

The MPC Elucidator: A Case Study in the Design for Human–Automation Interaction

Stephanie Guerlain, *Associate Associate Member, IEEE*, Greg A. Jamieson, Peter Bullemer, and Ronald Blair

Abstract—In this paper, we describe the design of a decision support system for operators of model-based predictive controllers (MPC). MPC is a form of advanced automatic control that is increasingly common in process operations due to its ability to control and optimize large sections of a process. A cognitive task analysis revealed that current operating displays, which rely on displaying tables of numeric information across several display pages, do not effectively support human operator monitoring, diagnosis, and control of MPC. This case study shows how we applied representation aiding and workspace management design principles to better support the human–automation interaction requirements of monitoring, understanding, and adjusting these complex, semi-autonomous process controllers. We show how effective user interface design can significantly reduce the complexity of operating with advanced automation, and can lead to improved understanding of how the automation works.

Index Terms—Automation, design methodology, display, graphical user interfaces, human factors, industrial plants, knowledge representation, process control, process monitoring.

I. INTRODUCTION

THE many challenges of human–automation interaction have been well documented. For example, automation can sometimes make easy tasks easier, while exacerbating hard tasks [1], [49]; automation can be “brittle,” only working well for the situations for which it is designed [15], [50]; and it may be difficult for operators of automation to maintain situation awareness [4], [10]–[12], [31], [32], maintain vigilance [31], or effectively calibrate the automation’s capabilities or current state [27], [33]–[35], [45], [50], [53]. Despite these issues, automation plays a clear role in improving the throughput, efficiency, and safety of many complex and dangerous operating environments. Empirical research is making steady inroads to understanding how human–automation interaction can be better supported (see e.g., [16], [17], [19], [23], [26], [29], [36]–[38]).

As advanced automation use proliferates in transportation, communications, and process control, familiar human–automation interaction problems tend to arise. In general, these can be stated as problems of how best to monitor the automation, diagnose any problems, and make effective control changes (see [31]). Moreover, it is becoming increasingly necessary to predict the future behavior of highly autonomous controllers [12].

Manuscript received July 7, 2000; revised November 16, 2001. U.S. Patents are pending regarding the subject matter discussed in this paper.

S. Guerlain is with the Department of Systems and Information Engineering, University of Virginia, Charlottesville, VA 22904-4747 USA (e-mail: guerlain@virginia.edu).

G. A. Jamieson, P. Bullemer, and R. Blair are with Honeywell Labs, Minneapolis, MN 55418 USA.

Publisher Item Identifier S 1083-4427(02)01485-6.

In this paper, we discuss the design of a user-interface for model-based predictive controllers (MPC), a class of automation technology that is being employed with increasing frequency in refining, pulp and paper, and grinding operations worldwide. We first identified human–automation interaction issues using a cognitive task analysis methodology. We then created a redesign based on the principles of 1) providing representational aids to show the constraints and interrelationships of the controller algorithms and 2) providing a coherent workspace that minimizes the amount of window manipulation and cognitive data integration required to support operators’ information needs. These techniques have rarely been applied to elucidate the workings of a highly automated system (although see [42] for a notable exception).

Designing an effective user interface for these types of controllers was extremely challenging due to the complexity of the controller algorithms, the amount of information that is potentially relevant, and the complex interactions of the underlying process. It is hoped that the presentation of this case study will be useful to other researchers and practitioners interested in applying representation aiding and workspace management to effectively support human interaction with highly automated systems.

In Sections II and III, we discuss the design strategies of representational aiding and minimizing workspace management requirements. We then move on to descriptions of model-based predictive control (MPC), the refining environment in which it is often employed, and the current state of the operator interfaces. Following this domain description, we relate the findings of a cognitive task analysis of MPC use in petrochemical refining, as well as a cognitive work analysis of the automation technology itself. The largest section of our paper is devoted to a case study of the design of the MPC Elucidator, a user interface designed to support human–automation interaction with MPC. We discuss in detail how the design addresses the information requirements more effectively than the original user interface design.

II. REPRESENTATION AIDING

The goal of representation aiding, broadly speaking, is to represent relevant domain, task, and system constraints through visual properties of the display, and thus encourage people to perceive these relationships with little cognitive effort (see [2], [3], [5], [7]–[9], [13], [18], [20], [21], [24], [25], [37], [39]–[41], [44], [46]–[48], and [55] for some examples and discussion). The representation of domain constraints and relationships through graphical user interface components

is based largely on the work of Gibson [14]; who argues that people can be attuned to invariant relationships in the environment without having to compute those relationships. Thus, if one can map the structure of such invariant domain properties onto a visual form, it may be easier for humans to recognize abstract concepts or relationships. These and various related concepts have been studied under the headings of representation aiding [52], ecological interface design [48], and the semantic mapping principle [2].

A. Representation Aiding Strategies

One strategy in representation aiding is to map higher-order relationships onto emergent features of the display that can be perceived by the viewer. Conversely, if no corresponding domain relationship exists, then one should avoid using a visual representation that has an emergent feature, as the visual representation has no corresponding meaning. Thus, extraneous use of graphics can be misleading and should be avoided.

A second representation aiding strategy is to represent the dimensional properties of a variable appropriately [55]. For example, object shape is a categorical representation. If one uses shape to represent an interval property such as magnitude, the observer must relate the meaning of a shape to the magnitude of the variable. It is more appropriate to use object size to represent such a variable, because the observer can directly perceive that a larger object is greater than a smaller object, which has direct relationship to the ordinal dimension of the variable being represented.

A third representation aiding strategy is to put data into its appropriate context. For example, in assessing the value of a parameter, one needs to know the current values versus the expected values, e.g., to know if a value is within normal constraints or exceeding constraints. Displaying a number without reference to expected ranges and setpoints takes the data out of context, and requires that people remember the setpoints and ranges of interest and make mental calculations to determine if a variable is “normal” or not. Often, displaying a value in analog form makes it easier to directly see how a variable is performing relevant to regions of interest.

Appropriately designed representational aids can significantly minimize the cognitive complexity of a task. Representations can be quite powerful, and very effective, if the information that is being represented accurately maps to the relevant information in the domain of interest. Zhang and Norman [55] have studied the *representational effect*, showing that people can easily solve a problem that is informationally equivalent to the normally difficult Tower of Hanoi problem if the constraints on the problem are represented externally through equivalent physical constraints, thereby making wrong moves impossible to perform.

B. Advantages

The major strength of analog representations of data as opposed to text-based display is that many relationships can be conveyed directly using visual properties of the display. In doing so, access to the embedded knowledge is granted through perception of the display, as opposed to doing mental calculations

to infer the desired information. For example, in monitoring a process, people are often interested in such higher-level questions as: How is the process behaving? Are key parameters increasing, decreasing, going out of safe limits, or operating normally? Sometimes, this kind of information can be mapped directly into a graphical form, so that the user can obtain relevant information by simply glancing at the display.

Visual representations are also good at preserving the spatial, topological, and geometric properties that are important for certain kinds of reasoning tasks, such as in solving physical or geographic problems. In trying to understand the representational benefits of a diagram, Larkin and Simon [25] noted that object properties could be indexed by their location, rather than by an explicit label. Furthermore, many properties can be represented simultaneously in the same location, and these properties may all be relevant to the problem.

In a text-based display that only shows raw data values in numerical form, the viewer must remember the set points and relationships of interest, compare those values with others that may or may not be displayed, and perform mental calculations to determine the required information. Over-reliance on digital forms makes it difficult to put data into context, makes it difficult to highlight events, creates the problem of fleeting data, and increases the need for the user to navigate through virtual data space to collect and integrate related data [30], [51].

C. Potential Challenges and Limitations

Despite the strengths of the representation aiding approach, there are several challenges to applying this technique. One potential problem with a representation aiding approach is that, whereas the designer of the display may be able to encode properties of the domain into properties of the display, users may have difficulty decoding that display, i.e., not knowing how to properly interpret the display [52]. Each representation makes some information about a problem salient while making other information more difficult to see [20], [26], [28]. Thus, we run a risk of highlighting some information at the expense of hiding other information.

A second issue is that, in many complex domains, the number of potentially important relationships is very large; sometimes too large to be represented with only one visual representation. In these situations, representation aiding can be preceded by modeling efforts that parse the information space in context sensitive ways [30]. The results of such analyses can be used to highlight important relationships as they become relevant for a given task situation. In this approach, one develops a set of representations that are sensitive to the person’s problem solving context, and makes more detailed data available as a person narrows down the search space.

A third limitation is that visual representations are not necessarily the best means to portray all types of information effectively. For example, procedural information may be better represented with a message-based display [38] or a hybrid display that uses both text and graphics [23]. Thus, the designer must be aware of when the use of representational aiding is appropriate given the information presentation requirements.

Finally, the design and implementation of effective representations takes considerable skill and insight. It is often difficult

even for experienced designers to develop effective representations and many of them require specialized, nonstandard programming techniques to implement. Thus, representational aids are often difficult to include in a design because they are too difficult and expensive to develop.

D. Conclusion

The preceding discussion demonstrates that there are many potential advantages and limitations to the use of representational aiding. It is important to point out that, in many cases, good design practice can lead to an exploitation of those advantages and avoidance of the limitations. Presenting successful design evolutions of representational aids is a valuable step in helping others to develop these design skills. This is particularly true in the case of representational aids for complex automation, because very few examples exist in the literature (cf. [39]).

III. WORKSPACE MANAGEMENT

Workspace management refers to the window manipulation, command input, and navigation activities required when working with computer-based systems [54]. Workspace management activities take away from the primary task (e.g., process monitoring and control), as users must ask themselves, “What information do I need, where is it located, and how do I call it up?” Due to the large amount of potential information that can be displayed on the screen and the generally fixed screen area, users must manage their workspace carefully. This involves making decisions as to what information to call to the forefront, at the expense of potentially missing important information on screens that are not being displayed [17], [30]. Woods and colleagues have likened this to looking through a “keyhole,” as only a small portion of the big picture can be viewed at a time [51].

