



A Heterarchical Urgency-Based Design Pattern for Human Automation Interaction

Axel Schulte¹, Diana Donath¹, Douglas S. Lange^{2(✉)},
and Robert S. Gutzwiller²

¹ Universität der Bundeswehr München (UBM), Neubiberg, Germany
{axel.schulte,diana.donath}@unibw.de

² Space and Naval Warfare Systems Center Pacific (SPAWAR),
San Diego, CA, USA
{dlange,gutzwill}@spawar.navy.mil

Abstract. We document a Human-Autonomy Teaming design pattern to provide a means for an task management assistant to mitigate errors that may occur due to changes in urgency levels of tasks. Urgency can increase or decrease due to changes in the task environment, or through failure to begin execution of a task at the correct time. We discuss the structure and key aspects of the pattern and provide a sample implementation. We also discuss the key aspects of the human partner's performance that must be measured and considered in implementing such a pattern. Finally, we discuss known issues and other related patterns.

Keywords: Human-autonomy teaming · Pattern · Task management

1 Intent

Our intent is to provide a means for an assistant/associate system to mitigate erroneous behavior of an operator by a stepwise, increasing intervention/support. The interventions of the assistant/associate system range from alerts, messages, and suggestions, up to overrides in order to transition a dangerous situation into a normative (safe) one. The stepwise approach strives to keep the operator vigilant with respect to the task, and responsible for as long as possible for task accomplishment. Another objective of stepwise intervention is to avoid a degradation of the final work result, which may only be possible as long as any error caused by the human has no direct/immediate negative effect on the overall work objective. For this reason, we suggest a stepwise error correction only for errors which are still repairable before degradation is realized.

We consider two kinds of erroneous behaviors as related to human performance when interacting with automation:

1. Errors which occur when the human fails to take a necessary action (errors of omission) and;
2. Errors caused by a wrongly selected, wrongly executed, or improperly timed executed action (errors of commission).

The rights of this work are transferred to the extent transferable according to title 17 U.S.C. 105.

These two types of errors are found in studies of human automation interaction [1] and are especially critical aspects of human interaction when automation may make decisions [2]. Intuitively, with increased time pressure, humans may be more likely to accept automation recommendations or rely on the system in order to conserve mental resources. Reliance on aids may increase as operators reach their task saturation limits [3], but it is more complicated in determining if urgency itself dictates this relationship. One study that manipulated time pressure showed no relationship with automated aid reliance in an air traffic control context [4]. Nevertheless, these results may not apply as they dealt in decisions to use the automation, whereas the current design pattern instantiates an automated solution automatically to keep damage or errors from occurring.

In many contexts, task urgency is highly related to safety outcomes. In driving, a key factor in accidents is time following some emergent information, such as a truck pulling out onto the highway – is there enough time to avoid it (a physical limitation), and what actions must be taken through to task completion within that amount of time to avoid a serious accident. Methods which “create” more time (through reduction in speed, heads-up alerts, etc.) are then generally successful at enabling the human to respond more effectively. In the figure below, this would represent pushing the threshold for a task completion (e.g., maneuvering away from the truck) further toward the right where the human has time to respond and the response will be effective (top Fig. 1). The system could also respond in this case, but as discussed above, may need to be left idle to ensure the human remains aware and engaged in driving. There remain situations on the leeward edge of the urgency “continuum,” in which so little time is available for a safety-critical task to be accomplished that full automation is used and justified (bottom Fig. 1). In that case, the system may even lock the human from responding altogether to avoid any interaction issues.

