ARTES project 0005-22
2000-05-23

Real-Time Response and Control of Autonomous Agents
ARTES Grant Proposal

Nancy E. Reed
Real-time Systems Laboratory
Department of Computer and Information Science
Linköping University, SE-581 83 Linköping, Sweden
nanre@ida.liu.se

May 22, 2000

1 Summary

Autonomous agents are systems that are embedded in an environment with sensors for obtaining data and effectors for performing actions. Agents are pro-active in that they have some purpose in their environment. Agents have recently been applied in a wide variety of domains, including economic markets, Internet searching, disaster management and defense applications. Agents are ideal for development, testing and training applications because they do not tire as people do and they can be easily replicated to work in many different locations simultaneously. Because of the embedded nature of autonomous agents, they must make intelligent decisions on their next action in real-time.

The basic cycle of operation of an autonomous agent is to sense the environment, decide on an action and then execute the action. This cycle is referred to as the *decision cycle* of the system. The intelligent systems will often have multiple, possibly conflicting, objectives which need to be achieved. The number of possible actions is very large. One result is that the possible actions an agent has available are not always pre-enumerated. Instead, an "intelligent" action must be found (selected or constructed) and performed within the time constraints. The penalty for a system that does not react within its time constraints depends on several factors. If an action is not selected at one particular time, the opportunity may be lost forever. For example, if a simulated pilot's aircraft is close to the ground or an obstacle and the action that turns it to avoid the obstacle is not taken or is taken too late, the aircraft could be destroyed.

Related to the response requirement, it is also desirable to be able to partially control or influence the behavior of an agent in real-time. This type of control is particularly valuable when the agents are actors in human training exercises and the instructor wants to alter an agent's behavior during a scenario. For this type of control or influence to be available, the instructor must be able to understand how the behavior is generated and alter that decision-making capability in real-time.

2 Problem Statement

We are investigating solutions to the problems of real-time response and control of autonomous agents. Action selection is the process of choosing a particular action given a set of *objectives*. Action selection is only one part of an intelligent systems decision making – deciding on objectives, execution monitoring and long term planning being examples of other things an intelligent autonomous system must do – but it is a critical part.

An action is said to be *achieving an objective* if the action's effects will lead toward the system fulfilling the objective. The *quality* of an action with respect to an objective is the degree to which the action will achieve the objective. Actions fulfilling the objective in an inefficient or otherwise undesirable manner have low quality with respect to the objective. Actions which fulfill an objective in a desirable manner have high quality with respect to the objective. An example of satisfying multiple objectives is for an aircraft to fly toward a waypoint and at the same time avoid all other aircraft. These objectives are not efficiently satisfied by doing them one at a time. The best action is one that avoids all other aircraft and obstacles, but still gets to the waypoint quickly and with a reasonably smooth path. This is different from traditional planning in real-time systems in that it is not a matter of scheduling individual pieces of actions, but rather synthesizing and comparing alternative actions. In addition, an agent must select outputs for several different effectors within the same time period.

The *action selection problem* is to find the best quality action with respect to all objectives. Briefly, the best quality action with respect to all the objectives is the action which maximizes a function of the quality of the action with respect to each of the objectives. A more formal description is given below.

EASE (End-user Actor Development Environment) [Scerri and Reed, 1999, Scerri and Reed, 2000] is currently being developed and tested by creating actors that simulate pilots for the TACSI aircraft simulator [(Missions and Analysis), 1998] and simulated soccer players for the RoboCup soccer competition [Kitano *et al.*, 1997]. An investigation of the RoboCup Rescue disaster management application area is also under consideration.

3 Main ideas

There are a variety of different algorithms for the action selection problem. Some of the algorithms are guaranteed to generate an optimal action (according to some definition of optimality). Other algorithms have weaker guarantees on the quality of the generated action but better computational performance. However, few if any, of the algorithms can guarantee an action will be selected within a certain time period. In this research an *anytime algorithm* [Zilberstein, 1996, Zilberstein *et al.*, 1999] is developed and extended which addresses the problem of real-time deadlines on the generation and evaluation of actions. The algorithm produces an initial action almost immediately then continuously improve the action as time permits. This type of algorithm is desirable when the agent is operating under real-time constraints, i.e. the agent can guarantee it will reach some decision under arbitrary time pressure but will use all available time to improve the quality of that decision.