There is a tradeoff between having more information available on one screen and preserving spatial separation between data elements. In the former case, the designer risks cluttering the display, while in the latter case, the designer risks forcing the user into excessive navigation as well as imposing a memory load. If the task does not require parallel access to the data, then having it spread across screens is not as problematic as when the task requires that the data be integrated by the user. Therefore, one design strategy is to determine what corresponding information needs to be viewed in parallel and group that information on the same screen. In general, the goal is to minimize workspace management as much as possible. A recent study by Burns showed that subjects were able to perform diagnoses better with an integrated display that overlapped all the required information in the same spatial area, than with a display that spread the same information onto separate windows [5]. The extreme density of information in Burns’ integrated display challenges commonly held notions about what is considered “cluttered.” Given that effectively combining related information into a coherent “picture” is one of the goals of representation aiding discussed above, these two design strategies should be complementary.

IV. REFINERY OPERATIONS AND MODEL-BASED PREDICTIVE CONTROLLERS

In Sections II and III, we discussed two design techniques for building interfaces for complex systems. In this section, we shift our attention to a description of one particular class of such systems and an increasingly common form of advanced control employed there.

A. Refinery Operations

Refineries and other process plants typically encompass a large physical area, with scores of multi-storey towers, hundreds of pumps and vessels, and thousands of sensors, controllers, instruments and valves. Such plants are typically subdivided into functional units. Operations teams comprised of field operators and board operators are tasked with controlling one or more units of the plant. Field operators are responsible for physical interaction with the unit (e.g., making rounds, taking readings and samples, and adjusting manual valves). Board operators use schematics, trend graphs, and alarm pages to monitor and control the process from a control room via a distributed control system. A board operator can change the setpoint of variables under regulatory control (e.g., flows, levels, pressures, or temperatures) within his/her unit. The operator can trend each variable, evaluate its associated alarms, and potentially see where that variable is in a schematic display.

B. Model-Based Predictive Control

Model-based predictive controllers (MPC) are multi-input, multi-output automatic controllers that take over much of the monitoring and control responsibility for a section of the process [22]. They are designed to optimize the process (e.g., maximize production variables or minimize utility costs) subject to various process parameter constraints. Well-designed MPCs can keep the process running smoothly and push production as well as, or better than, most operators. Because of its high profit potential, this advanced automation technology is being introduced into petrochemical, pulp and paper, and grinding operations throughout the world.

MPC uses an empirical process model to predict how changes in one process variable will affect others. There are three types of variables contained in the model, namely controlled variables (CVs), manipulated variables (MVs) and disturbance variables (DVs). A midsize MPC might have 20 to 30 CVs, 6–8 MVs, and 2–3 DVs.

- 1) *Controlled Variables (CVs)* are the process variables that MPC is trying to keep within constraints or at setpoint.
- 2) *Manipulated Variables (MVs)* are the variables (usually control valves) that MPC can adjust in order to keep all the CVs within their constraints while trying to meet optimization objectives.
- 3) *Disturbance Variables (DVs)* are those variables that have an impact on the process and can be measured, but not controlled (e.g., ambient air temperature). Knowledge of these independent variables can help MPC act to offset CV excursions before they take place.

Once installed, MPC is monitored and adjusted by board operators. The operator’s primary responsibility is to set high and

TABLE I
SAMPLE SITE VISIT AGENDA

1 hr.	Site control engineer(s) brief us on their process, indicating where MPC controllers have been being installed, and why.
1 hr.	Site control engineer(s) review current MPC operator interface(s) in use at their plant.
4 hrs.	Interviews with, and observations of, control engineers and operators to understand: <ul style="list-style-type: none"> • How training occurs and how effective it is. • How the introduction of MPC has changed operators' jobs. • How MPC failures are detected and diagnosed. • The changes and adjustments that should and should not be made. • The ease/difficulty of predicting the effect of controller changes. • User likes and dislikes and Frequently Asked Questions (FAQs)

low limits for each variable within a range specified by the process engineer. MPC then decides what the target value for each MV and CV should be based on its empirical model of the process, the constraints as defined by the engineers and adjusted by the operator, and optimization objectives. Thus, MPC is relying primarily on *range* control, delivering setpoints to the lower level regulatory controllers. Relieved of this lower level control task, the operator is theoretically available to control larger sections of the process. However, in addition to the traditional process schematics, trend, and alarm pages, the board operator must also monitor the MPC, using an additional set of display pages that are installed when the MPC is introduced to the unit.

V. COGNITIVE TASK ANALYSIS

A. Method

We conducted a cognitive task analysis [43] to understand how and when operators currently interact with, and ideally should interact with, MPCs. We performed a second analysis on MPC itself to understand how this control technology is engineered. This second analysis followed the premises of cognitive work analysis [47], although the modeling frameworks normally associated with that approach were not employed. Both of these analyses were based on data gathered via on-site interviews with seven control engineers and observations and targeted interviews with ten operators, for a total of 45 hours of on-site data collection. Multiple operators at several different units were interviewed to get a cross-section of user experience with MPC, and to understand how different MPC characteristics (robustness, size, etc.) affected their use. A similar agenda was followed at each site (see Table I).

In addition to the on-site activity, our team extensively reviewed the MPC documentation and spent approximately ten hours working with designers of the automation to gain insight into the engineering foundations of MPC. Throughout the design process, these domain experts critiqued the new interface in terms of its faithfulness to the automation technology.

B. From Cognitive Task Analysis to Information Requirements

We did not explicitly use a formal modeling technique (such as GOMS, abstraction hierarchy, or the operator function model, [6], [28], [30], [48]) in performing our cognitive task analysis/cognitive work analysis (CTA/CWA). Rather, we developed a comprehensive set of information requirements based on the activities described above. We conducted several

iterative reviews of the information requirements with control engineers and operators to ensure completeness.

Next, we analyzed the information available in the current displays to support these activities. We found several deficiencies in information availability or information aggregation (i.e., either the information was not available at all or it was cumbersome to gather and integrate as required for task demands). Finally, we used these information requirements to develop design requirements for the novel interface. The design itself served as our only formal "model" of the operator-relevant domain information properties. Section VI describes this design process in more detail.

C. Issues With the Use of MPC

The cognitive task analysis revealed that it is very difficult for operators to interact effectively with MPC. Specifically, operators have difficulty *monitoring*, *diagnosing*, and *controlling* these advanced controllers. This is due to a combination of the complexity of the controller algorithms, the complex coupling of the large number of variables contained in the controller, and the dynamic nature of the controller's interaction with the process. However, the current user interface is also not well designed to assist operators in these primary tasks. Table II shows the relationship between generic human-automation interaction needs, several MPC-specific examples of those information needs, and the means by which the operator could get that information (if at all) with the current displays. The purpose of this table is to show how our cognitive task and work analyses led to information requirements, which in turn led to the identification of deficiencies in the current user interface. We can see how poorly the current displays meet information requirements by analyzing the number of steps and mental calculations necessary in order to gather the information necessary to make informed decisions.

1) *Monitoring*: During normal operations, an operator needs to periodically monitor the controller to determine if it is running effectively. Operators refer to this as establishing the 'health' of the controller. This is difficult to ascertain with the present displays for two reasons. First, information about the controller is spread across multiple display pages. One Summary page (or set of pages in the case of a large controller) lists the current value, predicted value, and high and low limits for each of the MVs. A second Summary page lists similar information for each of the CVs, and a third shows the current value for DVs. Fig. 1 shows a sample of one of the CV Summary pages. The effect of this organization is

TABLE II
HUMAN-AUTOMATION INTERACTION WITH CURRENT MPC DISPLAYS (EXTENSIVE NAVIGATION AND DATA INTERPRETATION REQUIRED)

Monitoring		
GENERIC HUMAN-AUTOMATION INTERACTION QUESTION	SPECIFIC INFORMATION THAT HELPS ADDRESS GENERIC QUESTION	HOW INFORMATION NEED IS MET (OR NOT) WITH CURRENT DISPLAYS
What are the automation's objectives?	Which variables are being optimized (either minimized or maximized or targeted)?	For each CV, DV and MV listed, navigate to corresponding detail page. If LINEAR OBJ COEF is negative, variable is set to be maximized, and vice versa. If QUAD OBJ COEF is nonzero, number displayed is the targeted value.
How well are the automation's objectives being met?	Are optimized variables near optimum?	After performing above steps, check each optimized variable compared to target.
What is the current state of the automation?	Which MVs and CVs are at their limit?	For each CV, DV and MV listed, compare value to Lo and High Limits.
	Is MPC currently making large moves? If so, why?	Can possibly be inferred by trending the setpoint of variables in the controller.
	Are an abnormally large number of MVs at their limit? This is an indication that the controller is losing control.	For each MV, compare Value to Lo and High Limits. Count number of MVs at a limit. Compare to "normal" count, if known.
Diagnosis		
Why is the automation performing poorly? What factors are constraining the controller? Is it possible to relieve the automation's constraints?	What is limiting the controller? e.g., What is preventing it from increasing feed?	Look at Gain/Delay Page to determine which variables are influencing feed. Navigate to corresponding MV, CV, and DV pages to see which of these variables are limited, as described above. If limited, decide if this is what is limiting the controller. If nothing found, navigate to regulatory pages to see if a valves is "wound up".
Control		
How do I make adjustments, taking into account system constraints and objectives of previous operators?	What is the widest allowable range for this variable (as defined by plant engineers, process constraints or best practices)?	Found on detail pages rarely viewed by operators, and not viewable while making control changes.
	What are the delta soft "cushions" set by the engineer?	Found on detail pages rarely viewed by operators, and not viewable while making control changes.
	What recent limit changes have been made for this variable and why?	Not displayed.

that relevant information is highly distributed throughout the displays, resulting in a "keyhole effect" [51].

