MABA-MABA or Abracadabra?
Progress on human-automation coordination

Sidney W. A. Dekker*
Department of Mechanical Engineering, IKP
Linköping Institute of Technology
SE - 581 83 Linköping, Sweden
+46 - 13 - 281646 (phone)
+46 - 13 - 282579 (fax)
sidde@ikp.liu.se

David D. Woods
Cognitive Systems Engineering Laboratory
The Ohio State University
210 Baker Systems
1971 Neil Avenue
Columbus, OH 43210, USA
+1 - 614 - 292 1700
woods.2@osu.edu

*corresponding author

MABA-MABA or Abracadabra?
Progress on human-automation coordination

Abstract

In this paper we argue that substitution-based function allocation methods (such as MABA-MABA, or Men-Are-Better-At/Machines-Are-Better-At lists) cannot provide progress on human-automation coordination. Quantitative "who does what" allocation does not work because the real effects of automation are qualitative: it transforms human practice and forces people to adapt their skills and routines. Rather than re-inventing or refining substitution-based methods, we propose that the more pressing question on human-automation coordination is "how do we make them get along together".

Keywords: automation, cognition, computers, function allocation, human error, coordination

1. What should we automate?

There was a time when the question of what to automate had a simple answer: Automate everything you technically can (Chapanis, 1970; and see also Douglas, 1990). In one domain, commercial aviation, the Air Transport Association observed that "during the 1970's and early 1980's...the concept of automating as much as possible was considered appropriate" (ATA, 1989, p. 4). Chapanis (1970) even suggested that it is imperative to the engineering profession to aim to mechanize or automate everything. Not everyone subscribes to this of course, not least the users or consumers of the systems involved. Technologist attempts to automate everything has led to both predictable and unanticipated consequences in aviation and elsewhere (see Wiener, 1989; FAA, 1996; Billings, 1996; Moray, 1997; Sarter & Woods, 1997; Dekker & Hollnagel, 1999; Mouloua & Scerbo, 1999). The question of what to automate has no simple answer.

Engineers and others involved in automation development, however, are still led to believe that there *is* a simple answer, and in fact a simple way of getting the answer. MABA-MABA lists, or "Men Are Better At-Machines Are Better At" lists have appeared over the decades in various guises (e.g. Chapanis, 1965; Edwards & Lees, 1972; Mertes & Jenney, 1974; Swain & Guttman, 1980; Sheridan, 1987; Douglas Aircraft Co., 1990). Such work relies on a presumption of fixed human and machine strengths and weaknesses, and suggest an often quantitative division of work (you do this much, I do this much). Similar design guidance keeps appearing. For example Parasuraman, Sheridan & Wickens (2000) propose that designers divide the tasks between humans and machines by considering four different groups of system functions:

- information acquisition
- information analysis
- decision and action selection
- action implementation

A simple flow chart is presented that takes the engineer from the question "what should be automated" to an identification of the types of automation (a choice from the four functions above). Subsequently the engineer can choose from a list of levels of automation (see table 1 and also Sheridan, 1987).

----
insert table 1 about here

Choices among kinds and levels of automation are presented to the engineer as if inherently non-problematic and self-evident, fuelling the Abracadabra dream of MABA-MABA methods: put your allocation problem into our method, and the solution will emerge from the other end. Then, in classic MABA-MABA style, the advantages and disadvantages for automating parts of each of the four functions are discussed. For example, automating parts of decision and action selection and action implementation may create problems of "operator complacency" while automating information acquisition and analysis likely produces problems for "operator situation awareness" (Parasuraman *et al.*, 2000). These large motherhood labels for possible human performance decrements, know little or no consensus in the human factors community. Indeed, agreement on, or specification of what these constructs exactly mean is far off , and such terms are instead often based on folk models of human performance. They give the psychologically uninitiated (indeed the engineer who uses the method) an illusion of understanding; a mere fallacy of deeper access to human performance issues associated with his or her design.

