GUEST EDITORS' INTRODUCTION

AI Space Odyssey

Michael G. Shafto, *NASA Ames Research Center*

Maarten Sierhuis, *PARC*

The field of intelligent systems has witnessed dramatic advances since 1980. The Web has enabled an environment in which some of the basic problems identified in the 1970s are emerging in new, often more concrete and practical forms. Correspondingly, the opportunities for space exploration to benefit from intelligent systems are evolving in unexpected directions, with certain invariants in knowledge representation and process modeling.

Past: AI Dawns on the Space Program

In 2008, Patrick Stakem provided a partial bibliography of flight-software references dating back at least to 1956.[1] What follows is a brief sketch of the odyssey of AI in Space, in roughly one-decade increments, from 1980 to the present.

Advanced Automation, Machine Intelligence, and Robotics

Beginning in 1979, Carl Sagan chaired a study group on machine intelligence and robotics.[2,3] The distinguished participants included several who had won, or would go on to win, the ACM Turing Award—Ivan Sutherland, Marvin Minsky, Allen Newell, Herb Simon, Alan Perlis, and Raj Reddy (vice chair of the study group).

They noted that the goal of space exploration missions is collecting, analyzing, and disseminating information. Their primary recommendation was that space agencies

1541-1672/10/$26.00 © 2010 IEEE
Published by the IEEE Computer Society

"should adopt a policy of vigorous and imaginative research in computer science, machine intelligence, and robotics in support of broad [exploration] objectives."

Another cogent suggestion was that "[m]ission objectives should be designed flexibly to take advantage of existing and likely future technological opportunities." They detailed several key application areas where they felt intelligent systems should be considered for broader use: smart sensors, autonomous manipulators, fault management, teleoperations, on-board AI for rovers, computer architectures, formal methods, and improved software development tools.

Shortly after the Sagan Report was published, Robert A. Freitas, Jr., Timothy J. Healy, and James E. Long reported on a workshop that focused on advanced automation for space missions.[4] Like the Sagan Report, their conclusions and recommendations seem prescient in light of three decades of experience:

> Advanced machine technology is essential in realizing a major space program capability for extra-terrestrial exploration and resource utilization within realistic temporal and economic limits....
>
> (1) Machine intelligence systems with automatic hypothesis formation capacity are necessary for autonomous examination of unknown environments....
>
> (2) The development of efficient models of Earth phenomena and their incorporation into a world-model-based information system are required for a practical, user-oriented, Earth resource observation network.
>
> (3) A permanent manned facility in low Earth orbit is an important element of a future space program. Planning for such a facility should provide for a significant automated space manufacturing capability.
>
> [...]
>
> (6) General and special purpose teleoperator/robot systems are required for a number of manufacturing, assembly, inspection and repair tasks.
>
> (7) An aggressive ... commitment in computer science is fundamental to the acquisition of machine intelligence/automation expertise and technology required for the mission capabilities described earlier. This should include a program for increasing the number of people trained in the relevant fields of computer science and artificial intelligence.[4]

Human Factors in Automated and Robotic Space Systems

In 1987, Thomas B. Sheridan, Dana S. Kruser, and Stanley Deutsch edited the proceedings of a National Research Council Symposium on Human Factors in Automated and Robotic Space Systems.[5] (Tom Sheridan and Allen Newell participated in this symposium and in the Sagan study of machine intelligence.) Focal topics included human-system productivity in space, especially on the International Space Station; applications of mixed-initiative systems in space; natural language processing; decision-support systems; teleoperation and telepresence; computer-mediated communication; and human-computer interaction. Their conclusions and recommendations emphasized the need for long-range, sustained research investment in these key areas related to human-automation interaction.

Planning and Scheduling, Architectures, and Trust in Autonomy

Rich Doyle has contributed a number of articles and edited sections on AI in Space to various IEEE journals. In his Guest Editor's Introduction to a 1998 special issue,[6] he discussed the strategic value of intelligent autonomous systems in extending the number, capability, and affordability of future NASA missions. The articles in that issue covered a range of important topics, including formal methods, planning and scheduling, data mining, trust in autonomous systems, and intelligent systems architectures.

