ISE

ISE-TR-91-93

HUMAN-COMPUTER INTERACTION AND COOPERATION FOR SUPERVISORY CONTROL OF LARGE-COMPLEX SYSTEMS

by

Toshiyuki INAGAKI
and
Gunnar JOHANNSEN

September 25,1991

INSTITUTE
OF
INFORMATION SCIENCES AND ELECTRONICS
UNIVERSITY OF TSUKUBA

Human-Computer Interaction and Cooperation for Supervisory Control of Large-Complex Systems

Toshiyuki Inagaki* and Gunnar Johannsen

Laboratory for Man-Machine Systems (IMAT-MMS), University of Kassel (GhK)
D-W-3500 Kassel, F.R. Germany

* On leave from: Institute of Information Sciences and Electronics
University of Tsukuba, Tsukuba 305 Japan

1. INTRODUCTION

Correct and immediate fault management is vital for maintaining system safety and reliability. The possibility that a chain of incidents may lead to a catastrophic accident must be reduced. One of central issues there is how we should allocate functions between human operators and computers. When a serious accident occurs in a large-complex system such as a nuclear power plant or an aircraft, many people say that human operators or pilots have made serious mistakes and errors. However we should not blame them. In many past accidents, the post-accident investigations have found that inappropriate design of the human-machine interface made human operators misunderstand what was really going on in the systems [1].

Many modern large-complex systems have *supervisory control* configurations [2], where the final responsibility lies with human operators, even though responsibilities can be shared between human operators and computers [3, 4]. We cannot be sure that decisions made by human operators are always correct and appropriate. If the human-machine interface is poorly designed, human operators would feel difficulty in identifying the current system state when the system exhibits some unusual behavior. Human operators may misunderstand the current situation by making an inappropriate matching with situations which they have experienced before [5]. The situation becomes worse if the system is a fastly responding system or functionally very complex and human operators are under time-pressure. It is known that the time-pressure is one of major factors that cause human errors.

An extreme way to avoid human errors is to control systems in a fully automatic manner, which is not wise, however, because our knowledge about large-complex systems is not always complete and perfect. We cannot give machines intelligence which is superior to our own intelligence. Moreover, sensing devices can produce false alarm signals. Human operators are indispensable, at least at the present time, for avoiding inappropriate system shutdowns due to false alarms. There still remain some issues to be analyzed further for

developing supervisory control configurations which are useful in cases of emergency of large-complex systems.

This paper gives three different schemes of interaction and cooperation between the human operator and the computer for supervisory control of large-complex systems, where the *degree of automation* [2] differs from scheme to scheme depending on how the human operator and the computer cooperate. In the first scheme (Scheme 1), the system is shut down in a full automatic manner immediately when an alarm is given. In the second scheme (Scheme 2), the human operator initiates an *alarm analysis* for evaluating alarm validity and for locating faults in the system when he or she is provided with an alarm. After some amount of time for the alarm analysis, he decides whether to shut down the system or not. In the third scheme (Scheme 3), the computer takes a fault-compensation action on the system immediately when an alarm is given, and then the human operator performs his alarm analysis. The fault-compensation action in Scheme 3 is introduced in order to: (1) mitigate possible system damages which can become larger in size while the human operator is analyzing the alarm, and (2) lessen the time-pressure on the human operator. Our aim is to find an appropriate scheme of cooperation, or *responsibility allocation*, between the human operator and the computer for cases in which an alarm is given. Each scheme is analyzed in a probabilistic manner by evaluating the conditional expectation of possible damage to the system, given that an alarm is generated.

Alarm analysis is not an easy task if the system is large and complex. The human operator may identify the system state incorrectly. Moreover, the human operator may fail to reach a definite conclusion on the system state; he may not be able to say either that the system is safe or that the system is unsafe. This can happen if the system is functionally very complex. For this kind of situation, we distinguish the following two types of safety-control policies [6]: (1) *safety-preservation* safety-control policies, where the system is shut down when its safety cannot be assured definitely, and (2) *fault-warning* safety-control policies, where the system is shut down only when it is regarded as being unsafe. The distinction between these types of safety-control policies is significant for Schemes 2 and 3. We show that an optimal scheme for responsibility allocation between the human operator and the computer has a time-dependent characteristic. The optimal scheme varies depending on the time point at which an alarm is given. There exists no single scheme which is always superior to other schemes.

