(1998). Proceedings. Fourth Annual Symposium on Human Interaction with Complex Systems (Cat. No.98EX120)(pp.90-98). Los Alamitos, CA: IEEE Comput. Soc. ix+235pp.

Implications of a Control-Theoretic Approach to Human-Automation-Plant Interface Design

Greg A. Jamieson[1] and Kim J. Vicente[2]
Cognitive Engineering Laboratory
Department of Mechanical and Industrial Engineering
University of Toronto
5 King's College Rd.
Toronto, Ontario, Canada M5S 3G8
[1] gimlet@me.utoronto.ca
[2] benfica@ie.utoronto.ca, www.mie.utoronto.ca/labs/cel/kim.htm

Abstract

This article critically evaluates some of the design guidelines proposed by researchers investigating human-automation interaction. A control-theoretic framework is proposed to introduce a set of important conceptual distinctions that need to be respected in automation research and design. The framework is applied to a specific topic of recent research focus, modes in aviation automation. Although previous research in this area has advanced our understanding of key issues, the existing design guidelines lack the necessary specificity to contribute to major improvements in design practice. Our control-theoretic perspective helps clarify these issues, and demonstrates why existing design guidelines are unlikely to fully support automation designers in their efforts to reduce the frequency of mode-related errors. An alternative perspective, which we believe will yield more concrete design criteria and more appropriate tools, is suggested.

Introduction

The use of automation in complex sociotechnical systems has proved to be a double-edged sword. On the one hand, it promises unprecedented reliability, reduced workload, improved economy, and fewer errors. On the other hand, it whispers of less tangible, but no less real, costs to operators in terms of skill degradation, mental isolation, and supplemental monitoring tasks [1, 2]. Repeating patterns demonstrated in other technology-driven enterprises, designers have focused on the beneficial cutting edge of automation while ignoring or downplaying the costs to the human stakeholders who must necessarily remain as aspects of the system. How can we convince engineers that automation technology is a tool and not a panacea [3] for human-machine interaction problems? The ironic solution we adopt in this article is to use automation designers' own language, control theory, to point out the role of human factors considerations in the design of automation.

The purpose of this article is to propose a domain-independent, control-theoretic framework for investigating human factors issues in automation. This framework is used to evaluate some of the design guidelines that have been put forth in the literature and to discuss how these guidelines might be augmented to more effectively contribute to the design process. We also show the value of this generic framework by using it to critically review a particular body of automation research, that dealing with mode proliferation, mode transitions, mode awareness, and mode error in aviation automation.

A Control-Theoretic Framework For Studying Automation

Definitions

In examining the literature on human-automation interaction, we have noted frequent terminological inconsistencies. In order to avoid imprecise language ourselves, we offer the following definitions which will be employed for the duration of this article:

- Process: The thing being influenced by the Final Control Components.
- Final Control Components: Controllable equipment that is used to influence a Process.
- Plant: Final Control Components + Process.
- Controller: The means by which action is exerted on the Final Control Components.

(1998). Proceedings. Fourth Annual Symposium on Human Interaction with Complex Systems
(Cat. No.98EX120)(pp.90-98). Los Alamitos, CA: IEEE Comput. Soc. ix+235pp.

- Instrumentation: The means by which data about the Plant and Controllers are gathered.
- Displays: The devices through which Operators obtain information about the System.
- Controls: The devices through which Operators take Action on the System.
- Interface: Displays + Controls.
- System: Plant + Controllers + Instrumentation + Interface.