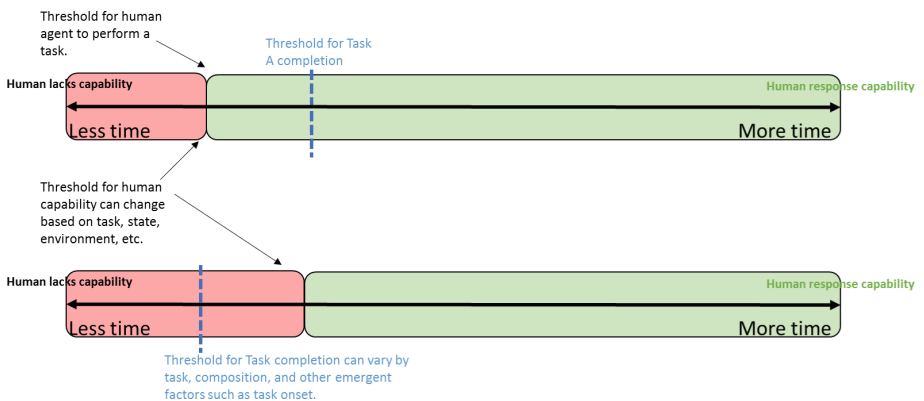


Fig. 1. Urgency continuum abstract representation to illustrate time limits on human capacity to perform tasks.

The current design pattern is better understood through examining the general cases – those in which intervention before error is possible by the human or the system. Other

domains that may make use of this continuum are in automated vehicles, in which it can be argued that humans need to be in a significant degree of control, in order to compensate for any system failures (using a full range of urgency). The type of intervention we recommend then lives within the urgency continuum, and depends on the urgency and better understanding it. As each task has a certain time window to be executed, the elapsed time of an omitted or wrong executed task plays a major role in the choice of the adequate intervention of any assistant system. Thus in order to enact any of these solutions, a taxonomy of sorts should be built or generated that allows for some speculation and characterization of task performance by the system and the human. Because these can be done, and methods are widely known to accomplish it [5, 6] we focus instead on the interactions rather than the task properties.

2 Motivation

Highly automated systems, which are able to detect human errors, are also typically designed in a way that they immediately correct human errors. This approach dispossesses the human operator of his/her task immediately, independent of whether the human operator still has enough time, mental resources, and/or the ability to correct the error on their own. If such an error correction occurs often and the correction is relatively reliable, this may cause complacency effects in the human. The human may put a miscalibrated, high amount of trust in the corrective actions of the automation, therefore neglect his own tasks and consequently lose vigilance or situation awareness [7]. In general, negative effects of automation may be avoided by actively involving the human operator in the error correction process itself. Therefore, within the current design pattern, we suggest a directed, stepwise-escalated error correction approach to support the human operator based on his/her needs, and the urgency of an emerging or an already occurred error.

This pattern will also provide a means for the human-autonomy team to adapt tasks and actions as urgency for completing tasks increases. This implies the need for repeated adaptation, as urgent tasks are completed, become less urgent, are abandoned, or are considered obsolete. Examples related to the importance of urgency are seen in autonomous assistants for aviation and driving. If the system determines that a collision risk is high from its sensor data, the system can then determine deadlines for the various forms of intervention based on the interactions between models of the autonomy and models of human performance. For our current purposes, we make the assumption that the autonomy's deadlines are generally later than the human's. As a human deadline approaches, urgency is increased and the system managing the tasks can invoke actions intended to reduce the risk that the deadline will be missed. These can include notifications, reprioritization of tasks, changes in methods for completing tasks, and task abandonment in the case of lower priority tasks (for example in terms of reward, or cost).

Important to this pattern is a definition of urgency, which we attempt to provide with regard to two task types. Consider a model of each available agent (human and/or machine) capable of performing tasks within a work process. For each task, the predicted performance is a probability density function representing the probability that an

instance of a task type will require any particular time by the agent for completion. For each task instance in each agent’s queue, there is an associated task type, and a required completion time. Tasks can be decomposed into subtasks and methods, such as in the structure in the figure below, if desired (and these form composite tasks), but is not necessary. Urgency of an atomic task is the simple probability that the task will not be completed on time given the current resource allocation (in this case, which agent is assigned and their capacity and state, for example) (Fig. 2).

