# A Systematic View of Agent-supported Simulation
## *Past, Present, and Promising Future*

Tuncer Ören[1], Levent Yilmaz[2] and Nasser Ghasem-Aghaee[3,4]
[1]*School of Electrical Eng. and Computer Science, Univ. of Ottawa, Ottawa, ON, Canada,*
[2]*Computer Science and Software Engineering, Samuel Ginn College of Engineering, Auburn University, Auburn, AL, U.S.A.*
[3]*Computer Eng. Dept. of Sheikh Bahaei Univ., Baharestan, Iran*
[4]*Univ. of Isfahan, Isfahan, iran*

Keywords: Agent-directed Simulation, Simulation for Agents, Agent Simulation, Software Agents for Simulation, Agent-supported Simulation, Agent-based Simulation, Agent-monitored Simulation, Agent-triggered Simulation.

Abstract: Agent-supported simulation involves the use of intelligent agents to enhance modeling and simulation (M&S) infrastructures and consists of support by software agents: (1) to provide computer assistance for front-end and/or backend interface functions in M&S environments; (2) to process elements of an M&S study symbolically (for example, for consistency checks and built-in reliability); and (3) to provide cognitive abilities to the elements of an M&S study, such as perception, anticipation, learning or understanding abilities. Several aspects of agent-supported simulation are clarified and references are provided.

# 1 INTRODUCTION

Agent-supported simulation is a special case of agent-directed simulation and involves the use of intelligent agents to enhance modeling and simulation (M&S) infrastructures. Agent-supported simulation involves use of software agents: (1) to provide computer assistance for front-end and/or backend interface functions in M&S environments; (2) to process elements of M&S studies symbol-lically (for example, for consistency checks and built-in reliability); and (3) to provide cognitive abilities to the elements of M&S studies, such as learning or understanding abilities.

Section 2 is a very brief overview of simulation and software agents. In section 3, we elaborate on synergies of simulation and agents to provide an appropriate perspective to conceive properly agent-supported simulation. In section 4, we focus and elaborate on several aspects of agent-suppor-ted simulation. Section 5 is a review of past and present realizations of agent-supported simulation as well as promising development areas. Section 6 includes conclusions and some future activities. Due to space limitations, only main aspects and references are given.

# 2 BACKGROUND

## 2.1 Simulation

Two aspects of simulation, i.e., experimentation and experience need to be emphasised for the scope of this article. So far as its *experimentation* aspect is concerned, simulation is performing goal-directed experiments with models of dynamic systems. So far as its *experience* aspect is concerned, (1) simulation is providing experience under controlled conditions for training, i.e., for gaining/enhancing competence in one of the three types of skills: (i) motor skills (virtual simulation), (ii) decision and/or communication skills (constructive simulation; serious game), and (iii) operational skills (live simulation) or (2) simulation is providing experience for entertainment purpose (gaming simulation). For further details, see Ören (2011a, b).

## 2.2 Software Agents

Software agents are autonomous software modules with perception and social ability to perform goal-directed knowledge processing over time, on behalf of humans or other agents in software and physical

Ören T., Yilmaz L. and Ghasem-Aghaee N..
A Systematic View of Agent-supported Simulation - Past, Present, and Promising Future.
DOI: 10.5220/0005138804970506
In *Proceedings of the 4th International Conference on Simulation and Modeling Methodologies, Technologies and Applications* (SIMULTECH-2014), pages 497-506
ISBN: 978-989-758-038-3

environments. When agents operate in physical environments, they can be used in the implementation of intelligent machines and intelligent systems and can interact with their environment by sensors and effectors. The core knowledge processing abilities of agents include: goal-directed knowledge processing, reasoning, planning, motivation, and decision making. The factors that can affect decision making abilities, such as personality, emotions, and cultural backgrounds can also be embedded in agents. Agents may have additional abilities such as understanding, including understanding and expressing emotions, awareness, as well as ethical behavior.

# 3 SYNERGIES OF SIMULATION AND AGENTS

## 3.1 Agent-directed Simulation

Agent-directed simulation refers to the synergy of software agents and simulation. As shown in Figure 1, there are three categories of possibilities that can be considered under two groups: (1) contribution of simulation to agents: which consists of agent simulation and (2) contribution of agents to simulation which consists of agent-supported simulation and agent-based simulation (Ören, 2001a; Yilmaz and Ören, 2009):

- *Agent simulation* is simulation of agent systems or simulation of systems modeled by using software agents.
- *Agent-supported simulation* is use of agents for at least one of the following purposes:
  (1) to provide agent assistance for front-end interface functions in a computer-aided modeling and simulation study;
  (2) to provide agent assistance for back-end interface functions in a computer-aided simulation study;
  (3) for symbolic processing of elements of a simulation study –for consistency checks, for example; and
  (4) to provide cognitive abilities to the elements of a simulation study –such as learning, understanding and/or hypothesis formulation.
- *Agent-based simulation* is use of agents for the generation and/or monitoring of agent behavior. (This is similar to the use of AI techniques –like qualitative simulation– for the generation of model behavior).

