldots s_k, \ldots$ that starts in *s* (i.e. $s_0 = s$) there exists some $n \geq 0$ for which $s_n \models_M \psi$ and $s_k \models_M \varphi$ for every $k \in [0..(n-1)]$;
- $s \models_M E_b(\varphi\ U\ \psi)$ iff for some *b*-run $s_0, \ldots s_k, \ldots$ that starts in *s* (i.e. $s_0 = s$) there exists some $n \geq 0$ for which $s_n \models_M \psi$ and $s_k \models_M \varphi$ for every $k \in [0..(n-1)]$.

The standard branching-time temporal logic *CTL* can be treated as ***Act-CTL*** with a single implicit action symbol.

Combined Logic of Knowledge, Actions and Time

There are many **combined polymodal logics** for reasoning about multiagent systems. Maybe the most advanced is **Belief-Desire-Intention (BDI) logic** (Wooldridge, 1996; Wooldridge, 2002). An agent's beliefs correspond to information the agent has about the world. (This information may be incomplete or incorrect. An agent's knowledge in BDI is just a true belief.) An agent's desires correspond to the allocated tasks. An agent's intentions represent desires that it has committed to achieving. Admissible actions are actions of individual agents; they may be constructed from primitive actions by means of composition, nondeterministic choice, iteration, and parallel execution. But semantics of BDI and reasoning in BDI are quite complicated for a short encyclopedia article.

In contrast, let us discuss below a simple example of a combined logic of knowledge, actions and time – namely **Propositional Logic of Knowledge and Branching Time for *n>0* agents *Act-CTL-K$_n$*** (Garanina, Kalinina, & Shilov, 2004; Shilov, Garanina & Choe, 2006; Shilov & Garanina, 2006). First we provide a formal definition of *Act-CTL-K$_n$*, then discuss some pragmatics, and then – in the next section – introduce model checking as a reasoning mechanism.

Let $[1..n]$ be a set of agents ($n > 0$), and *Act* be a finite alphabet of action symbols. Syntax of *Act-CTL-K$_n$* admits epistemic modalities K_i , and S_i for every $i \in [1..n]$, and branching-time constructs $A_bX, E_bX, A_bG, E_bG, A_bF, E_bF, A_b(\ldots U\ldots)$, and $E_b(\ldots U\ldots)$ for every $b \in Act$. Semantics is defined in terms of entailment in environments. An **(epistemic) environment** is a tuple $E = (D, \sim_1, \ldots \sim_n, I)$ such that $(D, \sim_1, \ldots \sim_n)$ is a model for **PLK$_n$** , and (D, I) is a model for ***Act-CTL***. **Entailment relation** $\models$ is defined by induction according to the standard definition for propositional connectives (see semantics of ***EPDL***), and the above definitions of epistemic modalities and branching time constructs.

We are mostly interested in trace-based perfect recall synchronous environments generated from background finite environments. "Generated" means that possible "worlds" are runs of finite-state machine(s). There are several opportunities how to define semantics of combined logics on runs. In particular, there are two extreme cases: **Forgetful Asynchronous Systems (FAS)** and **Synchronous systems with Perfect Recall (PRS)**. "Perfect recall" means that every agent has a log-file with all his/her observations along a run, while "forgetful" means that information of this kind is not available. "Synchronous" means that every agent can distinguish runs of different lengths, while "asynchronous" means that some runs of different lengths may be indistinguishable.

It is quite natural that in the FAS case combined logic $Act\text{-}CTL\text{-}K_n$ can express as much as it can express in the background finite system. In contrast, in the PRS case $Act\text{-}CTL\text{-}K_n$ becomes much more expressive than in the background finite environment. Importance of combined logics in the framework of trace-based semantics with synchronous perfect recall rely upon their characteristic as logics of agent's learning or knowledge acquisition. We would like to argue this characteristic by the following single-agent[4] Fake Coin Puzzle *FCP(N,M)*.

A set consists of (N+1) enumerated coins. The last coin is a valid one. A single coin with a number in [1..N] is fake, but other coins with numbers in [1...(N+1)] are valid. All valid coins have the same weight that differs from the weight of the fake. Is it possible to identify the fake by balancing coins M times at most?

