Autonomous Systems

David P. Watson and David H. Scheidt

Systems that can change their behavior in response to unanticipated events during operation are called "autonomous." The capability of such systems and their domains of application have expanded significantly in recent years, with high-profile successes in both civilian and military applications. These successes have also been accompanied by high-profile failures that compellingly illustrate the real technical difficulties associated with seemingly natural behavior specification for truly autonomous systems. The rewards are great for advancing this technology, however. Autonomous systems technology is truly transformational, with potential benefits in both cost and risk reduction. The technology also holds the potential for enabling entirely new capabilities in environments where direct human control is not physically possible. Note also that autonomy development is a discipline that cuts broadly across traditional engineering domains and system life-cycle phases, a true systems engineering discipline. For all of these reasons, APL has identified autonomous systems technology as an important element of its science and technology vision and a critical area for future development.

BACKGROUND

History

The idea of a machine intelligence embodied in an actuated physical system is not new. In fact, early Greek myths of Hephaestus and Pygmalion include concepts of animated statues or sculptures.[1] These ideas have persisted throughout history, with periodic attempts to achieve some limited set of functionality using the technology of the time. Although these efforts sometimes produced extremely complex mechanical devices ("automata") that mimicked human action, they are more properly characterized as works of art than of engineering.[2] In more recent history, the field of cybernetics was born in 1940 when Norbert Wiener, a mathematics professor at MIT working to develop automated rangefinders for anti-aircraft guns, began to ponder the seemingly "intelligent" behavior of these servomechanisms and the apparent similarity in both their nominal and anomalous (failed) operation to biologic systems.[3] This work led to the formalization of a theory of feedback control and its generalization to biologic (human) systems. This theory motivated the first generation of autonomous systems research in which simple sensors and effectors were combined with analog control electronics to create systems that could demonstrate a variety of interesting reactive behaviors. In 1964, for example, the APL Adaptive Machines Group, led by Leonard Scheer, demonstrated an autonomous rover system that could navigate APL's hallways, identify electrical outlets in the wall, and then plug itself in to recharge its battery cells (Fig. 1).

With the advent of digital control electronics in the 1970s and increased interest in automated perception and cognition within the new field of "artificial intelligence," additional advances were made in autonomous systems that could plan and execute relatively complex operations with little or no human interaction. As the cost of sensors, actuators, and most significantly, processors has dropped over the past two decades, there has been significant growth in autonomous systems research for all operational modalities: air, surface, undersea, and space. Today, we are witnessing the maturation and transition of this research into a variety of systems that presents transformational civilian and military capabilities. This article describes some of these systems and presents a brief survey of the state of the art in a variety of autonomy domains. We also highlight some critical science and technology (S&T) challenges in expanding the application of autonomous systems in the future.

Critical Cross-Cutting Science and Technology

Research into autonomy has historically been tied closely to particular application domains. A fundamental premise of the APL vision for this S&T area is that there is significant benefit to focusing on cross-domain solutions. For example, note the similarity in autonomy requirements between spacecraft and underwater vehicles. For low-Earth-orbit missions, human operators have a periodic high-quality communications link with the system, allowing them to perform almost all high-level planning, control, and health management functions. In deep-space missions, however, communications link quality can be extremely low. In such situations, system designers must address these functional requirements in the absence of human support. Similar, if not identical, challenges face the undersea vehicle system engineer. Although certain missions may provide high-quality communications, permitting low-level supervisory control of vehicle systems, it is more likely that such systems will be limited to very low-quality communications links to the surface, resulting in a set of functional requirements identical to that of the deep-space vehicle, i.e., the ability to

- Develop a well-defined, yet modifiable, mission plan
- Execute the mission plan, modifying it if necessary
- React appropriately, if not optimally, to anomalous events
- Coordinate with human controllers

These functional requirements are also shared by ground and air systems that must operate with minimal human interaction in dynamic environments. There is a core set of technology areas that can address these current autonomous systems requirements as well as some future capabilities such as the ability to

- Improve performance through learning
- Coordinate with peer autonomous systems in mission operations

A final, key aspect of the autonomous systems we consider in this article is interaction with the physical world. Although "mobile" software constructs (e.g., viruses, agents) exist that can operate without direct human interaction, we do not specifically include them within the scope of this vision element. There are certainly analogies between software agent behaviors and autonomous systems in the physical world, and some overlap in component technologies and architectures, but we are particularly concerned here with the unique problem of interaction with an open physical world in the accomplishment of a complex performance goal.