2. "Function allocation by substitution"

2.1. The substitution myth

There are many problems with the beliefs (and the consequent messages sent to engineers) that are sustained in MABA-MABA-like methods. One problem is that the level of granularity of functions to be considered for function allocation is arbitrary. For example, it depends on the model of information processing that underlies the MABA-MABA method is based (Hollnagel, 1999). In Parasuraman *et al*. (2000), the four stages of information processing reflect such an arbitrary decomposition (acquisition, analysis, selection, response). This particular decomposition, common in MABA-MABA methods, is a retread of ancient beliefs in which the human-machine ensemble resembles a linear input-output device that takes in an impoverished stimulus from the world and converts it into a response by adding meaning through various distinct processing stages along the way (cf. Neisser, 1976; Flach, 2000). In cases where it is not a model of information processing that determines the categories of functions to be swapped between human and machine, the technology itself often determines it (Hollnagel, 1999). MABA-MABA attributes are then cast in mechanistic terms, derived from technological metaphors. For example, Fitts (1951) applies terms such as "information capacity" and "computation" in his list of attributes for both the human and the machine. If the technology gets to pick the battlefield (i.e. determine the language of attributes) it will win most of them back for itself. This results in human-uncentered systems where typically heuristic human abilities (filtering irrelevant information, scheduling and reallocating activities to meet current constraints, anticipating events, making generalisations and inferences, learning from past experience, collaborating) easily fall by the wayside, misleading the designer who relies on the list (Norman, 1990; Hollnagel, 1999).

MABA-MABA lists practice what Hollnagel (1999) calls "function allocation by substitution". They foster the idea that new technology can be introduced as a simple substitution of machines for people—preserving the basic system while improving it on some output measures (lower workload, better economy, fewer errors, higher accuracy, etc.). Indeed, Parasuraman *et al.* (2000) define automation precisely in this sense: "automation refers to the full or partial replacement of a function previously carried out by the human operator" (p. 287). Sheridan's list of levels of automation support (see Table 1) completes the quantitative substitution argument by listing the complimentary degrees to which machines and people make contributions to system processing and output, where human and automation control over the process becomes symmetrically apportioned as viewed from top to bottom. The list of levels indicates the varying degrees of possible supervisor involvement and alludes to the nature of the human task at each of the levels. But neither the list nor much of the accompanying supervisory control literature explains the cognitive work that might be involved in deciding how and when to intervene or how to switch from level to level. The list of supervisory control levels leaves unspecified how humans should decide when and whether to intervene or when to back off (Dekker & Woods, 1999).

2.2. Qualitative effects

Behind the substition myth (and any MABA-MABA list) lies the false idea that people and computers have fixed strengths and weaknesses and that the task of an engineer is to capitalize on the strengths while eliminating or compensating for the weaknesses. Capitalizing on some strength of automation does not replace a human weakness. It creates new human strengths and weaknesses—often in unanticipated ways (Bainbridge, 1987). For instance, the automation strength to carry out long sequences of action in pre-determined ways without performance degradation (because of fatigue), amplifies classic and well-documented human vigilance problems (e.g. Broadbent, 1958). It also exacerbates the system's reliance on the human strength to deal with the parametrization problem (automation does not have access to all relevant world parameters for accurate problem solving in all possible contexts), but systems may be hard to direct even if the human knows what s/he wants it to do (Billings, 1996; Sarter & Woods, 1997). In addition, allocating a particular function does not absorb this function into the system without further consequences. It creates new functions for the other partner in the human-machine equation—functions that did not exist before, for example typing, or searching for the right display page. The quest for *a-priori* function alloction, in other words, is intractable (Hollnagel & Woods, 1983).

Engineers who follow the substitution method of function allocation can become spectacularly ill-calibrated with respect to the real consequences of technology change in their domain of work (Roesler et al., 2001). That is, they will envision that the predicted consequences (e.g. lower workload, higher accuracy) and only the predicted consequences of automation will occur (see also Norman, 1990). Parasuraman *et al.* (2000) list predictable consequences that an engineer may expect if s/he follows function allocation by substitution, for example lower workload, problems with complacency and situation awareness, time savings and higher response accuracy. But this is only a vague and bleak reflection of the real impact of automation, or any technology change for that matter. For example, one folk argument is that complacency (whatever it might be) presents a safety risk in automated systems. The operator is painted as a passive monitor, whose greatest safety risks lie in deskilling, complacency, vigilance decrements and the inability to intervene successfully in deteriorating circumstances. Parasuraman et al. (2000, p. 291) contend that "the operator may not monitor the automation and its information sources and hence fail to detect the occasional times when the automation fails", just as Kern (1998, p. 240) claims that "as pilots perform duties as system monitors, they will be lulled into complacency, lose situational awareness, and not be prepared to react in a timely manner when the system fails". But none of these folk claims have a strong empirical basis, and in fact the inverse may be true. First, automation hardly ever "fails" in a binary sense. In fact, manufacturers consistently point out, in the wake of accidents, how their automation behaved as designed (FAA, 1996). Second, instead of being the result of people "slipping out of the loop", accidents with automated systems, for example in aviation (Strasbourg (METT, 1993); Nagoya (NTSB, 1994); Toulouse (DGA, 1994) Cali (Aeronautica Civil, 1996)) are preceded by practitioners being highly active managers—operators who are fully tied up in typing, searching, programming, planning, responding, communicating, questioning—trying to coordinate their intentions and activities with those of other people and the automation, exactly like they would in the pursuit of success and safety.