The algorithm presented here is adapted from Pirjanian [Pirjanin, 1998, Pirjanian and Mataric, 1999]. Pirjanian's algorithm is an efficient method for generating actions with a certain quality level. The generated action will be *pareto optimal*, which intuitively means that when compared to any other action, an improvement in the quality with respect to one objective implies a drop in quality with respect to another objective. In general there will be zero or more pareto optimal actions available to the intelligent system. The Pirjanian algorithm further guarantees that the quality of the action with respect to the highest priority objective will be at least as high as the quality of any other action with respect to the highest priority action. If there are multiple pareto optimal actions with equal quality for the highest priority action then the quality with respect to the second highest priority objective is compared, and so on.

The Pirjanian algorithm works by first filtering all possible actions for the set of *feasible* actions. A feasible action is one where all objectives are at least partially fulfilled by the action. The feasible actions are then filtered to remove actions for which there is another action that is strictly "better", i.e. the set of pareto optimal solutions is found. Finally the "best" action is found by comparing the quality of the actions with respect to the highest priority objective, second priority objective, etc.

The algorithm generates high quality solutions reasonably efficiently, however the algorithm must run it's course before any action is generated. For some applications this is unacceptable. The algorithm described here compromises somewhat on the guarantees of quality given by the Pirjanian algorithm in favor of being able to generate actions within arbitrary time limits. The algorithm differs from the Pirjanian algorithm in the following way: rather than working with sets of actions one action is assessed at a time and compared to the best action found so far. Whenever the algorithm is halted the best action found so far is the outcome of the action selection. While the algorithm is allowed to continue a better action may be found.

3.1 Action Selection

Mathematically an action can be represented by a vector $\mathbf{a} = \{a_1, a_2, \ldots, a_n\},\ where\ a_j$ represents the jth decision variable. Each decision variable is related to one degree of freedom of the intelligent system. The domain of a_j is the range of possible values for that degree of freedom. For example the domain of an engine speed decision variable will be the range of possible output voltages and the domain of a decision variable for radar status will be the different modes that the radar can have. In a normal action selection problem there may be many decision variables, each of which can have values in a potentially large range. Hence, $A = D_1 \times D_2 \ldots \times D_n,\ where\ D_i$ is the domain of a_i, is the domain of **a**, and will often be very large.

For each objective there is assumed to be a mapping which relates an action to the quality with respect to an objective:

$$quality = q(\mathbf{a}) \tag{1}$$

Q is the set of all quality functions, i.e. one for each objective.

Lower quality values mean that the actions do not achieve the objective while higher values mean that the action efficiently achieves the objective. Intermediate values represent intermediate levels of objective fulfillment.

The problem of action selection can then be defined as finding the action which maximizes some function of the quality of the action with respect to each objective[Pirjanin, 1998], i.e.:

$$\arg\max_{\mathbf{a}\in\mathbf{A}} [q_1(\mathbf{a}), q_2(\mathbf{a}), \ldots, q_n(\mathbf{a})] \tag{2}$$

3.2 Real-time control

The other aspect of autonomous agents in real-time environments is that of selective human control or influence over the behavior of an agent during its execution. The capability to partially control or influence the actions of an agent without completely taking control is called adjustable autonomy. We are developing and approach to this problem within the EASE development environment [Scerri *et al.*, 1999, Reed and Scerri, 1999]. In EASE, each simulated pilot is called an actor. The actions of each actor are controlled by a multi-agent system. Our approach uses a tool called “the boss” which includes graphical interfaces to monitor the behavior of the agents within each actor. Another part of the boss allows the creation, deletion or suspension of individual agents (or a hierarchy of agents) controlling an actor. Ths approach needs development and testing with respect to real-time performance issues.

4 Expected results and impact

The anytime algorithm being developed is integrated within EASE as the means of action selection by EASE actors. There are important improvements to be made to the algorithm to make the responses in real-time. This requires a complete analysis of the real-time requirements of the environments where EASE actors perform - specifically the TACSI aircraft simulator.