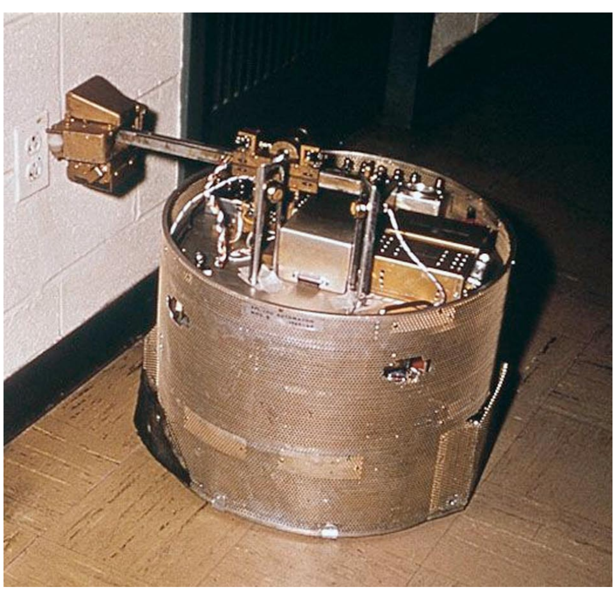


Figure 1. The APL "beast" (circa 1965).

APPLICATION DOMAINS

We begin by discussing some relevant application domains in terms of current capabilities and particular S&T challenges. As described above, APL has had a long history of involvement with autonomous systems and technologies. This involvement continues today in systems operating in maritime, ground, air, and space domains.

Maritime

Autonomy in the maritime domain has been focused primarily on submersible systems, for both shallow-water and deep-submergence applications. Torpedo guidance and control capabilities have become increasingly sophisticated and have formed the basis for some work in more general autonomy, including sophisticated systems such as the Mk 30 Mod 2 acoustic training target that can execute scripted "mission plans" and respond to real-time events. In general, such systems are assumed to operate in well-characterized areas and to have good connectivity with human operators through either reliable acoustic links or optical tethers. Deep-submergence vehicle systems, in contrast, may need to operate in unfamiliar areas and cannot always rely on a communications link to the surface. In this aspect, they have autonomy requirements similar to those of a science spacecraft operating in deep space.

APL has supported the development of unmanned underwater vehicle (UUV) systems and technologies since the mid-1980s for the Defense Advanced Research Projects Agency (DARPA), the Navy Program Office for Unmanned Undersea Vehicles (PMS-403), the Navy Program Office for Explosive Ordnance Disposal (PMS-EOD), the Office of Naval Intelligence, the Office of Naval Research, and others. The Laboratory was recently designated the Systems Engineering Agent for PMS-403 in support of their acquisition of a 21-in.-dia. Mission Reconfigurable UUV System. Laboratory staff were also members of the core team that wrote the Navy UUV Master Plan in 2000 and its update in 2004.[4] The plan describes nine critical mission capabilities, with highest priority for intelligence, surveillance, and reconnaissance (ISR) and mine countermeasures. In a comprehensive analysis on enabling technologies for these mission capabilities, autonomy was cited as one of the critical technologies requiring investment in the future.

Ground

The terrestrial operating domain provides certain advantages as well as particular challenges in comparison to other domains. Reliable, high-quality communication between the system and its control station is less of an issue. Also, the relative stability of the terrestrial environment can provide the opportunity to suspend active control in order to maintain safety or perform additional processing. This option does not generally exist in other domains, where active control is required for system stability. Once a ground system is mobile, however, the terrestrial world presents a significantly more challenging operating environment in terms of obstacles and terrain than other operational domains. For these reasons, sensing and mobility have been much more of a focus in ground robotics research than high-level autonomy. The DoD's Joint Robotics Program Master Plan[5] has identified "semi-autonomous mobility" as one of five critical technology development priorities and "increased autonomy in manipulation and control" as a critical unfunded technology requirement.

Perhaps the most visible ground autonomy work in the DoD today is being done by DARPA as part of the Army's Future Combat System. This work has again focused on vehicle design (the Unmanned Ground Combat Vehicle program) and off-road navigation, sensing, and mobility (the Perception for Off-road Robotics program). In addition, the Army Research Laboratory Robotics Collaborative Technology Alliance has identified three technology areas essential to the development of semi-autonomous mobility: (1) perception, (2) intelligent control architectures and tactical behaviors, and (3) human–machine interfaces. The autonomy technologies we discuss in this article, in particular the "layered control" ideas we consider later, are in complete alignment with these requirements.