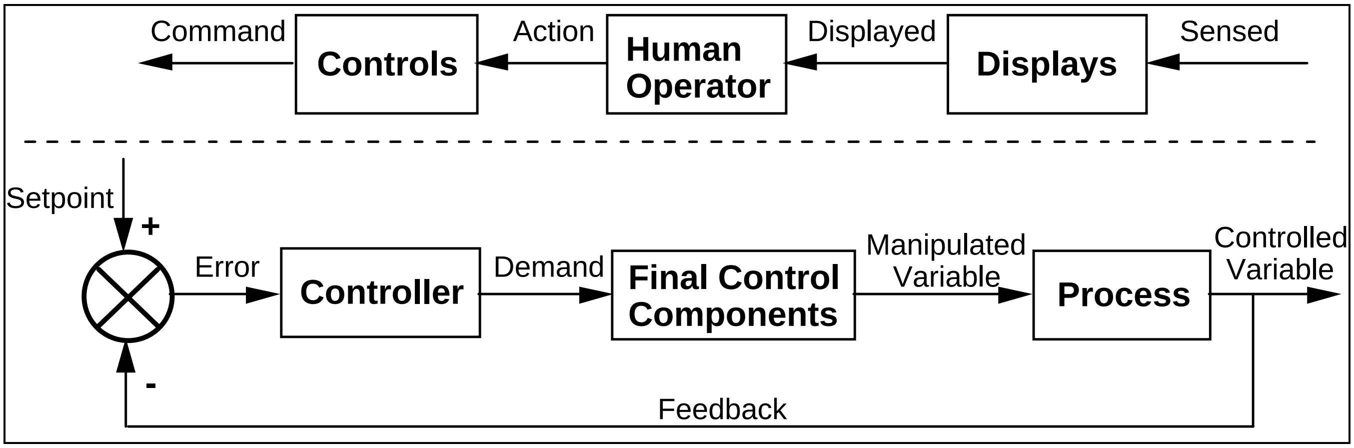

Figure 1. A negative feedback control loop.

Throughout this article we have capitalized these terms. The relationships between the terms should become more clear as we place them in the context of a negative feedback control loop.

Negative Feedback Loop

Figure 1 presents a very simple but conceptually powerful model of automation based on a standard negative feedback control loop (adapted from [4]). In this figure, boxes represent System elements and arrows denote signals between elements. The lower portion of Figure 1 is a traditional regulatory feedback loop composed of three distinct elements (the upper portion will be described later). If all of the elements are working as planned, then the behavior of the feedback loop can be described in a relatively straightforward fashion. Tracing through the diagram from left to right, the comparator subtracts the Feedback signal from the Setpoint (i.e., the goal) to create an Error signal. This Error signal is sent to an automatic Controller which generates (via programmed logic) a Demand signal that is sent to the Final Control Components. The Final Control Components (e.g., a valve) map this Demand signal onto a change in the state of the Manipulated Variable (e.g., valve position) which then influences the Process itself, causing the Controlled Variable (e.g., flow rate) to change as well. The Instrumentation (e.g., a flow meter) gathers information about the Controlled Variable and feeds it back to the comparator.

The upper portion of Figure 1 serves as a placeholder for the Interface and the Human Operator, two System elements that play a role in feedback control loops that are not fully automated. Moving from right to left, Sensed data signals are fed into the Displays element and Displayed to the Human Operator. The Operator takes Action by manipulating the Controls in the Interface which generates a Command signal. The manner in which the elements in the upper portion of the figure are connected to the rest of the feedback control loop (i.e., what signals serve as inputs and outputs) cannot be described generically because there is no process that is agreed upon by automation designers for identifying which connections are required. Connecting the upper and lower portions of Figure 1 involves making a set of design decisions. One of the contributions of this paper is to propose how these decisions should be made.

Implications

While control engineers will find this feedback model highly oversimplified, it serves to make five important points. First, there are four different elements that must be distinguished conceptually: the automation which is the Controller, the Final Control Components that the automation acts on, the Process itself, and the Interface. Frequently, researchers do not discriminate clearly and consistently between these elements. A common approach is to use the catch-all label, “system”. In some cases, “system” seems to mean automation, whereas in other cases it seems to include the Final

(1998). Proceedings. Fourth Annual Symposium on Human Interaction with Complex Systems (Cat. No.98EX120)(pp.90-98). Los Alamitos, CA: IEEE Comput. Soc. ix+235pp.

Control Components and the Process together. Furthermore, in some cases, it seems that authors are inadvertently switching between these two uses of the term "system" in the same publication. We say that this "seems to be the case" because we can only make inferences based on the surrounding text; a definition of "system" is rarely given.

The second implication is that a failure can occur in different System elements. For example, it is possible for malfunctions to occur in the Controller, the Final Control Components, or the Process. These are qualitatively different events that have varying implications for Operators' fault management activities because they may require different types of Actions (e.g., turning off the automation vs. isolating the failed component vs. shutting down the Process). If only the catch-all phrase "system" is used, then it will merely be possible to say that there is a problem "somewhere". It will not be possible to localize the problem to a particular element.

Third, all of the signals in the lower portion of Figure 1 must be displayed in the Interface if operators are to discriminate between the different types of failures just described. For example, to determine if there is a problem with the Controller, Operators must know what the current Error signal is, know what the current Demand signal is, and then use their knowledge of how the Controller is supposed to work to determine if the Demand is what it should be, given the current input to the Controller. Basically, this process requires that a normative model (in this case, of the Controller) be used to check if the System element is behaving as expected. This logic is known as <u>analytical redundancy</u> in control theory [5]. The same general approach can be applied to other System elements, but only if the requisite signals are available.

