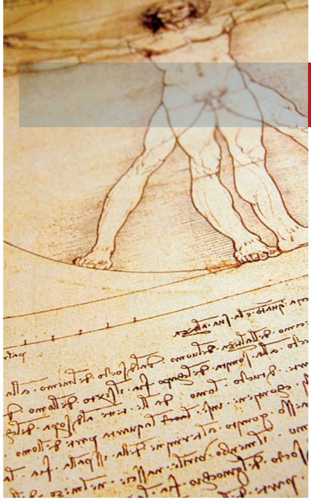

Editor: **Robert R. Hoffman, Jeffrey M. Bradshaw, and Kenneth M. Ford,**
Institute for Human and Machine Cognition, rhoffman@ihmc.us

Beyond Cooperative Robotics: The Central Role of Interdependence in Coactive Design

Matthew Johnson, Jeffrey M. Bradshaw, Paul J. Feltovich, and Robert R. Hoffman,
Institute for Human and Machine Cognition
Catholijn Jonker and Birna van Riemsdijk, *Delft University of Technology*
Maarten Sierhuis, *Xerox PARC*

As automation becomes more sophisticated, the nature of its interaction with people will need to change in profound ways. Inevitably, software and robotic agents will become so capable that they will function less like tools and more like teammates.[1–3]

Many approaches to designing more team-like cooperation between humans and machines have been proposed, including function allocation, supervisory control, adaptive automation, dynamic task allocation, adjustable autonomy, mixed-initiative interaction—most recently regrouped under the rubric of *cooperative robotics*. All these approaches rely on the levels of autonomy concept as the benchmark for machine performance and the criterion for decisions about human-machine task allocation.

In this article, we argue that the concept of levels of autonomy is incomplete and insufficient as a model for designing complex human-machine teams, largely because it does not sufficiently account for the interdependence among their members. Building on a theory of joint activity,[4,5] we introduce the notion of coactive design,[6] an approach to human-machine interaction that takes interdependence as the central organizing principle among people and agents working together as a team.

What Is Autonomy?

The word *autonomy*, derived from a combination of Greek terms signifying self-government (*auto* means self, and *nomos* means law), today has two basic senses in everyday use.[7] The first sense, self-sufficiency, is about the degree to which an entity operates without outside help. For example, a Roomba robot can vacuum a room without assistance. The second sense refers to an entity's self-directedness, or the degree of freedom from outside control. The Mars Rover, which was tightly controlled by NASA engineers, is such as example.

In our discussion, we will use the terms self-sufficiency and self-directedness to distinguish between these two senses of autonomy.

Pervasiveness of the Levels of Autonomy Concept

The concept of levels of autonomy is usually attributed to the pioneering work of Thomas Sheridan and William Verplank.[8] Their ideas were derived from a teleoperation study with underwater robots. Although the original 1978 work is often cited, the original three page table is usually condensed and simplified as shown in Table 1. The "levels" were used to describe the space of design options, as they saw them. They range from tedious and error-prone manual operation, where humans are required to do everything (level 1), to fully autonomous operations, where the machine can perform the entire task without assistance or direction (level 10). Sheridan and Verplank realized the unlikelihood of achieving a completely autonomous solution because they "simply [did] not

1541-1672/11/$26.00 © 2011 IEEE
Published by the IEEE Computer Society

Table 1. Levels of automation.*

Level	Description
High	10. The computer decides everything, acts autonomously, ignoring the human.
	9. The computer informs the human only if it, the computer, decides to.
	8. The computer informs the human only if asked, or
	7. The computer executes automatically, then necessarily informs the human, and
	6. The computer allows the human a restricted time to veto before automatic execution, or
	5. The computer executes that suggestion if the human approves, or
	4. The computer suggests one alternative
	3. The computer narrows the selection down to a few, or
	2. The computer offers a complete set of decision/action alternatives, or
Low	1. The computer offers no assistance; the human must take all decisions and actions.

*Adapted from an earlier work.[11]

have available at [that] time such devices or the understanding to build such devices" (p. 1–10) for their demanding environment. Given this realization, they suggested two things:

- levels of automation as a means to gain some of the benefits of autonomy while not requiring a fully autonomous solution and
- supervisory control, in which humans allocate tasks to one or more machines and then monitor them.