Air

APL's work in ground robotics systems (including current efforts in ground robotics swarming) has been limited to one or two small prototype endeavors. In contrast, the Laboratory has a long history of work in unmanned air vehicle (UAV) systems, ranging from small radio-controlled vehicles developed for ISR applications to significant efforts in mission planning and control as part of the Tomahawk missile program. Most recently, the Laboratory has been selected as the Common Operating System Integrator/Broker for the Joint Unmanned Combat Air Systems (J-UCAS) program. This combined DARPA/Navy/Air Force program is developing prototype aircraft to demonstrate the technical feasibility, utility, and value of networked high-performance autonomous air vehicles in combat missions that currently require manned aircraft (Fig. 2). The J-UCAS Common Operating System will push the limits in all aspects of the autonomy technology we discuss in this article.

The J-UCAS program is motivated by the very visible successes of high-performance UAV systems such as Predator and Global Hawk in recent ISR and combat actions. At the other end of the warfare spectrum, small UAVs are now experiencing unprecedented levels of use in tactical applications. Small systems such as Dragon Eye are routinely used to provide electro-optic, IR, or low-light video imagery directly to warfighters at the company/platoon level. Unfortunately, these small systems require direct human supervision for both control and data analysis. Furthermore, they operate as standalone systems with no direct connection to backbone networks or other tactical systems. The move toward "fire-and-forget" autonomy for these systems and the development of ad hoc sensor/control networking capabilities are key research challenges in this area.

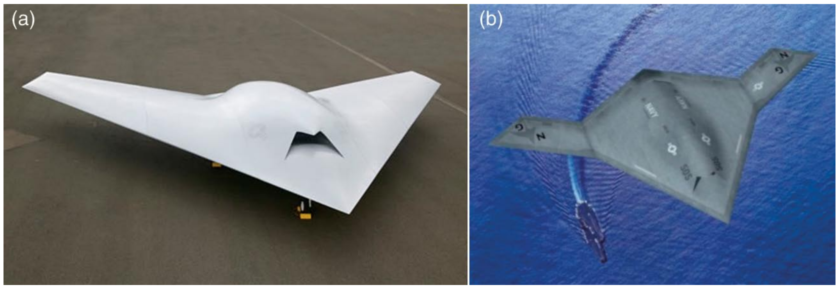


Figure 2. J-UCAS prototype aircraft: (a) Boeing X-45C and (b) Northrop Grumman X-47B.

Space

As with the other operational domains we have discussed, space presents unique enablers and challenges to the problem of autonomous operation. First, it is important to appreciate the distinction between near-Earth-orbit missions and deep-space missions. In the former, communications with the system can be assumed to be relatively reliable and of moderate to high quality in terms of bandwidth and latency. This provides the opportunity to perform most supervisory operations under direct, real-time human control (though this may be cost prohibitive). As with UAV systems, high-rate control loops are still required onboard to maintain stability, but all higher-level planning, scheduling, maintenance, and anomaly management functions can be done on the ground and uploaded to the spacecraft for execution. Deep-space missions, in contrast, present a variety of significant communications issues, including low-bandwidth channels that can be accessed only intermittently and have significant latencies—up to a 4-h delay for a one-way communication with the APL New Horizon spacecraft when it reaches rendezvous with Pluto, for example (Fig. 3). In this case, there is no alternative but to perform certain high-level decision making onboard without direct human supervision. New Horizon's mission is particularly problematic, for example, as traditional anomaly response actions may be inappropriate during the actual planetary flyby.

Space systems engineers have been dealing with these problems for many years, generally relying on two strategies: exhaustive analysis of potential mission events with *a priori* design of appropriate response actions, and overall simplification of system design. These strategies have worked well in the past, but as mission requirements (and the resulting systems) become more complex, it will be necessary to automate and embed increasingly higher levels of autonomous decision-making capability on the spacecraft itself. But this autonomy must be amenable to verification and validation, consistent with all other elements of spacecraft systems engineering. Although significant autonomy research has been performed in the space community, much of it has failed to transition into mission applications as a result of verification and validation issues.

AUTONOMY RESEARCH AND TECHNOLOGY

There are domain-specific and domain-independent aspects to the S&T required to address the requirements described above. Domain-specific aspects include sensing and perception, manipulation, mobility, power, navigation, and communications. Although these cannot be completely decoupled from autonomy, our focus here is on aspects of autonomy that may be invariant across operational domains. These include automated planning, layered control, model-based and reactive control, and behavior coordination.