We further discuss the problem whether the computer may be allowed to override the human operator for maintaining system safety. The human operator sometimes hesitates to shut down the system, even when an alarm is existent, because he fears to receive bad reputation from his company or society if he causes unnecessary shutdowns based on false alarms. Or, he may even cancel or ignore alarms if he has experienced many false alarms before. A catastrophe would be inevitable if the ignored alarm was a correct alarm. We show that there can exist some cases in which the computer must take countermeasure actions on the system for

avoiding catastrophic accidents, even when the human operator cancels or ignores alarms and gives the computer no command for taking the countermeasure action. We give necessary and sufficient conditions under which the computer should override the human operator's alarm cancellation. This sort of computer's override should be viewed as a kind of cooperation between the human operator and the computer. Allowance of this type of computer's override never reduces the importance of the human operator, who is indispensable in any supervisory control configuration.

2. MODEL DESCRIPTION

2.1. Schemes for Human-Computer Interaction and Cooperation

Consider a supervisory control system which consists of four units: a) system, b) alarm subsystem, c) computer, and d) human operator. The system, which is dynamic in nature, is monitored by the alarm subsystem continuously in time. The alarm subsystem consists of multiple divergent sensors which monitor different system variables. While the system is regarded as being safe, the computer controls the system according to instructions which have been given and approved by the human operator; the operator is a supervisor of the computer.

The alarm subsystem generates an alarm when an alarm criterion is matched. Alarm criteria can be described by a collection of production rules which combine multiple sensor signals. The alarm criteria are set initially by the human operator based on his knowledge about the system which might be represented by *fault trees* [7, 8] or some other related tools.

When an alarm is generated, the alarm is transmitted to the human operator and the computer. Who should be responsible in taking necessary actions to the system when an alarm is given? The human operator, or the computer? It depends on a scheme which specifies responsibility allocation between the human operator and the computer. We distinguish between the following three schemes for human-computer cooperation with different degrees of automation:

Scheme 1 : The computer shuts down the system in a fully automatic manner immediately when the computer receives an alarm from the alarm subsystem.

Scheme 2 : Upon receiving an alarm, the human operator initiates an alarm analysis in order to locate faults in the system. The alarm can be false, of course. The alarm subsystem sometimes generates a false alarm which can be caused by incorrect sensor data (sensors are

not always correct) or inadequate settings of alarm criteria [9]. The human operator must evaluate the validity of the alarm at the very beginning of the alarm analysis. After some amount of time, say T units of time, which is allowed for the alarm analysis, the operator decides whether to shut down the system or not, and sends necessary commands to the computer for doing so.

Scheme 3 : When the human operator receives an alarm, he firstly orders the computer by sending a command to take a fault-compensation action on the system, and then he initiates an alarm analysis. An example of fault-compensation actions may be to reconfigure or partially shut down the system for leading it into a milder operating condition. The aim of the fault-compensation is to: (1) mitigate possible system damages which may grow while the human operator is analyzing the alarm without taking any countermeasure action to the system, and (2) lessen the time-pressure for the human operator. After T units of time for the alarm analysis, the human operator decides whether to totally shut down the system or not, and sends necessary commands to the computer.

Scheme 1 has the highest degree of automation , and Scheme 2 has the lowest among these three schemes.

2.2. Safety-Control Policies

If the system is functionally very complex, it can happen that the human operator is not sure about safety or unsafety of the system; the human operator cannot say either that the system is safe or that the system is unsafe. What kind of action should he take to the system then, under Scheme 2 or 3, if no more information on the system is available? We distinguish the following two types of safety-control policies [6]:

(1) *Safety-Preservation* (SP) safety-control policy, in which the human operator can let the system be operative only when he is definitely sure about safety of the system. If the human operator cannot be sure of system safety, he must shut down the system, even when he has no definite evidence which proves unsafety of the system.

(2) *Fault-Warning* (FW) safety-control policy, in which the human operator shuts down the system only when he regards the system as being unsafe. The human operator lets the system run as far as he does not have a definite evidence which proves unsafety of the system.

2.3. Assumptions

1. The system states are classified into two categories: (1) Safe (S) states, where the system is in its normal operating conditions, and (2) Unsafe (U) states, where the system is in its failed operating conditions and, thus, an appropriate countermeasure, such as an immediate system shutdown, is necessary to avoid a catastrophic accident.