For the second suggestion, once control is given to the machine, it is ideally expected to complete the tasks without human intervention. The job of the machine's designer is to determine what needs to be done and then provide the capability (self-sufficiency) for the machine to do it. This is often described as finding the appropriate level of autonomy.

Although the supervisory-control approach fulfilled its initial purpose, its static nature did not address requirements for variable task allocation in different situations, which spurred interest in research on dynamic and adaptive function allocation. Dynamic interaction of this sort has been suggested as a unifying theme in human-robot interaction[9] and has led to numerous proposals for dynamic adjustment of autonomy level[10]—in this case, the self-directedness aspect. Such approaches have been variously called adjustable autonomy, dynamic task allocation, sliding autonomy, flexible autonomy, and adaptive automation. In each case, the system must decide at runtime which functions to automate and to what level of autonomy.[11]

Mixed-initiative interaction is defined as "a flexible interaction strategy, where each agent can contribute to the task what it does best" (p. 14).[12] Its contribution is in the perspective that people can work in parallel alongside autonomous systems, so it adopts the stance that the perception, problem-solving, and task-execution processes are subject to an ongoing give and take that can be initiated by either the human or the machine, rather than explicitly determined by the original system designer. Although it is more sophisticated in some ways than function allocation, in practice this approach still tends to be autonomy-centric, focusing on fluid management of task assignment and the authority to act—the self-directedness aspect of autonomy. The influence of the levels of autonomy concept is apparent in James Allen's proposal for mixed-initiative interaction levels.[12]

The classic Sheridan-Verplank levels are widely cited and have had a significant impact on the outlook of robot designers. A recent survey of human-robot interaction concluded that "perhaps the most strongly human-centered application of the concept of autonomy is in the notion of level of autonomy" (p. 217).[9] This seems counterintuitive. Why should the independence of a given robotic partner play a more dominant role in human-centered design of joint activity than the interdependence among the set of human-robotic team members?

Problems with the Levels of Autonomy Concept

Significant nuances in the original Sheridan-Verplank work have been forgotten through frequent use of the simplified list shown in Table 1. As a basis for our discussion, Figure 1 illustrates the richer detail in the original work. In this excerpt from the complete model, we have altered Sheridan's level 6 by adding the tell functions and associated text from level 8. We did this to incorporate all the basic elements in a single level for discussion purposes, but it does not significantly alter the original intention because the original table had a footnote indicating other possible variations.

The first column is the description that corresponds to an item on the simplified version of the list from Raja Parasuraman, Thomas Sheridan, and Christopher Wickens.[11] The second column represents the human functions in the activity and the third represents the functions the computer performs. Interestingly, arrows were used between the second and third columns in the original work, creating a small causal diagram. This representation more clearly shows that two parties are involved in the activity, as opposed to the list in Table 1, which focuses solely on the computer. Additionally, these arrows represent a workflow with dependencies connecting the functions. Insightfully, Sheridan and Verplank understood that even their original richer description had limitations and stated that "as computer control and artificial intelligence become more sophisticated, certain human functions in teleoperation may be replaced, but greater need and demand will be placed upon other human functions, and in these respects the need for improved man-computer interaction will increase, not diminish" (p. 1–10).[8]

With this in mind, we have outlined several problems with the simplified concept of levels of autonomy as it is usually formulated.

Problem 1: Functional Differences Matter

There are significant differences between performing an action and making a decision as well as between different kinds of actions. Sheridan and Verplank's original work provided a table of behavior elements that can be used to characterize a system. Their list included request options, get options, select action, approve action, start action, and tell functions. In this regard, the original levels model mixes apples and oranges—task work and teamwork. For example, in their level 1, the human handles the entire task without automation by performing the get options, select action, and start action functions. These are task-work components. On the other hand, the request options, approve action, and tell elements engage both parties in a simple form of teamwork.