Planning and Scheduling

Autonomous behavior begins with the establishment of a plan to accomplish some desired goal or set of goals subject to some given set of resources and constraints. Imagine a spacecraft transiting interplanetary space, with high-level instructions to image a set of objects in the free time between trajectory correction maneuvers. Each imaging operation requires a complex sequence of interdependent attitude thruster and instrument warm-up and initialization commands. Today this sequence would be constructed by the mission operations team and uploaded to the spacecraft well before the event, but true autonomy would enable the spacecraft controller itself, using formal models of subsystem capabilities and constraints, to establish the sequence and modify it if necessary during flight. Automation of planning processes such as these has been a central problem in the field of artificial intelligence for more than 30 years, and a number of important approaches, including state-space search and hierarchical task decomposition, have

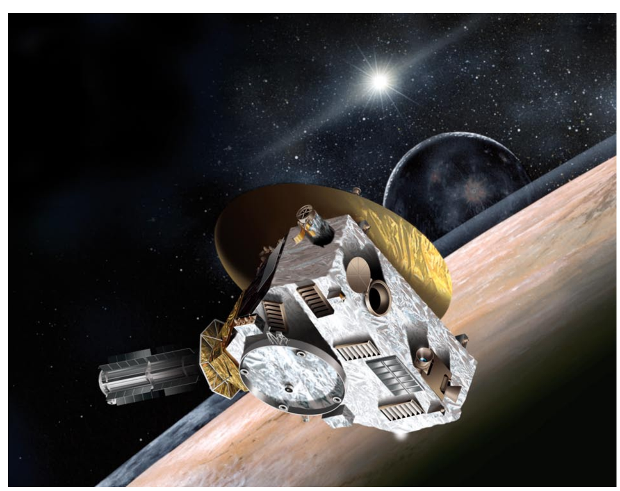


Figure 3. Artist's conception of the New Horizons spacecraft at Pluto.

evolved as a result of work in this area.[6] More recently, related work from the operations research community modeling sequential decision-making problems using Markov decision processes[7] has also been applied to the planning problem. There have been many successful applications of this technology, in domains ranging from transportation and military logistics scheduling to manufacturing process control.

Layered Control

The majority of research in automated planning has framed the problem as an offline process that can be addressed independent of system operation. Given an initial state, a goal state, and a set of operators or actions to work with, the objective is to derive a complete, optimal plan that can then be executed. The problem must be framed differently for autonomous systems operating in open (i.e., not completely modeled) environments, however. In that case, planning must generally proceed in parallel with plan execution in order to address the occurrence of unanticipated events.

Imagine that our opportunistic imaging spacecraft in the previous example is in the process of executing an image capture plan for some previously detected object when the detection sensor reports a newer, more interesting object to image. The previous plan must be terminated and a new one constructed as soon as possible, all the while maintaining overall system stability. The problem here is that planning algorithms generally strive for global optimality over the known set of resources and constraints. This implies computational complexity and, indeed, the general problem of state-space search planning, for example, is known to be "nonpolynomial-hard."[8] It is not feasible to put such a process in the real-time feedback control loop of an autonomous system.

To address this problem, a canonical architecture has emerged in the research community based on the notion of layered control loops that address the control problem at various timescales (and levels of abstraction) to provide a combination of responsiveness and robustness. Figure 4 shows a simplified example of such an architecture, where the lowest-level control loops are used to provide feedback control with deterministic responsiveness. This control is reactive in the sense that the system will be driven to some local optima with respect to the overall behavior goals (maintaining system safety, for example). This local control set point is determined by the next level of the architecture, which can be characterized as a *plan executor*. This loop uses filtered perception data to assess the progress of the system through a preplanned sequence of states. Responsiveness at this level is limited to contingency plans that have been prepared in advance and the event conditions that trigger them. At the highest level of control, a *deliberative planning process* uses explicit, declarative models of the system and its environment, combined with state information from the plan execution layer, to determine if the current situation requires global replanning. All layers operate asynchronously in parallel to produce the controlled behavior. This architecture has become almost ubiquitous in autonomy systems ranging from underwater vehicles to exploratory spacecraft.[9] An elaborated version has even been proposed as a standard for intelligent manufacturing systems.[10]

Model-Based Reasoning

Within the general framework of planning and control that we have described, there are a number of dimensions or functional aspects of behavior tailored to the particular autonomy domain. These include navigation, mobility, power, health management, and payload-specific operations. Although these tend to be primarily domain-specific technologies, we can make some general observations. Each functional area can have dedicated sensing, control, and actuation requirements, yet these cannot be strictly partitioned in the system architecture because of the inherent coupling of subsystems through shared power, mechanical, and computing resources. For example, at some point the navigation subsystem on an autonomous ground vehicle may require time and mobility resources to obtain a GPS fix, and these must be coordinated with other, perhaps higher-priority, requirements from a surveillance payload package. This