2. The alarm subsystem has three states: (1) Normal (NM) state, where the alarm subsystem identifies the system state correctly, (2) Positively failed (PF) state, where the alarm subsystem regards the system as being unsafe when the system is actually safe, and (3) Negatively failed (NF) state, where the alarm subsystem regards the system as being safe when the system is actually unsafe.

3. When an alarm is given to the human operator, he is expected to perform an alarm analysis. He may, however, ignore or cancel the alarm without performing an actual alarm analysis by thinking that the alarm must be false.

4. The human operator can make two types of errors in estimating the system state in the alarm analysis. (1) He regards the unsafe system as being safe, even though the alarm was correct. (2) He regards the safe system as being unsafe, although the alarm was false.

5. The human operator may encounter difficulty in judging the current state of the system in the alarm analysis. He cannot judge whether the system is safe or unsafe. This is not always a human error. Inadequate human-machine interface design for the control room can cause this kind of situation.

6. If the system, which is becoming unsafe, is shut down immediately upon an alarm, no damage is assumed on the system. The system suffers from damages if: (1) the safe system is shut down unnecessarily, or (2) the unsafe system is not shut down.

7. The system is a *fastly responding system.* If the system is in its unsafe operating conditions when an alarm is given, the system state can become worse while the human operator is analyzing the alarm. If the human operator takes Scheme 2 or 3 and the unsafe system is shut down T time units after the alarm generation, then the system damage is larger than in the case in which the system is shut down immediately upon the alarm under Scheme 1.

3. PROBABILISTIC ANALYSIS OF SCHEMES

We analyze properties of each Scheme k (k=1,2,3) by evaluating the conditional expectation $D_k = E[Z \mid A, \text{Scheme } k]$ of system damages Z (random variable), given that an alarm has been generated, where A denotes the event that an alarm is generated.

3.1. Scheme 1

The conditional expectation of system damages under Scheme 1 is evaluated as follows:

$$D_1 = E[Z \mid A, \text{Scheme 1}] = z_s\, p_f + 0 \cdot p_c = z_s\, p_f \qquad (1)$$

where z_s = E[Z | safe system is shut down], p_f = P[false alarm | A], p_c = P[correct alarm | A].

3.2. Scheme 2

The performance of Scheme 2 depends on the choice of the safety-control policy. If the human operator adopts the safety-preservation (SP) safety-control policy, we have:

$$D_{2SP} = E[Z \mid A, \text{Scheme 2, SP safety-control policy}] =$$

$$= w\, z_a\, p_c + (1\text{-}w)\{z_s\, p_f\, (P["U"|S] + P["N"|S]) + z_a\, p_c\, P["S"|U] + z_d\, p_c\, (P["U"|U] + P["N"|U])\} \qquad (2)$$

where w = P[operator ignores the alarm | A], z_a = E[Z | unsafe system is not shut down], z_d = E[Z | unsafe system is shut down with time delay T], P["X"|Y] = P[operator judges that "the system is in state X" | system is actually in state Y], where the system state can be either safe (S) or unsafe (U), and "N" denotes the event that the operator hesitates to draw his conclusion on the current system state.

The alarm ignorance probability w takes various values depending on how often the human operator was faced with false alarms before. The frequency of false alarms depends on sensor characteristics and alarm criteria. P["N"|S] and P["N"|U] are dependent on the alarm criteria as well as on the inadequacy of the human-machine interface design. The probabilities of erroneous judgements, P["U"|S] and P["S"|U], represent the degree of how complex the system is.

If the human operator takes the fault-warning (FW) safety-control policy, we have:

$$D_{2FW} = E[Z \mid A, \text{Scheme 2, FW safety-control policy}] =$$

$$= w\, z_a\, p_c + (1\text{-}w)\{z_s\, p_f\, P["U"|S] + z_a\, p_c\, (P["S"|U] + P["N"|U]) + z_d\, p_c\, P["U"|U]\} \qquad (3)$$

3.3. Scheme 3

As in the case of Scheme 2, the performance of Scheme 3 is dependent on the choice of the safety-control policy. If the human operator takes the SP safety-control policy, then we have:

$$D_{3SP} = E[Z \mid A, \text{Scheme 3, SP safety-control policy}] = w\, z_a\, p_c +$$

$$+ (1\text{-}w)\{z_s\, p_f\, (P["U"|S] + P["N"|S]) + z_c\, p_f\, P["S"|S] + z_a\, p_c\, P["S"|U] + z_{dc}\, p_c\, (P["U"|U] + P["N"|U])\} \quad (4)$$

where $z_c = E[Z \mid$ fault-compensation action is taken unnecessarily to safe system$]$, $z_{dc} = E[Z \mid$ unsafe system to which a fault-compensation action was taken is shut down with time delay T$]$. It is natural to assume that $z_{dc} < z_d = E[Z \mid$ unsafe system is shut down with time delay T$]$.