The second reason that it is difficult to assess the health of the controller is that the data are presented in discrete elements. In order to assimilate higher level information about the controller, users must navigate through the displays to see what variables are currently at a limit, if a particular variable is predicted to be outside its limits, and to note which variables are set to be optimized. Because the current and predicted values of a parameter are displayed in numeric format next to the high and low limits for that variable, operators must do mental calculations to determine if a variable is closer to one limit or another, and to infer how wide the allowable range is.

Noting which variables are being optimized is a particularly intensive task. Operators must select each variable in turn to call up its detail page to see if a nonzero linear or quadratic optimization coefficient is set for that variable (see Fig. 2). To

do this, the operator would have to navigate to each CV and MV Summary Page and then click to see the detail page for each CV and MV. Once on these detail pages, the operator needs to check if the LINEAR OBJ COEF is nonzero. If so, then the operator needs to know that a negative value means that the variable is being maximized, and a positive value means that the variable is being minimized. Further, if the QUAD OBJ COEF is set to nonzero, the operator needs to know that the number displayed is the targeted value. These are a large number of steps, each requiring mental processing by the operator. For a controller with 36 variables, at least 40 screens would have to be viewed, with a minimum of 75 workspace navigation activities.

To get a sense of how the controller has been behaving over time, the user's only option is to call up a trend of one or more variables contained in the controller. There are no summary statistics or graphs that the operator can consult. Consequently, operators often rely on alarms or anomalous changes in the more

CV SUMMARY Page 1 of 3										
RX / REGEN CTL	ON	OFF	WARM	OPTIMIZING						
CV DESCRIPTION	STAT	VALUE	SS VAL	LO LIMIT	HI LIMIT	SETPOINT				
1	REACTOR BED TEMP	GOOD	980.75	980.48	975.00	990.00				
2	REGEN BED TEMP	GOOD	1344.2	1344.2	1260.0	1765.0				
3	REGEN EXCESS O2	GOOD	2.2390	2.2605	1.0000	6.0000				
4	RX/REGEN DELTA P	GOOD	4.0950	4.0000	3.2000	4.8000				
5	REGEN CAT SLV DP	GOOD	3.0229	2.8394	20.000	1.0000				
6	SPENT CAT SLV DP	GOOD	4.2597	4.3956	1.0000	20.000				
7	STRIPPER LEVEL	GOOD	55.986	55.405	20.000	100.00				
8	BLOWER AMP's	GOOD	285.11	284.59	0.0000	307.00				
9	WET GAS RPM's	DROP	752.67	768.41	720.00	780.00				
10	FEED HDR-PRESS	GOOD	79.984	80.027	0.0000	115.20				
11	FRAC BTMS TEMP	GOOD	661.89	661.85	660.00	665.00				
12	FRAC DELTA PRESS	GOOD	0.8120	0.8110	0.5000	0.8650				
13	BLOWER VLV OP	GOOD	54.417	54.174	0.0000	102.50				
14	WET GAS VLV OP	GOOD	39.342	39.493	5.0000	50.000				
15	RX PRED OCTANE	GOOD	91.837	91.830	87.000	100.00				

APPLCN	PROCSS	CV	MV	DV	STATUS	MV	CV	GAIN/	TREND
MENU	DISPLY	DISPLY	DISPLY	DISPLY	MSG	TUNING	TUNING	DELAY	DISPLY

Fig. 1. Example CV summary display.

CV DETAIL									
RX / REGEN CTL	ON	OFF	WARM	OPTIMIZING					
TAG	25ATCV01								
DESC	DCO YIELD								
SOURCE	25ATCV01.PV								
PV VALUE	579.3	STATUS	GOOD						
PRED VAL	579.36								
FUTURE	579.38	SP.LIM TRACKS PV	YES	NO					
SS VALUE	581.36	UPDATE FREQUENCY	=	<	CV LO ERROR WEIGHT	1.00			
		CRITICAL CV	YES	NO	CV HI ERROR WEIGHT	1.00			
SETPOINT		CONTROL THIS CV	YES	NO	PERFORMANCE RATIO	1.00			
LO LIMIT	400.00	# OF BAD READS ALLOWED	5		CLS LOOP RESP INT	54.800			
ACTIVE	400.00				FF TO FB PERF RATIO	0.50			
		LO LIMIT RAMP RATE	10.000		SETPOINT GAP	0.00			
HI LIMIT	600.00	HI LIMIT RAMP RATE	10.000		NUMBER OF BLOCKS	10.0			
ACTIVE	600.00	UNBIASED MODEL PV	379.35						

APPLCN	PROCSS	CV	MV	DV	STATUS	MV	CV	GAIN/	TREND
MENU	DISPLY	DISPLY	DISPLY	DISPLY	MSG	TUNING	TUNING	DELAY	DISPLY

Fig. 2. Example CV detail display.

traditional regulatory displays to alert them to potential problems with the controller.

2) *Diagnosis*: The difficulty of gathering and assimilating information also has an impact on diagnostic activities. In the course of their monitoring activities, operators will sometimes notice that the controller is behaving in an unusual manner. This is often due to the controller becoming constrained. A constrained controller is one that has exhausted its available degrees of freedom.¹ If MPC becomes constrained, it may take what seem like drastic measures, such as cutting overall production, in order to keep all of the variables in the controller

within constraints. If the operator can diagnose which variable is presenting the problem, it is often a simple solution to change the limit range on that variable, take that variable temporarily out of control, or make changes to other parts of the process to relieve that constraint.

Engineers often diagnose the root cause of constraint problems off-line by examining the model algorithm. They do this by using the gain/delay matrix that defines how the model parameters interact. Some board operators will mimic this diagnostic process at their workstation, by using the gain/delay matrix screen on their displays (see Fig. 3). However, conducting this kind of analysis with the current user interface requires navigating through several pages to gather information, hold it in memory, and then make inferences from the collected information. This is because the gain/delay page does not show the current status of each of the variables, so an operator must continuously move back and forth between the gain/delay page that

¹Available degrees of freedom are calculated as follows: Sum the MVs that are not at constraints. Subtract the sum of the CVs that are either at constraints or constrained to setpoint. If the resulting value is greater than zero, then the controller can maintain control. If the value falls below zero then the controller will shut down. Prior to shutting down, however, the controller will act to retain degrees of freedom. The resulting behavior can be very confusing to an operator.

ONLINE GAIN AND DELAY CHANGE											
RX / REGEN CTL	ON	OFF	WARM	OPTIMIZING							
CV DESCRIPTION	MV01	MV02	MV03	MV04	MV05	MV06	MV07	MV08	MV09	MV10	DV01
1	REACTOR BED TEMP	-1.0	2.0	-3.5	4.2		6.1	-0.5	0.25		
2	REGEN BED TEMP	4.0				5.9					
3	REGEN EXCESS O2	0.3		-1.0	2.0	-3.5	4.2		6.1	-0.5	0.25
4	RX/REGEN DELTA P	.12	-1.0	2.0	-3.5	4.2		6.1	-0.5	0.25	
5	REGEN CAT SLV DE	10.0	-3.0	1.0	-2.5	4.2		6.1	-0.7	0.70	
6	SPENT CAT SLV DE	-0.4		7.2		9.0					
7	STRIPPER LEVEL	12.0					-8.0		-2.0		6.9
8	BLOWER AMP's	-.60		3.0	5.2	-2.5	9.0			1.5	3.6
9	WET GAS RPM's	1.2						-3.5			
10	FEED HDR-PRESS	3.0			-5.5			.02	6.2		-8.3
11	FRAC BTMS TEMP	2.2		-7.3			4.5				2.1
12	FRAC DELTA PRESS	.04									
13	BLOWER VLV OP	5.1		4.4		2.6			-9.0		-0.06
14	WET GAS VLV OP	3.2	6.3		4.0			6.2		-.25	
15	RX PRED OCTANE	-0.4					4.3		7.0	-8.2	
Gain Multiplier		1.000		Deadtime Bias		0.000					
Gain		3.750		Deadtime		0.000		Max Deadtime		2.00	
APPLCN	PROCSS	CV	MV	DV	STATUS	MV	CV	GAIN/	TREND		
MENU	DISPLY	DISPLY	DISPLY	DISPLY	MSG	TUNING	TUNING	DELAY	DISPLY		

Fig. 3. Example gain/delay display.

shows the model information and the relevant MV and CV pages that show the current status of those variables. Remembering the relevant information while navigating across displays adds to the difficulty of this reasoning task.

3) *Control*: The primary operator interaction with MPCs is changing the high and low limit for any variable in the controller. This is accomplished by entering text values in the HILIM and LOLIM fields of the detail displays. The user interface offers little assistance in letting the operator know what is an acceptable limit change. The display does not show a history of previous limit changes, nor why those were made. Further, the trend graphs of a particular variable are not shown in relationship to the limits that were in effect, so it is not possible to see if a variable has been operating within, beyond, or close to limits. This type of information must be assembled from data on various display pages.

Operators are also given little information to assess how the health of the controller will be affected by changes to limits. Sometimes the limit change will have little immediate effect but will cause the controller to become unduly constrained in the future when process conditions change. Over a period of several weeks, operators tend to “clamp” the limits (make the allowable operating region tighter) due to local operating conditions. Eventually, these latent problems will limit the controller’s ability to control effectively in all situations.

D. Conclusion

Our cognitive task analysis provides insight into the challenges experienced by operators interacting with MPCs. In general, these challenges are typical of other forms of human-automation interaction. Operators interacting with automated systems must know the objectives of the automation, whether the objectives are being met, how to diagnose what might be limiting the performance of the automation, and how to assist the automation. When interacting with the automation, it is necessary to know what acceptable, feasible options are available,

taking into account safety constraints and operating objectives, so as to keep the overall system running in a safe and effective manner. As we have shown, the current MPC operating displays provide limited support for these generic human-automation interaction requirements. We therefore sought to design a set of interface displays that would help operators more effectively monitor, diagnose, and control MPC. Our goal was to use the representation aiding approach to make the task more perceptual for all phases of cognitive activity and to support seamless transition from one phase to another through improved workspace management. The result is the MPC Elucidator.