Instead of the single values which the original algorithm supported we now support hierarchically data structures. EASE currently allows fairly arbitrary hierarchies. The branches in the hierarchy can be mutually exclusive.

Because we are producing an actor's behavior and because reasonable behavior is sometimes enough, different action construction methods might work better under different conditions. We plan to test different methods of generating new values and evaluating the resulting behavior. E.g. genetic algorithms, simulated annealing, random search, hill climbing, etc. Basically the problem then becomes a real-time local search problem. Different scenarios and domains might work better with different methods. Additionally with different time restrictions different methods might be better.

For example a GA will often get very good solutions but will sometimes get very bad results. On the other hand hill climbing will always get reasonable solutions - which is better? Simulated annealing solutions jump around more than hill climbing which may mean that the resulting actor's behavior is too “jumpy” for some applications.

5 Project Plan

This research will be conducted by Paul Scerri and a new graduate student in the Real-time systems laboratory under the direction of Nancy Reed and Jörgen Hansson. Paul Scerri is expected to finish his Ph.D less than one year into the project - in the Spring of 2001.

Milestones: year one. The first stage of the project will update a survey of real-time techniques currently used in autonomous agents as well as real-time techniques that may be adapted to autonomous agents with particular emphasis on the action selection problem. The Ph.D. dissertation of Paul Scerri is expected in the Spring or Summer of 2001 with results of several experiments with EASE.

Milestones: year two. Analysis of the specific real-time requirements of at least two environments interfaced with EASE actors. Testing and analysis of real-time action selection mechanisms developed and integrated within EASE. Interface the new EASE system to Saab's TACSI aircraft simulator and RC Rescue disaster management simulator.

Milestones: years three and four. Generalization of the results from interfacing and testing the action selection mechanisms with several environments. A licentiate thesis is expected for the new graduate student in approximately year 3.

Deliverables. Journal and conference papes will be produced throughout the project. The source code for EASE will be continually updated and available on the RTSLAB web pages. Academic work planned includes the Ph.D. dissertation by Paul Scerri as well as a Licentiate thesis for the new graduate student.

Project timing. The project is expected to start in approximately October, 2000 and last for 5 years - until the Fall of 2005.

6 Preliminary Budget

The total project budget is estimated as 2400 KSEK over the five-year period. This will cover the funding of one Ph.D. student for 5 years, assuming a maximum of 20 % teaching duties. Approximately 6 months of the first year of funding is intended for Paul Scerri, with the remaining funding going to a new graduate student that Paul will help bring up to speed on the research. Funding will be used for student salaries, travel and mobility programs. We are applying for 600 KSEK for each year.

7 Related Research

Autonomous agents are being applied in more and more application areas. Good introductions to the field of agents can be found in [Huhns and Singh, 1997, Wooldridge and Jennings, 1994, Bradshaw, 1997]. Additional information is available from the *American Association for Artificial Intelligence*, (http://www.AAAI.org/) and *AgentLink*, (http://www.AgentLink.org/), a new ESPRIT funded agents organization in Europe.

There a number of large agent related conferences including *Autonomous Agents*, *International Conference on Multi-Agent Systems (ICMAS)* and *Agent Theories, Architectures, and Languages (ATAL)*. There are also agent related sections at most artificial intelligence conferences and distributed computing conferences.

Previous work has not focused much on the real-time aspects of agent response or control with the notable exception of a recent symposium on real-time autonomous systems chaired by David Musliner [Musliner, 2000].

8 Relationship to the grant CFP

This research addresses the following relevant questions most directly. "How to specify and design a heterogeneous real-time system?" and "What are the implications for processes, methods and tools for real-time systems. Specific topics most relevant to action selection in and control of autonomous agents are resource handling and distributed systems.

9 Industrial Relevance

Autonomous agents are being used in more and more application areas, from space exploration to manufacturing to development. Systems are becoming more and more complex and we are looking for methods to improve the development and reliability of these complex systems. Autonomous agents technology offers "intelligent" behavior for the actors. If the actors can also respond within real-time constraints, the actors can then replace human experts in simulations and tests, reducing costs as well as freeing the experts from the tedious and time-consuming task of trying to exactly repeat the same sequence of actions several times for different trainees or scenarios.