If the human operator takes the FW safety-control policy, then we have:

$$D_{3FW} = E[Z \mid A, \text{Scheme 3, FW safety-control policy}] = w\, z_a\, p_c +$$

$$+ (1\text{-}w)\{z_s\, p_f\, P["U"|S] + z_c\, p_f\, (P["S"|S] + P["N"|S]) + z_a\, p_c\, (P["S"|U] + P["N"|U]) + z_{dc}\, p_c\, P["U"|U]\} \quad (5)$$

3.4. Criteria for Selecting an Optimal Scheme

We show that an optimal scheme varies depending on the time point when an alarm is given. The type of safety-control policies (SP or FW) is determined by the following conditions:

$$D_{2SP} < D_{2FW} \iff p_c^{-1}\, p_f < z_s^{-1}\, z_a\, P["N"|S]^{-1}\, P["N"|U] \quad (6)$$

$$D_{3SP} < D_{3FW} \iff p_c^{-1}\, p_f < (z_s - z_c)^{-1}\, (z_a - z_{dc})\, P["N"|S]^{-1}\, P["N"|U] \quad (7)$$

The above preference order relations are time-dependent, because $p_c^{-1}\, p_f$ is a function of time and is evaluated at the time point when an alarm is given:

$$p_c^{-1}\, p_f = a_{PF}\, f_a(t)\, \{1 - F_a(t)\}^{-1}\, f_s(t)^{-1}\, \{1 - F_s(t)\} \quad (8)$$

where $a_{PF} = P[$alarm subsystem enters PF state $\mid$ alarm subsystem leaves NM state$]$, f_Y and F_Y denote the probability density and the distribution function, respectively, for the life of unit Y

(Y= a for the alarm subsystem, Y = s for the system). The values of f_Y and F_Y are evaluated at the time t when an alarm is given. No specific probability distribution, such as an exponential or a Weibull distribution, is assumed for the life of the units.

In real systems which are currently in service, the type of safety-control policies (SP or FW) is usually fixed explicitly or implicitly: We may be allowed to say that the FW safety-control policy is a common practice, because human operators usually tend to avoid a system shutdown and to let the system work as long as possible. The preference order relations (6) and (7) state that we must use the safety-control policies properly depending on the time point at which an alarm is given. It is not wise to fix a single safety-control policy in advance and to use it at any circumstance. The choice of a safety-control policy should be time-dependent and flexible.

The following is a partial list of conditions which describe preference order relations among Schemes 1 through 3:

$$D_{2SP} < D_1 \Leftrightarrow p_c^{-1}p_f\{w + (1-w)P["S"|S]\} > z_s^{-1}z_a\{w + (1-w)P["S"|S]\} + z_s^{-1}z_d(1-w)\{1 - P["N"|S]\} \quad (9)$$

$$D_{2FW} < D_1 \Leftrightarrow p_c^{-1}p_f\{1 - (1-w)P["U"|S]\} > z_s^{-1}z_a\{1 - (1-w)P["U"|U]\} + z_s^{-1}z_d(1-w)P["U"|U] \quad (10)$$

$$D_{3SP} < D_{2SP} \Leftrightarrow p_c^{-1}p_f < z_c^{-1}(z_d - z_{dc})P["S"|S]^{-1}(P["N"|U] + P["U"|U]) \quad (11)$$

$$D_{3FW} < D_{2FW} \Leftrightarrow p_c^{-1}p_f < z_c^{-1}(z_d - z_{dc})(P["N"|S] + P["S"|S])^{-1}P["U"|U] \quad (12)$$

By checking these preference order relations, we can determine an optimal scheme (with a proper safety-control policy, SP or FW) at each time point when an alarm is given.