VI. MPC ELUCIDATOR

The MPC Elucidator user interface is shown in Fig. 4.² The user interface was motivated by the following design principles [18].

Design Principle 1: Use representation aiding design principles to map domain properties into corresponding graphical elements.

- 1.1: Use consistent color coding throughout the display.
- 1.2: Show variable information relative to limits.

Design Principle 2: Create a workspace that supports monitoring, diagnosis, and control.

- 2.1: Support periodic monitoring of the controller through the design of an overview display.
- 2.2: Support direct navigation from the overview display to more detailed information.
- 2.3: Show important context information when the user makes a control change, including past operator changes and predicted behavior.

²The MPC Elucidator was designed to be consistent with current display capabilities in petrochemical control rooms. The display is designed for a 21" full color monitor at a resolution of 1280 × 1024. Such displays are standard in current practice. The screen images shown in this article are prototype drawings rendered in Visio Technical 5.0. The displays are implemented as ActiveX Controls through Visual Basic 6.0.

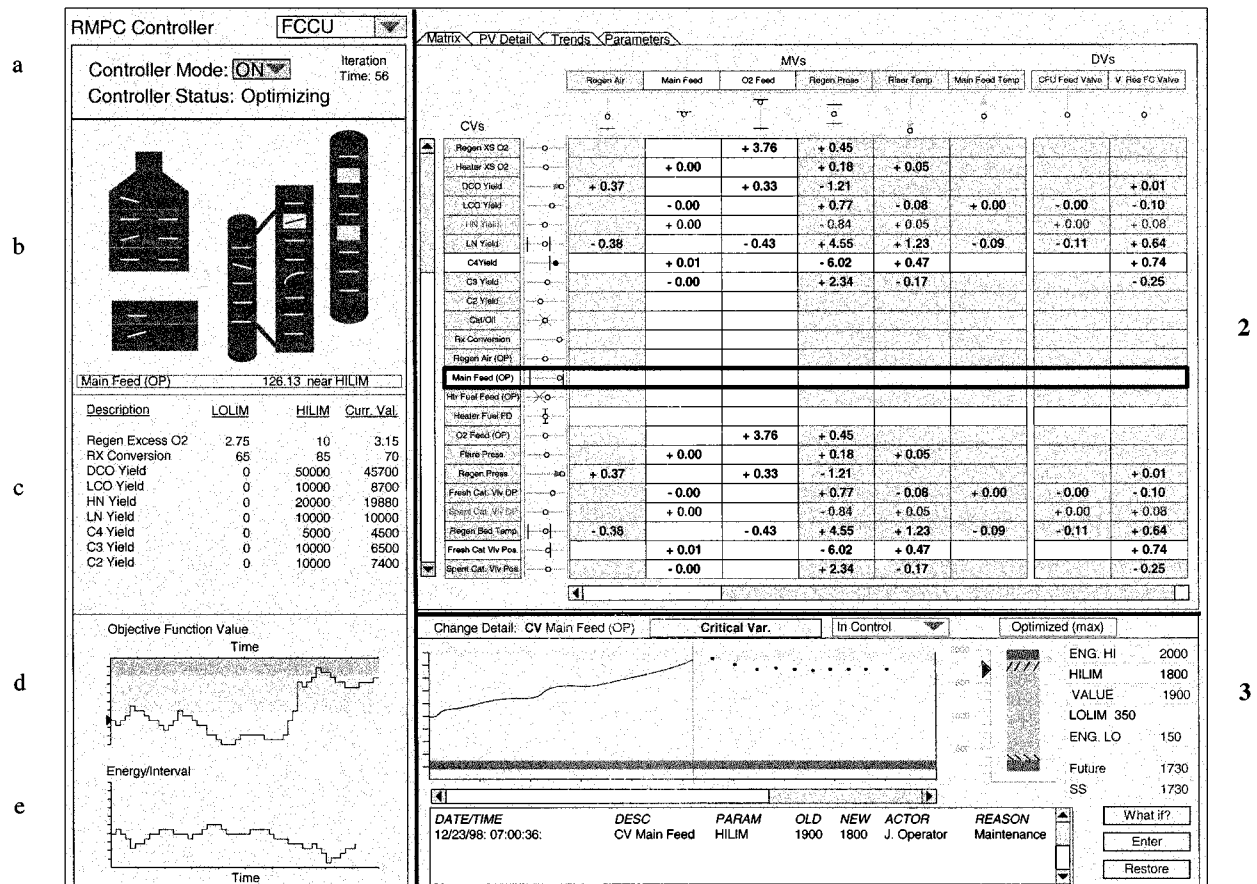


Fig. 4. Elucidator user interface.

The user interface is split into three Viewports³ (see Fig. 4). The left third of the screen (Viewport 1) is dedicated to overview information. This area is intended to alert the operators to anomalies that may need further examination. It is also intended to support infrequent viewing of the controller displays (as task demands only allow operators to check MPC periodically) by providing relevant overview and recent history information that would indicate a potential problem that requires additional investigation. The second viewport (Viewport 2) contains a tabbed dialogue view, which takes up the top two thirds of the remaining area of the screen. This view is designed to support more detailed monitoring activities of the controller, and to assist operators in understanding the interactions between variables in the controller to support diagnosing why the controller might be constrained. The third viewport (Viewport 3) shows more specific information about a particular variable, useful when changing a variable's limits. Thus, the three viewports of the screen generally support the three major cognitive activities of operators: monitoring, diagnosis, and control.

A. Overview

The Overview pane (Viewport 1) is designed to present information that the user will want to have access to at all times (to support monitoring). It is composed of five process views

presented in parallel. The PV Overview (1b in Fig. 4) and the two trend plots (1d and 1e in Fig. 4) are described in more detail below.

B. The Process Variable (PV) Overview Display

The PV Overview (see Fig. 5 for a detail) is a novel display that has been adapted from the work on mass-data-displays (MDDs) [3]. MDDs rely on the human's ability to detect abnormalities in visual patterns to alert operators to changes in a process. In the PV Overview, a *signature trend plot* represents each variable in the controller. This signature trend plot relies on an algorithmic technique that maps the recent behavior of the variable into one of the seven standard first and second order trend patterns: steady state, ramping up, ramping down, increasing at an increasing rate, increasing at a decreasing rate, decreasing at an increasing rate, or decreasing at a decreasing rate. When all of the variables are at steady state, the signature trend plots form a consistent pattern of horizontal lines. As changes occur, anomalous data breaks that pattern.

In addition, if the variable is within 1% of a limit, the signature trend plot turns yellow, and if the variable is outside its limits, the signature trend plot turns red. These colors are consistent with the existing alarm color-coding scheme and are repeated several times in the Elucidator. Further, the background highlighting of the icon indicates the results of *abnormality assessment* algorithms. These algorithms are designed to identify variables that exhibit short-term behaviors that are incon-

³The terminology employed in describing the interface is taken from [52].

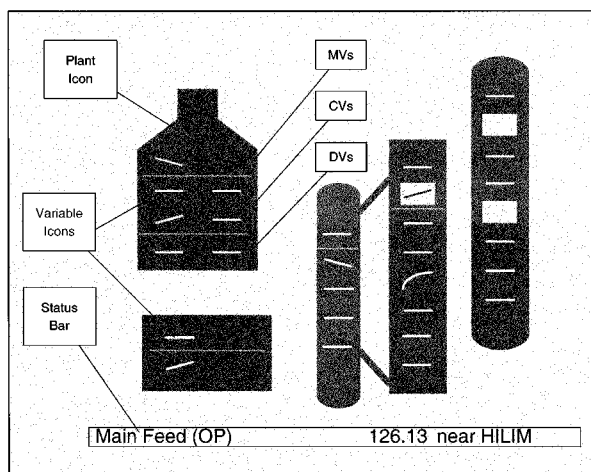


Fig. 5. PV overview display.

sistent with their long-term behaviors. For example, a variable that normally hovers near its low limit and has recently started trending toward its high limit might be flagged as abnormal. The variable need not be in violation of a control limit or in an alarm state. Rather, the algorithms are designed to raise operator awareness of dynamic behaviors that may lead to future problems (i.e., prognostic information). In summary, if all is normal, each variable in the PV overview display will have a steady state plot (i.e., a horizontal line), with no background shading or line coding. As abnormalities or changes take place, the icons represent those changes, alerting the operator to explore the situation in more detail.

The individual icons populate a static background graphic that depicts the major pieces of plant equipment (following [3]). The purpose of this arrangement is to help the operator localize a variable to a region of the unit under control. Within the background forms, the icons are arranged with MVs at the top, CVs below them, and DVs (if any) at the bottom.

The PV Overview display also serves as a stepping stone for further investigation or control. We do this in two ways: 1) by allowing the operator to “mouse over” a variable of interest, and obtain a status indicator below to identify the variable and any abnormality, if applicable, and 2) by allowing the operator to click on a variable to navigate to more detailed information about that variable in the corresponding process views on the right hand side of the screen. When a variable is selected in the PV Overview display, it will also be selected in the Matrix Display in Viewport, and the details about that variable will be displayed in the Change View in Viewport. This allows an operator to quickly navigate to details about a variable in corresponding views on the screen.

C. Objective Function Value and Energy Plots

Two plots are shown at the bottom of the Overview viewport. The first, the Objective Function Value Plot (1d in Fig. 4), provides insight into how well the controller is optimizing the process. Using this plot, the operator can determine if the optimization performance is slipping, initiating a targeted search for an explanation. To assist rapid monitoring, colored bands appear behind the plot to convey qualitative characterizations

of the objective value performance. These bands are shades of gray when the optimizer is performing well and the value of the function is low. As the value rises, and enters one of the gray regions, the band changes color (yellow and then red) to alert users that the optimizer performance is slipping. This technique of context sensitive display augmentation is employed again in the trending function in Viewport 3.