The model also mixes reasoning (get options), decisions (select action), and actions (start action). Moreover, the entire approach reinforces the erroneous notion that "automation activities simply can be substituted for human activities without otherwise affecting the operation of the system."[13]

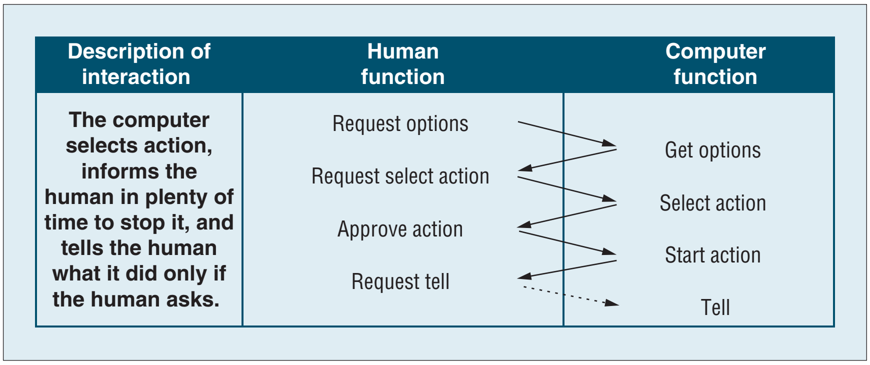

Figure 1. Altered excerpt of Sheridan-Verplank's level 6 automation. Our goal was to incorporate all the basic elements in a single level for discussion purposes and more clearly show that two parties (computer and human) are involved in the activity. The solid arrows depict hard constraints that enable or prevent the possibility of an activity. The dashed arrow indicates soft interdependence, which includes optional commands. (Adapted from an earlier work.[8])

Parasuraman, Sheridan, and Wickens' work attempted to address some of these problems by associating activity types with the 10 levels.[11] They proposed four types (acquisition, analysis, decision, and action), but this merely highlights the importance of functional differences between the elements and ignores the issues of interdependence relating to such activities.

Problem 2: Levels Are Neither Ordinal nor Representative of Value

Another problem is that the term *level* implies an ordinal relationship. Authors who reproduce the condensed version often add the low and high labels to levels 1 and 10, respectively, as in our Table 1. These labels imply that the levels are of increasing autonomy, but are they really? The get options function seems like a lower level of autonomy than the select option. However, if the "getter" of the options can filter the options and the receiver has no other means to know what the options are, is it really a lower level? Who holds the power in this relationship? Which has a higher value: a start action or tell? It probably depends on the criticality of what is being started and the importance of what is being told. For these and other reasons, it is more productive to think about autonomy in terms of multiple task-specific dimensions rather than in terms of a single, unidimensional scale.[7]

The perspective in which we view a system can also affect our assessment of autonomy. For example, ambiguity about the term autonomy comes into play in Figure 1. Because the level shown is six out of 10, we could consider the machine semiautonomous—that is, at a mid-level of autonomy. However, with respect to the self-sufficiency perspective on autonomy, the machine could be viewed instead as fully autonomous because it can perform all aspects of the task work. On the other hand, from a self-directedness perspective, a machine functioning at this level would have no autonomy since the performance of its task work is completely subject to the direction and initiative of the human.

Our assessment of a system's autonomy also depends on the way we define the boundaries of its sphere of action. Consider the vehicles that competed in the DARPA Urban Challenge, which were designed to find their way over a given course in "fully autonomous" fashion. Although fully autonomous with respect to this one particular task, they might be far from autonomous with respect to related tasks, such as going to the store and getting groceries.

This also applies in the other direction. Several entries in the Urban Challenge were unsuccessful at completing the task but were successful at aspects of the task. For example, some could follow the road but not deal with traffic. These might be called semiautonomous, but all this term tells us is that the machine could not do everything on its own. If we redefine the task as something simpler, such as following a road without traffic, then we could once again describe the car as fully autonomous. In fact, virtually any machine could be considered fully autonomous if we define the grain size of its task to be sufficiently small. These examples make it obvious that the property of autonomy is not a mere function of the machine, but rather a relationship between the machine and a task in a given situation.