3.5. May Computer Override Decisions of Human Operator?

When an alarm is given, the human operator may ignore the alarm by hitting an alarm cancellation button without performing an alarm analysis if he has experienced many false alarms before; this is the case which we analyzed in Sections 3.2 and 3.3.

3.5.1. Scheme 2

Consider here that the computer is allowed to shut down the system even if the human operator hits the alarm cancellation button. It is natural for the computer to regard the system as being unsafe, once it receives an alarm from the alarm subsystem. Suppose that the human

operator takes Scheme 2 with the SP safety-control policy and that the computer shuts down the system if the human operator hits the alarm cancellation button. Then the conditional expectation of system damages, denoted as D_{2SP}*, is given by:

$$D_{2SP}^* = w\, z_s\, p_f + (1\text{-}w)\{z_s\, p_f\, (P["U"|S] + P["N"|S]) + z_a\, p_c\, P["S"|U] + z_d\, p_c\, (P["U"|U] + P["N"|U])\} \quad (13)$$

If the human operator takes Scheme 2 with the FW safety-control policy under the same situation, then the conditional expectation of system damages, D_{2FW}*, is evaluated as:

$$D_{2FW}^* = w\, z_s\, p_f + (1\text{-}w)\{z_s\, p_f\, P["U"|S] + z_a\, p_c\, (P["S"|U] + P["N"|U]) + z_d\, p_c\, P["U"|U]\} \quad (14)$$

By comparing (13) or (14) with (2) or (3), respectively, we obtain the following necessary and sufficient condition under which the computer is allowed to override the human operator's alarm cancellation:

$$p_c^{-1}\, p_f < z_s^{-1}\, z_a \quad (15)$$

which is a time-dependent condition and applies to both safety-control policies, SP and FW.

3.5.2. Scheme 3

Scheme 3 in Section 3.3 assumed that the human operator orders the computer to take a fault-compensation action on the system when an alarm is given. If the human operator cancels the alarm, no further action is taken to the system by the human operator nor by the computer. One way to prohibit the human operator's alarm cancellation under Scheme 3 is to modify the scheme so that the computer can take an automatic fault-compensation action to the system whenever the computer receives an alarm from the alarm subsystem. Once the fault-compensation action is taken automatically by the computer, the human operator cannot ignore the alarm any more. The human operator must analyze the alarm at least to judge whether the fault-compensation action was right or not.

If the human operator takes this modified type of Scheme 3 with the SP safety-control policy, then the conditional expectation D_{3SP}* of system damages is given by:

$$D_{3SP}^* = z_s\, p_f\, (P["U"|S] + P["N"|S]) + z_c\, p_f\, P["S"|S] + z_a\, p_c\, P["S"|U] + z_{dc}\, p_c\, (P["U"|U] + P["N"|U]) \quad (16)$$

Under the modified Scheme 3 with the FW safety-control policy, the conditional expectation D_{3FW}* of system damages is evaluated as:

$$D_{3FW}^* = z_s\, p_f\, P["U"|S] + z_c\, p_f\, (P["S"|S] + P["N"|S]) + z_a\, p_c\, (P["S"|U] + P["N"|U]) + z_{dc}\, p_c\, P["U"|U] \quad (17)$$

Thus, the necessary and sufficient condition under which the human operator is not allowed to cancel the fault-compensation action of the computer is given by:

$$p_c^{-1}\, p_f < \{z_s\, (1 - P["S"|S]) + z_c\, P["S"|S]\}^{-1}\, (z_a - z_{dc})\, (P["N"|U] + P["U"|U]) \quad (18)$$

when the human operator takes the SP safety-control policy, and

$$p_c^{-1}\, p_f < \{z_s\, P["U"|S] + z_c\, (1 - P["U"|S])\}^{-1}\, (z_a - z_{dc})\, P["U"|U] \quad (19)$$

under Scheme 3 with the FW safety-control policy.