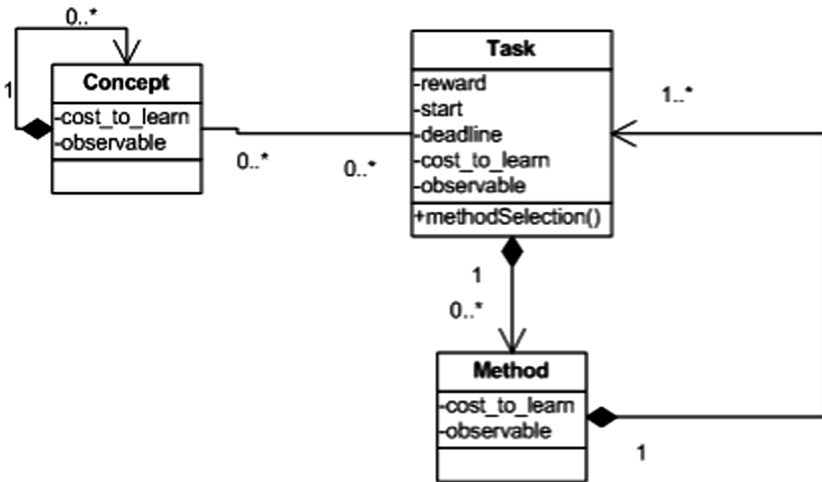


Fig. 2. Possible task structure.

Urgency for a composite task in contrast must be propagated based on a task decomposition pattern. Composite Tasks (one model of hierarchical WProc) are composed of methods, with an ‘or’ relationship. Any of the methods can be selected to complete the task. Methods are composed of tasks, all of which must be completed to complete the method. Therefore, our definitions are as follows:

- A task’s urgency is the probability that it will not be completed in time if it is atomic; else if
- A method’s urgency is the maximum urgency present among its tasks.
- A hierarchical/composite task’s urgency is the (median | mean | max | min | E(x) | CoV@R(r, x)) urgency of the methods it is composed of. The exact method is up to the system designer.
- A networked WProc can be modelled using composite tasks for the purposes of urgency. There are at least 3 subtasks (receive input, perform task, and send output). “Perform task” is likely to be further decomposed to account for aspects that must periodically wait for input.

3 Applicability

Use this pattern when you want to mitigate complacency effects in a human operator, which might be caused if technical systems always attempt to perform an immediate and automated error correction.

Do not use this pattern for time critical support of human operators, where you need immediate function or task adoption by the technical system. This design pattern is more apt for tactical or strategic tasks where the human can contribute, rather than reactive tasks.

4 Structure

Figure 3 illustrates the collaboration between human and agents or automation within a work system, which enables a “step-by-step” error correction. It uses a graphical language, which is defined in [8]. On the left hand side both the human and the assistant system are workers. Workers know the given work objective. They are able to understand and pursue the work objective according to their abilities. The relation between both workers (human and assistant system) is a heterarchical one (blue connection); therefore within this cooperation schema there is no hierarchical order guiding the involved workers. Instead, each worker acts on its own initiative to pursue the overall mission goal. In this example case, the assistant system continuously monitors both the human and the work process, and supports the human step-by-step in the achievement of the overall mission goal. While pursuing the work objective both workers use the available tools. These tools are subordinate to the worker, as shown on the right hand side of the work process. These tools are in a delegation/supervisory control relationship to the worker (green connection). Workers receive tasks, instructions, or commands which they have to execute in order to achieve the given/delegated tasks. Within this work system we describe two distinct kind of agents, each which have differing purposes [9]:

- Purpose of the delegate agent: Control of conventional automation to reduce or remove human task loading.
- Purpose of the assistant system: Mitigation, i.e. prevention or correction of erroneous behavior of the human, in order to maximize safety but avoid complacency behaviors.