If any one of these signals is not available, then it will not be possible to determine precisely where the problem is. For example, if the Controlled Variable does not match the goal Setpoint, we do not know if it is because: a) the Controller is sending appropriate Demand signals but the Final Control Components are not responding as they should (e.g., an actuator failure); b) the Final Control Components are functioning properly, and are merely reacting to inappropriate Demand signals from a faulty Controller (e.g., faulty control logic); or c) both the Controller and Final Control Components are behaving properly, but there is a fault in the Process (e.g., a leaky tank). Unless all of the signals in the lower portion of Figure 1 feed into the Interface element, the situation is underspecified. Unfortunately, at least some, and perhaps many, automation interfaces seem to suffer from this problem (e.g., [7]).

Fourth, the same Plant can conceivably be regulated by one of many different Controllers. Thus, the design of the Controller need not be taken as a given, but can instead be seen as an element whose design should be specified by taking into account, among other things, human factors considerations [8, 9]. This perspective differs from the traditional view that the design of automatic Controllers is strictly a technical issue that must be completed before human factors professionals can start their job of Interface design [10].

Fifth and finally, the internal structure of the Controller is different from the internal structure of the Final Control Components or of the Process. Thus, when researchers recommend that Interfaces should be designed so as to make the "system" transparent, it is not clear what this really means. For example, a visualization of the Controller will look very different from a visualization of the Process. After all, the former is the thing doing the controlling whereas the latter is the thing being controlled. What do we want to make transparent, the Controller, the Final Control Components, the Process, or all three? Will the acknowledged problems with automation be remedied by making any combination of these elements more transparent?

In summary, the control-theoretic framework in Figure 1, while simple, leads to several points which have significant implications for both research and design. In the next section, we will show the value of this generic framework by using it to review a specific body of research, that related to modes in aviation automation.

Modes, Mode Transition, Mode Awareness, And Mode Error

What is a Mode?

A mode is a manner of device behavior [11]. A device may have multiple modes of operation, although only one may be active at any given moment. That is to say, some devices can behave in more than one manner, although only one such behavior can be exhibited at a time. These multiple manners of behavior can serve at least two purposes. First, they can allow a single device to accomplish <u>multiple tasks</u>. Take, for example, a digital watch. It may have modes that tell time, set an alarm, display the date, or act as a stop watch. Second, multiple behaviors can allow the device to accomplish a <u>single task</u> using any one of a number of <u>strategies</u>. For example, that digital watch might allow the user to choose between civilian or military time displays. This important

(1998). Proceedings. Fourth Annual Symposium on Human Interaction with Complex Systems
(Cat. No.98EX120)(pp.90-98). Los Alamitos, CA: IEEE Comput. Soc. ix+235pp.

distinction between tasks and strategies is rarely, if ever, drawn in the literature on modes.

Modes can be present in the Plant, the Controllers, or the Interface of a System. For example, an automobile transmission (Plant) can be configured in three modes; neutral, forward (perhaps in more than one gear), or reverse. A flight management system (Controller) can be configured to adjust speed via pitch or thrust. A cockpit display for a radar (Interface) can be set to provide rose (360 degree) or arc coverage. Note that in the flight management system (FMS), the mode of control does not change the mode of the Plant (i.e., the aircraft itself). The various Controller modes are imposed by designers who try to anticipate behaviors that the pilots will want the Plant to assume. Similarly, having a radar's Interface present data in an arc rather than a circle does not change the manner in which the radar itself (the Plant) operates. The various Display modes are imposed by designers who try to anticipate the structure of information that the flight crew will want to observe.

Some of these distinctions have been recognized in the literature. For example, Sarter and Woods [12] distinguish system (which, in our terms, seems to correspond with Controller) and interface modes. Also, Degani and Kirlik [13] separate interface/format and control modes. Note, however, that none of these authors have explicitly distinguished the third category of modes; those of the Plant. This distinction is very important because it leads to another principle not often directly expressed in the literature; modes are not just a feature of automation. They are a potential feature of all three System elements independently and must be dealt with as such by designers.

Mode Transition

Because a Plant, Controller, or Interface can exhibit several modes of operation, designers must also provide for transitions between modes. In most contexts, this means that users are required to take actions to make such transitions. However, as the applications in which mode options are available become more complex, transitions also become more complex [14]. Plants, Controllers, or Interfaces are capable of partially and fully autonomous mode transitions. The variety of automated subsystems, their interrelated mode settings, and the mixed autonomy of transitions can increase overall system complexity exponentially.