Problem 3: Autonomy Is Relative to the Context of the Activity

Autonomous capabilities are relative to the context of the task for which they were designed. When a designers consider what level of autonomy is appropriate, they are assuming some level of granularity and using that to define activity boundaries. Sheridan and Verplank's original table title was "Levels of automation in man-computer decision making for a single elemental decisive step." In other words, level 10 represents full autonomy relative to the single elemental decisive step or activity. Unfortunately, over time researchers have generalized this to all activity in complex systems involving teams of humans and machines. This goes far beyond the original scope and might explain Sheridan's comment that "surprisingly, the level descriptions as published have been taken more seriously than were expected" (p. 206).[14]

Functions are not automated in isolation from task context. Therefore, when system designers automate a subtask, they are really performing a type of task distribution and, as such, have introduced novel elements of interdependence within the system. This is the lesson to be learned from studies of the *substitution myth*,[13] which states that reducing or expanding the role of automation in joint human-automation systems can change the nature of interdependent and mutually adapted activities in complex ways. To effectively exploit automation's capabilities (versus merely increasing automation), we must coordinate the task work—and the interdependence it induces among players in a given situation—as a whole.

As an example, consider the major assumption underlying the Sheridan-Verplank levels that the human, in a supervisory role, is the initiator of the activity and has an implied obligation to monitor the activity. Although this is not explicit in the model, it can be derived from the fact that the request options action is only available to the human and that the tell option is only available to the computer. Roles are not simple titles; rather they are mechanisms by which we describe capabilities and their interdependence.

Problem 4: Levels of Autonomy Encourage Reductive Thinking

Previous essays in this department have raised the issue of "keeping things too simple" in the design of cognitive systems.[15] The levels of autonomy concept demonstrates several of these oversimplifications. Some have already been mentioned, such as ignoring functional differences, which could include treating heterogeneous elements as homogeneous and ignoring task context. Another problem is the tendency to view activity as sequential when it is actually simultaneous. Although task work often entails sequential dependencies and can be reasonably decomposed by looking at individual capabilities, we cannot uniquely describe or design teamwork in this way. Teamwork is necessarily based on the interaction among the participants, whereas a simplifying notion of levels treats elements as cleanly separable.

Using Figure 1 as an example again, there seems to be a sequential ordering of the task elements. This might be appropriate for some tasks but not in general. Most teamwork occurs concurrently. Looking at the description of level 6 in the first column of Figure 1, it includes the phrase "informs the human in plenty of time to stop it." This implies the human is concurrently monitoring and assessing the computer's activity on some level. It would also suggest the need for a stop function, although none is included. The simplification here might explain the apparent oversight of including a stop behavioral element, and it is indicative of the problems faced when using a model with a solitary focus on levels of autonomy.

Problem 5: The Levels of Autonomy Concept Is Insufficient to Meet Future Challenges

Many of the challenges facing designers are related to teamwork. An earlier article in this department proposed 10 challenges for making automation a "team player."[5] These challenges include directability, transparency, and predictability. These challenges deny the intrinsic validity of any levels of autonomy concept. Each of these challenges must be addressed not by making the machines more independent, but by making

Things the human does	Things the human depends on the computer for	Things the computer depends on the human for	Things the computer does
Request options	Computer status and availability		
		Initiation by request Environment status Supervisor status	Get options
Request select action	A list of options Factors effecting option selection Ranking of options		
		Initiation by request Preferences	Select action
Approve action	An action to approve Reasoning behind decision Confidence level		
		Approval Constraints on activity	Start action
Request tell	Action status Environment status Progress appraisal		
		What to tell When to tell Who to tell How to tell	Tell

Figure 2. Example of an interdependence analysis based on the Figure 1 example. We added some potential interdependence and allow the sequential-work-flow assumption to persist only to maintain consistency in the discussion. The solid arrows depict hard constraints, and the dashed arrow indicates soft interdependence. (Adapted from an earlier work.[8])

them more capable of supporting system interdependence.

Many supportive behaviors are what might be called soft system constraints and are not essential to task completion—that is, although the performer is, strictly speaking, self-sufficient, it can benefit from support. Joint activity is not exclusively about the hard constraints that enable or prevent the possibility of an activity, as the solid arrows in Figure 1 depict. Joint activity also includes soft interdependence, which includes optional commands, such as the ability to request the final status of the action (see the dashed arrow in Figure 1). Soft interdependence also includes helpful things that a participant might do to facilitate team performance. For example, team members can signal progress appraisals[16] ("I'm running late"), warnings ("Watch your step"), helpful adjuncts ("Do you want me to pick up your prescription when I go by the drug store?"), and observations about relevant unexpected events ("It has started to rain").