The Energy Plot (1e in Fig. 4) provides the operator with an indication of how hard the controller is working to adjust the manipulated variables. The controller calculates a single energy value at each control interval. Through the presentation of the time history of this value, the user can detect when the controller is making larger than normal adjustments to the process. This indication has two uses, depending on the context. If the operator is expecting the controller to make a large move, a spike on this plot would confirm that the controller is doing that. In contrast, if the controller makes a sudden large move that is not anticipated by the operator, the spike on the plot helps call his or her attention to it.

The inclusion of these two plots reveals the value of analyzing both the operator tasks (CTA) and the functionality of the automation (CWA). Both the objective function and energy information have always been components of the controller algorithm, but neither has been displayed to the operator previously. By performing both types of analysis, we were able to identify readily available information that would support an operator task that was not displayed directly with the previous interface.

D. Matrix View

The Matrix View is the primary process view that appears in Viewport 2. (Three other process views may appear in this Viewport, none of which are described here). The Matrix View is comprised of two graphical forms, the Gains Matrix and the Bubble Gauge Display (see Fig. 6 for detail). Each is described in turn.

1) *Gains Matrix*: The Gains Matrix is a table showing the predicted interaction between the variables in the controller. These values indicate how a one-unit increase in a manipulated or disturbance variable will affect each of the controlled variables. For example, for the controller shown in Fig. 6, a one-unit increase in regenerator air flow (Regen Air) will increase the regenerator pressure (Regen Press) by 0.37 units. This information can be used for diagnosis or predicting controller behavior. We have added a number of features to put the static model information into the context of the current status of the controller variables. This design approach minimizes the need to navigate to other displays to find this information.

Variables selected in the Gains Matrix (or in other views) are highlighted with a rectangle drawn around the entire column or row. This correspondence helps the user to locate references to common variables between views. Under normal conditions, the values in the matrix are displayed in black text on a white background. However, grayed out text in a column or row indicates that a variable has been dropped from control. A row or column whose background has been highlighted in white indicates a variable in a state that is costing the controller a degree of freedom (e.g., a CV constrained to setpoint or violating a limit).

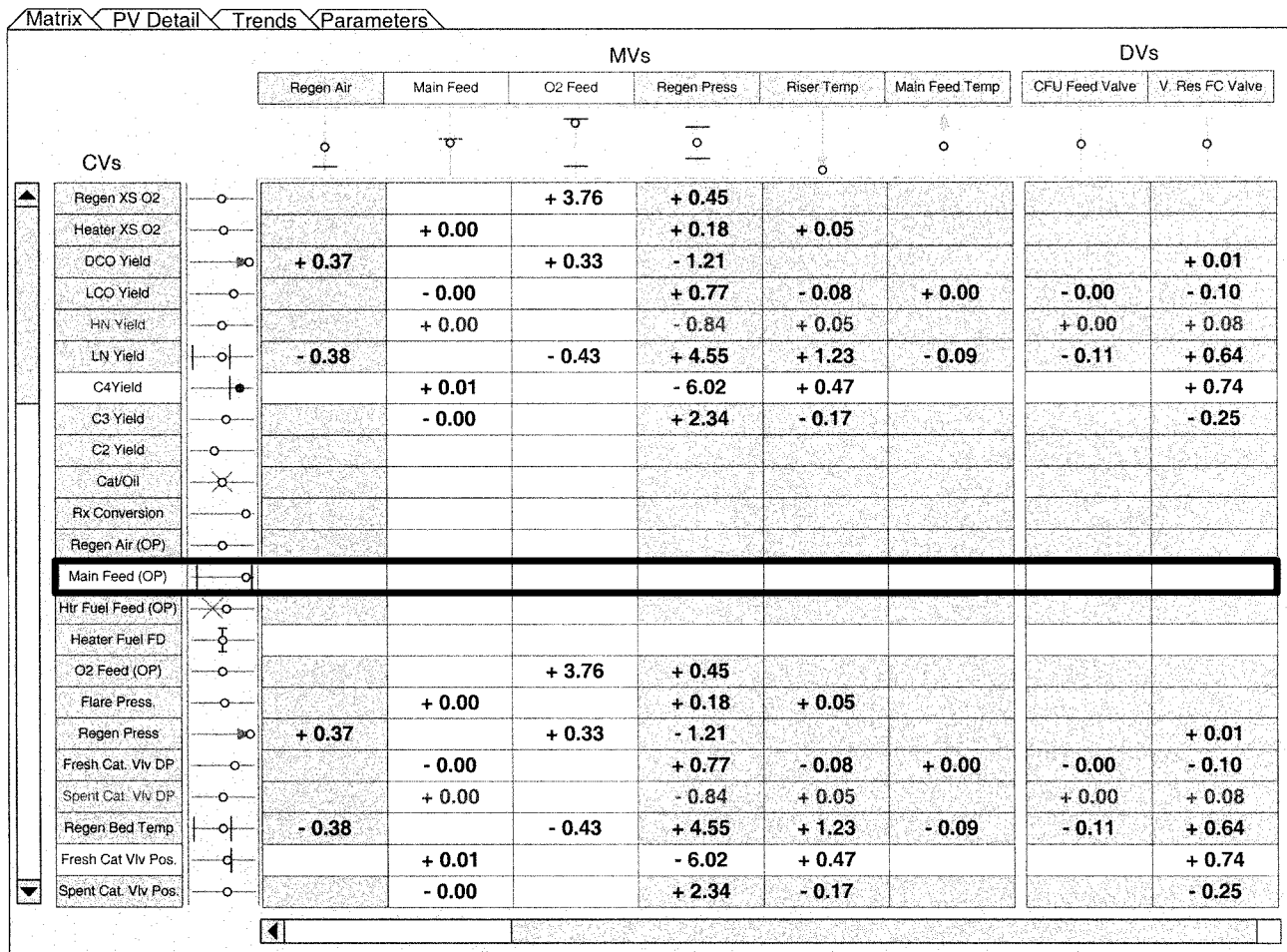


Fig. 6. Matrix display.

2) *Bubble Gauge*: A Bubble Gauge⁴ is shown for each variable (row and column) in the Matrix display. The gauge shows the current value of the variables in the context of both its control limits and its optimization objectives. The Bubble Gauge consists of an axis, a small hollow circle to represent the current value of the variable, black lines for both engineering and operator limits, and gray crosses and arrows to represent optimization information.

The span of the axis for each variable is normalized to the engineering limits (those limits that represent the widest allowed range for that variable as defined by the control engineer). These limits do not formally exist in the current MPC database, but the cognitive task analysis showed these limits would provide the operators with a context for knowing how the current limit values compare to the range deemed safe by a process engineer. Since each bubble gauge is normalized to this range, it is easy to scan across the bubble gauges in the matrix display to see which variables have had their limits “clamped” (tightened) by an operator.

Fig. 7 shows a representative set of potential states that a bubble gauge can assume. Examples a, b, and c show different normal states. The alarm color codes discussed in the signature trend plots of the PV Overview display are mapped onto the cur-

rent value circle (examples d and e). If the operator constrains a variable to setpoint, then the variable limits are shown with wing tips (example f). A wound up control valve takes on the appearance of example g. In the existing display scheme, the operator would have to identify each of these states by comparing several digital values.

The bubble gauge shows optimization information as well. If a variable is set to be maximized or minimized (i.e., linear optimization), a gray arrow is depicted in the direction of optimization (examples h and i). Quadratic optimization is depicted by a gray cross at the target value (example j). The current MPC displays show this optimization information indirectly on a variable’s detail page that operators rarely consult. On this detail page, the operator must know that a negative linear coefficient means “maximization,” and a positive linear coefficient means “minimization.” The bubble gauge alleviates the need to mentally encode this counter-intuitive relation.

The bubble gauge is a good example of the benefits of using a representation aiding approach. In isolation, each bubble gauge shows the variable’s current value with respect to its context. Scanning across the bubble gauges in the matrix display, one can quickly see which variables are at either a high or low limit, which have been constrained to setpoint, which have an optimization parameter associated with them, which have been “clamped” by an operator, and which are out of range. This is in

⁴The bubble gauge design was inspired by the redesigned box plot of Tufte [46, p. 62].

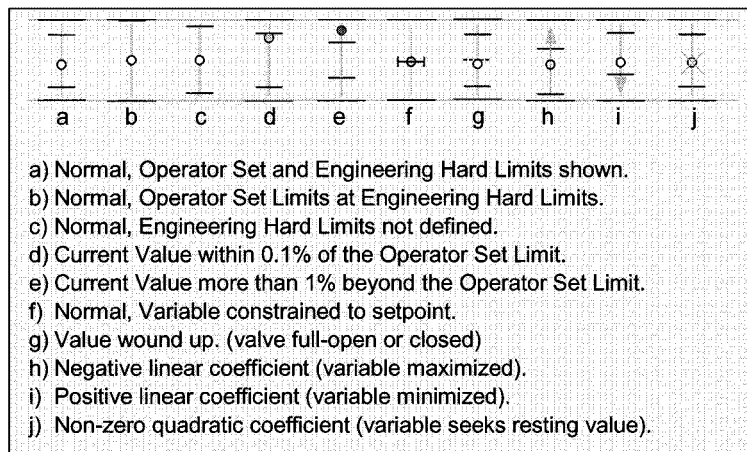


Fig. 7. Example bubble gauge states.