Present: Scaling Up to Enterprise and Mission Levels

This special issue covers semantic technologies, including natural language processing, sophisticated onboard fault management, human-robotic interaction, and agent-based simulation architectures. These topics complement the articles in Doyle's 1998 special issue, and they share its focus on strategic value. Specifically, they reflect the topics of data mining and intelligent systems architectures in ontology-related work that incorporates text mining and in large-scale intelligent-agents architectures. Building trust in autonomy is also a theme of ours.

Jane T. Malin, Christopher Millward, Fernando Gomez, and David Throop's article "Semantic Annotation of Aerospace Problem Reports to Support Text Mining" describes a semantic annotation approach that allows NASA analysts to find sets of related problem reports. Text mining of problem reports is also improved by their innovative method by overcoming some shortcomings of automated extraction of semantic information from written reports.

Semantic technologies across the spectrum of text mining, ontology, and natural language processing offer a new capability that combines flexible human processing with logical structure and automated reasoning.[7] This type of technology has broad applicability not only to analysis of problem reports, but also to mission design and planning.

"The TacSat-3 Vehicle Systems Management Experiment" by Ryan Mackey, Lee Brownston, Joseph P. Castle, and Adam Sweet addresses one of the issues discussed in Doyle's GEI[2]—trust in automated reasoning systems, in this case fault detection and diagnosis systems. Automated fault-management systems can provide benefits in space operations. Questions remain, however, as to whether high-reliability systems can be implemented within current on-board computing limitations. To address this problem, NASA and the US Air Force Research Laboratory (AFRL) used the TacSat-3 spacecraft as an example of an intelligent space system. Their tests demonstrate proper function in high-fidelity simulation, providing credible evidence that the proposed intelligent system could be trusted in any foreseeable spacecraft performance environment.

Following the pioneering work of Bill Clancey and his colleagues,[8] Debra Schreckenghost, Tod Milam, and Terrence Fong's article "Measuring Performance in Real Time during Remote Human-Robot Operations with Adjustable Autonomy" reports quantitative analyses of human-robotic team performance in a space-analog environment. Such analyses are crucial for building confidence that NASA missions can conduct complex surface exploration activities using interactive robots. In one plausible class of scenarios, Earth-based operators will remotely supervise multiple robots performing planned tasks. An important aspect of such operations is the flexible allocation of tasks between the robots and their operators, called adjustable autonomy. Metrics are needed to evaluate the effectiveness and efficiency of such operations.

When evaluating human-agent teams in a dangerous, complex environment, it can be impossible to evaluate system performance in the actual context. There are limits on the involvement of actual end users in the evaluation; astronauts and flight controllers are overworked and have no time to participate in HITL simulations. The adaptive agents and human actors operating in a dynamic environment make it difficult to control the experiment. In "Assessing Human-Agent Teams for Future Space Missions," Nanja J.J.M. Smets, Jurriaan van Diggelen, Mark A. Neerincx, Jeffrey M. Bradshaw, Catholijn M. Jonker, Lennard J.V. de Rijk, Pieter A.M. Senster, Ot ten Thije, and Maarten Sierhuis propose an agent-based simulation platform that makes it possible to run computer simulations of work practices early in the development process and to chart a reliable path from simulation to implementation.

Future: Mathematics Chasing Complexity

Earlier this year, David Parnas provided a thorough discussion and historical perspective on the use of mathematical methods in software engineering and analysis.[9] There are many attractive aspects of formal methods, for example, provable correctness, interoperability with other software and hardware systems, and much better tools for software development. However, current methods and tools have not been widely adopted, and usability seems to be a chronic problem. Rigorous mathematical methods must be supported by techniques for step-by-step processes. These processes must be understandable by other engineers in order for them to have confidence in complex software components.

> [E]verything that we can derive from an abstraction must be true of the real thing. If we can derive something that is not actually true, what we have is not an abstraction but a lie. Our role-model should be engineers, not philosophers or logicians. Engineers use mathematics in very different ways from pure mathematicians and logicians. Mathematicians who prove theorems use axiom systems that allow them to search for a proof. Engineers usually evaluate expressions.[9]

A valuable extension of Parnas's ideas is available in Johann Schumann and Yan Liu's 2010 book, which is one of the first attempts to apply advanced mathematical methods to the design of complex, embedded intelligent systems.[10] Their work begins to address the difficult problems of formally verifying and validating adaptive control systems. This line of work might eventually help justify greater confidence in the reliability of intelligent systems, thereby opening new opportunities for more advanced software to be incorporated into future aerospace systems.