3. Transformation and adaptation

Automation does not just have quantitative consequences, it produces qualitative shifts. It will transform people's practice and force them to adapt in novel ways. “It alters what is already going on—the everyday practices and concerns of a community of people—and leads to a resettling into new practices” (Flores et al., 1988, p. 154). New technologies "alter the tasks for which they were designed, indeed alter the situations in which the tasks occur and even the conditions that cause people to want to engage in the tasks” (Carroll and Campbell, 1988, p. 4). Unanticipated consequences are the result of these much more profound, qualitative shifts. For example, during the Gulf War in the early 1990's, "Almost without exception, technology did not meet the goal of unencumbering the personnel operating the equipment. Systems often required exceptional human expertise, commitment, and endurance" (Cordesman and Wagner, 1996, p.25).

Where automation is introduced, new human roles emerge. The original belief is that new technology transforms the tools of people, who will then have to adapt. For example, according to Albright *et al.* (1996), the removal of paper flight progress strips in Air Traffic Control represents a transformation of the workplace, to which controllers only need to adapt (they "compensate" for the lack of flight progress strips). In reality, however, it is people's practice that gets transformed by the introduction of new tools. New technology, in turn, gets adapted by people in locally pragmatic ways so that it will fit the constraints and demands of actual practice (Cook & Woods, 1996). For example, controlling without flight progress strips (relying more on the indications presented on the radar screen) asks controllers to develop and refine new ways of managing airspace complexity and dynamics. In other words, it is not the technology that gets transformed and the people who adapt. Rather, people's practice gets transformed and they in turn adapt the technology to fit their local demands and constraints.

The key for designers is to accept that automation *will* transform people's practice and to be prepared to learn from these transformations as they happen. This is by now common in design that practices forms of contextual inquiry (e.g. Beyer & Holtzblatt, 1998). Here the main focus of system design is not the creation of artifacts per se, but getting to understand the nature of human practice in a particular domain, and changing those work practices rather than just adding new technology or replacing human work with machine work. Designers have to recognize:

- that design concepts represent hypotheses or beliefs about the relationship between technology and human cognition/collaboration;

- that they need to subject these beliefs to empirical jeopardy by a search for disconfirming and confirming evidence;
- that these beliefs about what would be useful have to be tentative and open to revision as they learn more about the mutual shaping that goes on between artifacts and actors in a field of practice.

4. Progress towards better teamplay

Designers need to depart from the quantitative, substitutional practice of function allocation (Norman, 1990, Hollnagel, 1999). Substitution assumes a fundamentally uncooperative system architecture in which the interface between human and machine has been reduced to a trivial "you do this, I do that" barter. The question for successful automation is not "who has control over what or how much". It is "how do we get along together". Indeed, where designers really need guidance today is how to support the coordination between people and automation. In complex, dynamic, non-deterministic worlds, people will continue to be involved in the operation of highly automated systems. The key to a successful future of these systems lies in how they support cooperation with their human operators—not only in foreseeable standard situations, but also during novel, unexpected circumstances.

One way to frame the question is how to turn automated systems into effective team players Christoffersen and Woods (2000). Good teamplayers make their activities observable for fellow teamplayers, and are easy to direct. To be observable, automation activities should be presented in ways that capitalize on well-documented human strengths (our perceptual system's acuity to contrast, change and events; our ability to recognize patterns and know how to act on the basis of this recognition (e.g. Klein, 1998). For example:

- Event-based: representations need to highlight changes and events in ways that the current generation of state-oriented displays do not;
- Future-oriented: in addition to historical information, human operators in dynamic systems need support for anticipating changes and knowing what to expect and where to look next;
- Pattern-based: operators must be able to quickly scan displays and pick up possible abnormalities without having to engage in difficult cognitive work (calculations, integrations, extrapolations of disparate pieces of data). By relying on pattern- or form-based representations, automation has an enormous potential to convert arduous mental tasks into straightforward perceptual ones.