In addition to inequalities (9)-(12), some more criteria are available for the preference order among schemes. For example, we have:

$$D_{2SP}^* < D_1 \Leftrightarrow p_c^{-1} p_f\, P["S"|S] > z_s^{-1} z_a\, (P["N"|U] + P["U"|U]) + z_s^{-1} z_d\, P["S"|U] \quad (20)$$

$$D_{2FW}^* < D_1 \Leftrightarrow p_c^{-1} p_f\, (P["N"|S] + P["S"|S]) > z_s^{-1} z_a\, (P["N"|U] + P["S"|U]) + z_s^{-1} z_d\, P["U"|U] \quad (21)$$

$$D_{3SP}^* < D_1 \Leftrightarrow p_c^{-1} p_f\, (z_s - z_c)\, P["S"|S] > z_a\, P["S"|U] + z_{dc}\, (P["N"|U] + P["U"|U]) \quad (22)$$

$$D_{3SP}^* < D_{2SP}^* \Leftrightarrow p_c^{-1} p_f\, (w\, z_c - z_s)\, P["S"|S] > w\, z_a\, P["S"|U] + \{z_{dc} - (1-w) z_d\}\, (P["N"|U] + P["U"|U]) \quad (23)$$

Note that the allowance of computer's override has no effect on the preference order relations between SP and FW safety-control policies within the same single scheme: Inequalities (6) and (7) hold without any modification even for cases where the computer is allowed to override the human operator's alarm cancellation.

4. NUMERICAL EXAMPLES

Assume that we are given the following set of data: $w = 1/10$, $P["S"|S] = P["U"|U] = 8/10$, $P["U"|S] = P["S"|U] = 1/10$, $P["N"|S] = P["N"|U] = 1/10$, $z_a = 500$, $z_d = 100$, $z_s = 100$, $z_{dc} = 50$ and $z_c = 20$. We illustrate, by some examples, how an optimal scheme (with an appropriate safety-control policy) is selected.

Example 1: Suppose that our available alternatives are Schemes 1 and 2, and that the computer is not allowed to override the human operator in any way.

We can find the best scheme by evaluating the criteria for order relations among schemes in Sections 3.4 and 3.5. Figure 1 illustrates how an optimal scheme varies depending on $p_c^{-1} p_f$, the value of which is evaluated by (8) at the time point when an alarm is given.

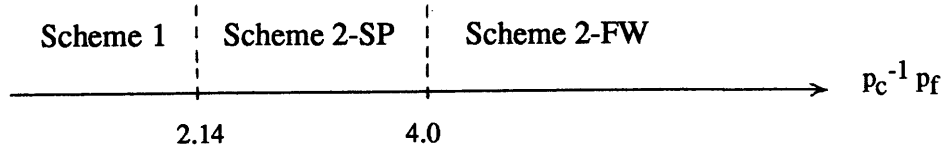

Fig.1. Optimal scheme for Example 1

As the value of $p_c^{-1} p_f$ becomes larger, we have a higher possibility that the given alarm is a false alarm. The above result fits well to our intuition that Scheme 1 is the strictest rule for maintaining system safety and Scheme 2 with the FW safety-control policy (denoted as Scheme 2-FW in Figure 1) is the least.

Example 2: Suppose that our available alternatives are still Schemes 1 and 2 only. However, let us allow the computer to override the human operator's alarm cancellation; the computer can shut down the system, if necessary, even when the human operator hits the alarm cancellation button upon receiving an alarm.

An optimal scheme, which varies depending on the value of $p_c^{-1} p_f$, is illustrated in Figure 2, where the asterisk symbol (*) expresses that the computer must shut down the system even if the human operator hits the alarm cancellation button. Under the scheme without an asterisk, the computer does not shut down the system when the human operator hits the alarm cancellation button.

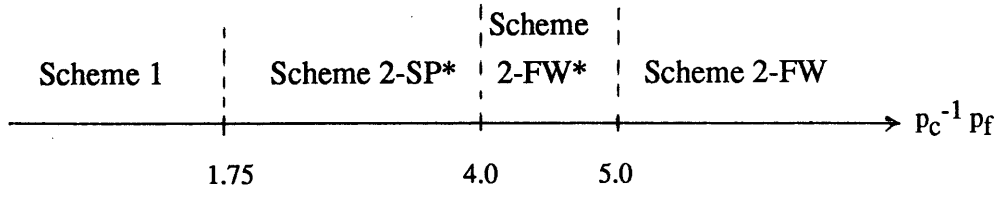

Fig. 2. Optimal scheme for Example 2

Comparing Examples 1 and 2, we can visually recognize the figure of merits for allowing the computer to override the human operator's alarm cancellation which can be inappropriate.