In *FCP(N,M)* the agent (i.e. a person who have to solve the puzzle) does not know neither a number of the fake, nor whether it is lighter or heavier than the valid coins. Nevertheless, this number is in *[1..N]*, and the fake coin is either lighter (*l*) or heavier (*h*). The agent can make balancing queries and read balancing results after each query. Every balancing query is an action $b_{(L,R)}$ which consists in balancing of two disjoint sets of coins: with numbers $L\subseteq[1..N+1]$ on the left pan, and with numbers $R\subseteq[1..N+1]$ on the right pan, $|L| = |R|$. There are three possible balancing results: "<", ">", and "=", which means that the left pan is lighter, heavier than or equal to the right pan, respectively. Of course, there are initial states (marked by *ini*) which represent a situation when no query has been made.

Let us summarize. The agent acts in the environment generated from a finite space $[1..N]\times\{l,h\}\times\{<, >, =, ini\}$. His/her admissible actions are balancing query $b_{(L,R)}$ for disjoint $L, R\subseteq [1..N+1]$ with $|L| = |R|$. The only information available for the agent (i.e., which gives him/her an opportunity to distinguish states) is a balancing result. The agent should learn *fake_coin_number* from a sequence which may start from any initial state and then consists of *M* queries and corresponding results. Hence single agent logic $Act\text{-}CTL\text{-}K_1$ seems to be a very natural framework for expressing *FCP(N,M)* as follows: to validate or refute whether

$$s \models_E (E_B X \ldots_{M-times} \ldots E_B X (\vee_{f\in[1..N]} K_1(fake\_coin\_number = f))\ldots))$$

for every initial state *s*, where *E* is a PRS environment generated from a finite space $[1..N]\times\{l,h\}\times\{<, >, =, ini\}$, and *B* is a balancing query $\cup_{L,R\subseteq[1..N+1]} b_{(L,R)}$.

FUTURE TRENDS: MODEL CHECKING FOR COMBINED LOGICS

The **model checking problem** for a combined logic ($Act\text{-}CTL\text{-}K_n$ in particular) and a class of **epistemic environments** (ex., PRS or FAS environments) is to validate or refute $s \models_E \varphi$, where *E* is a finitely-generated environment in the class, *s* is an "initial state" of the environment *E*, and φ is a formula of the logic. The above re-formulation of *FCP(N,M)* is a particular example of a **model checking problem** for a formula of $Act\text{-}CTL\text{-}K_n$ and some finitely-generated perfect recall environment.

Papers (Meyden & Shilov, 1999) and (Garanina, Kalinina & Shilov, 2004) have demonstrated that if the number of agents $n>1$, then the model checking problem in perfect recall synchronous systems is very hard or even undecidable. In particular, it has non-elementary[5] upper and lower time bounds for $Act\text{-}CTL\text{-}K_n$. Papers (Meyden & Shilov, 1999) and (Shilov, Garanina & Choe, 2006) have suggested a tree-like data structures to make "feasible" model checking of combinations of temporal and action logics with propositional logic of knowledge PLK_n. Alternatively, (van der Hoek & Wooldridge, 2002; Lomuscio & Penczek, 2003) have susggested either to simplify language of logics to be combined, or to consider agents with "bounded" recall.

M

CONCLUSION

Combinations of temporal logics and logics of actions with logics of knowledge become an actual research topic due to the importance of study of interactions between knowledge and actions for reasoning about real-time multiagent systems. A comprehensive survey of logics, techniques, and results was out of scope of the article. The primary target of present article was to provide semi-formal introduction to the field of combined modal logics, discuss their utility for reasoning about multiagent systems. The emphasis has been done on model checking of trace-based knowledge-temporal specifications of perfect recall synchronous systems.

REFERENCES

Clarke, E., Grumberg, O., & Peled, D. (1999). *Model Checking.* MIT Press.

Bull, R. & Segerberg, K. (2001) Basic Modal Logic. *Handbook of Philosophical Logic*, v.3. D. Gabbay and F. Cuenthner editors. Kluwer Academic Publishers.

Dixon, C., Nalon, C., & Fisher, M. (2004). Tableau for Logics of Time and Knowledge with Interactions Relating to Synchrony. *Journal of Applied Non-Classical Logics*, 14(4), 397-445.

Emerson, E.A. (1990). Temporal and Modal Logic. *Handbook of Theoretical Computer Science (B)*, J. van Leeuwen, A.R. Meyer, M. Nivat, M. Paterson, D. Perrin editors. Elsevier and The MIT Press.

Fagin, R., Halpern, J.Y., Moses, Y., & Vardi, M.Y. (1995). *Reasoning about Knowledge*. MIT Press.