Our observations suggest that good teams can be distinguished from great ones by how well they support requirements arising from soft interdependence. Although social science research on teamwork indicates it as an important factor in team performance,[17] interdependence (particularly soft interdependence) has not received adequate attention in the research literature.[6]

Teamwork is largely about enhancing each member's performance, not merely effective task distribution. In response to the MABA-MABA (men-are-better-at/machines-are-better-at) Fitts' List model,[18] an alternative human-centered view was expressed in this department as the Un-Fitts List.[19,20] The intent was to emphasize the ways in which people and machines cannot simply divide up the work, but rather mutually enhance their competencies and mitigate their limitations. Such a view is consistent with our view of interdependence and its role in design.

Consider the hypothetical level 6 in Figure 1. If we consider the interdependence in the activity, we can concoct a table patterned after the Sheridan-Verplank levels of automation but based on the Un-Fitts List (see Figure 2). We have added some potential interdependence that might be appropriate for such an activity. We allow the sequential-work-flow assumption to persist only to maintain consistency in the discussion. The focus of Figure 2 is the diversity of interdependence among the activities.

Although we apply this process to a single level within the original Sheridan-Verplank list here, it can be applied to any of the levels with different results, based on the varying interdependence within the activity. If we move beyond the single decisive element portrayed by the Sheridan-Verplank list toward activity to support the future envisioned roles, the interdependence become much more complex and generating such a table becomes even more interesting. Such a construction calls out the ways in which changes to the level of autonomy affect interdependence and how the interdependence affects the total work system. Levels by themselves do not provide this information, which leads to the next problem.

Problem 6: Levels Provide Insufficient Guidance to the Designer

Levels of autonomy do not provide principles or guidelines for designers as they build human-machine systems. Previous articles have discussed the challenge of bridging the gap from cognitive engineering products to software engineering.[21] The levels of autonomy concept provides no assistance here. Parasuraman, Sheridan, and Wickens suggested using levels of autonomy in combination with human performance as an evaluative criterion for automation design.[11] Although we agree that human-performance measures are important and useful, it is unclear what value the descriptive levels of autonomy provide other than as a labeling mechanism. They provide no assistance to the designer, whose only option is to build it and try it, then build something else and compare the results.

Interdependence, however, affords a great deal of predictive power. It can inform the designer of what is and is not needed, what is critical, and what is optional. Most importantly, it can indicate how changes in capabilities affect relationships.

This extends the human-centered approaches where designers typically ask, "How can we keep the human in the loop?" or "How do we reduce the burden on the human?" These types of questions lead designers to focus on usability issues. Understanding the interdependence in the human-machine system in the context of the anticipated activity can provide a wealth of guidance to a designer. In fact, we posit that it is through understanding the dynamic interdependence within the macrocognitive work that the system developer can answer such questions as "What should be automated?" and "How do we reduce the burden on the human?" More importantly, it has the potential to answer richer questions, such as "How will this change affect the work system?"

As an example, consider our level 6 in Figure 1. What is the impact of allowing the computer to move from the get options to select action functions without requiring the human request function? Here, some amount of risk analysis might be required to assess the consequences of leaving it completely to the system. Making this change might enable a higher level of autonomy, but is it better? How does it affect the system?

Now look at Figure 2. Identifying the interdependence suggests several impacts. Not only does allowing the computer to select the action reduce the directability of the automation by eliminating the computer's dependence on the human to initiate action selection, it also reduces transparency because the human no longer has access to the options. Both of these limit the work system's ability to leverage the human's ability to improve the overall work system's effectiveness.

Coactive design takes interdependence as the central organizing principle among people and agents working together as a team.

Toward Coactive Design

Building on the theory of joint activity,[4,5] we are working on a *coactive design approach*[6] that is intended to provide prescriptive guidance to designers of sophisticated human-machine systems. Coactive design takes interdependence as the central organizing principle among people and agents working together as a team. The approach also embraces the idea that effective coordination in human-machine activity has much to learn from the various forms of social regulation that enable people to work well together.[22]

Besides implying that two or more parties are participating in an activity, the term coactive is meant to convey the reciprocal and mutually constraining nature of actions and effects that are conditioned by coordination. In joint activity, individual participants share an obligation to coordinate, to a degree sacrificing their individual autonomy in the service of progress toward group goals.