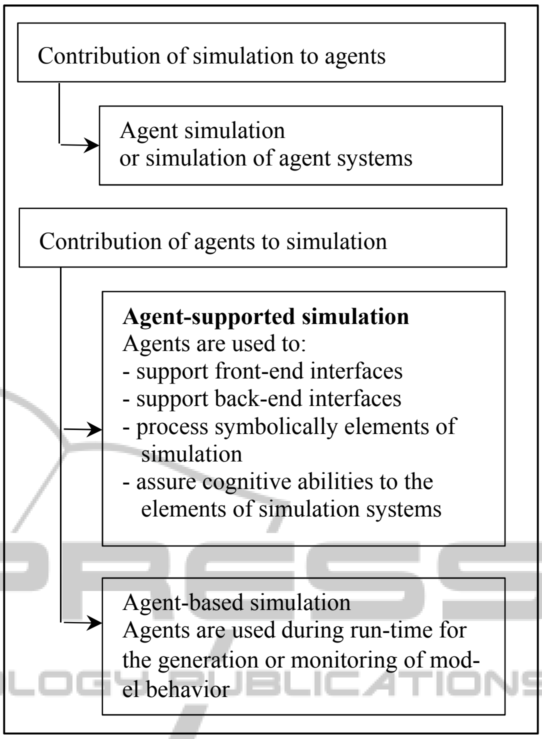


Figure 1: Three categories of Agent-directed Simulation.

## 3.2 Agent Simulation

Agents provide a natural paradigm to represent intelligent entities. *Agent simulation* is simulation of natural or engineered entities represented by agents (Yilmaz and Ören, 2009).

## 3.3 Agent-based Simulation

Agent-based simulation is the use of software agents during run time to monitor and generate model-behavior. This is similar to the use of AI techniques for the generation of model behavior, e.g., qualitative simulation and knowledge-based simulation. The possibilities include agent-triggered simulation, agent-monitored simulation; agent monitored couplings, agent-monitored multi-model transitions… The term agent-based simulation is also used to mean agent simulation when the two other possibilities of contribution of agents to simulation are not taken into consideration

# 4 AGENT-SUPPORTED SIMULATION

Agent-supported simulation–which is the focus of this article– is the use of agent technology to support simulation activities in modeling and simulation environments as well as simulation-based problem solving environments (or simulative problem solving environments).

The possibilities for agent support in modeling and simulation environments are outlined in Table 1. The agent support can be for the following elements of simulation studies: goal of the study, parametric models, model parameters, design of experiments, experimental conditions, simulation program, and the behavior and recommendations. For each element, the support can be for the generation, specification/editing, as well as processing of the element. For each category of knowledge and knowledge processing knowledge (Ören 1990), an important category of knowledge processing activity is to ensure the reliability of the associated elements.

Table 1: Possibilities for agent support in modeling and simulation environments.

| For | For specification/ generation/editing | For processing |
|---|---|---|
| Goal | -goal specification / editing<br>-goal generation<br>-hypothesis formulation | -goal processing<br>--goal seeking<br>--modification<br>--evaluation<br>--selection |
| Parametric model | -modelling<br>--model composition<br>--model editing | -model-base management<br>-model analysis<br>--characterization<br>--evaluation<br>-model transformation |
| Model parameters | -parameter estimation/ calibration<br>-editing<br>--parameters<br>--auxiliary parameters | -symbolic processing<br>--parameters<br>--auxiliary parameters |
| Design of experi-ments | - design/editing of experiments | -processing of design of experiments<br>--evaluation<br>--selection |
| For every experiment | -specification/editing of experimental conditions<br>--initial conditions of state variables<br>--behavior generator<br>--behavior generation parameters | -automatic selection<br>--behavior generator<br>--behavior generation parameters<br>-reliability |
| Simulation program | -transformation of problem specifications into a simulation program<br>-automated editing of simulation programs | - processing sim. programs (legacy programs, new programs)<br>-program understanding<br>-program reliability |

## 4.1 Agent Support for Front-end Interfaces

Table 2 outlines the front-end functionalities for the elements of a modeling and simulation environment. Front-end interface functionalities include: anticipation of user's needs, help, just-in-time-learning, explanation, awareness, assistance, guidance, (un)solicited advice, advanced types of inputs such as perception, speech input, body language, facial expression, deictic input, and haptic input. Front-end interface functionalities are applicable to goals, parametric models, model parameters, experimentation conditions, simulation pro-grams, and model behaviors.

Table 2: Some front-end interface functionalities.

- anticipation of user's needs
- help formulate/specify problems
- awareness, just-in-time-learning, explanation
- assistance, guidance, (un)solicited advice
- abilities to process advanced types of inputs:
  -- perception (focusing), speech input
  -- body language, deictic input, haptic input

## 4.2 Agent Support for Back-end Interfaces

Back-end interfaces are used by systems to communicate to the users the primary and auxiliary outputs of the system. Table3 outlines the back-end functionalities for the elements of modeling and simulation environments. Back-end interface functionalities provide support for behavior display, instrumenting/monitoring, processing, evaluation, and advice. Advanced types of outputs such as augmented/enhanced reality and virtual reality are part of the

Table 3: Some back-end interface functionalities.