Teamplayers are directable when the human operator can easily and efficiently tell them what to do (see also Sarter & Woods, 1997). Designers could borrow inspiration from how practitioners successfully direct other practitioners to take over work. These are intermediate, cooperative modes of system operation that allow human supervisors to delegate suitable sub-problems to the automation, just like they would be delegated to human crewmembers. The point is not to make automation into a passive adjunct to the human operator who then needs to micro-manage the system each step of the way. This would be a waste of resources, both human and automation. Human operators must be allowed to preserve their strategic role in managing system resources as they see fit given the circumstances.

Even completely automated systems almost always have a human operator somewhere, at some level. Chapanis' 1970 engineer's dream of a fully mechanized world is illusory. Questions about fully automated systems are misguided as they re-frame the debate about the human-machine relationship in the language of a gradual marginalization of human input—indeed in the way of Sheridan's levels. Once again, the question is pitched as one of uncooperative all-or-none control. What matters is the extent to which powerful automation allows teamplay with its human operators. What matters is how observable the automation makes its behavior for its human counterparts, and how easily and efficiently it allows itself to be directed, even (or especially) during busy, novel episodes.

5. Conclusion

Automation is almost always justified on the basis of its presumed benefits for system performance (e.g. Parasuraman et al. 2000). As such it embodies a hypothesis (or set of hypotheses) about what would be useful for a field of practice. The accuracy of the designer’s hypothesis hinges on (1) how well the prediction of automation effects is grounded in the human factors research base, and (2) on whether the designer is willing and able to take an experimental stance towards automation development in his/her chosen field of practice. In guiding their design decisions (and thus generating their hypotheses), system developers should abandon the traditional “who does what” question of function allocation. Instead, the more pressing question today is how to make humans and automation get along together.

References

Aeronautica Civil de Colombia (1996). Aircraft accident report: Controlled flight into terrain, American Airlines flight 965, Boeing 757-223, N651AA near Cali, Colombia, December 20, 1995. Bogota, Colombia: Aeronautica Civil.

Air Transport Association of America. (1989, April). National plan to enhance aviation safety through human factors improvements. Washington, DC: ATA.

Albright, C. A., Truitt, T. R., Barile, A. B., Vortac, O. U., & Manning, C. A. (1996). How controllers compensate for the lack of flight progress strips (Final report DOT/FAA/AM-96/5). National Technical Information Service, Virginia.

Bainbridge, L. (1987). Ironies of automation. In J. Rasmussen, K. Duncan, J. Leplat (Eds.), New technology and human error, pp. 271-283. Chichester: Wiley.

Beyer, H., & Holtzblatt, K. (1998). Contextual design: Defining customer-centered systems. San Diego, CA: Academic Press.

Billings, C. E. (1996). Aviation automation: The search for a human-centered approach. Hillsdale, N.J.: Lawrence Erlbaum Associates.

Broadbent, D. E. (1958). Perception and communication. London, UK: Pergamon.

Carroll, J.M. and Campbell, R.L. (1988). Artifacts as psychological theories: The case of human-computer interaction. IBM Research Report RC 13454, Watson Research Center, Yorktown Heights, NY.

Chapanis, A. (1965). On the allocation of functions between men and machines. Occupational Psychology, 39, 1-11.

Chapanis, A. (1970). Human factors in systems engineering. In K. B. De Greene (Ed.), Systems psychology, pp. 51-78. New York: McGraw-Hill.

Christoffersen, K., & Woods, D. D. (2000). How to make automated systems team players. Columbus, OH: Institute for Ergonomics, The Ohio State University.

Cook, R.I., and Woods, D.D. (1996). Adapting to new technology in the operating room. Human Factors, 38(4), 593-613.

Cordesman, A. H. and Wagner, A. R. (1996). The Lessons of Modern War, Vol.4: The Gulf War. Boulder, CO: Westview Press.

Dekker, S. W. A., & Hollnagel, E. (Eds.) (1999). Coping with computers in the cockpit. Aldershot, UK: Ashgate Publishing Co.

Dekker, S. W. A., & Woods, D. D. (1999). Automation and its impact on human cognition. In S. W. A. Dekker & E. Hollnagel (Eds.), Coping with computers in the cockpit, pp. 7-28. Aldershot, UK: Ashgate Publishing Co.

Directorate General of Armaments (France) (1994). Investgation committee report on A330 accident in Toulouse on 30 June 1994. Paris: DGA.

Douglas Aircraft Co. (1990). Functional decomposition of the commercial flight domain for function allocation (Final report to NASA Langley Research Center, Contract No. NASI-18028). Washington, DC: NASA.

Edwards, E., & Lees, F. P. (1972). Man and computer in process control. London: Institute of chemical engineers.

Fitts, P. M. (1951). Human engineering for an effective air navigation and traffic control system. Washington, DC: National Research Council.