5 Participants

As depicted in Fig. 3 the participants are a human and at least one intelligent agent, the agent on the worker side, referred to as “assistant system.” The human is in charge of the achievement of the given mission objective. The assistant system lets the human accomplish his/her tasks as long as no errors (errors of omission, errors of commission) emerge. In case these errors occur, the assistant system chooses an adequate intervention strategy according to the urgency of the task, which has either to be

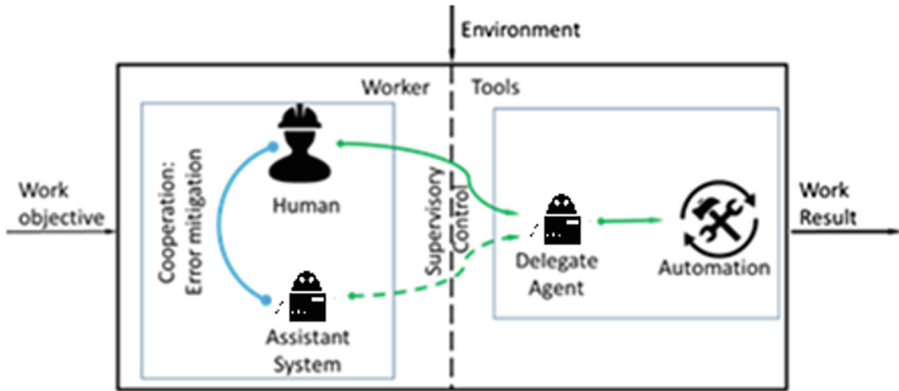


Fig. 3. The high-level elements for a design pattern, “increasing urgency/step-by-step error correction” (Color figure online)

accomplished on time or corrected for an error. A working agreement (see section on this pattern) between the human and the assistant can be agreed and in place. This allows the human to establish the parameters of how the assistant will behave as urgency increases, and informs and constrains the actions of the assistant to ensure the human’s mental model matches that of the agent when actions are needed [see for example, 10].

The “delegate agent” that appears on the tool side of Fig. 3 is optional. This agent is able to accomplish given tasks by use of available automation. If this agent is not available, the interaction of the human or the assistant system is on the level of commands for the available conventional automation.

6 Collaborations

The fundamental requirement for this pattern is that there be at least one participant agent that is aware of the urgency and priority of the tasks. The participant should have an agreement with any humans or other agents concerning the rules under which actions may be taken. Such agreement can be built into the system, or more flexibly created using a working agreement design pattern [please see 11, for required details].

A working agreement is a task-centric, shared understanding of how task performance is to be split and shared between partners. These styles of agreement can be found in air traffic control, for example, in splitting up airspace responsibilities [e.g., 12]. Working agreements between humans and automation should be accompanied with several benefits to the each agent as well as the system overall – first, the development of the agreements helps articulate the tasks and methods required to perform them for both the agent and the system (a step not always taken in system design). Second, an agreement helps in understanding how these tasks should be allocated effectively and allows for evaluation (agreement A versus B). Third, the definitions of agreements allows their codification into system- and human-understandable display. In other

words, agents in the system get clarification on what other agents are doing and supposed to do given a set of conditions [11]. Usually the human does not have this level of awareness in a system, leading to mental model mismatching.

7 Consequences and Justification

This pattern mainly affects the cooperation between the human and the assistant system. The assistant system in general behaves like a restrained human teammate. Within normal situations, the assistant system is no more than a silent observer. Only in situations which require an action of the assistant system to prevent a degradation of the overall mission performance, the assistant system becomes active with a situation adequate intervention, falling within the working agreement structure. This restrained behaviour of the assistant system will leave the human in charge of task accomplishment, as long he/she is able to do his/her task on their own according to estimations and current projections. In these times where it is necessary for the human to take actions (e.g. recognition of a effecting change in situation, necessary execution in tasks) the assistant system makes an appearance, by giving alerts, hints, or messages without wresting the human from his task. The human will be kept in the loop and supported as long as the human has enough time, resources and capabilities to solve the situation on his/her own.

In many ways, this positive benefit harkens to “lockout” or constraint methods of processing, in which certain actions that are harmful are literally prevented by manipulating the interaction capabilities (or removing a capability altogether under certain circumstances). An example is the “grey out” of action buttons on an interface; not only does this prevent the user from making an inappropriate response, but it can also communicate that the system believe it is inappropriate. Similarly, other changes in design and lockouts – such as those used to prevent sudden unintended gear changes in vehicles, and those made to physical equipment (such as changing the fittings on operating room equipment to avoid connecting the wrong gas tanks to patients) provide major safety improvements that greatly reduce the burden on the human operator to “avoid error.” These system-driven error reduction methods come highly recommended from other engineering domains and are at the heart of major theoretical advances in human error mitigation [13, 14].