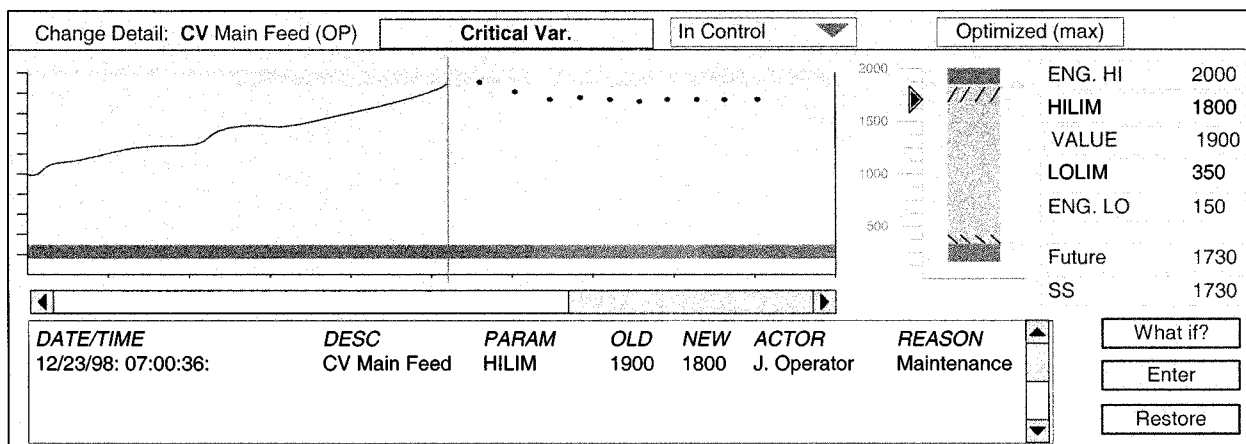


Fig. 8. Change view.

contrast to the current operating displays, which show the same elemental data in several pages of numeric text. With the bubble gauge, we have sacrificed the precision of numeric values for the advantage of showing several variables simultaneously. Given that our task analysis showed that numeric comparisons across variables were rarely necessary, we developed an analog representation that aids operators in making higher-level assessments about controller health. If exact values are needed, (such as when changing the variable’s limits), then the operator can click on the variable of interest in the matrix display, and all the details about that variable are displayed in the Change View (described as follows).

Taken together with the display of the gain/delay matrix, the operator can more readily see which variables are constraining the controller, and what other variables may be influencing a particular variable’s performance. We observed operators trying to do this kind of analysis with the current displays, but it is a cumbersome and difficult task. With the Elucidator, an operator can see the relevant information together on one screen, significantly easing the task difficulty.

E. Change View

The final viewport in the Elucidator interface (Viewport 3 in Fig. 4) is populated with information specific to a single process

variable. This Change View (see Fig. 8 for a detail) allows the user to view and manipulate the current limits in the context of the engineering limits, delta limits, and the current value of the variable. A trio of buttons allows the user to 1) view predictions of the impact a limit change will have using a “What If” prediction algorithm, 2) instruct the controller to carry out those changes, or 3) restore the limits to the values used in the previous control interval. The Trend History/Prediction Plot provides trending of the variable *in relation to operator set limits*. This is a noteworthy improvement upon current displays that do not show these limits. A corresponding history log shows details about previous limit changes to give the current operator a historical context for changing the limits.

1) *PV Gauge*: The PV Gauge in the Change View displays the current, future, and steady state values in relationship to all the relevant limits on the variable. We incorporate a direct manipulation feature to allow the user to change the LOLIM and HILIM settings (see Fig. 9 for detail) by clicking and dragging on the flags that extend from the gauge. Alternatively, the operator can choose to set the limits by typing into the fields next to the gauge.

The limits, current value, and future values are shown on a single scale to create a uniform frame of reference. The dark gray bar represents the magnitude of the engineering limit

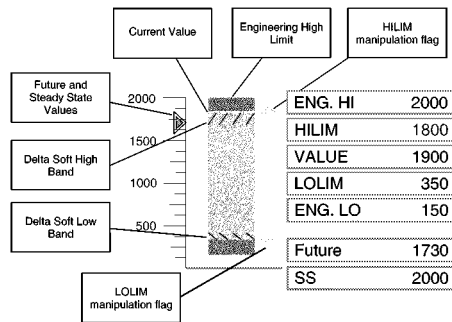


Fig. 9. The PV gauge control and its supporting fields.

range. A lighter gray bar is drawn inside that bar to show the range defined by the operator set limits. Thus, the dark gray bar represents room that the operator has to adjust the interior limits. Hashmarks inside the light grey region represent controller off-sets (called “delta-limits”) that MPC engineers define. The controller will try to maintain the variable within the light grey range, avoiding the hashmark region as a safety cushion. A thin arrow pointing to the scale from the right denotes the current value of the variable. The predicted future state value (at the end of the control horizon) and predicted steady state value (at the end of the prediction horizon) are also shown on the scale as two concentric triangles pointing to the scale from the left. These values show where the variable is predicted to go in the nearer and longer term, respectively. Color coding of these indicators is consistent with the signature trend plot and bubble gauge. In the example shown in Fig. 10(a), the current value is shown in yellow, but the predicted and steady state values are shown in black, indicating that the controller will be bringing that variable back into a safer region. In Fig. 10(b), one can see that the variable is predicted to go down, but stay within the allowable region. In Fig. 10(c), the current value is right on target, but the variable is predicted to go above its high limit.

2) *Trend History/Prediction Plot*: Trending packages are commonly employed in the process industries for monitoring, diagnosis, and evaluating the effectiveness of control actions. The Trend History/Prediction Plot in the Elucidator Change View (see Fig. 11 for detail) displays the historical values of a variable, the historical trend of the limits, and a prediction of the anticipated behavior. The vertical scale of the Trend History/Prediction Plot matches the scale specified by the PV Gauge control for this variable. This equivalence makes it easier to compare the trend to the PV Gauge control.

Two bars on the top and bottom edge of the plot depict the history of the constraint relationships. The bars reflect the difference between the operator set limits and the respective engineering limits. The current difference is projected across the prediction plot as well. Note that as the distance between the engineering limits and operator set limits increases, this bar becomes thicker. It is easy to see when a user has changed a limit because of the abrupt change in the width of the limit bar. Further, by displaying the limits in conjunction with the actual value, it is easy to see when a variable is, has been, or is predicted to be at or beyond its limit.

3) *Change Log*: The PV Gauge and trend controls are complemented by a change log just below the trend that automat-

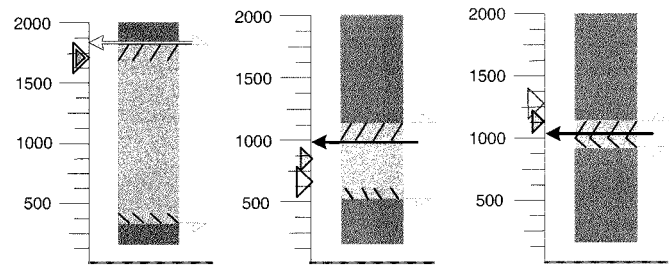


Fig. 10. Example PV gauge states.

ically documents critical information about limit changes and encourages users to give explanations for them. When the user enters a limit change, Elucidator creates a log entry with fields specifying the variable, parameter, old and new values, a date and time stamp, and the user. A cursor is then placed in the reason column. The intention of this feature is to document not only a limit change, but also the reason for it so that operators on later shifts can make decisions about whether that limit should be changed again. Clicking on one of these entries scrolls to the corresponding time on the trend graph above.

VII. HAVE WE BEEN SUCCESSFUL?

Table III presents a summary of how the Elucidator provides the information required for effective human–automation interaction with MPC. Comparing Table III with Table II, one can see the significant reduction in the number of steps and data interpretations. Information that previously required scores of coordinated mental and navigation activities can now be accessed directly at the main level of the display. For example, in our previous example, determining whether each of the optimized variables are meeting their objectives would require a minimum of 75 workspace navigation activities (moving from screen to screen). This can now be performed by glancing across the bubble gauges in the matrix display to see if any of the circles are not “on top” of any targeted value (represented by a gray X or arrow).

Unfortunately, despite our arguments for rigorous user testing, the product development group was convinced that our design is “obviously better” than the existing display suite and, as such, will be implementing Elucidator in a new product release. (This is an undoubtedly a common “Catch 22” for user interface designers. On the one hand, the designers promote the design in the hope that it will be accepted by management and development, while on the other hand, they need to argue that the designs should be tested and validated by users prior to release). We took what we knew from the literature, and from our cognitive task and work analysis of MPC, and employed our best engineering knowledge to design a system that should be successful according to the research on which it was based.

The Elucidator has undergone several iterations of comments and review by the developers of the MPC technology, and has been greatly enhanced by their inputs. These engineers’ in-depth knowledge of the controller algorithms enabled them to quickly identify missing portions of our design and offer suggestions for adding information to the displays. For example, our original bubble gauge design depicted minimum and

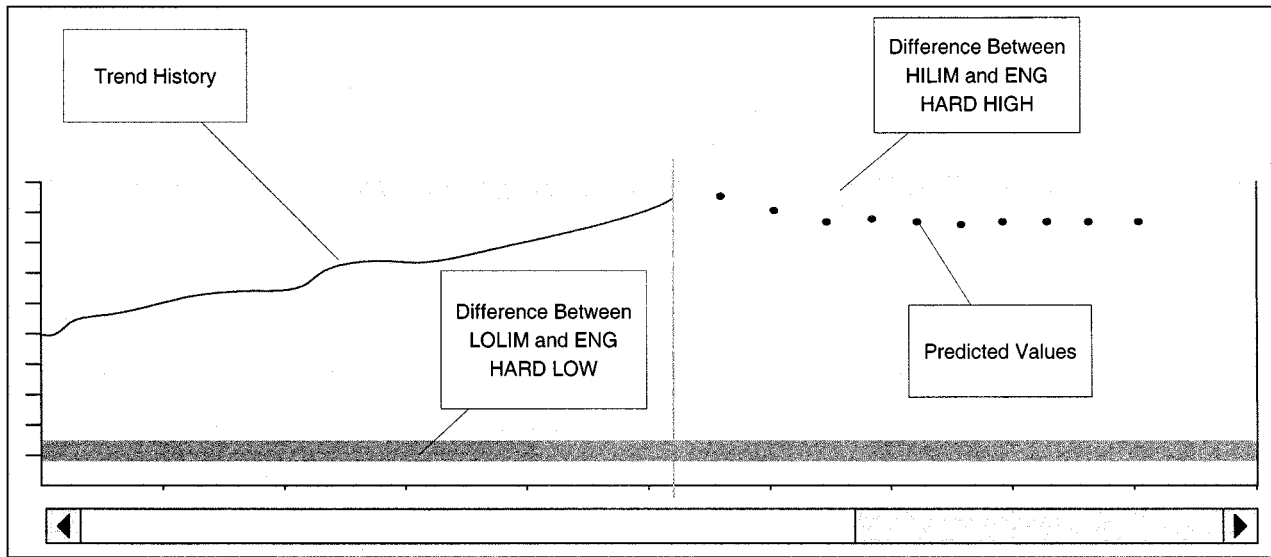


Fig. 11. Trend plot.