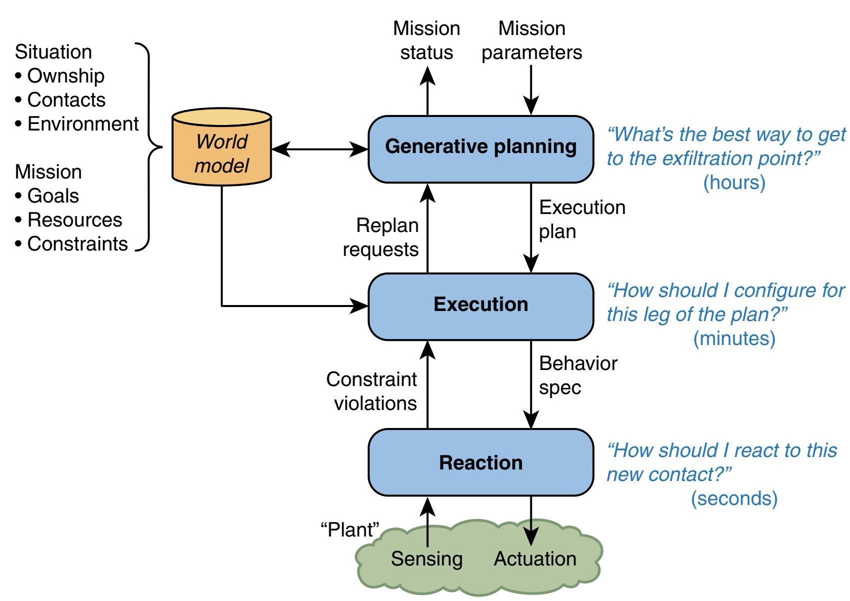


Figure 4. Layered autonomy model.

cross-coupling implies some centralized assessment and decision making at the highest level of system control for both mission and health management. The modeling and manipulation of these various facets of subsystem performance within a single integrated framework is a critical challenge for autonomous systems and an active area of current research.

In classical control theory, models of system behavior are an essential starting point in the derivation of optimal control policy. However, as systems expand in complexity (becoming, in fact, systems of interrelated subsystems), the construction of a single analytical model that characterizes the system in the classical sense may become infeasible. But the requirement to control such systems in a coordinated manner, as described above, is critical in autonomous systems. This has led to various heuristic, or ad hoc, approaches that generally amount to sets of control rules of the form: "If A is true, then do B." The construction of these rule sets may be driven by a rigorous systems engineering process (such as a Failure Modes, Effects, and Criticality Analysis), but it is fundamentally limited by the combinatoric expansion of subsystem interactions that must be considered in any complex system operating in a dynamic environment. Thus such approaches to autonomous control have historically been subject to failure as a result of inadequate or (less frequently) inconsistent rule sets. The NASA Accident Review Board for the recent Mars Polar Lander failure concluded that the spacecraft's autonomous control rules incorrectly shut down the descent engines based on landing leg sensor data before reaching the planet surface, despite having the necessary information onboard (independent altimeter data) to correctly reason that the leg sensor data were spurious transients and not truly indicative of a landed condition. The onboard rule set did not address this scenario because autonomy engineers did not anticipate it at the time the system was designed.

It is simply not feasible for systems engineers to reason through all potential subsystem interactions for autonomous systems of significant complexity. This realization was the motivation behind a new approach to autonomy design based on the use of explicit system models. Although this method resembles a control theoretic approach in the abstract, the types of system models and their use in an embedded controller are very different.

Current approaches to model-based reasoning in autonomous systems have their roots in early work by Randall Davis at MIT in digital circuit diagnosis.[11] Davis proposed a functional constraint satisfaction framework for system characterization which later proved amenable to extension into a comprehensive autonomy architecture that addresses system diagnosis, fault management, and top-level behavior control. Mature instantiations of this approach were tested in a deep-space experiment[12] and form the basis for new system development frameworks under development at the NASA Jet Propulsion Laboratory[13] and MIT.[14] The general idea, as illustrated in Fig. 5, is that autonomous behavior is controlled through a continuous process of "state estimation" and "state control," where system state and associated attributes, constraints, and transitions are defined in a set of declarative component models that capture both the nominal and failed behavior of all subsystems. These models take the place of rule sets in such controllers. Instead of a direct mapping from subsystem telemetry to command, an additional inference step is introduced that transforms the telemetry values into a state estimate, which is then used to derive commands to drive this estimated state to the current goal state. Structurally, this approach is identical to that used in modern control theory. The significant difference is in the runtime synthesis of control actions. To date this has been accomplished in a very limited manner using state-space search techniques, and extension of the state controller synthesis idea into a general theory of autonomous control represents a primary research challenge.