As discussed, the difficulties here lie in determining what those actions are during system design and not in hindsight after an accident or devastating error is committed. Presumably, we can account for a large portion of both, but never all of either type. This leads to conditions when the human may need access and the design blocks it; or times when the design fails to block an action that leads to mistakes.

Another possible downside might be, that a human can adapt to the restrained behavior of the assistant system – in other words complacency. This means the human could wait until no more time is available to do the task on his own, when the assistant system would then stand in by a full task adoption from the human.

8 Implementation

For the realization of an assistant system, which provides a stepwise increasing intervention policy, the assistant system has to have the following capabilities [7]:

- Monitoring of the environment and detection and analysis of danger
- Monitoring of the human and interpretation of the observed data with respect to the human's cognitive state(s)
- Planning and scheduling of interventions
- Execution of interventions, i.e. of the actual interaction with the human

The interventions of the assistant system can be supportive but reserved. The overall goal is, to keep the human in the loop as long as possible and responsible for task accomplishment. To enable this requirement the assistant system shall express the following desired behavior:

- The human shall be given as much time as possible to find own solution
- Interventions shall provide input that helps with the current problem (but may not solve it as optimally as a human expert)
- Dangerous situations shall be resolved before fatal/critical damage is inflicted
- The input given by an intervention shall not exceed the current problem.

In order to identify the emerging conflict situation, the urgency, and selection of the adequate intervention strategy, the assistant system has to continuously:

- Determine if the current situation is dangerous and, if so, at what time damage (a violated threshold of certain performance parameters) will be inflicted. A situation is dangerous if the further development, without intervention by the human or the assistant system will lead to damage (e.g. degradation of the overall work result).
- Determine what the human should do to resolve the dangerous situation. The resolution typically consists of giving a certain command (sequence) either to an existing delegate agent or to conventional automation.
- Estimate the current cognitive state of the human. The cognitive state includes mental resources such as situation awareness, vigilance, workload and focus of attention. It also includes the state of information processing, i.e. the current task(s) and the associated cognitive processes. These estimates should be based on a model of the human's information processing but could be informed by real-time inputs and measures.
- Compute the transitions of the human's cognitive states leading from the current situation to the resolution of the dangerous situation, and identify the conditions for each transition: What steps will the human's mind have to go through to effect the desired action, beginning with its current state? These steps and estimates of their duration (with buffers and worst-case assumptions) should be based on a model of the human's information processing.
- Arrange these computed mental steps along a timeline, beginning with the earliest one. Arrange them in a way that the final step (the desired action) takes place

immediately before the moment of damage, i.e. barely in time. The position of the left end, i.e. the starting point of the sequence, will then determine the point in time at which the pilot must begin working on the problem in order to prevent damage in the worst predicted case

- Determine whether the starting point of the sequence is in the future, or not?
 - (Yes): There is still time left for the human to find own solutions. The system should do nothing.
 - (No): The human should have reacted by now. Intervene by enforcing the current transition, i.e. the first step, using any available means.

In Fig. 4, we provide an example which shows how the assistant system derives necessary interventions. Within this example, a human operator has to enter new commands to his own aircraft to avoid a collision with a foreign aircraft. So the task for the operator is to enter the right evasion commands (2) to avoid the collision. This has to be happened latest immediate before time (1), which is the last chance to avoid the damage. In fact, the human operator is actually analyzing his tactical map (3). By monitoring the human the assistant system could detect, that the human did not yet detect the foreign aircraft. The assistant system determines that the sequence of detection, information processing and action, leading from the actual task of the operator to the desired action steps (4, 5) will likely not be completed in time (6a). Therefore, the assistant system intervenes (6b): It enforces the transition from the human operator's current mental state to the next state (detection of a relevant change in the tactical environment) by alerting the human operator about an incoming other aircraft