10 Relation to other SSF programmes

Other funding from a variety of sources is being sought by members of the lab for related projects.

11 Context

The Laboratory for Real-time Systems at Linköping university has undergone a change recently with Dr. Anders Törne moving to part-time status last year and Dr. Simin Nadjm-Tehrani starting this year as the new lab leader and a fourth senior researcher joining the lab - Dr. Jörgen Hansson. The budget of the lab for the year 2000 is approximately 4800 KSEK, roughly half of it externally funded by TFR, NUTEK and EU projects. The lab is internationally active in the areas of autonomous agents, formal methods, hybrid systems, real-time databases and scheduling, and systems engineering.

11.1 Research Group

Dr. Törne, Paul Scerri, and Dr. Reed have been working in the area of autonomous agents for several years. This funding will be used to fund a small part of the remaining work by Paul Scerri as well as start a new graduate student to continue the work with more focus on real-time issues in autonomous agents.

11.2 Complementary activities and funding

Funding for the complementary project "Specification of Agents in Interactive Simulation Environments" comes from the Swedish National Board for Industrial and Technical Development (NUTEK), under grants IK1P-97-09677, IK1P-98-06280 and IKIP-99-6166. Funding for "Agents in Complex Simulation Environments" comes from the Center for Industrial Information Technology (CENIIT) under grant 99.7.

11.3 Research Cooperation

Group members have been and are cooperating with several international groups. Paul Scerri is currently spending 6 months working with Prof. Milind Tambe's group as USC's Information Sciences Institute.

Dr. Reed submitted a membership application and is now the contact person for Linköping University in AgentLink. She is the chair of a workshop titled "Teams with Adjustable Autonomy" to be held at the Pacific Rim Conference on Artificial Intelligence in August this year.

11.4 Industrial cooperation

This work is supported by Saab AB. Our primary contact is: Tomas Karlsson, Manager, Operational Analysis Saab AB Grippen, 581 88 Linköping, phone: 013 18 4625, fax: 013 18 5332, email: tomas.karlsson@saab.se.

References

[Bradshaw, 1997] Jeffery Bradshaw. An introduction to software agents. In *Software Agents*, pages 3–49. MIT Press, 1997.

[Huhns and Singh, 1997] Michael Huhns and Munidar Singh. *Readings in Agents*, chapter Agents and Multiagent systems: themes approaches and challenges. Morgan Kaufmann, 1997.

[Kitano *et al.*, 1997] Hiraoki Kitano, Minoru Asada, Yasuo Kuniyoshi, and et. al. RoboCup: A challenge problem for AI. *AI Magazine*, 18(1):73–85, Spring 1997.

[(Missions and Analysis), 1998] Saab (Missions and Systems Analysis). Proposed study subjects in Saab (GDIO) - LiTH (RTSlab) cooperation. Internal Memo, March 1998.

[Musliner, 2000] David Musliner. Real-time autonomous systems. Technical report, AAAI, 2000. Spring Symposium Series, Working Notes.

[Pirjanian and Mataric, 1999] Paolo Pirjanian and Maja Mataric. A decision theoretic approach to fuzzy behavior coordination. In *Proceedings, IEEE International Symposium on Computational Intelligence in Robotics & Automation (CIRA-99)*, Monteray, CA, Nov. 1999.

[Pirjanin, 1998] Paolo Pirjanin. *Multiple objective action selection and behavior fusion voting.* PhD thesis, Department of Medical Informatics and Image Analysis, Aalborg university, 1998.

[Reed and Scerri, 1999] Nancy E. Reed and Paul Scerri. Adjustable autonomy in simulated pilots. In *International Joint Conference on Artificial Intelligence, Adjustable Autonomy Systems Workshop*, pages 56–59, August 1999.

[Scerri and Reed, 1999] Paul Scerri and Nancy Reed. The ease actor development environment. Technical Report TACSIM-99-01, Computer and Information Science Department, Linköping University, October 1999.

[Scerri and Reed, 2000] Paul Scerri and Nancy E. Reed. Creating complex actors with ease. In *Fourth International Conference on Autonomous Agents (Agents 2000)*, page to appear. ACM Press, June 2000.