Mode Awareness

With mode-induced flexibility comes an added responsibility -- management. Once given the opportunity to choose a mode, an Operator must make a decision, keep track of it, and act in a way that is consistent with it. This added responsibility is frequently referred to as mode awareness: "knowledge and understanding of the current and future status and behavior of automated...systems" ([15], p. 239). Mode confusion is the failure to maintain mode awareness; a misidentification of machine behavior and transitions between behaviors [11]. Sarter and Woods [16] have referred to the unanticipated system response associated with mode confusion as "automation surprises." Note that mode confusion does not necessarily correspond to mode error unless an erroneous action is taken.

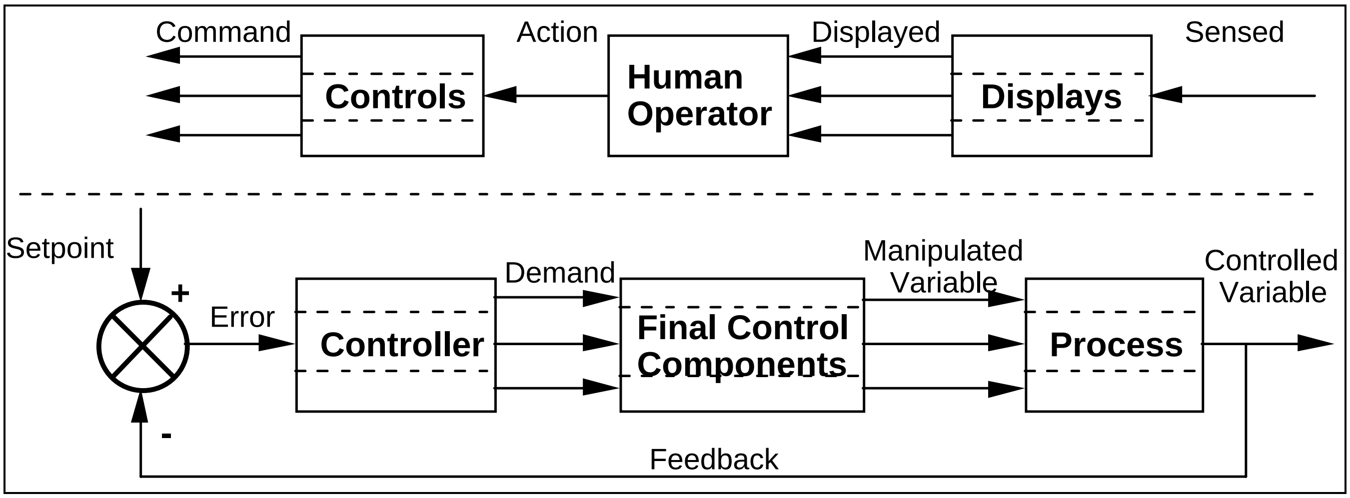

Figure 2. The feedback control loop for a System with modes.

(1998). Proceedings. Fourth Annual Symposium on Human Interaction with Complex Systems (Cat. No.98EX120)(pp.90-98). Los Alamitos, CA: IEEE Comput. Soc. ix+235pp.

Mode Error

A mode error occurs when an Operator takes an action that is appropriate in one mode, but inappropriate in the present (active) mode [17, 11]. Mode error has been cited as a contributor to a number of fatal aviation incidents [18] and is recognized as a problem with successive generations of cockpit automation [19]. Mode error is an inherently human-machine problem; it necessitates a System element with modes and an Operator who loses track of them [12]. Thus, mode error is not a property of a person or of a device. Rather, it is an emergent phenomenon. Any approach that fails to acknowledge this systems perspective and takes a reductionistic approach to the problem will more than likely fail to recognize this issue.

Modes in the Feedback Control Process Model

Figure 2 demonstrates how the presence of modes can complicate the negative feedback model introduced earlier in Figure 1. The relationships between signals are now conditional upon the active mode. Figure 2 reflects this conditionality by segmenting System elements into modes using dotted lines and adding different signal paths between elements for each mode segment. Thus, depending on the active mode of Controller operation, the Demand signal may change for the very same Error signal (e.g., an FMS can send Demand signals to the engines or to the control surfaces, depending on the engaged vertical navigation mode). Similarly, depending on the Displays mode, the same Sensed signal might be Displayed differently (e.g., a signal from a radar unit is configured in either arc or rose format).