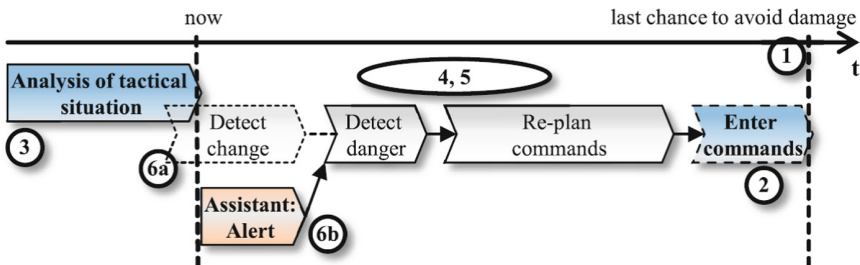


Fig. 4. Example for planning and scheduling of interventions by the assistant system

For each possible dangerous situation, the assistant system has to have a repertoire of different intervention possibilities, e.g.:

- neutral alerts without hints to the emerging error situation,
- directed alert towards the emerging problem,
- messages and suggestions to give hints to the human how to solve the situation,
- proposals to adopt part-tasks to support the human in task accomplishment,
- complete task adaptations in temporal critical situations

One possible consideration is that behavior may be different when urgency is increasing and when it is decreasing. The working agreement with the task manager should specify what these behaviors should be [15]. Below is an example table of behaviors that may explain this better than a formal definition (Table 1).

Table 1. Sample urgency decision table.

Notional urgency range	When urgency increasing since last time step...	When urgency decreasing since last time step...
0–24	No change from standard working agreement	Treat as having indicated priority. Remove requests to abandon or messages and failure and replace with explanation dialog. If activity on method has not been initiated by human and another method requiring more human action is available that will keep task within this urgency range, then change methods
25–49	If human activity has not been initiated on this method and another method is available that reduces urgency, then select new method	Remove highlights in lists. Remove requests to abandon or messages and failure and replace with explanation dialog
50–74	Use alert highlighting in lists. Treat as having 50% higher priority in sorts	If a request to abandon task or a failure message has been presented, remove message. Provide dialog explaining reason for removal
75–99	If there are no task dependent, then request approval to abandon task	If a failure message presented, remove message and provide dialog explaining removal
100	Failure message to human. Request extension on deadline or abandonment of task	Not possible

9 Examples and Known Uses

This pattern has been applied to the domain of unmanned air reconnaissance conducted by a single human pilot in a ground control station.

The work objective of the single human pilot was to gain reconnaissance information on certain objects (buildings, persons, vehicles) in a hostile area. The required information could be obtained by using the sensors attached to an unmanned aircraft. These sensors provided video and imaging data to the human pilot. Beside the task of gathering and evaluation of sensor data to gain the required information, the pilot has

furthermore to manage the flight of the unmanned aircraft. The reconnaissance targets were given to the pilot beforehand, but could have changed during the mission. The execution was also constrained by airspace regulations (boundaries and corridors), threats (possible unexpected hostile air defenses), and resource limitations (fuel). As it was a single pilot station, the pilot had to carry out the tasks of flight management, sensor management and interpretation of sensor data in parallel. Therefore, the pilot was supported by an assistant system according to the described design pattern.

Within this use-case, the assistant system had to prevent, among others, the following effects of erroneous behaviour of the human pilot:

- Violation of airspace regulations by the unmanned aircraft
- Loss of unmanned aircraft by exhaustion of fuel reserves during flight
- Loss of unmanned aircraft by entry into the threat radius of hostile air defense sites
- Ineffective reconnaissance (inadequate fulfilment of the mission objective)

To avoid these effects, the assistant system was allowed to intervene. It was integrated into the control station's systems and had direct access to the pilot's GUI. Depending on the information processing step of the human determined by the assistant system, the assistant system was able to display general alerts and iconic or textual messages, highlight certain screen elements, direct the pilot's attention to other screens, or override commands if necessary. The assistant system gathered all necessary information to plan, schedule and execute an intervention from the subsystems. For a more detailed description of the implementation of the required functionality, please refer to [9].