-*primary outputs*
  --(un)processed behavior
  --performance measure
  --evaluation
  --advice on the problem
-*auxiliary outputs*
  --automated documentation
  --explanation

with abilities to process advanced types of outputs such as:
--virtual reality,
--augmented reality
--holographic displays

functionalities of back-end interfaces. Back-end interface functionalities are applicable to behavior displays, instrumenting, pro-cessing, evaluating, explanation, and warning/advice.

### 4.3 Agent Support for Symbolic Processing of Elements of Simulation Studies

Intelligent agents can provide support in various stages of the overall simulation development lifecycle. For instance, in Model-Driven Engineering (MDE) that involves automated transformation of platform-independent abstract models, agents can serve as transformation engines, by which increasingly concrete and platform-dependent models and simulations can be generated. Agents can also function as mediators and brokers for distinct simulations by bridging the syntactic and semantic gap between their representations. To support goal-directed experimentation, agents can bring transparency to the overall experiment design (Ören 2001b), execution, analysis, and adaptation process for various types of experiments such as sensitivity analysis, variable screening, understanding, optimization, and decision-support. Next, for illustrative purposes, we discuss these three application areas.

#### 4.3.1 Agent-supported Model Transformation

The common strategy in MDE is based on the application of a sequence of transformations starting with platform-independent models down to the concrete realization of the simulation system. Besides the reuse of models and deployment of designs in alternative platforms, MDE improves the reliability of simulation systems through correctness preserving transformations that allow substantiating the accuracy of realizations with respect to explicit constraints and assumptions defined in the abstract models. To facilitate the application of the MDE methodology shown in Figure 2, models are defined in terms of an explicit modeling language, which in turn is specified in terms of a meta-modeling language. The transformations are executed by agents using a set of rules, which are specified by using the constructs of a specific transformation language

An agent with understanding capabilities as presented in (Ören et al., 2007) can be used to map constructs of a source meta-model to equivalent features of the target meta-model. Such templates can be customized and applied by agents upon models by matching rules to constructs and elements of the

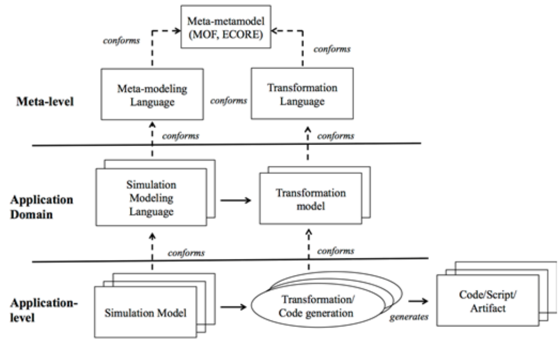

Figure 2: Model-Driven Engineering.

models. The strategies for defining such production rules can vary depending on the sophistication of the MDE environment. Transformation rules can be produced by agents with understanding capabilities from scratch or can be a refinement of a generic specification template applicable to selected source and target modeling languages. Alternatively, transformation rules can be derived automatically out of higher-level mappings rules between models. This strategy requires (1) defining/discerning a mapping of elements of one model to another model (e.g., model weaving) and (2) automating the generation of the actual transformation through an agent interpreter or matcher that takes as input two model definitions and the mapping rules between them to produce the concrete transformations.

#### 4.3.2 Agent-supported Interoperability

The above mechanism can be extended (see Figure 3) to utilize agents to support interoperability. In distributed simulation, interoperability refers to the ability of system models or components to exchange and interpret information in a consistent and meaningful manner. This requires both syntactic and semantic congruence between systems either through standardization or mediators that can bridge the syntactic and semantic gap between peer components.

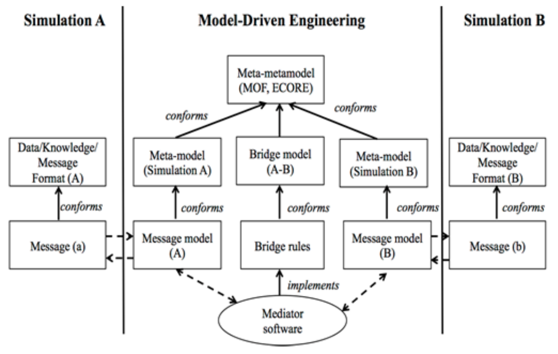

Figure 3: Agent-supported Simulation Interoperability.

Such mediator or bridge agents ensure both data-level interoperability (i.e., metadata/data interchange) and operational-level interoperability (i.e., behavioral specification). Common strategies for developing adapters involve the provision of standard APIs and connecting components through published interfaces that explicitly specify the required and provided services. This low-level connectors are often limited and do not achieve full data interoperability.

As shown in Figure 3, the use of agent-supported transformation can provide a sound and comprehensive framework for defining bridges. By making the internal schema (i.e., meta-model) of each system explicit and then aligning them via agents by matching or weaving concepts, we can leverage model-to-model transformations exploiting the matching information to export data conforming to the meta-model of the target system or component. While internal schema, structural specifications, and behavioral models may be available along with the implementation of the simulation system, in their absence agents can also be supportive in deducing such models for transformation. By extracting the abstract syntax of the Platform-Specific Model of a system and then transforming it into a set of PSMs using agent transformation rules would be a first step to automate the derivation of high-level specifications. Such specifications could then be used to generate bridge rules in terms of model transformation language, which serves as the meta-model for the translation rules that map the source data/behavioral specification to the target specification. Such mapping rules can be used as bridge implementations in terms of mediator software agents.