There are, however, several advantages of this approach for autonomy specification. First, the computational burden of considering all possible subsystem state interactions during operation is removed from the system designer, who can now focus on the specification of individual subsystem behaviors (both nominal and faulted). This specification leads to the potential for correct system response even to unanticipated operational scenarios. Indeed, the use of explicit behavior models forms the basis for a significant body of current research into the formal validation of autonomous control system performance, a critical aspect for use of this technology

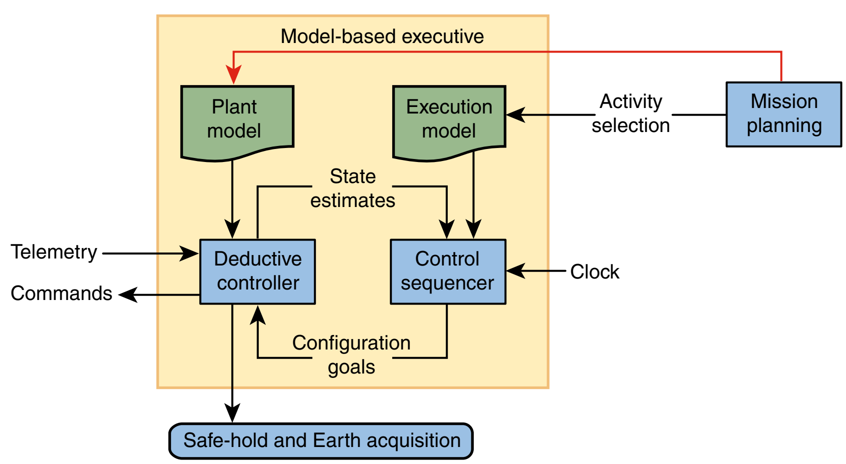


Figure 5. Model-based executive concept.

in high-reliability systems. The explicit computation and manipulation of abstract system states in the autonomy controller also provides a natural high-level interface to human supervisors and collaborators (see the article by Gersh et al., this issue). Finally, the use of explicit models in the controller provides the potential for model adaptation and learning during operation.

Despite the significant advantages of adopting a model-based programming and execution paradigm for autonomous systems, critical research and engineering issues must be addressed. Most fundamental is the issue of the representational framework. A variety of alternative approaches have been proposed in the controls and computer science communities, including various discrete-event dynamic system formulations (finite state automata, Petri networks, and Markov Decision Processes), synchronous programming languages, and constraint satisfaction logic. Each has certain advantages and disadvantages with regard to the key representational issues of nondeterminism, hybrid discrete/continuous behavior, temporal constraint, and computability. This last area, computability, is particularly critical as the assumption in model-based reasoning systems is that the complexity associated with subsystem interactions (which can scale exponentially with the number of subsystems) must be managed at run-time by the system executive. This rich set of challenges in model-based reasoning, combined with the associated benefits of the technology, is the rationale behind selecting model-based reasoning as one of the two major thrusts in the APL autonomy S&T vision.

Reactive Behavior

Referring back to the conceptual architecture shown in Fig. 4, note that the model-based execution and planning technology we have been discussing must operate in parallel with the reactive control processing that is responsible for maintaining local system stability. In the simplest case, this consists of a real-time control loop as discussed earlier. In a broader sense, reactive processing can include higher-level functionality, even decision logic, to achieve locally optimal (or safe) behavior. The key attribute of processing at the reactive level is speed, which must be subject to hard, deterministic bounds. A classic higher-level function of the reactive layer is vehicle "safing." This generally consists of a periodic test to see that a system telemetry vector lies within a desired operating envelope. If the test fails, the system is immediately commanded to a known safe state. This can be nonoptimal from a global mission planning perspective, but is intended to "buy time" for higher-level reasoning to synthesize a more optimal response/recovery command set.

Researchers have also explored the bounds of reactive control as a general paradigm for autonomy. As noted above, early work in closed-loop control revealed "emergent" behavior that, while not explicitly specified by the system designer, was an appropriate, even seemingly "intelligent," response to the system environment. In the mid-1980s Rodney Brooks proposed the concept of a "subsumption" architecture that could produce a variety of complex autonomous behaviors through a purely reactive control process.[15] Subsumption rejected the use of global system models, focusing instead on the layering and combination of locally optimal controllers (e.g., "drive toward light source," "follow wall edges") to achieve system-level performance goals. Although the approach eliminated many of the difficulties associated with model specification, implementation, and computation, it was ultimately limited in the scope and complexity of behaviors that could be implemented. Despite these limitations, however, reactive autonomous controllers are appropriate for a broad range of system applications (Brooks' company, iRobot, has achieved significant public recognition for recently putting into production an autonomous home vacuum cleaner) and are gaining renewed interest as an approach to autonomous behavior coordination, our last major topic area.