Previously, our control-theoretic perspective led to five generic research and design implications. We now examine how the introduction of modes inflates those implications. First, the need to distinguish between System elements is increased. Since each element can have multiple modes, identifying those elements requires that their modes be distinguished as well. Second, we have added a potential new source of failure. Whereas before we noted that failure could occur at any System element, we now have the added potential for failures localized to modes within elements. Third, locating a fault in the System is potentially much more difficult because the number of signal paths at which failures can occur has been multiplied. Before we required normative models of the System elements in order to achieve analytical redundancy. Now we need normative models for each mode configuration of each System element. Fourth, the number of degrees of freedom open to the automation designer has increased. Not only is the design of the Controller open to human factors considerations; so is the selection of mode configurations to be incorporated in that Controller. Fifth, if we are to pursue the idea of making the internal structures of the elements transparent, we have more details which must be specified (i.e., the various mode-induced relationships between signals).

The control-theoretic approach has provided a framework through which we have raised a number of issues for research and design of human-automation interaction. Our discussion of modes, their associated phenomena, and their impact on the feedback control process model has demonstrated that evolutionary trends in automation pose increasingly complex challenges for designers. Our ultimate goal is to support designers in employing these new capabilities in ways that are more beneficial to Human Operators. In the next section, we will review some guidelines that have been proposed in the automation literature in the light of our control-theoretic framework.

Proposed Guidelines: Mental Models And Feedback

A number of design guidelines for improving human-automation interaction have been offered in the literature, specifically in relation to mode issues. These guidelines typically revolve around two principles (e.g. [16]), namely that mode awareness problems and mode error can be reduced if: 1) the operator's faulty mental model of the "system" is corrected; and 2) more feedback about what the automation is doing is provided. We will review each of these guidelines in turn, demonstrate their limitations in terms of our control-theoretic framework, and propose an alternative approach that we believe provides the necessary specificity to contribute to significantly improved design for human-automation-plant interaction.

Improving the mental model

Mode confusion leads to the following questions about automation: "What is it doing now?", "Why is it doing that?", "What will it do next?", and "How in the world did we get into this mode?" [16]. How should automation designers address these questions? Consider the following: "If pilots were provided with an overall mental representation of the functional structure of the FMS, they would be better able to manage and utilize the automated systems in unusual or novel situations" [12, p. 320]. What do Sarter and Woods [12] mean by an "overall mental representation of the functional structure"? Perhaps the following serves to clarify their

(1998). Proceedings. Fourth Annual Symposium on Human Interaction with Complex Systems
(Cat. No.98EX120)(pp.90-98). Los Alamitos, CA: IEEE Comput. Soc. ix+235pp.

intention, "The monitoring of an automated system requires an adequate mental model of the structure of the system" [19 p.182]. This is a statement with which we agree; however, the "system" to which Sarter and Woods are referring seems to be the Controller (i.e., the FMS). They are suggesting that it is necessary to make explicit to the Operator the way in which the automation functions. Note that they do not recommend that the Operator also have a mental model of the Plant (i.e., the aircraft being controlled), perhaps because the Plant falls outside of the boundaries they define for their "system".

What insights can the control-theoretic framework offer on Sarter and Woods' [12, 19] recommendation? The first implication we drew from our model was that the Controller and the Plant should be distinguished from each other. It is not sufficient to require that Operators have functional mental models of the Controller without explicitly relating that knowledge to a model of the Plant itself. At no point in their article do Sarter and Woods [12] acknowledge this important point. We believe that in order to answer questions like "What is it doing now?", "Why is it doing that?", "What will it do next?", and "How in the world did we get into this mode?", the Operators must have access to veridical models of both the Controller and the Plant.