[Scerri *et al.*, 1999] Paul Scerri, Nancy E. Reed, and Anders Törne. An approach to directing intelligent agents in real-time. In *Proceedings of the AAAI Spring Symposium on Agents with Adjustable Autonomy*, pages 114–115, Stanford University, March 1999. American Association for Artificial Intelligence.

[Wooldridge and Jennings, 1994] Michael Wooldridge and Nicholas Jennings. *Intelligent Agents*, chapter Agent Theories, Architectures and Languages: A Survey. Springer-Verlag, 1994.

[Zilberstein *et al.*, 1999] S. Zilberstein, F. Charpillet, and P. Chassaing. Real-time problem-solving with contract algorithms. In *Proceedings of 16th joint conference on artificial inteligence*, Stockholm, Sweden, August 1999.

[Zilberstein, 1996] S. Zilberstein. Using anytime algorithms in intelligent systems. *AI Magazine*, 1996.

Appendices

A. CV of Nancy Reed

B. Support letter from Saab - to come under separate cover.

Nancy E. (Lundgren) Reed, Ph.D.

Department of Computer and Information Science
Linköping University
SE-581 83 Linköping, Sweden
Phone: +46 13 28 2668, Fax: +46 13 28 4020
nanre@ida.liu.se
http://www.ida.liu.se/~nanre/

Department of Computer Science
University of California, 2063 Eng. II
One Shields Avenue, Davis, CA 95616-8562, USA
Phone: (530) 752-3128, Fax: (530) 752-4767
nereed@ucdavis.edu
http://wwwcsif.cs.ucdavis.edu/~reedn/

Education

Ph.D. **1995**. Computer Science, University of Minnesota, Minneapolis. Title: Diagnosing Multiple Interacting Defects with Cue Combination Descriptions.

M.S. **1988**. Computer Science. University of Minnesota, Minneapolis. Title: Strategies and Inexact Models in Computer Hardware Diagnosis.

B.S. **1977**. Biology. University of Minnesota, Minneapolis.

Professional Experience

[1998-] **Assistant Professor (Forskarassistent, Lektor)**, Computer and Information Science Department, Linköpings Universitet, Sweden. Research in autonomous agents, interactive simulation environments, diagnosis, and real-time systems. Operating Systems (TDDB12 and TDDB63) course instruction.

[1996-] **Adjunct Assistant Professor** (on leave 1998-99, 99-00), [94-95, 96] **Lecturer**, [95] **Researcher**, Computer Science Department, University of California, Davis. Introduction to Computers (ECS15), Introduction to Software Development (ECS40), Discrete Mathematics (ECS100), Data Structures (ECS110), and Decision-Support Systems (PMD207) course instruction. Knowledge-based system development for the Remote Technical Assistance (RTA) project.

[1993-94] **Lecturer**, Computer and Information Science Department, Sonoma State University, Rohnert Park, California. Introduction to Computers laboratory (CIS101) and Artificial Intelligence (CIS480) course instruction.

[1989-91] **Assistant**, Artificial Intelligence Laboratory, Swiss Federal Institute of Technology, Lausanne, Switzerland. Software development for education. Supervision of student diplome (M.S.) projects. Unix system administration.

[1985-88] **Research Assistant**, University of Minnesota. Compilation of a bibliography on expert system explanation and query generation. Translation of the Galen expert system shell from Franz Lisp to Common Lisp. Research on the reasoning of computer hardware diagnosis experts.

[1986] **Computer Programmer,** Control Data Corporation, Arden Hills, Minnesota. Feasibility study and prototype expert system implementation for a computer hardware diagnosis task.

[1985-86]**Teaching Assistant**, University of Minnesota. Development and grading of assignments and tests. Assisting students. Courses: Pascal, Lisp, and Artificial Intelligence.

[1983-84] **Professional Research Assistant**, Molecular, Cellular and Developmental Biology Dept., University of Colorado, Boulder. Research on animal tumor viruses. Supervision of student employees.

[1982-83] **Physical Science Technician**, U. S. A. Environmental Hygiene Agency, Fitzsimons Army Medical Center, Aurora, Colorado. Chemical analysis of water and soil samples.