### 4.3.3 Agent-supported Experimentation

An agent-coordinated support system could greatly enhance the experimental design process in several ways, but mainly by providing expert knowledge that the user might lack (Ören, 2001). The agent can decide which designs best fit the experiment's objective, as well as which factors should be kept or discarded after each iteration of the experiment's life-cycle. Concomitantly, the agent guides the process by requesting the information it needs in order to construct an experiment, verify the validity of the user's input and ensure the integrity of the experiment elements.

As shown in Figure 4, the agent-supported Simulation Experiment Management System (SEMS) is a software tool that allows users to design, execute, store, and share computer simulation experiments. An ontology-assisted interface builder managed by an interface agent that is aware of experiment ontology guides the simulation experiment design. The experiment design procedure is governed by a KEPLER Scientific Workflow process (Ludäscher et al., 2006) that implements the experiment life-cycle.

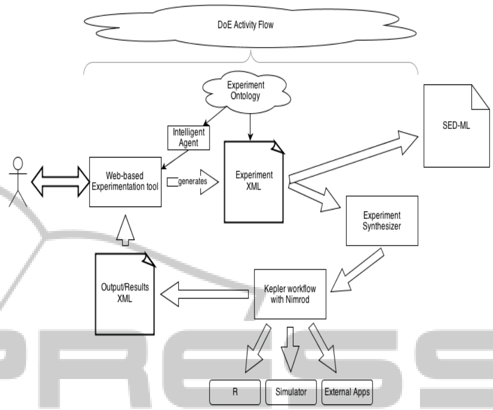


Figure 4: Simulation Experiment Management System Components (Data-Flow View).

At each step of the life-cycle, the user inputs, with the help of an intelligent agent, the information required to construct an experiment. This experiment structure is stored in an XML file that is used by a synthesizer agent, which "translates" the XML source into executable NIMROD (Abramson et al., 1995) (experiment execution engine) code. Following the execution of the experiment, the synthesizer agent collects the results and forwards them to statistical software to generate an XML representation of the experimental results.

## 4.4 Agent Support to Provide Cognitive Abilities to the Elements of a Simulation

Software agents can provide cognitive abilities such as perception (Ören, 2001), anticipation (Ören and Yilmaz, 2012), and understanding (Ören, Ghasem-Aghaee and Yilmaz, 2007) to the elements of simulation studies. Table 11 includes some additional possibilities.

# 5 AGENT-SUPPORTED SIMULATION: PAST, PRESENT, AND FUTURE

## 5.1 Past and Present

Ándras Jávor did pioneering work by using “demons” in modeling and simulation. In those early days, even the term “software agent” was not yet introduced in the scientific literature. (Jávor 1990, 1992; Jávor and Szűcs, 1998). Another early contributor to the field using demons was Hogeweg (1979).

Tables 4 - 7 contain, respectively, samples of references for agent-supported simulation front-end interfaces, back-end interfaces, symbolic processing of elements of simulation studies, and cognitive abilities for the elements of simulation systems.

Table 4: Agent-supported simulation for front-end interfaces.

| To support | Author(s) | Year |
|---|---|---|
| -airline ticket assistance | Groves & Gini | 2013 |
| -animated interface agent | Rist et al. | 1997 |
| -behaviour-based control | Alexander et al. | 2010 |
| -collaborative interface | Eisenstein & Rich | 2002 |
| | Rich & Sidner | 1997 |
| -experimental design | Ören | 2001b |
| -intelligent interface | Pitts & Ping Hwang | 1999 |
| | Bikovska et. al. | 2006 |
| | Tuchinda & Knoblock | 2004 |
| -natural language interface | Moran et al. | 1997 |
| -sensor/emitter model | Dryer | 1997 |
| | Presser et al. | 1999 |
| -visualization environment | Wasfy et al. | 2004 |

Table 5: Agent-supported simulation for back-end interfaces.

| To support | Author(s) | Year |
|---|---|---|
| -end-user individual-based modeling | Ginot et al. | 2002 |
| -explanation | Haynes et.al. | 2009 |
| | Vasconcelos et al. | 2004 |
| -natural language advice | Kuhlmann et al. | 2004 |
| -route advice | Rogers et al. | 1999 |

Table 6: Agent-supported simulation for symbolic processing.

| To support | Author(s) | Year |
|---|---|---|
| -distributed symbolic computation | Schimkat et al. | 2000 |
| -emergence of inquiry conversation | Omori & Nishizaki | 1999 |
| -simulation-based systems engineering | Yilmaz & Ören | 2010 |
| -symbolic and behavioral processing of data | Chella et al. | 1997 |
| -symbolic performance and learning in continuous valued environment | Rogers | 1997 |
| -test and refine models | Kennedy & Theodoropoulos | 2006 |
| -verification and validation | Balci | 2004 |

Table 7: Agent-supported simulation for cognitive abilities for the elements of simulation systems.