Flach, J. M. (2000). Discovering situated meaning: An ecological approach to task analysis. In J. M. Schraagen, S. F. Chipman, & V. L. Shalin (Eds.): Cognitive task analysis, pp. 87-100. Mahwah, NJ: Lawrence Erlbaum Associates.

Flores, F., Graves, M., Hartfield, B. and Winograd, T. (1988). Computer systems and the design of organizational interaction. ACM Transactions on Office Information Systems, 6, 153-172.
Hollnagel, E. and Woods, D. D. (1983). Cognitive systems engineering: New wine in new bottles. Int. J. Man-machine studies, 18, 583-600.
Hollnagel, E. (1999). From function allocation to function congruence. In S. W. A. Dekker & E. Hollnagel (Eds.), Coping with computers in the cockpit, pp. 29-53. Aldershot, UK: Ashgate Publishing Co.
Howard, M. (1999). Visualizing automation behavior. In S. W. A. Dekker & E. Hollnagel (Eds.), Coping with computers in the cockpit, pp. 55-67. Aldershot, UK: Ashgate Publishing Co.
Klein, G. (1998). Sources of power: How people make decisions. Cambridge, MA: MIT Press.
Mertes, F., & Jenney, L. (1974). Automation applications in an advanced air traffic management system: Vol III, Methodology for man-machine task allocation (Rep. No. DOT-TSC-OST-74-14-III). McLean, VA: TRW, Inc.
Ministère de l'equipment, des transports et du tourisme (1993). Rapport de la commission d'enquete sur l'accident survenu le 20 Janvier 1992 près du Mont Saint Odile (Bas Rhin) a l'Airbus 320 immatriculé F-GGED exploité par la compagnie Air Inter. Paris, France: METT.
Moray, N. (1997). Human factors in process control. In G. Salvendy (Ed.), Handbook of human factors and ergonomics, pp. 1944-1971. New York: Wiley Interscience.
Mouloua, M. & Scerbo, M. W. (Eds.) (1999). Automation technology and human performance. Mahwah, NJ: Lawrence Erlbaum Associates.
National Transportation Safety Board. (1994). Safety recommendations A-94-164 through 166 concerning China Airlines Airbus A-300-600R accident at Nagoya, Japan, April 26, 1994. Washington, DC: NTSB.
Neisser, U. (1976). Cognition and reality. San Francisco, CA: Freeman.
Norman, D. A. (1990). The 'problem' with automation: inappropriate feedback and interaction, not 'over-automation'. Phil. Trans. R. Soc. Lond, B327, 585-593.
Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). A model for types and levels of human interaction with automation. IEEE transactions on systems, man, and cybernetics—part A: systems and humans, 30(3), 286-297.
Roesler, A., Feil, M., Puskeiller, A., Woods, D. D., & Tinapple, D. (2001). Design is telling stories about the future. Columbus, OH: Cognitive Systems Engineering Laboratory, The Ohio State University.
Sarter, N. B., & Woods, D. D. (1997). Teamplay with a powerful and independent agent: Operational experiences and automation surprises on the Airbus A-320. Human Factors, 39(4), 553-569.
Sarter, N. B., Woods, D. D., & Billings, C. E. (1997). Automation surprises. In, G. Salvendy (Ed.). Handbook of human factors/ergonomics (2nd. ed., pp. 743-766). New York: Wiley.
Sheridan, T. B. (1987). Supervisory control. In: I. Salvendy (Ed.), Handbook of human factors. New York, NY: Wiley.
Swain, A. D., & Guttman, H. F. (1980). Handbook of human reliability analysis with the emphasis on nuclear power plant applications (NRC NUREG-CR-1278). Albuquerque, NM: Sandia National Laboratories.

Wiener, E. L. (1989). Human factors of advanced technology ("glass cockpit") transport aircraft (NASA contractor report No. 177528). Moffett Field, CA: NASA Ames Research Center.

Levels of Automation Support The automation:
1. offers no assistance: human supervisor must do it all;
2.offers a complete set of action alternatives, and
3. narrows the selection down to a few, or
4. suggests one, or
5. executes that suggestion if the supervisor approves, or
6. allows the supervisor a restricted time to veto before automatic execution, or
7. executes automatically, then necessarily informs the supervisor, or
8. informs him after execution only if he asks, or
9. informs him after execution if the subordinate decides to.
10. decides everything and acts autonomously, ignoring the supervisor.

Table 1: A list of levels of supervisory control (after Parasuraman, Sheridan & Wickens, 2000 and Sheridan, 1987).