Behavior Coordination

The coordination of individual autonomous systems to accomplish a single goal is another key functional requirement for this technology. Figure 6 shows the set of autonomy levels defined by the Unmanned Aerial Vehicles Roadmap.[16] Notice that higher levels are characterized by coordinated group behavior (note also the exponential technology advancement that is expected within the next 10 years). Similar requirements and expectations can be found in other domains, from underwater glider formations designed to perform environmental characterization, to microsatellite constellations that can form extremely large virtual space sensor apertures. Obviously, behavior coordination represents a critical challenge in autonomous systems and thus has been selected as the second major thrust area within the Laboratory's autonomy S&T vision.

There are many current directions to research in autonomous behavior coordination. At its most abstract, the problem has been investigated within the software "agents" community in terms of communications infrastructure, coordination languages, and formal representation of knowledge for use in those languages. This work has not generally been applied to the coordination of autonomous physical systems but may prove valuable in the future. Distributed control research, in contrast, has focused on more limited functional capabilities (such as relative motion control for vehicle formations), with the traditional emphasis on provable system characteristics.

As with the research in integrated autonomous control systems that we have discussed earlier, distributed autonomy research can be broadly classified as

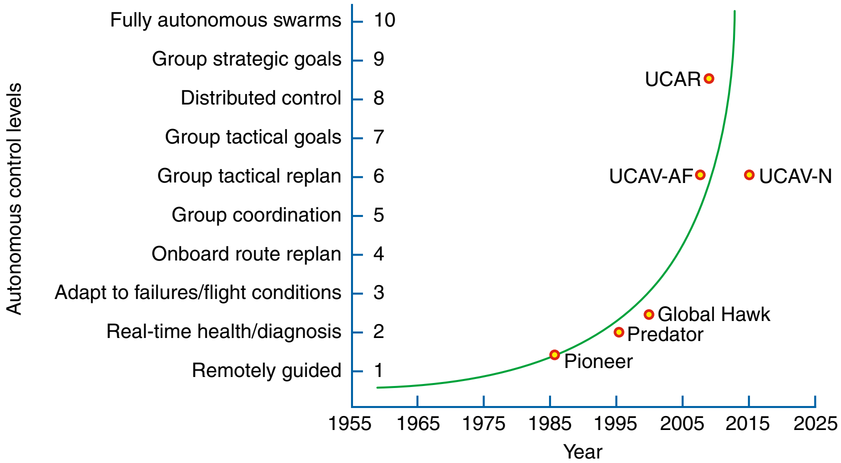


Figure 6. Autonomy levels of UAVs defined by the Office of the Secretary of Defense (UCAR = unmanned combat aerial rotocraft, UCAV = unmanned combat air vehicle).

"model-based" and "reactive." Model-based approaches have been developed, for example, to synthesize the minimal amount of communication required to maintain unambiguous global state knowledge in a set of distributed discrete event controllers.[17] The scaling problems associated with model-based approaches in single-vehicle systems are exponentially worse in distributed multi-vehicle systems, however, and this has motivated an alternative research thrust focused on reactive methods for behavior coordination. In the early 1990s a number of researchers began working with biologically inspired control approaches to behavior coordination. This work has been labeled "swarm intelligence" in reference to the social insect behaviors it seeks to emulate.[18] Classic examples of swarming behaviors include ant foraging, bird flocking, and wolf pack hunting. In each case, researchers have been able to replicate the behavior by using simple reactive control algorithms for each agent, without the requirement for an explicit coordinating plan or global communications. In these cases, the behaviors are generally robust with regard to variations in the environment and failures of individual agents.

The success of these early experiments has motivated current work to codify useful distributed autonomy applications (e.g., search, pursuit, formation flying) in a way that is amenable to solution through swarm intelligence. This work shows significant promise in the near term but will be limited, as all purely reactive methods are, in the scope of behaviors that can be achieved. A critical research challenge in the future will be the integration of these methods with model-based deliberative coordination methods to enable increased operational complexity in addition to robust, reactive behavioral synchronization.