The issue of mental models is also discussed by Vakil et al. [20], who have pointed out that aviation automation has evolved progressively, adding new features in generational increments without surrendering older features (see also [3]). "The ... effect of this entropic growth is to create a system [Controller] that appears to lack a simple, consistent, global model. This lack requires pilots to create their own ad-hoc models" ([20], p. 429) Elsewhere in their article, Vakil et al. [20] state that there is a "lack of underlying structure to the automation, making it difficult for pilots to develop consistent mental models" (p. 428). From the perspective of our control-theoretic framework, these two statements prompt a few comments. First, it must be emphasized that some sort of structure <u>must</u> exist in the automation. Otherwise it would not be capable of reducing the degrees of freedom in the Plant to that required for action. Disorder can only be reduced by regularity (i.e., structure). From the viewpoint of a Human Operator, that structure may look very complex and difficult to understand, but it is nevertheless still structure. Second, if we are talking about modeling complex systems, the Law of Requisite Variety [21] tells us that a simple model is not likely to be of much use -- complex systems require complex controllers. Therefore, overly simplified models may actually be misleading to Operators and thus detrimental to performance. Third, if we infer that, by "consistent" and "global" Vakil et al. [20] mean internally coherent and accurate, then we are left with another problem. It seems that the complex structure of existing aviation automation is a by-product of an evolutionary design process. This sort of 'legacy' structure can be faithfully modeled, but it will still appear as ad-hoc as the process which produced it. In other words, a good model cannot compensate for structural deficiencies in the modeled system.

This brief critique shows that the control-theoretic perspective can clarify and point out the limitations of statements made in the literature. But to be of constructive value, our framework should also point to a better approach to these challenging issues. Perhaps a more constructive approach would be ensuring that the automation has an adequate 'mental' model of the Plant. This means that automation designers would have to cooperate with Plant designers to ensure that the Controllers, the Final Control Elements, and the Process are driven by integrated functions and compatible intentions [22]. In such a design approach, the responsibility of the Interface designer would be to ensure that the model of the Plant that served as the basis for designing the Controller is available to the Operator. In this way, the structure (and the emerging behavior) of the Controller can be related to the functionality of the Plant (which would also be modeled in the Interface). In other words, the Interface designer should externalize both a model of the Controller and a model of the Plant [23]. Ideally, a single model of the Plant would: a) be used to design the Controller; b) be represented in the Interface; and c) be used by Operators to understand the Plant. Such an approach should ensure that Plant, Controller, and Operator have a common, integrated design basis that provides for effective coordination between all three elements.

Improving feedback

On the heels of an account of mode confusion or mode error one will typically find a statement about improving the feedback provided by the Interface. For example, "To support the increased need for coordination, systems would need to be more transparent. Feedback design would need to be improved to support the Human Operator in keeping track of and in anticipating the status and behavior of his machine counterpart" [19, p. 181]. This recommendation has some merit. If Operators are not in touch with what is taking place in the System then an appropriate response is to give them more data. However, Norman notes that in most aviation incidents in which automation is implicated, the data required to detect problems were available [2]. This point is echoed by Sarter and Woods [16]. Calling for more feedback begs the question, "Feed back what?"

(1998). Proceedings. Fourth Annual Symposium on Human Interaction with Complex Systems (Cat. No.98EX120)(pp.90-98). Los Alamitos, CA: IEEE Comput. Soc. ix+235pp.

The quick answer is, “Feed back the structure and state of the automation.” Earlier, however, we suggested that the legacy structure of modern aviation automation may be complex and difficult to understand. Making such automation structure transparent is likely to result in something visibly complex. Such a representation is as likely to contribute to Operator confusion as it is to resolve it. We stated before that good modeling cannot make up for structural complexity in the system being modeled. Neither can good Interface design.

There is a second problem with the “increased feedback” design guideline. “What is needed is continual feedback about the state of the system, in a normal natural way, much in the manner that human participants in a joint problem-solving activity will discuss the issues among themselves” ([2], p. 591). This statement raises a new question, how should feedback be presented to the Operator? The challenge to Norman’s recommendation is that Human Operators seem to be far more effective communicating with each other than with automation. How can this be replicated? “The task of presenting feedback in an appropriate way is not easy to do. Indeed, we do not yet know how to do it” ([2], p. 591). The issue, as Norman [2] correctly points out, is that automation is not sensitive enough to provide information to the operator that is relevant to the context. Feedback needs to be meaningful, not just available. The problem is not that the Operators do not have enough data, but that they do not have enough information (c.f. [4]). “Feedback? Yes, but how?”

Both of the questions emanating from the above discussion can be couched in terms of our control-theoretic perspective. Specifically, which signals in Figure 1 or 2 need to be fed into the Display element (Feed back what?) and in what form should they be Displayed (Feedback how?). It should be evident that a guideline such as, “In order to reduce mode error, provide more feedback to the Operator,” lacks the specificity required to contribute to improved design practice. Fortunately, the control-theoretic perspective provides more concrete direction about feedback content. In Systems where Operators are required to shut down the plant when there is an automation failure (as in some nuclear power plants), then minimal feedback is required. However, in Systems where Operators must keep the Plant on-line and compensate for the fault (as in some petrochemical plants), then more feedback is necessary. In order to achieve analytical redundancy with respect to any System element, the input and output signals for that element must appear in the Interface along with a normative model of the element’s operation.