Example 3: Suppose that every one of Schemes 1 through 3 is available, but that the computer is never allowed to override the human operator's alarm cancellation. Figure 3 illustrates an optimal scheme:

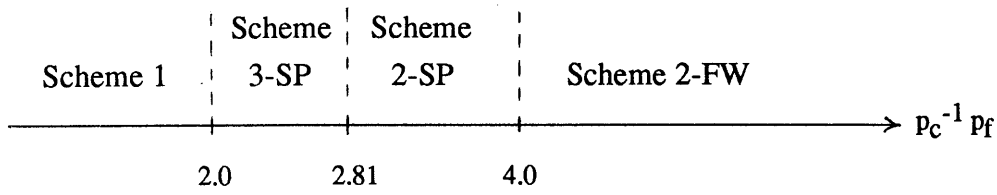

Fig. 3. Optimal scheme for Example 3

Comparison with the result of Example 1 would be useful for recognizing the effectiveness of a fault-compensation action for the fastly responding system, which was anticipated.

Example 4: Finally, consider the case in which we have the whole set of schemes and the computer is allowed to override the human operator's alarm cancellation if necessary. Figure 4 depicts an optimal scheme:

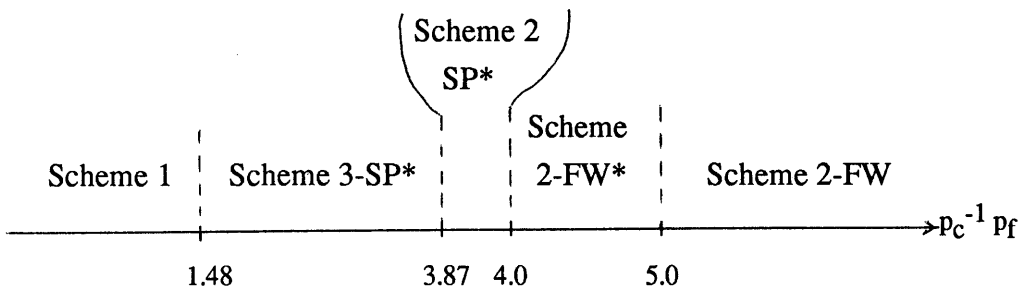

Fig. 4. Optimal scheme for Example 4

In the above Figure 4, we see again the significance of allowing the computer to override, when appropriate, the human operator's alarm cancellation.

5. CONCLUDING REMARKS

In this paper, we have analyzed some schemes for responsibility allocation between the human operator and the computer for cases of emergency in a fastly responding system. Even though we have restricted our consideration only to a fastly responding system, our analysis applies directly to a *slowly responding system* by just setting z_d and z_{dc} at some values which are much smaller than those for a fastly responding system. The state of a slowly responding system does not become worse very rapidly while the human operator is analyzing an alarm, even when the system is in its unsafe operating conditions. If the system is an extremely slowly responding system, we may even be allowed to assume that $z_d = z_{dc} = 0$, which makes the analysis for responsibility allocation quite easy.

As shown in Section 3 and illustrated with numerical examples in Section 4, an optimal scheme is situation-dependent. It should be chosen properly depending on various parameters which describe reliability characteristics of system components, possibility of the human operator's misinterpretation of given situations, and possible system damages in several different settings. Moreover, the optimal scheme is time-dependent, in the sense that the value of $p_c^{-1} p_f$, which is a factor in deriving an optimal scheme, is determined by the values of probability density and distribution functions for the life of the system and the alarm subsystem at the time point when an alarm is given (although this type of time-dependency is *macroscopic* compared to the *microscopic* time-scale for an alarm analysis in which the system behavior during the period of, for example, 10 to 30 minutes prior to the alarm can be a matter of concern for locating possible faults). It is, thus, almost impossible to derive an optimal scheme in an intuitive manner. One of the contributions of this paper lies in giving a systematic method for deriving an optimal scheme.

The complexity of the system makes it difficult for the human operator to recognize the system state correctly. Inadequate design of the human-machine interface enlarges the difficulty. We have given probabilistic models for analyzing how these factors can degrade system safety. Although we have given only simple examples in Section 4, there exists no necessity that parameter values must be simple. For example, even though we have considered the case in which P["N"|S] = P["N"|U] in Section 4, P["N"|S] can take a different value from P["N"|U], which depends on the specific design of the human-machine interface. Extensive analyses based on real data would be stimulative and useful from both theoretical and practical viewpoints.