[1978-81] **Research Laboratory Technician**, Gastroenterology Research Unit, Mayo Clinic, Rochester, Minnesota. Gastrointestinal hormone research.

Funding

PI: Nancy E. Reed, *Diagnosing Interacting Defects Using Cue Combination Types,* NSF Grant# 9870454, $75,000, Sept 1998 - Feb 2000.

PIs: Anders Törne and Nancy E. Reed, *Specification of Agent Behavior in Complex, Interactive, Simulation Environments,* Swedish National Board for Industrial and Technical Development (NUTEK) Grant# 1K1P-97-09677, SEK 425,000 (approx. $55,000), Apr 1998 - Mar 1999. Matching support is provided by Saab AB, Operational Analysis Division.

PI: Nancy E. Reed, *Specification of Agent Behavior in Complex, Interactive, Simulation Environments*, Swedish National Board for Industrial and Technical Development (NUTEK) Grant#s 1K1P-98-06280 and IK1P-99-6166, SEK 850,000 (approx. $105,000) Jan 1999 - Dec 2000. Matching support is provided by Saab AB, Operational Analysis Division.

PI: Nancy E. Reed, *Interactive Simulation Environments Research Group,* Center for Industrial Information Technology (CENIIT) Grant# 99.7, SEK 750,000 (approx. $95,000), Jan 1999 - Dec 2000.

Publications

I. Reviewed Journal Publications

[1] Nancy E. Reed and Paul E. Johnson. 1993. Analysis of expert reasoning in hardware diagnosis. *The International Journal of Man-Machine Studies*, Vol. 38, No. 2, pp. 251-280.

[2] Richard F. Walters and Nancy E. Reed. 1996. Distance learning, Can we use it to teach M programming? *M Computing*, Vol. 4, No. 1, pp. 20-24.

[3] Nancy E. Reed, Maria Gini, and Paul E. Johnson. 1996. Robust strategies for diagnosing manufacturing defects. *Applied Artificial Intelligence*, Vol. 10, No. 5, pp. 387-406.

[4] Richard F. Walters and Nancy E. Reed. 1996. Outcome Analysis of Distance Learning: A Comparison Between Conventional and Independent Study Instruction. On The Horizon, Integrating Information Technology Tools in Instruction. Online:
`http://horizon.unc.edu/projects/monograph/CD/Science_Mathematics/Walters.asp` or
`http://www.ida.liu.se/~nanre/pubs/distance-learning97.html`
Also available on CD-ROM from Microsoft.

[5] Nancy E. Reed, Maria Gini, Paul E. Johnson, and James H. Moller. 1997. Diagnosing Congenital Heart Defects using the Fallot Computational Model. *Artificial Intelligence in Medicine* (Special issue on Knowledge-based systems in Cardiovascular Medicine) Vol. 10, No. 1, pp. 25-40.

[6] Paul Scerri and Nancy E. Reed, 2000. Engineering Characteristics of Autonomous Agent Architectures, to appear in the *Journal of Experimental and Theoretical Artificial Intelligence*

II. Reviewed International Conference Publications

[1] Nancy E. Reed, Elizabeth R. Stuck, and James B. Moen. August 22-26, 1988. Specialized strategies: an alternative to first principles in diagnostic problem solving. *Proceedings of the Seventh National Conference on Artificial Intelligence (AAAI-88)*, pages 364–368, St. Paul, MN.

[2] Dmitry Grivas and Nancy E. Reed. June, 1990. A framework for knowledge-based applications in pavement management. *VTT Symposium 116: OECD Workshop on Knowledge-Based Systems in Transportation*, pages 401–423, Espoo, Finland.

[3] Nancy E. Reed and Paul E. Johnson. August 6-10, 1990. Generative knowledge for computer troubleshooting. *Proceedings of the Ninth European Conference on Artificial Intelligence (ECAI-90)*, pages 535–540, Pitman Publishing, Stockholm, Sweden.

[4] Nancy E. Reed. March 21-23, 1994. Robust strategies for diagnosing manufacturing defects. *Proceedings of the AAAI Spring Symposium on Detecting and Resolving Errors in Manufacturing Systems*, pp. 129-133, Stanford University.