By its nature, joint activity implies the greater parity of mutual assistance, enabled by intricate webs of complementary, reciprocal affordances, and obligations. Thus, coactive design considers the mutual interdependence of the all parties instead of merely focusing on the dependence of one of the parties on the other. It recognizes the benefits of designing agents with the capabilities they need to be interdependent.

As we try to design more sophisticated human-machine work systems, we move along a maturity continuum from dependence to independence to interdependence. The process is a continuum because a small level of agent independence through autonomy is a prerequisite for interdependence. However, independence is not the supreme achievement in human-human interaction,[23] nor should it be in human-machine systems. Imagine a completely capable, autonomous human possessing no skills for coactivity—how well would such a person fit in most everyday situations?

This maturation process cannot only be seen in individual systems but also in the human-machine systems field as a whole. Consider the history of unmanned aerial vehicle (UAV) R&D. The first goal in development was a standard engineering challenge to make the UAV self-sufficient for some tasks (such as stable flight and waypoint following). As the capabilities and robustness increased, the focus shifted to the problem of self-directedness by the machine ("What am I willing to let the UAV do autonomously?"). The future developments of UAVs suggest yet another shift, as discussed in the "Unmanned Systems Roadmap,"[24] which states that unmanned systems "will quickly evolve to the point where various classes of unmanned systems operate together in a cooperative and collaborative manner" (p. 2). This requires a focus on interdependence ("How can I get multiple UAVs to work effectively as a team with their operators?").

This progression of development is a natural maturation process that applies to any form of sophisticated automation. Awareness of interdependence was not critical to the initial stages of UAV development, but it becomes an essential factor in realizing a system's full potential.

We believe that increased effectiveness in human-agent teamwork hinges not merely on trying to make machines more independent through their autonomy, but also in striving to make them better team players[5] by making them more capable of sophisticated interdependent joint activity with people.