Garanina, N.O., Kalinina, N.A., & Shilov N.V. (2004) Model checking knowledge, actions and fixpoints. *Proceedings of Concurrency, Specification and Programming Workshop CS&P'2004*. Humboldt Universitat, Berlin, Informatik-Bericht, 170, 351-357.

Halpern, J.Y., & Vardi, M.Y. (1986). The Complexity of Reasoning About Knowledge and Time. *Proceedings of the eighteenth annual ACM symposium on Theory of computing*. 304-315.

Halpern, J. Y., van der Meyden, R., & Vardi, M.Y. (2004). Complete Axiomatizations for Reasoning about Knowledge and Time. *SIAM Journal on Computing*, 33(3), 674-703.

Harel, D., Kozen, D., & Tiuryn, J. (2000). *Dynamic Logic.* MIT Press.

Hintikka, J. (1962). *Knowledge and Belief.* Cornell University Press.

van der Hoek, W., & Wooldridge, M.J. (2002). Model Checking Knowledge and Time. *Lecture Notes in Computer Science*, 2318, 95-111.

Lomuscio, A., & Penczek, W. (2003). Verifying Epistemic Properties of Multi-agent Systems via Bounded Model Checking. *Fundamenta Informaticae*, 55(2), 167-185.

van der Meyden, R., & Shilov, N.V. (1999). Model Checking Knowledge and Time in Systems with Perfect Recall. *Lecture Notes in Computer Science*, 1738, 432-445.

Rescher, N. (2005). *Epistemic Logic. A Survey of the Logic of Knowledge.* University of Pitsburgh Press.

Shilov, N.V., Garanina, N.O., & Choe, K.-M. (2006). Update and Abstraction in Model Checking of Knowledge and Branching Time. *Fundameta Informaticae*, 72(1-3), 347-361.

Wooldridge, M. (1996) Practical reasoning with procedural knowledge: A logic of BDI agents with know-how. *Lecture Notes in Artificial Intelligence*, (1085), 663–678.

Wooldridge, M. (2002). *An Introduction to MultiAgent Systems*. John Wiley & Sons Ltd.

KEY TERMS

Environment: A labeled transition system that provides an interpretation for logic of knowledge, actions and time simultaneously.

Labeled Transition Systems or Kripke Model: An oriented labeled graph (infinite maybe). Nodes of the graph are called states or worlds, some of them are marked by propositional symbols that are interpreted to be valid in these nodes. Edges of the graph are marked by relational symbols that are interpreted by these edges.

Logic of Actions: A polymodal logic that associate modalities like "always" and "sometimes" with action symbols that are to be interpreted in labeled transition systems by transitions. A so-called Elementary Propositional Dynamic Logic (EPDL) is sample logic of actions.

Logic of Knowledge or Epistemic Logic: A polymodal logic that associate modalities like "know" and "suppose" with enumerated agents or groups of agents. Agents are to be interpreted in labeled transition systems by equivalence "indistinguishability" relations. A so-called Propositional Logic of Knowledge of n agents (PLK_n) is sample epistemic logic.

Logic of Time or Temporal Logic: A polymodal logic with a number of modalities that correspond to "next time", "always", "sometimes", and "until" to be interpreted in labeled transition systems over discrete partial orders. For example, Linear Temporal Logic (LTL) is interpreted over linear orders.

Model Checking Problem: An algorithmic problem to validate or refute a property (presented by a formula) in a state of a model (from a class of Kripke structures). For example, model checking problem for combined logic of knowledge, actions and time in initial states of perfect recall finitely generated environments.

Multiagent System: A collection of communicating and collaborating agents, where every agent have some knowledge, intensions, enabilities, and possible actions.

Perfect Recall Synchronous Environment: An environment for modeling a behavior of a perfect recall synchronous system.

Perfect Recall Synchronous System: A multiagent system where every agent always records his/her observation at all moments of time while system runs.

ENDNOTES

1. Due to pioneering papers of Saul Aaron Kripke (born in 1940) on models for modal logics.
2. A symmetric, reflexive, and transitive binary relation on *D*.
3. That is for the last state *s* there is no state *s'* such that $(s,s')\in I(b)$.
4. For multiagent example refer Muddy Children Puzzle (Fagin, Halpern, Moses & Vardi, 1995).
5. I.e. it is not bounded by a tower of exponents with any fixed height