The manner in which our proposed framework answers the “How” question has already been suggested, but in a different context. The appropriate means of communicating between designers, automation, and Operators is through a shared model of the Plant [22]. To the extent that all three agents can “converse” with a common understanding of the functions of the Plant, they will be communicating information as opposed to mere data. Thus, a better guideline for designers is, “In order to achieve effective communication of feedback, present it in the context of a model shared by all stakeholders.”

A Control-Theoretic Approach To Human-Automation-Plant Design

In the preceding section, we critically examined two common design guidelines for human-automation interaction. The control-theoretic perspective suggests some modifications to these guidelines to make them more specific. However, the potential contributions of our framework are not limited to revising existing guidelines. In this section, we summarize the research and design implications that are derived from our control-theoretic approach.

Research and Design Implications

1. Designers and researchers must distinguish between all of the System elements in the feedback control loop. Vague references to the “system” do not support researchers’ efforts to investigate the nature of human-automation interaction, nor do they serve Operators attempting to localize disturbances to specific System elements. Clear distinctions will support researchers attempting to understand the nature of such problems and will provide meaningful organizing principles to Interface designers. These distinctions must be made explicit in the Interface which is employed to convey the structure of System elements to the Operator. In mode-capable Systems, individual modes within elements must also be distinguished.

2. In order to support analytical redundancy, all of the signals in the feedback loop must be Displayed in the Interface. The output signal from each element must be compared to the expected value, given the input signal and a normative model of the element. This technique will allow the Operator to localize sources of disturbance in the System (in cases where root cause diagnosis is required). In mode-capable Systems, normative models of each mode are required as well.

(1998). Proceedings. Fourth Annual Symposium on Human Interaction with Complex Systems
(Cat. No.98EX120)(pp.90-98). Los Alamitos, CA: IEEE Comput. Soc. ix+235pp.

3. Automation designers must recognize the tradeoff between flexibility and management responsibility in mode-capable Systems; an increase in the former invariably incurs more of the latter. One key to this tradeoff is the distinction between single-function/multiple-strategy modes and multi-function modes. In the case of single-function/multiple-strategy modes, designers must be able to justify inclusion of such flexibility. Providing 'another way of doing the same task' cannot be considered sufficient justification for burdening Operators with additional monitoring duties.

4. The simplest suggestion for limiting mode confusion is to reduce the number of modes. Figure 2 demonstrates that such an approach would reduce the number of possible signal paths through the System. However, no criteria have been established for determining how many modes is too many. The control-theoretic perspective suggests that having more Controller or Interface modes than Plant modes may be indicative of excessive flexibility. Automated Controllers and Interfaces which offer a range of options that substantially exceed the range of modes inherent in the Plant are suspect. If designers can match Controller modes to Plant modes, then there will be natural mappings between System elements. The design goal, then, is to reduce the number of Controller and Interface modes to the number of Plant modes.

5. Plant, Controller, and Operator should have a well-defined, integrated design basis that provides for effective coordination between all three elements. This can be accomplished by adopting a common Plant model to be employed by Operators, designers, and automation. To achieve this goal, we need one design framework that incorporates both human factors and control engineering factors associated with the design of automated systems [22].

Conclusions

The claims of automation developers (e.g. reduced workload, fewer errors, increased productivity) often go unchecked [24]. Unfortunately, so have the design solutions offered by some researchers. In part, the "improve the mental model" and "provide feedback" solutions have not been evaluated empirically because they are not concrete enough to consider as design strategies. In this article, we have presented a theoretical perspective which clarifies the System elements and agents which must be coordinated in order for automated systems to function effectively. Our control-theoretic perspective serves researchers by offering some conceptual clarity and precision and by suggesting focal points for empirical investigation. Furthermore, it serves designers by clarifying existing design recommendations and by offering new implications that can lead to improved human-automation interaction.

Acknowledgments

We are indebted to Asaf Degani, Stephanie Guerlain, and John Lee for their insightful comments on an earlier version of this manuscript.

References

1. Hirschhorn, L. (1984). Beyond mechanization: Work and technology in a postindustrial age. Cambridge, MA: MIT Press.

2. Norman, D. A. (1990). The 'problem' with automation: Inappropriate feedback and interaction, not 'over-automation'. Philosophical Transactions of the Royal Society of London, B 327, 585-593.