References

1. J.M. Bradshaw, P.J. Feltovich, and M. Johnson, "Human-Agent Interaction," *Handbook of Human-Machine Interaction*, G. Boy et al., eds., Ashgate, 2011, pp. 283–302.
2. C. Breazeal et al., "Social Robots: Beyond Tools to Partners," *Proc. 13th IEEE Workshop Robot and Human Interactive Communication*, IEEE Press, 2004, pp. 551–556.
3. T.W. Fong, *Collaborative Control: A Robot-Centric Model for Vehicle Teleoperation*, tech. report, Robotics Inst., Carnegie Mellon Univ., 2001.
4. G. Klein et al., "Common Ground and Coordination in Joint Activity," *Organizational Simulation*, W.B. Rouse and K.R. Boff, eds., John Wiley, 2004, pp, 139–184.
5. G. Klein et al., "Ten Challenges for Making Automation a 'Team Player' in Joint Human-Agent Activity," *IEEE Intelligent Systems*, 2004, vol. 19, no. 6, pp. 91–95.
6. M. Johnson et al., "Coactive Design," to be publishing in *Coordination, Organisation, Institutions and Norms 2010*, M.D. Vos et al., eds., Springer, 2012.
7. J.M. Bradshaw et al., "Dimensions of Adjustable Autonomy and Mixed-Initiative Interaction," *Agents and Computational Autonomy*, M. Klusch, G. Weiss, and M. Rovatsos, eds., Springer, 2004, pp. 17–39.
8. T.B. Sheridan and W. Verplank, *Human and Computer Control of Undersea Teleoperators*, tech. report, Man-Machine Systems Lab., Dept. of Mechanical Eng., Massachusetts Inst. of Technology, 1978.
9. M.A. Goodrich and A.C. Schultz, "Human-Robot Interaction: A Survey," *Foundations and Trends in Human-Computer Interaction*, vol. 1, no. 3, 2007, pp. 203–275.
10. P.A. Hancock and S.F. Scallen, "Allocating Functions in Human-Machine Systems," *Viewing Psychology as a Whole*, R.R. Hoffman, M.F. Sherrick, and J.S. Warm, eds., Am. Psychological Assoc., 1998, pp. 509–539.
11. R. Parasuraman, Y. Sheridan, and C. Wickens, "A Model for Types and Levels of Human Interaction with Automation," *IEEE Trans. Systems, Man and Cybernetics, Part* A, vol. 30, no. 3, 2000, pp. 286–297.
12. J.F. Allen, "Mixed-Initiative Interaction," *IEEE Intelligent Systems*, vol. 14, no. 5, 1999, pp. 14–16.
13. K. Christoffersen and D.D. Woods, "How to Make Automated Systems Team Players," *Advances in Human Performance and Cognitive Engineering Research*, vol. 2, E. Salas, ed., Elsevier Science, 2002, pp. 1–12.
14. T.B. Sheridan, "Function Allocation: Algorithm, Alchemy or Apostasy?" *Int'l J. Human-Computer Studies*, vol. 52, no. 2, 2000, pp. 203–216.
15. P.J. Feltovich, R.R. Hoffman, and D.D. Woods, "Keeping it Too Simple: How the Reductive Tendency Affects Cognitive Engineering," *IEEE Intelligent Systems*, vol. 19, no. 3, 2004, pp. 90–95.
16. P.J. Feltovich et al., "Progress Appraisal as a Challenging Element of Coordination in Human and Machine Joint Activity," *Engineering Societies for the Agents World VIII*, A. Artikis et al., eds., Springer, 2007, pp. 124–141.
17. E. Salas, N.J. Cooke, and M.A. Rosen, "On Teams, Teamwork, and Team Performance: Discoveries and Developments," *Human Factors*, vol. 50, no. 3, 2008, pp. 540–547.
18. P.M. Fitts et al., *Human Engineering for an Effective Air-Navigation and Traffic-Control System*, Committee on Aviation Psychology, Nat'l Research Council, 1951.
19. R.R. Hoffman et al., "A Rose by Any Other Name … Would Probably Be Given an Acronym," *IEEE Intelligent Systems*, vol. 17, no. 4, 2002, pp. 72–80.
20. D.D. Woods, "Steering the Reverberations of Technology Change on Fields of Practice: Laws that Govern Cognitive Work," *Proc. 24th Ann. Meeting of the Cognitive Science Soc.*, Erlbaum, 2002, pp. 14–16.
21. R.R. Hoffman, "Influencing versus Informing Design, Part 2: Macrcognitive Modeling," *IEEE Intelligent Systems*, vol. 23, no. 6, 2008, pp. 86–89.
22. P. Feltovich et al., "Social Order and Adaptability in Animal and

Human Cultures as an Analogue for Agent Communities: Toward a Policy-Based Approach," *Engineering Societies in the Agents World IV*, LNAI 3071, Springer-Verlag, 2004, pp. 21–48.
23. S.R. Covey, *The Seven Habits of Highly Effective People*, Free Press, 1989.
24. "Unmanned Systems Roadmap: 2007–2032," memo, US Office of the Secretary of Defense, 10 Dec. 2007.

Matthew Johnson is a researcher at the Institute for Human and Machine Cognition and a PhD student at the Delft University of Technology, The Netherlands. Contact him at mjohnson@ihmc.us.

Jeffrey M. Bradshaw is a senior research scientist at the Institute for Human and Machine Cognition. Contact him at jbradshaw@ihmc.us.

Paul J. Feltovich is a research scientist at the Institute for Human and Machine Cognition. Contact him at pfeltovich@ihmc.us.

Robert R. Hoffman is a senior research scientist at the Institute for Human and Machine Cognition. Contact him at rhoffman@ihmc.us.

Catholijn Jonker is the head of the Man Machine Interaction Group of the Department of Mediametics at the Delft University of Technology, The Netherlands. Contact her at C.M.Jonker@tudelft.nl.

Birna van Riemsdijk is an assistant professor in the Man Machine Interaction Group of the Department of Mediametics at the Delft University of Technology, The Netherlands. Contact her at m.bvanriemsdijk@tudelft.nl.

Maarten Sierhuis is area manager in the Intelligent Systems Laboratory at the Palo Alto Research Center. Contact him at Maarten.Sierhuis@parc.com.