TABLE III
HUMAN-AUTOMATION INTERACTION WITH REDESIGNED MPC DISPLAYS (MINIMAL NAVIGATION AND DATA INTERPRETATION REQUIRED)

Monitoring		
GENERIC HUMAN-AUTOMATION INTERACTION QUESTION	SPECIFIC INFORMATION THAT HELPS ADDRESS GENERIC QUESTION	HOW ELUCIDATOR DISPLAYS MEET NEED
What are the automation's objectives?	Knowledge of which variables are being optimized (either minimized or maximized or targeted)	Look to see if a gray up arrow (maximization) or down arrow (minimization) or X (targeted value) is shown for any of the bubble gauges in Viewport 2.
How well are the automation's objectives being met?	Are optimized variables near optimum?	See if circles on bubble gauges are on top of arrows and X's.
What is the current state of the automation?	Which MVs and CVs are at their limit?	Look for yellow or red circles in the bubble gauges in Viewport 2 OR for similarly colored signature trend plots in Viewport 1.
	Is MPC currently making large moves? If so, why?	See if there are any non-horizontal lines in Mass Data Display in Viewport 1 OR look for a spike in the Energy Plot in Viewport 1.
	Are an abnormally large number of MVs at their limit?	Look for yellow or red circles in the bubble gauges in Viewport 2 OR for similarly colored signature trend plots in Viewport 1.
Diagnosis		
Why is the automation performing poorly?	What is limiting the controller? e.g., What is preventing it from increasing feed?	Use Gain/Delay Matrix and bubble gauges to see which variables are in fact at a limit (shown with a red or yellow circle) or have a wound-up variable (shown by a dashed horizontal line across Bubble Gauge).
What factors are constraining the automation?		
Is it possible to relieve the automation's constraints?		
Control		
How do I make adjustments, taking into account system constraints and objectives of previous operators?	What is the widest allowable range for this variable (as defined by plant engineers, process constraints or best practices)?	Look at range of bubble gauge in Viewport 2 OR range of PV Gauge in Viewport 3.
	What are the delta soft "cushions" set by the engineer?	Look for hashmarks on PV Gauge in Viewport 3.
	What recent limit changes have been made for this variable and why?	Look in scrolling change window in Viewport 3 OR on trend plot in Viewport 3.

maximum optimization parameters, but neglected to include quadratic optimization. A design review by the MPC developers revealed this shortcoming and prompted us to add the X marker on the bubble gauge. This type of iterative revision happened several times throughout the design and review process. The unexpected lesson in this experience is that the emerging design itself served as a good communication device between the design team and the developers of the controller algorithms. The cognitive work analysis and cognitive task analyses were actually enriched through the design of the display. Thus, the representations served as a model of our understanding of the important domain properties and relationships. When both groups agreed to a final design it was because both had come to an agreement on a) the functionality of the automation and b) the role of the operator.

The direction of knowledge sharing between the designers and the development team was not one-way. Whereas the development team was initially resistant to some of our design suggestions, we were able to demonstrate their utility and win acceptance. For example, engineering limits were previously considered almost incidental because they did not contribute to the MPC control algorithms. However, our demonstration and explanation of how they enhanced the context in which operators must make control decisions convinced the development team of their utility. This example serves to emphasize that human factors engineers have a contribution to make to the engineering design process.

VIII. CONCLUSION

This case study describes the development of a representation aiding system intended to help operators monitor, diagnose, and control a complex automated controller. The design was guided by cognitive task and work analyses, which helped define the information requirements of the user interface. This case study is the first of its kind that we know of that illustrates the process of developing a representation aiding decision support system for understanding and interacting with complex algorithmic automation. In addition, particular attention was paid to the design of the overall workspace, so as to support correspondence of information across displays. By reducing the need to navigate from one screen to another, operators can more smoothly transition from monitoring to diagnosis to control. Although our particular design is specific to the features of this type of model-based predictive controller, we believe that several of the individual representational aids have applicability to many process monitoring applications. Further, many of these design features may be applicable to other domains that use optimization algorithms.

Because the user interface was designed based on the cognitive task and work analyses conducted, it essentially served as our model of what was operationally relevant for the users of this automation. No other formal modeling representation was used, beyond developing a detailed list of information requirements based on our understanding of the automation's behavior and the cognitive activities and context of use by operators of the

automation. The representation aiding approach was thus useful not only for designing the human-automation user interface, but also for conveying our understanding of the automation's features and properties. As these representations were developed, they became more detailed and more correct, because the domain experts could see what was missing from our designs, explain the gap, and often suggest design revisions to make them more accurate.

REFERENCES

- [1] L. Bainbridge, "Ironies of automation," *Automatica*, vol. 19, pp. 775–779, 1983.
- [2] K. Bennett and J. Flach, "Graphical displays: Implications for divided attention, focused attention, and problem solving," *Hum. Factors*, vol. 34, pp. 513–533, 1992.
- [3] C. Beuthel, B. Boussoffara, P. Elzer, K. Zinser, and A. Tißen, "Advantages of mass-data-displays in process S&C," in *Int. Fed. Automatic Control Symp.*, Cambridge, MA, 1995.
- [4] C. Billings, "Human-Centered Aircraft Automation: A Concept and Guidelines," NASA Ames Res. Ctr., Moffett Field, CA, 103 885, 1991.
- [5] C. Burns, "Putting it all together: Improving display integration in ecological displays," *Hum. Factors*, vol. 42, pp. 226–241, 2000.
- [6] S. Card, T. Moran, and A. Newell, *The Psychology of Human-Computer Interaction*. London, U.K.: Lawrence Erlbaum, 1983.
- [7] B. Chandrasekaran, N. Narayanan, and Y. Iwasaki, Reasoning with Diagrammatic Representations: A Report on the AAAI Spring Symp., Mar. 25–27, 1992.
- [8] P. Cheng, "Problem solving and learning with diagrammatic representations in Physics," in *Forms of Representation*, D. Peterson, Ed. Bristol, U.K.: Intellect Books, 1996, pp. 47–66.
- [9] W. Cole and J. Stewart, "Human performance evaluation of a metaphor graphic display for respiratory data," *Meth. Inform. Med.*, vol. 33, pp. 390–396, 1994.
- [10] R. Cook, D. Woods, E. McColligan, and M. Howie, "Cognitive consequences of 'clumsy' automation on high workload, high consequence human performance," in *SOAR 90, Space Operations, Applications and Research Symp.*, 1990.
- [11] M. Endsley and E. Kiris, "The out-of-the-loop performance problem and level of control in automation," *Hum. Factors*, vol. 37, pp. 381–394, 1995.
- [12] M. Endsley, "Automation and situation awareness," in *Automation and Human Performance. Theory and Applications*, R. Parasuraman and M. Mouloua, Eds. Mahwah, NJ: Lawrence Erlbaum, 1996, pp. 163–181.
- [13] J. Flach and K. Bennett, "A theoretical framework for representational design," in *Automation and Human Performance*, R. Parasuraman and M. Mouloua, Eds. Mahwah, NJ: Lawrence Erlbaum, 1996, pp. 65–87.
- [14] J. Gibson, *The Ecological Approach to Perception*. Hillsdale, NJ: Lawrence Erlbaum, 1986.
- [15] D. Gregory, "Delimiting expert systems," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-16, pp. 834–843, 1986.
- [16] S. Guerlain, P. Smith, S. Gross, T. Miller, J. Smith, J. Svirbely, S. Rudmann, and P. Strohm, "Critiquing vs. partial automation: How the role of the computer affects human-computer cooperative problem solving," in *Human Performance in Automated Systems: Current Research and Trends*, M. Mouloua and R. Parasuraman, Eds. Hillsdale, NJ: Lawrence Erlbaum, 1994, pp. 73–80.
- [17] S. Guerlain and P. Bullemer, "User-initiated notification: A concept for aiding the monitoring activities of process control operators," in *Proc. 1996 Annu. Meeting of the Human Factors and Ergonomics Society*, Philadelphia, PA, 1996, pp. 283–287.
- [18] S. Guerlain, G. Jamieson, and P. Bullemer, "Visualizing model-based predictive controllers," in *Proc. IEA 2000/HFES 2000 Congr.*, 2000, pp. 3-511–3-514.
- [19] S. Guerlain, "Interactive advisory systems," in *Human Performance, Situation Awareness and Automation: User-Centered Design for the New Millennium*, Savannah, GA, 2000, pp. 166–171.
- [20] E. Hutchins, *Cognition in the Wild*. Cambridge, MA: MIT Press, 1995.