| To support | Author(s) | Year |
|---|---|---|
| -adaptive elements | Crain | 1999 |
| -adaptive mesh generation | Hilaire et al. | 2000 |
| -agent decision making | Brouwers & Verhagen | 2003 |
| -agent intelligence to support human org. | Knoblock et al. | 2008 |
| -assessment model | Krywkow et al. | 2002 |
| -automated evaluation of Internet business | Chong & Cho | 2001 |
| -cognitive emergence | Castelfranchi | 1998 |
| -cooperation tools for supply chain management | Klingebiel et al. | 2001 |
| -decision assistant | Itmi et al. | 2002 |
| -decision support | Yilmaz & Tolk | 2008 |
| -design of experiments | Ören | 2001b |
| -dynamic reasoning | Kazar et al. | 2000 |
| -help/documentation | Fujishima | 1997 |
| -HLA-based distributed virtual environments | Wang et al. | 2003 |
| -information warfare | Mack & Alzone | 1997 |
| -integration of databases using mobile code | Claro & Sobral | 2000 |
| -intelligent matchmaking for information agents | Lu & Sterling | 2000 |
| -interoperation | Yilmaz & Paspuleti | 2005 |
| -mediation | Novais et al. | 2000 |
| -multi-sensor planning | Hodge & Kamel | 2001 |
| -office automation | Thomas & Fischer | 1997 |
| -processes controlled by agents | Kruzel & Vondrak | 2000 |
| -resource location on the World Wide Web | Grey et al. | 2000 |
| -scheduling | Pesenti et al. | 2001 |
| -selection recognition | Pandit & Kalbag | 1997 |
| -social models | Moss | 1998 |
| -traffic intersection control | Dresner & Stone | 2005 |
| -understanding the design requirements | Cohen et al. | 1989 |

## 5.2 Some Promising Research and Development Areas

Some promising research and development (R&D) areas to fully benefit from the synergy of simulation and agents are outlined in tables T8-11.

Table 8: Promising R&D areas: Front-end interfaces.

| |
|---|
| * Intelligent interface agents<br>--context and situation awareness<br>-- anticipation of user's needs<br>---help formulate/specify problems<br>-- just-in-time-learning, explanation<br>--assistance, guidance, (un)solicited advice<br>* Abilities to process advanced types of inputs<br>--perception (focusing)<br>--natural language input<br>--body language interface<br>--emotional inputs<br>--deictic input, haptic input<br>--thought input<br>* Holographic avatars in front-ends |

Table 9: Promising R&D areas: Back-end interfaces.

| |
|---|
| * Holographic avatars in back-ends<br>* Help to select solutions<br>* Clarification of solution<br>* Spoken output |

Table 10: Promising R&D areas: Symbolic processing of elements of simulation.

| |
|---|
| * Reliability of simulation studies and agents<br>--agent-based built-in reliability<br>--agent-based verification and validation<br>--agent-based failure avoidance<br>* Program generation from specifications<br>* Program integration |

Table 11: Promising R&D areas: Cognitive abilities for the elements of simulation systems.

| |
|---|
| * Cognitive abilities to the elements of simulation, such as perception, anticipation, understanding, learning, and/or hypothesis formulation<br>* Program understanding for documentation and/or maintenance purposes<br>* Agents in simulation-based problem solving environments<br>* Holons for goal-directed co-operation and collaboration (including "principled holons" who can refuse certain types of cooperation)<br>* Simulation-based predictive displays for social and financial systems:<br>--to train future policy/decision makers<br>--to predict abnormal deviations and<br>--to test and select possible corrective actions<br>* Simulation to test and evaluate autonomous decisions by agents<br>* Agent-based ubiquitous (mobile) simulation (including agent-based mobile cloud simulation)<br>--selection of models<br>--selection of matching scenarios for experimentation |

## 6 CONCLUSION

Software agents represent powerful computational as well as modeling paradigms for autonomous entities. In this article, we focused on agent supported simulation and discussed the important benefits they bring to modeling and simulation. We also elaborated on the past contributions, the state-of-the-art and promising and important research and development areas. We are planning to explore, in a sequel paper, advanced possibilities of contributions of agents during run time, i.e., of agent-based simulation (e.g., agent-triggered simulation, agent-monitored simulation, agent-monitored dynamic coupling).

## REFERENCES

Abramson, David, Rok Sosic, Jonathan Giddy, and B. Hall. "Nimrod: a tool for performing parametrised simulations using distributed workstations." In *High Performance Distributed Computing, 1995., Proceedings of the Fourth IEEE International Symposium on*, pp. 112-121. IEEE, 1995.

Alexander, G., Heckel, F., W., P., Youngblood, G., M., Hale, D., H., Ketkar, N., S. (2010). Rapid Development of Intelligent Agents in First/third-person Training Simulation via Behavior-based Control, Proc. of the 19th Conference on Behavior Representation in Modeling and Simulation, Charleston, SC, 21-24 March 2010.

Balci, O. (2004). Quality Assessment, Verification, and Validation of Modeling and Simulation Applications, Proceedings of the 2004 Winter Simulation Conference, R.G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters, eds.

Bikovska, J., Merkuryeva, G., Grubbstrom, R. (2006). Enhancing Intelligence of Business Simulation Games, Proc. of 20th European Conference on Modelling and Simulation, Wolfang Burutzky, Alessandra Orsoni, ECMS 2006.