An example of an escalating sequence of interventions of the assistant system in response to potential emerging violations is shown in Fig. 5 below.

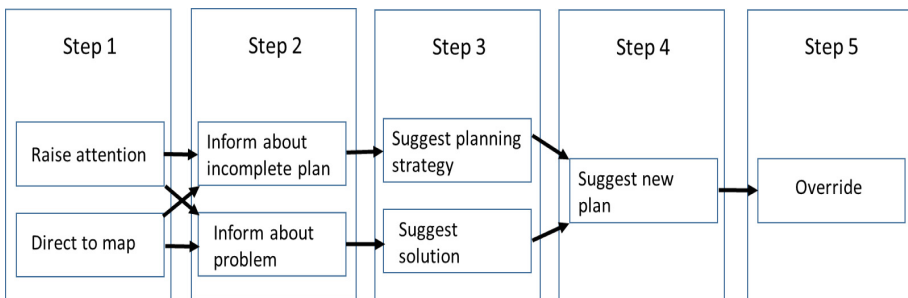


Fig. 5. Example for planning and scheduling of interventions by the assistant system

The following pictures (Fig. 6) illustrate one realization of the cooperation between the human operator and the assistant system by applying the stepwise escalating intervention sequence.

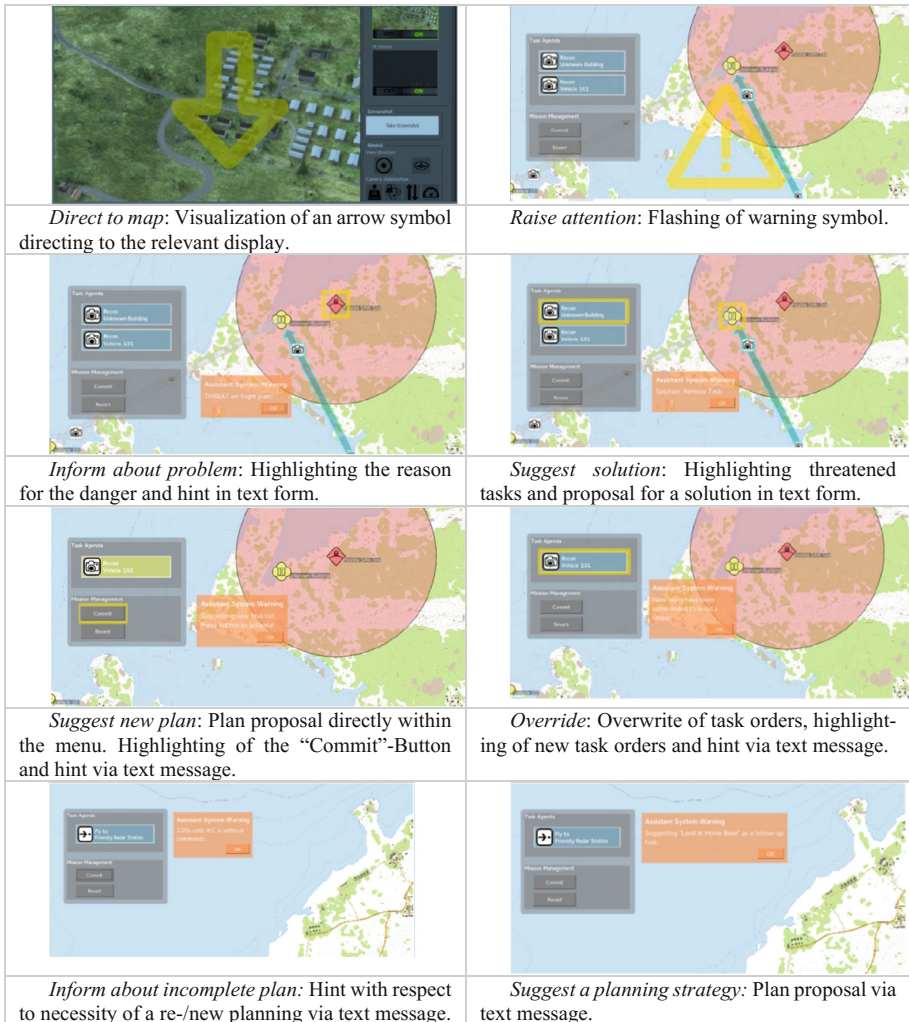


Fig. 6. Applying the stepwise escalating intervention sequence.