[5] Nancy E. Reed. July 31 - August 4, 1994. Diagnosing multiple interacting defects with combination descriptions. *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94)*, p. 1486, Seattle, Washington.

[6] Nancy E. Reed, Maria Gini, and Paul E. Johnson. March 25-27, 1996. Diagnosing Multiple Interacting Defects with Examples in Pediatric Cardiology. *Proceedings of the AAAI Spring Symposium on Artificial Intelligence in Medicine*, Stanford University, pp. 139-143.

[7] Steven P. Fonseca and Nancy E. Reed. August 4-8, 1996. Integration of an Expert Teaching Assistant with Distance Learning Software. *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI-96)*, p. 1388, Portland, Oregon.

[8] Nancy E. Reed. 1998. Constructing the Correct Diagnosis When Symptoms Disappear. July 1998. *Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI-98)*, pages 151–156, Madison, WI.

[9] Paul Scerri and Nancy E. Reed. 1999. An Approach to Directing Intelligent Agents in Real-time. March 1999. *Proceedings of the AAAI Spring Symposium on Agents with Adjustable Autonomy*, pages 114–115, Stanford University.

[10] Paul Scerri and Nancy E. Reed. 1999. Adapting an Agent to a Similar Environment May 1999. *Proceedings of the Third International Conference on Autonomous Agents (Agents 99)*, pages 420–421, Seattle, WA.

[11] Paul Scerri and Nancy E. Reed. 1999. Requirements for a General Agent Architecture for Agent-Based Simulation Environments, May 1999. *Proceedings of the Workshop on Autonomy Control Software*, Autonomous Agents '99, pages 102–108, Seattle, WA.

[12] Nancy E. Reed and Paul Scerri. 1999. Adjustable Autonomy in Simulated Pilots, August 1999. *Proceedings of the Workshop on Adjustable Autonomy Systems,* Sixteenth International Joint Conference on Artificial Intelligence (IJCAI99) pages 56-59, August 1, Stockholm, Sweden.

III. Other Publications

[1] G. Johnson, A.J. McCullough, N. Reed, and V.L.W. Go. May, 1980. Nutrient Specific Immunoreactive Gastric Inhibitory Polypeptide (IRGIP) Subspecies Response. *Gastroenterology*, p. 1330, Vol. 78, No. 5, Part 2.

[2] Nancy E. Reed. March, 1988. *Strategies and Inexact Models in Computer Hardware Diagnosis.* M.S., University of Minnesota, Computer Science Dept., 136 pages.

[3] Nancy E. Reed. Winter, 1993. FOCUS: Fallot. American Association for Artificial Intelligence - Artificial Intelligence in Medicine Subgroup Newsletter, Vol. 2, No. 2, pp. 1-2.

[4] Nancy E. Reed. June, 1995. *Diagnosing Multiple Interacting Defects with Cue Combination Descriptions.* Ph. D. Thesis, University of Minnesota, Computer Science Dept., 203 pages.

[5] Nancy E. Reed. August, 1996. Diagnosing multiple interacting defects with combination descriptions. Technical report CSE-96-19, Computer Science Department, University of California, Davis. Updated online version of a poster presented at the Twelfth National Conference on Artificial Intelligence (AAAI-94), Seattle, Washington. Online:
`http://www.ida.liu.se/~nanre/pubs/cse-96-19.html`.

[6] Steven P. Fonseca and Nancy E. Reed. August, 1996. Integration of an Expert Teaching Assistant with Distance Learning Software. Technical report CSE-96-20, Computer Science Department, University of California, Davis. Online version of a poster presentation at the Thirteenth National Conference on Artificial Intelligence (AAAI-96), Portland, Oregon. Online:
`http://www.ida.liu.se/~nanre/pubs/cse-96-20.html`.

[7] Nancy E. Reed. 1997. Report from a Grace Hopper Celebrant. *;login:* Vol. 22, No. 6, p. 5.

[8] Paul Scerri, Nancy Reed and Anders Törne. 1999. EASE – Enduser Agents Specification Environment, Technical Report TACSIM-99-01, Computer and Information Science Department, Linköping University, October. Online:
`http://www.ida.liu.se/~rtslab/publications/1999/tacsim9901ease.pdf.gz`.