- [21] G. Jamieson and K. Vicente, "Ecological interface design for petrochemical applications: Supporting operator adaptation, continuous learning, and distributed, collaborative work," *Comput. Chem. Eng.*, to be published.
- [22] G. Jamieson and S. Guerlain, "Operator interaction with model-based predictive controllers in petrochemical refining," in *Human Performance, Situation Awareness and Automation Conf.*, Savannah, GA, 2000, pp. 172–177.
- [23] P. Jones and C. Mitchell, "Model-based communicative acts: Human-computer collaboration in supervisory control," *Int. J. Hum.-Comput. Stud.*, vol. 41, pp. 527–551, 1994.
- [24] M. Lanze, W. Maguire, and N. Weisstein, "Emergent features: A new factor in the object-superiority effect?," *Percept. Psychophys.*, vol. 38, pp. 438–442, 1985.
- [25] J. Larkin and H. Simon, "Why a diagram is (sometimes) worth ten thousand words," *Cogn. Sci.*, vol. 11, pp. 65–99, 1987.
- [26] C. Layton, P. J. Smith, and E. McCoy, "Design of a cooperative problem-solving system for enroute flight planning: An empirical evaluation," *Hum. Factors*, vol. 36, pp. 94–119, 1994.
- [27] J. Lee and N. Moray, "Trust, control strategies, and allocation of function in human-machine systems," *Ergonomics*, vol. 35, pp. 1243–1270, 1992.
- [28] J. Lee and T. Sanquist, "Augmenting the operator function model with cognitive operations: Assessing the cognitive demands of technological innovation in ship navigation," *IEEE Trans. Syst., Man, Cybern. A*, vol. 30, pp. 273–285, May 2000.
- [29] C. Mitchell, "Design strategies for computer-based information displays in real-time control systems," *Hum. Factors*, vol. 25, pp. 353–369, 1983.
- [30] C. Mitchell and R. Miller, "A discrete control model of operator function: A methodology for information display design," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-16, pp. 343–357, 1986.
- [31] N. Moray, "Monitoring behavior and supervisory control," in *Handbook of Perception and Human Performance*, K. Boff, L. Kaufman, and J. Thomas, Eds. New York: Wiley, 1986, vol. 2, Cognitive Processes and Performance, pp. 40.1–40.51.
- [32] D. Norman, "The 'problem' with automation: Inappropriate feedback and interaction, not 'overautomation'," in *Human Factors in Hazardous Situations*, D. Broadbent, A. Baddeley, and J. Reason, Eds. Oxford, U.K.: Clarendon, 1990, pp. 569–576.
- [33] R. Parasuraman and M. Mouloua, "Automation and human performance. Theory and applications," in *Human Factors in Transportation*, B. Kantowitz, Ed. Mahwah, NJ: Lawrence Erlbaum, 1996, p. 514.
- [34] R. Parasuraman and V. Riley, "Humans and automation: Use, misuse, disuse, and abuse," *Hum. Factors*, vol. 39, pp. 230–253, 1997.
- [35] R. Parasuraman, T. Sheridan, and C. Wickens, "A model for types and levels of human interaction with automation," *IEEE Trans. Syst., Man, Cybern. A*, vol. 30, pp. 286–296, May 2000.
- [36] R. Parasuraman, "Designing automation for human use: Empirical studies and quantitative models," *Ergonomics*, vol. 43, pp. 931–951, 2000.
- [37] W. Pawlak and K. Vicente, "Inducing effective operator control through ecological interface design," *Int. J. Hum.-Comput. Stud.*, vol. 44, pp. 653–688, 1996.
- [38] S. Potter, "The development of temporal and functional information displays to support cooperative fault management," Ph.D. dissertation, Ohio State Univ., Columbus, 1994.
- [39] D. Ranson and D. Woods, "Animating computer agents," in *Proc. 3rd Annu. Symp. Human Interaction with Complex Systems*, Los Alamitos, CA, 1996, pp. 268–275.
- [40] D. V. Reising and P. Sanderson, "Designing displays under ecological interface design: Toward operationalizing semantic mapping," in *Proc. Human Factors and Ergonomics Soc. 42nd Annu. Meeting*, Santa Monica, CA, 1998, pp. 372–376.
- [41] J. Retief, K. Lynch, and W. Pearson, "Panning for genes—A visual strategy for identifying novel gene orthologs and paralogs," *Genome Res.*, vol. 9, pp. 373–382, 1999.
- [42] V. Riley, B. DeMers, C. Misiak, and B. Schmalz, "The cockpit control language: A pilot-centered avionics interface," in *HCI-Aero*, Montreal, QC, Canada, 1998.
- [43] E. Roth and D. Woods, "Cognitive task analysis: An approach to knowledge acquisition for intelligent system design," in *Topics in Expert System Design*, G. Guida and C. Tasso, Eds. New York: ElsevierB. V., 1990, pp. 233–264.
- [44] P. Sanderson, J. Flach, M. Buttigieg, and E. Casey, "Object displays do not always support better integrated task performance," *Hum. Factors*, vol. 31, pp. 183–198, 1989.
- [45] N. Sarter and D. Woods, "Decomposing automation: Autonomy, authority, observability and perceived animacy," in *Human Performance in Automated Systems: Current Research and Trends*, M. Mouloua and R. Parasuraman, Eds. Hillsdale, NJ: Lawrence Erlbaum, 1994, pp. 22–27.
- [46] E. Tufte, *Envisioning Information*. Cheshire, CT: Graphics, 1990.
- [47] K. Vicente, *Cognitive Work Analysis*. Mahwah, NJ: Lawrence Erlbaum, 1999.
- [48] K. Vicente and J. Rasmussen, "Ecological interface design: Theoretical foundations," *IEEE Trans. Syst., Man, Cybern.*, vol. 22, pp. 589–606, 1992.
- [49] E. Wiener, "Human Factors of Advanced Technology ('Glass Cockpit') Transport Aircraft," NASA, 117 528, 1989.
- [50] R. Will, "True and false dependence on technology: Evaluation with an expert system," *Comput. Hum. Beh.*, vol. 7, pp. 171–183, 1991.
- [51] D. Woods, E. Roth, W. Stubler, and R. Mumaw, "Navigating through large display networks in dynamic control applications," in *Proc. Human Factors Soc. 34th Annu. Meeting*, Santa Monica, CA, 1990, pp. 396–399.
- [52] D. Woods, "Toward a theoretical base for representation design in the computer medium: Ecological perception and aiding human cognition," in *Global Perspectives on the Ecology of Human-Machine Systems*, J. Flach, P. Hancock, J. Caird, and K. Vicente, Eds. Hillsdale, NJ: Lawrence Erlbaum, 1995, pp. 157–188.
- [53] —, "Decomposing automation: Apparent simplicity, real complexity," in *Automation and Human Performance: Theory and Applications*, R. Parasuraman and M. Mouloua, Eds. Mahwah, NJ: Lawrence Erlbaum, 1996, pp. 1–16.
- [54] D. Woods and J. Watts, "How not to have to navigate through too many displays," in *Handbook of Human-Computer Interaction*, Second ed. M. Helander, T. Landauer, and P. Prabhu, Eds. Amsterdam, The Netherlands: Elsevier, 1997, pp. 617–650.
- [55] J. Zhang and D. Norman, "Representations in distributed cognitive tasks," *Cogn. Sci.*, vol. 18, pp. 87–122, 1994.



Stephanie Guerlain (M'95) was born in Norwalk, CT, in 1967. She received the B.S. degree in engineering psychology from Tufts University, Medford, MA, in 1990 and the M.S. and Ph.D. degrees in cognitive systems engineering from The Ohio State University, Columbus, in 1993 and 1995, respectively.

She was Principal Research Scientist at Honeywell Technology Center, Minneapolis, MN, from 1995 to 1999 and is now an Assistant Professor of human factors, Department of Systems and Information Engineering, University of Virginia, Charlottesville.

Dr. Guerlain is a member of the Human Factors and Ergonomics Society and the IEEE Systems, Man, and Cybernetics Society and serves as an officer and technical reviewer for both societies. She received a technical achievement award while working at Honeywell, and she and her colleagues received the 1999 Jerome Ely "Best Paper" award for a paper published in *Human Factors*.



Greg A. Jamieson was born in Cincinnati, OH, in 1971. He received the B.S. degrees in mechanical engineering and in psychology (with distinction) from the University of Illinois at Urbana-Champaign in 1996. He received the M.A.Sc. degree in mechanical and industrial engineering in 1998 from The University of Toronto, Toronto, ON, Canada, where he is currently pursuing the Ph.D. degree.

In 1996, he was a Research Associate with the Battelle Seattle Research Center, Seattle, WA. From 1998 to 1999, he was a Research Intern at Honeywell Laboratories (formerly Honeywell Technology Center), Minneapolis, MN, and has held the position of Research Scientist since 1999.

Mr. Jamieson is a member of the Human Factors and Ergonomics Society, having held several offices at the local chapter level. In 1999, he was recognized for an Outstanding Contribution to the Human Factors and Ergonomics Society by a Student Member.



Peter Bullemer was born in Minnesota in 1960. He received his Ph.D. in cognitive psychology from the University of Minnesota, Minneapolis, in 1985.

He is a Senior Principal Research Scientist at Honeywell Labs, Minneapolis. He directs the Abnormal Situation Management research and development consortium, and leads several user interface design and development projects. His research interests are in decision support and intelligent tutoring.

Dr. Bullemer received the Honeywell Lund Award for Leadership in 1998 and the Honeywell Technical

Achievement Award in 1999.

Ronald Blair was born in Denver, CO, in 1940. He received the B.S. degree in computer science from the University of California, Berkeley, in 1976 and the M.S. degree in computer science from California State University, San Luis Obispo, in 1977.

He was Principal Research Engineer at Honeywell Technology Center, Minneapolis, MN, from 1986 to 2000 and is now an independent software consultant. He has also taught in the computer science postgraduate program at the University of St. Thomas, Minnesota.

Mr. Blair received several technical merit awards while working at Honeywell.

This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright. In most cases, these works may not be reposted without the explicit permission of the copyright holder.