3. Billings, C. E. (1997). Aviation automation: The search for a human-centred approach. Mahwah, NJ: Erlbaum.

4. Wade, H. L. (1994). Regulatory and advanced regulatory control. Research Triangle Park, NC: Instrument Society of America.

5. Frank, P. M. (1990). Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy - A survey and some new results. Automatica, 26, 459-474.

6. Ashby, R. W. (1956). An introduction to cybernetics. London: Chapman and Hall.

7. Mumaw, R. J., Roth, E. M., Vicente, K. J., & Burns, C. M. (1996). A model of operator cognition and performance during monitoring in normal operations (AECB Final Contract Report). Pittsburgh, PA: Westinghouse Science and Technology Center.

8. Vicente, K. J. & Rasmussen, J. (1990). The ecology of human-machine systems II: Mediating "direct perception" in complex work domains. Ecological Psychology, 2, 207-250.

9. Riley, V. (1996). What avionics engineers should know about pilots and automation. In Proceedings of the 14th AIAA/IEEE Digital Avionics Systems Conference (pp. 252-257). Piscataway, NJ: IEEE.

10. Vicente, K. J., Christoffersen, K., & Hunter, C. N. (1996). Response to Maddox critique. Human Factors, 38, 546-549.

(1998). Proceedings. Fourth Annual Symposium on Human Interaction with Complex Systems
(Cat. No.98EX120)(pp.90-98). Los Alamitos, CA: IEEE Comput. Soc. ix+235pp.

11. Degani, A., Shafto, M., & Kirlik, A. (1995). Mode usage in automated cockpits: Some initial observations. In Proceedings of the 6th IFAC/IFIP/IFORS/IEA Symposium on Analysis. Design, and Evaluation of Man-Machine Systems (pp. 401-407). Cambridge, MA: MIT.

12. Sarter, N. B., & Woods, D. D. (1992). Mode error in supervisory control of automated systems. In Proceedings of the 36th Annual Meeting of the Human Factors Society (pp. 26-29). Santa Monica, CA: Human Factors Society.

13. Degani, A., & Kirlik, A. (1995). Modes in human-automation interaction: Initial Observations about a modeling approach. In Proceedings of the 1995 IEEE International Conference on Systems, Man, and Cybernetics (pp. 3443-3450). Piscataway, NJ: IEEE.

14. Degani, A. (1996). Modeling human-machine systems: On modes, error, and patterns of interaction (Unpublished doctoral dissertation). Atlanta, GA: Georgia Institute of Technology.

15. Sarter, N.B. (1995). 'Knowing when to look where': Attention allocation on advanced automated flight decks. In R. S. Jensen (Ed.), Proceedings of the Eighth International Symposium on Aviation Psychology (pp. 239-242). Columbus, OH: Ohio State University.

16. Sarter, N. B., & Woods, D. D. (1995). How in the world did we ever get into that mode? Mode error and awareness in supervisory control. Human Factors, 37, 5-19.

17. Norman, D. A. (1988). The psychology of everyday things. New York: Basic Books.

18. Aviation Week and Space Technology (1995). Automated Cockpits Special Report, Parts I and II. January 30 & February 7, 1995.

19. Sarter, N. B., & Woods, D. D. (1995). Autonomy, authority, and observability: Properties of advanced automation and their impact on human-machine coordination. In Proceedings of the 6th IFAC/IFIP/IFORS/IEA Symposium on Analysis. Design, and Evaluation of Man-Machine Systems (pp. 181-184). Cambridge, MA: IFAC.

20. Vakil, S. S., Hansman, R. J., Midkiff, A. H., & Vaneck, T. (1995). Feedback mechanisms to improve mode awareness in advanced autoflight systems. In Proceedings of the Eighth International Symposium on Aviation Psychology (pp. 243-248). Columbus, OH: Ohio State University.

21. Ashby, R. W. (1956). An introduction to cybernetics. London: Chapman and Hall.

22. Rasmussen, J. & Goodstein, L. P. (1987). Decision support in supervisory control of high-risk industrial systems. Automatica, 23, 663-671.

23. Vicente, K. J. & Rasmussen, J. (1992). Ecological interface design: Theoretical foundations. IEEE Transactions on Systems, Man, and Cybernetics, 22, 589-606.

24. Woods, D. D. (1996). Decomposing automation: Apparent simplicity, real complexity. In R. Parasuraman & M. Mouloua (Eds.), Automation and human performance: Theory and applications (pp. 3-17). Mahwah, NJ: Erlbaum.