Recently there has been increasing discussion of a class of complex systems that seem to go beyond neural nets and other kinds of adaptive systems. These are referred to as ultra large-scale systems (ULS) or—reusing a term from decades earlier—self-organizing architectures.[11]

Future software systems might have the capability to adapt to changing operating conditions or user requirements. Bottom-up and top-down solutions are both possible. In either case, system components adapt their behavior to changing conditions. Both approaches have been able to realize self-adaptive systems, but both face challenges in dealing

THE AUTHORS

Michael G. Shafto is a deputy chief technologist at the NASA Ames Research Center. His research interests include modeling and simulation, intelligent agents, data analysis, and semantic technologies. Shafto has a PhD in experimental psychology (psycholinguistics) from Princeton University. Contact him at mike.shafto@nasa.gov.

Maarten Sierhuis is the Area Manager for Knowledge, Language, and Interaction at the Palo Alto Research Center (PARC). His research interests include multiagent environments, agent-based modeling and simulation of organizations and work practice, and human-systems interaction. Sierhuis has a PhD in social science informatics from the University of Amsterdam, the Netherlands. Contact him at sierhuis@parc.com.

with nondeterminism and with global incoherence, incompressibility, and chaos.

Joseph Licklider said, “People tend to overestimate what can be done in one year and to underestimate what can be done in five to ten years.”[12]

There is no doubt that intelligent systems research and technology have advanced dramatically since 1979. The Web has offered a new domain for intelligent systems to inhabit. Ontology and agents are of great practical and economic importance today—no longer abstract concepts relegated to a sub-subset of logical and philosophical arcana. Future space missions will be dramatically different as a result of this progress, different in how they are conceived, planned, designed, executed, and communicated to a worldwide constituency, a new generation of enthusiastic explorers. ■

References

1. P.H. Stakem, “Partial Bibliography of Spacecraft Computer/Flight Software References,” MEI Technologies, 2008.
2. C. Sagan et al., “Machine Intelligence and Robotics: Report of the NASA Study Group. Executive Summary,” NASA report 730-51, Jet Propulsion Lab., 1979.
3. C. Sagan et al., “Machine Intelligence and Robotics: Report of the NASA Study Group,” final NASA report 715-32, Jet Propulsion Lab., 1980.
4. R.A. Freitas, Jr., T.J. Healy, and J.E. Long, *Advanced Automation for Space Missions*, Space Initiative, 1980.
5. T.B. Sheridan, D.S. Kruser, and S. Deutsch, eds., *Proc. Symp. Human Factors in Automated and Robotic Space Systems*, Commission on Behavioral and Social Sciences and Education, Nat’l Research Council, 1987.
6. R.J. Doyle, “Guest Editor’s Introduction: Spacecraft Autonomy and the Missions of Exploration,” *IEEE Intelligent Systems*, vol. 13, no. 5, 1998, pp. 36–44.
7. T. Gruber, “Collective Knowledge Systems: Where the Social Web Meets the Semantic Web,” *Web Semantics: Science, Services and Agents on the World Wide Web*, 2007; doi:10.1016/j.websem.2007.11.011.
8. W.J. Clancey et al., “Automating CapCom using Mobile Agents and Robotic Assistants,” *Proc. Am. Inst. Aeronautics and Astronautics 1st Space Exploration Conf.*, NASA, 2007.
9. D.L. Parnas, “Really Rethinking ‘Formal Methods,’” *Computer*, vol. 43, no. 1, 2010, pp. 28–34.
10. J. Schumann and Y. Liu, eds., *Application of Neural Networks in High Assurance Systems*, Springer, 2010.
11. D. Weyns et al., *Proc. 1st Int’l Workshop Self-Organizing Architectures*, Springer, 2010.
12. J. Licklider, *Libraries of the Future*, MIT Press, 1965, p. 17.

IEEE Computer Society Publications Office
10662 Los Vaqueros Circle, PO Box 3014
Los Alamitos, CA 90720-1314

Submissions: For detailed instructions and formatting, see the author guidelines at www.computer.org/intelligent/author.htm or log onto *IEEE Intelligent Systems*’ author center at Manuscript Central (www.computer.org/mc/intelligent/author.htm). Visit www.computer.org/intelligent for editorial guidelines.