CONCLUSION

We have described a variety of autonomous system domains and discussed some common technology themes across them, highlighting some particular research challenges. Autonomous systems are playing an increasingly important role in both civilian and military applications. The continuing advance of processing, sensing, mobility, and navigation technologies, coupled with the fixed (perhaps increasing) cost and limited availability of human controllers, ensures that requirements for autonomous system control will only increase in the future. Yet today, the technology is relatively immature in real-world application. We have described a number of promising directions and noted that APL S&T development efforts are directed toward several of them. It is the systems engineering aspect of autonomy, however, that makes it a particularly compelling area for APL focus now. The ability to develop innovative operational concepts based on a deep understanding of the available technology, the definition of development requirements from those concepts, and the ability to perform rigorous test and evaluation of the resulting systems are all areas that leverage historical APL strengths, and all are critical challenges in maturing this transformational technology.

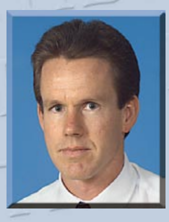
REFERENCES

1. Cohen, J., *Human Robots in Myth and Science*, Allen and Unwin, London (1966).
2. Wood, G., *Living Dolls: A Magical History of the Quest for Mechanical Life*, Faber and Faber, London (2002).
3. Heims, S. J., *John von Neumann and Norbert Wiener: From Mathematics to the Technologies of Life and Death*, MIT Press, Cambridge, MA (1980).
4. Navy UUV Master Plan, http://www.npt.nuwc.navy.mil/UUV/UUVMP.pdf.
5. Department of Defense Joint Robotics Program, http://www.jointrobotics.com/activities_new/masterplan.shtml.
6. Weld, D., "Recent Advances in AI Planning," *AI Mag.* **20**(2), 93–123 (1999).
7. Puterman, M., *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, Wiley, New York (1994).
8. Bylander, T., "The Computational Complexity of Propositional STRIPS Planning," *Artif. Intell.* **69**, 165–204 (1994).
9. Gat, E., "On Three-layer Architectures," in *Artificial Intelligence and Mobile Robots*, AAAI Press, pp. 195–210 (1997).
10. Albus, J., and Meystel, A., "A Reference Model Architecture for Design and Implementation of Intelligent Control in Large and Complex Systems," *Int. J. Intell. Control Syst.* **1**(1), 15–30 (1996).
11. Davis, R., "Diagnostic Reasoning Based on Structure and Behavior," *Artif. Intell.* **24**, 347–410 (1984).
12. Muscelttola, N., Nayak, P., Williams, B., and Pell, B., "Remote Agent: To Boldly Go Where No AI System Has Gone Before," *Artif. Intell.* **103**, 5–47 (1998).
13. Dvorak, D., Rasmussen, R., and Starbird, T., "State Knowledge Representation in the Mission Data System," in *Proc. IEEE Aerospace Conf.*, Vol. 1, IEEE Press, pp. 195–210 (2002).
14. Williams, B., "Model-based Programming of Intelligent Embedded Systems and Robotic Space Explorers," *Proc. IEEE* **91**(1), 212–237 (Jan 2003).

[15]Brooks, R. A., *A Robust Layered Control System for a Mobile Robot*, Tech. Rep. A. I., Memo 864, MIT, Artificial Intelligence Laboratory (Sep 1955).
[16] Office of the Secretary of Defense, *Unmanned Aerial Vehicles Roadmap 2002–2027*, Washington, DC (Dec 2002).
[17]Barrett, G., and Lafortune, S., "Decentralized Supervisory Control with Communicating Controllers," *IEEE Trans. Autom. Control* **45**(9), 1620–1638 (2000).
[18]Bonabeau, E., Dorigo, M., and Theraulaz, G., *Swarm Intelligence*, Oxford University Press, New York (1999).

THE AUTHORS

David P. Watson and **David H. Scheidt** jointly lead APL's science and technology thrust in autonomous systems. Mr. Watson joined APL in 1995 and is the Supervisor of the Systems and Information Sciences Group in the Milton S. Eisenhower Research and Technology Development Center. He holds an M.A. in applied mathematics from California State University and has performed research and development in autonomous systems for underwater, air, and space applications. Mr. Scheidt is the Group Scientist for Information Technologies in the National Security Technology Department. He joined APL in 1999. He holds a B.S. in computer engineering from Case Western Reserve University and has performed research and development in autonomous systems for air, sea surface, ground, space, and railroad applications for the Office of Naval Research, DARPA, and Class I Railroads. Further information on autonomy research at APL can be obtained from Mr. Watson. His e-mail address is david.watson@jhuapl.edu.



David P. Watson

David H. Scheidt