Brouwers, L., Verhagen, H. (2003). Applying the consumat model to flood management policies, In: Proc.

4th Workshop on Agent-Based Simulation, Müller, J. P., Seidel, M., M. (eds.), Montpellier, France, April 28-30, 2003, pp. 29-33.
Castelfranchi, C. (1998). Simulating with cognitive agents: the importance of Cognitive Emergence, A workshop forming part of Agent's World - Multi-agent Systems and Agent-Based Simulation (MABS), Paris, France, July 4-6, 1998, pp. 26-44,
Chella, A., Gaglio, S., Sajeva, G., Torterolo, F. (1997). An architecture for autonomous agents integrating symbolic and behavioral processing, Second Euromicro Workshop on Advanced Mobile Robots (EUROBOT'97).
Chong,Y.G., Cho, S.B. (2001). Web structure analysis agents for automated evaluation of Internet business, In: JSAI 2001 International Workshop on Agent-based Approaches in Economic and Social Complex Systems (AESCS 2001), Matsue City, Shimane, Japan, May 21-22, 2001. http://www.nda.ac.jp/cs/aescs2001/program.html.
Claro, D.B., Sobral, J.B.M. (2000). Integration of databases using the mobile code. In: Proc. Workshop on Agent-Based Simulation, Passau, Germany, May 2-3, 2000.
Cohen, M.D. (2001). The interaction topology of simulated agents: new research lines and their problems of accumulation, In: JSAI 2001 International Workshop on Agent-based Approaches in Economic and Social Complex Systems (AESCS 2001), Matsue City, Shimane, Japan, May 21-22, 2001, http://www.nda.ac.jp/cs/aescs2001/program.html.
Crain, C.R. (1999). The design of experiments in discrete-event models containing agent-based adaptive elements, In: Proc. of the Industrial & Business Simulation Symposium, Ades, M. (ed.), 1999 Advanced Simulation Technologies Conference, San Diego, California, April 11-15, pp. 144-148.
Dresner, K., Stone, P. (2005). Multiagent Traffic Management: An Improved Intersection Control Mechanism, AAMAS'05, July 25029, 2005, Utrecht, Netherlands.
Dryer, D. C. (1997). Wizards, guides, and beyond: Rational and empirical methods for selecting optimal intelligent user interface agents, IUI'97 Proceedings of the 1997 International Conference on Intelligent User Interfaces, Jan. 6-9, 1997, Orlando, Florida, USA, ACM, 1997, pp. 265-268.
Eisenstein, J., Rich, C. (2002). Agents and GUIs from task models . In Proc. of 2002 ACM Conference on Intelligent User Interfaces (IUI 2002).
Fujishima, Y. (1997). An interface agent for nonroutine tasks, IUI'97 Proceedings of the 1997 International Conference on Intelligent User Interfaces, Jan. 6-9, 1997, Orlando, Florida, USA, ACM, 1997, pp. 231-216.
Ginot, V., Le Page, C., Souissi S. (2002). A multi-agents architecture to enhance end-user individual-based modeling. Ecological Modeling 157: 23-41.
Grey. D.J., Dunne, P., Ian Ferguson, R. (2000). Agent seek: a means of efficiently locating resources on the World Wide Web using mobile, collaborative agents, In: Proc. Workshop on Agent-Based Simulation, Passau, Germany, May 2-3, 2000.
Groves, W., Gini, M. (2013). An Agent for Optimizing Airline Ticket Purchasing (Extended Abstract), In Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013), Ito, Jonker, Gini, and Shehory (eds), May, 6-10, 2013, Saint Paul, Minnesota, USA.
Haynes, S.R., Cohen, M.A. Ritter, F.E. (2009). A Designs for explaining intelligent agents, International Journal of Human-Computer Studies, Vol. 67, Issue 1, Jan. 2009 pp. 90-110.
Hilaire, V., Lissajoux, T., Koukam, A., Creput, J.C. (2000). A multi-agent approach to adaptive mesh generation, In: Proc. Workshop on Agent-Based Simulation, Passau, Germany, May 2-3, 2000.
Hodge, L., Kamel, M. (2001). A simulation environment for multi-sensor planning. In: Simulation, Vol. 76, No. 6, June 2001, pp. 371-380.
Hogeweg, P. and B. Hesper (1979), Heterarchical self-structuring simulation systems: concepts and applications in biology. In: Methodology in systems modelling and simulation. (Zeigler B.P., Elzas, M.S., Klir, G.J., Ören, T.I. (eds.) North Holland. pp. 221-2312.
Itmi, M., Elamri, F., Pecuchet, J.P. (2002). A decision making intelligent assistant: A procedure for the analysis of the results of simulation, In: Proc. of the 2002 Summer Computer Simulation Conference, July 14-18, 2002, pp. 29-33.
Jávor, A. (1990). Demons in simulation: A novel approach, systems analysis, modeling, Simulation 7, (1990), pp. 331-338.
Jávor, A. (1992). Demon controlled simulation, mathematics and computers in Simulation 34, 1992, pp. 283-296.
Jávor, A., Szűcs, G. (1998). Intelligent Demons with Hill climbing strategy for optimizing simulation models, Summer Computer Simulation Conference, Reno, Neveda, July 19-22, 1998, pp. 99-104.
Kazar, O., Zaidi, S., Frécon, L. (2000). Dynamic reasoning agent, In: Proc. Workshop on Agent-Based Simulation, Passau, Germany, May 2-3, 2000.
Kennedy, C., Theodoropoulos, G. (2006). Intelligent Management of Data Driven Simulations to Support Model Building in the Social Sciences, V. N. Alexandrov et al. (Eds.): ICCS 2006, Part III, LNCS 3993, pp. 562-569.
Klingebiel, K., Hoenen, M., Hellingrath, B. (2001). Multi-agent systems as cooperation tools for supply chain management, In: Proc. Workshop on Agent-Based Simulation II, Passau, Germany, April 2-4, 2001, pp. 42-47.
Knoblock, C., Ambite, J.L., Carman, M., Michelson, M., Szekely, P., Tuchinda, R. (2008). Beyond the Elves: Making Intelligent Agents Intelligent, AI Magazine Vol. 29 Number 2.