We have also discussed whether the computer may override the human operator. We have shown that there exist cases in which the computer should be allowed to override the decision of the human operator. The situation which we set there was quite simple. Our model serves well, however, to prove the following:

We must not assume beforehand that the computer should be always subordinate to the human operator, or vice versa. Cooperation or responsibility allocation between the human operator and the computer should be situation- and time-dependent.

We need further investigations for establishing better human-computer partnership.

ACKNOWLEDGMENT

The research of this work was partially supported by the Alexander von Humboldt-Foundation, Federal Republic of Germany.

REFERENCES

1. J. Reason, *Human Error*, Cambridge University Press, Cambridge, 1990.

2. T.B. Sheridan, "Supervisory control", in G. Salvendy (Ed.), *Handbook of Human Factors*, pp. 1243-1268, Wiley, New York, 1987.

3. G. Johannsen, "Fault management, knowledge support, and responsibility in man-machine systems", In J.A. Wise and A. Debons (Eds.), *Information Systems: Failure Analysis*, pp. 205-209, Springer-Verlag, Berlin, 1987.

4. T.B. Sheridan, and others, "Supervisory control, mental models and decision aids", In J. Ranta (Ed.), *Analysis, Design and Evaluation of Man-Machine Systems,* (Proc. 3rd IFAC/IFIP/IEA/IFORS Conf.) pp. 429-435, Pergamon Press, Oxford, 1988.

5. D.A. Norman, *The Psychology of Everyday Things*, Basic Books, New York, 1988.

6. T. Inagaki, "Interdependence between safety-control policy and multiple-sensor scheme via Dempster-Shafer theory," *IEEE Trans. Reliability*, vol. 40, no. 2, pp 182-188, 1991.

7. J.W. Hickman, and others, *PRA Procedures Guide*, NUREG/CR-2300, USNRC, 1981.

8. E.J. Henley, H. Kumamoto, *Reliability Engineering and Risk Assessment*, Prentice-Hall, New York, 1981.

9. R.D. Sorkin, "Why are people turning off our alarms?", *Journal of the Acoustical Society of America*, Vol. 84, pp. 1107-1108, 1988.

INSTITUTE OF INFORMATION SCIENCES AND ELECTRONICS
UNIVERSITY OF TSUKUBA
TSUKUBA-SHI, IBARAKI 305 JAPAN

REPORT DOCUMENTATION PAGE	REPORT NUMBER ISE-TR-91-93

TITLE

Human-Computer Interaction and Cooperation for Supervisory Control of Large-Complex Systems

AUTHOR(S)

Toshiyuki Inagaki:
Institute of Information Sciences and Electronics
University of Tsukuba, Tsukuba 305 Japan

Gunnar Johannsen:
Laboratory for Man-Machine Systems (IMAT-MMS)
University of Kassel (GhK), D-W-3500 Kassel, F.R. Germany

REPORT DATE	NUMBER OF PAGES
September 25, 1991	14
MAIN CATEGORY: Reliability theory	CR CATEGORIES

KEY WORDS

human-computer interaction, responsibility allocation, supervisory control, safety control, fault-warning and safety preservation

ABSTRACT

Correct and immediate fault management is vital for maintaining system safety and reliability. One of central issues there is how we should allocate functions between human operators and computers. This paper gives three different schemes of interaction and cooperation between the human operator and the computer for supervisory control of large-complex systems, where the final responsibility lies with human operators and the *degree of automation* differs from scheme to scheme depending on how the human operator and the computer cooperate. Our aim is to find an appropriate scheme of *responsibility allocation* between the human operator and the computer for cases in which an alarm is given. Each scheme is analyzed in a probabilistic manner by evaluating the conditional expectation of possible damage to the system, given that an alarm is generated. We show that an optimal scheme for responsibility allocation between the human operator and the computer has a time-dependent characteristic. We further discuss the problem whether the computer may be allowed to override the human operator for maintaining system safety. We show that there can exist some cases in which the computer must take countermeasure actions on the system for avoiding catastrophic accidents, even when the human operator cancels or ignores alarms and gives the computer no command for taking the countermeasure action. We give necessary and sufficient conditions under which the computer should override the human operator's alarm cancellation.

SUPPLEMENTARY NOTES