10 Related Patterns

This pattern can make use of working agreements [10, 11] to establish the rules by which urgency will be addressed. The formalization takes features from a pattern that might be titled tasks with deadlines and rewards. In that pattern, rewards are only received in full if the task is completed prior to its deadline.

This pattern has a complex interaction with the pattern human takes control upon autonomy failure. That pattern requires that a method requiring human attention be selected for a task that formerly was being performed by the autonomy. This can cause an immediate increase in urgency. This may be how the user is notified of the need to

take control (e.g., a new method is selected when the autonomy fails, this method requires urgent attention, so the task manager tries to reduce urgency by going to the autonomy). To avoid infinite loops, the methods allowing action by autonomy need to be marked as unavailable through some means.

References

1. Mosier, K., Skitka, L., Heers, S., Burdick, M.: Automation bias: decision making and performance in high-tech cockpits. *Int. J. Aviat. Psychol.* **8**(1), 47–63 (1998)
2. Parasuraman, R., Manzey, D.: Complacency and bias in human use of automation: an attentional integration. *Hum. Factors* **52**(3), 381–410 (2010)
3. Wickens, C., Hollands, J., Banbury, S., Parasuraman, R.: *Engineering Psychology and Human Performance*, 4th edn. Pearson, Upper Saddle River (2013)
4. Trapsilawati, F., Qu, X., Wickens, C., Chen, C.: Human factors assessment of conflict resolution aid reliability and time pressure in future air traffic control. *Ergonomics* **58**(6), 897–908 (2015)
5. Crandall, B., Klein, G., Hoffman, R.: *Working Minds: A Practitioner’s Guide to Cognitive Task Analysis*. MIT Press, Cambridge (2006)
6. Endsley, M., Jones, D.: *Designing for Situation Awareness: An Approach to Human-Centered Design*, 2nd edn. CRC Press, New York (2012)
7. Parasuraman, R., Molloy, R., Singh, I.: Performance consequences of automation-induced ‘complacency’. *Int. J. Aviat. Psychol.* **3**, 1–23 (2009)
8. Schulte, A., Donath, D., Lange, D.S.: Design patterns for human-cognitive agent teaming. In: Harris, D. (ed.) EPCE 2016. LNCS (LNAI), vol. 9736, pp. 231–243. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-40030-3_24
9. Theiβing, N., Schulte, A.: Designing a support system to mitigate pilot error while minimizing out-of-the-loop-effects. In: Harris, D. (ed.) EPCE 2016. LNCS (LNAI), vol. 9736, pp. 439–451. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-40030-3_43
10. de Greef, T., Arciszewski, H., Neerincx, M.: Adaptive automation based on an object-oriented task model: Implementation and evaluation in a realistic C2 environment. *J. Cogn. Eng. Decis. Mak.* **4**(2), 152–182 (2010)
11. Gutzwiller, R.S., Espinosa, S.H., Kenny, C., Lange, D.S.: A design pattern for working agreements in human-autonomy teaming. In: Cassenti, D.N. (ed.) AHFE 2017. AISC, vol. 591, pp. 12–24. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-60591-3_2
12. US Department of Transportation Federal Aviation Administration, Air Traffic Organization Policy: Section 3. Letters of Agreement (2010)
13. Rasmussen, J.: Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. *IEEE Trans. Syst. Man Cybern.* **3**, 257–266 (1983)
14. Reason, J.: *Human Error*. Cambridge University Press, Cambridge (1990)
15. Lange, D.S., Gutzwiller, R.S.: Human-autonomy teaming patterns in the command and control of teams of autonomous systems. In: Harris, D. (ed.) EPCE 2016. LNCS (LNAI), vol. 9736, pp. 179–188. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-40030-3_19