Kruzel, M., Vondrak, I. (2000). Processes controlled by agents, In: Proc. Workshop on Agent-Based Simulation, Passau, Germany, May 2-3, 2000.

Krywkow, J., Valkering, P., Van der Veen, A., Rotmans, J. (2002). Coupling an agent-based model with an integrated assessment model to investigate social aspects of water management, In Proc. 3rd Workshop on Agent- Based Simulation, Passau, Germany, April 7-9, 2002. pp. 79-84.

Kuhlmann, G., Stone, P., Mooney, R., Shavlik, J. (2004). Guiding a Reinforcement Learner with Natural Language Advice: Initial Results in RoboCup Soccer, Proc. of the AAAI-2004 Workshop on Supervisory Control of Learning and Adaptive Systems. pp. 30-35, San Jose, CA, July 2004.

Lu, H., Sterling, L. (2000). Intelligent matchmaking for information agent's cooperation on the World Wide Web. In: Proc. Workshop on Agent-Based Simulation, Passau, Germany, May 2-3, 2000.

Ludäscher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger, E., Jones, M., and Zhao, Y. (2006). Scientific workflow management and the Kepler System. *Concurrency and Computation: Practice and Experience*, *18*(10), 1039-1065.

Mack, G., Alzone, M. (1997). Software agents in analytical simulations, in: Proc. of the Summer Computer Simulation Conference, Obaidat, M., Illgen, J. (eds.), July 13-17, 1997, Arlington, Virginia, pp. 591-596.

Moran, D. B., Cheyer, A., Julia, L., Martin, D. L., Park, S. (1997). Multimodal user interfaces in the open agent architecture, IUI'97 Proceedings of the 1997 International Conference on Intelligent User Interfaces, Jan. 6-9, 1997, Orlando, Florida, USA, ACM, 1997, pp. 61-68.

Moss, S. (1998). Social simulation models and reality: three approaches. A workshop forming part of Agent's World - Multi-agent Systems and Agent-Based Simulation (MABS), Paris, France, July 4-6, 1998., Springer-Verlag, In Sichman, Conte and Gilbert, editors, Multi-agent systems and Agent-Based Simulation, LNAI Series, Vol. 1534, Dec. 1998, Berlin: Springer-Verlag.

Novais, P., Brito, L., Neves, J. (2000). Experience-based mediator agents as the basis of an electronic commerce system, In: Proc. Workshop on Agent-Based Simulation, Passau, Germany, May 2-3, 2000.

Omori, T., Nishizaki, M. (1999). Incremental knowledge acquisition architecture that is driven by the emergence of the inquiry conversation, Proc. of IEEE System Man & Cybernetics, Oct., 1999.

Ören, T.I. (1990). A Paradigm for Artificial Intelligence in Software Engineering. In: Advances in Artificial Intelligence in Software Engineering - Vol. 1, T.I. Ören (ed.), JAI Press, Greenwich, Connecticut, pp. 1-55.

Ören, T.I. (ed.) (2001a). Software agents and simulation (Guest Editor's Introduction). Special Issue of Simulation Journal, 76:6 (June 2001), pp. 328.

Ören, T.I. (2001b). Software agents for experimental design in advanced simulation environment, In: Ermakov, S.M., Kashtanov,Y.N., Melas, V. (eds.), Proc. of the 4th St. Petersburg Workshop on Simulation, June 18-23, 2001, pp. 89-95.

Ören, T.I. (2011a). The Many Facets of Simulation through a Collection of about 100 Definitions. SCS M&S Magazine, 2:2 (April), pp. 82-92.

Ören, T.I. (2011b). A Critical Review of Definitions and About 400 Types of Modeling and Simulation. SCS M&S Magazine, 2:3 (July), pp. 142-151.

Ören, T.I., Ghasem-Aghaee, N., and L. Yilmaz (2007). An Ontology-Based Dictionary of Understanding as a Basis for Software Agents with Understanding Abilities. Proceedings of the Spring Simulation Multiconference (SpringSim'07). Norfolk, VA, March 25-29, 2007, pp. 19-27.

Ören, T.I. and L. Yilmaz (2012). Agent-monitored anticipatory multisimulation: A systems engineering approach for threat-management training. Proceedings of EMSS'12 – 24th European Modeling and Simulation Symposium, F. Breitenecker, A. Bruzzone, E. Jimenez, F. Longo, Y. Merkuryev, B. Sokolov (Eds.), September 19-21, 2012, Vienna, Austria, pp. 277.282.

Pandit, M. S., Kalbag, S. (1997). The Selection Recognition Agent: Instant Access to Relevant Information and Operations, IUI'97 Proceedings of the 1997 International Conference on Intelligent User Interfaces, Jan. 6-9, 1997, Orlando, Florida, USA, ACM, 1997, pp. 47-52.

Pesenti, R., Castelli, L., Santin, P. (2001). Scheduling in a realistic environment using autonomous agents: a simulation study, in: Proc. Workshop on Agent-Based Simulation II, Urban, C. (ed.), Passau, Germany, April 2-4, 2001, pp. 149-154.

Pitts, G., Ping Hwang, S. (1999). An intelligent interface agent: Simulation/modeling made simple, in: the proc. of the 1999 Summer Computer Simulation Conference, Obaidat, M.S., Nisanci, A., Sadoun, B. (eds.), July 11-15, 1999, Chicago, Illinois, pp. 103-105.

Presser, C., Girad, D., Rose, J., Smith, W. (1999). A distributed agent environment system for simulating a naïve sensor/emitter model, in: Proc. of the 1999 Summer Computer Simulation Conference, Obaidat, M.S., Nisanci, A., Sadoun, B. (eds.), July 11-15, 1999, Chicago, Illinois, pp. 359-363.

Rich, C., Sidner, C. L. (1997). Segmented interaction history in a collaborative interface agent, IUI'97 Proceedings of the 1997 International Conference on Intelligent User Interfaces, Jan. 6-9, 1997, Orlando, Florida, USA, ACM, 1997, pp. 23-30.

Rist, T., Andre, E., Muller, J. (1997). Adding Animated Presentation Agents to the Interface, IUI'97 Proceedings of the 1997 International Conference on Intelligent User Interfaces, Jan. 6-9, 1997, Orlando, Florida, USA, ACM, 1997, pp. 79-86.

Rogers, S. O. (1997). Symbolic performance and learning in continuous valued environment, PhD Thesis, the University of Michigan, Dept. of Computer Science and Electrical Engineering, Jan. 1997.

Rogers, S., Flechter, C.N., Langley, P. (1999). An adaptive interactive agent for route advice. In O. Etzioni, J. P. Müller, and J. M., Bradshaw, editors, *Proceedings* *of the Third International Conference on Autonomous Agents (Agents'99)*, pages 198–205, Seattle, WA, USA, 1999. ACM Press.

Schimkat, R., Blochinger, W., Sinz, C., Friedrich, M., Küchlin, W. (2000). A Service-Based Agent Framework for Distributed Symbolic Computation,

Thomas, C. G., Fischer, G. (1997). Using Agents to Personalize the Web, IUI'97 Proceedings of the 1997 International Conference on Intelligent User Interfaces, Jan. 6-9, 1997, Orlando, Florida, USA, ACM, 1997, pp. 53-60.

Tuchinda, R., Knoblock, C.A. (2004). Agent wizard: building information agents by answering questions. In Proceedings of the 9th International Conference on Intelligent User Interfaces (IUI 2004), pp. 340-342, New York, NY, USA.

Yilmaz, L., T.I. Ören (2009). Agent-Directed Simulation (ADS). In L. Yilmaz and T.I. Ören (eds.). Agent-Directed Simulation and Systems Engineering. Wiley Series in Systems Engineering and Management, Wiley-Berlin, Germany, pp. 111-143.

Yilmaz, L., Ören, T. (2010). Intelligent Agent Technologies for Advancing Simulation-based Systems Engineering via Agent-Directed Simulation, SCS M&S Magazine, July 2010.

Yilmaz, L., Paspuletti, S. (2005). Toward a meta-level framework for agent-supported interoperation of defence simulation. Journal of Defence Modeling and Simulation, 2(3), p. 161-175.

Yilmaz L., Tolk, A. (2008). A Unifying Multimodel Taxonomy and Agent-Supported Multisimulation Strategy for Decision-Support, in Studies in Computational Intelligence (SCI) – Intelligent Decision Making: An AI-based Approach (Eds. Phillips-Wren, G., Ichalkaranje, N., and Lakhmi, J.). Vol. 97, pp. 189-222.

Vasconcelos, E., Pinheiro,V., Furtado, V. (2004). Mining Data and Providing Explanation to Improve Learning in Geosimulation, Intelligent Tutoring Systems, Springer.

Wang, F., Turner, S. J., Wang, L. (2003). Integrating agents into HLA-based distributed virtual environment, In: Proc. 4th Workshop on Agent-Based Simulation, Müller, J. P., Seidel, M.M. (eds.), Montpellier, France, April 28-30, 2003, pp. 9-14.

Wasfy, H., M., Wasfy, T., M., Noor, A., K. (2004). An interrogative visualization environment for large-scale engineering simulations, Journal of Advances in Engineering Software 35, Elsevier, (2004) 805-813.