TRUST IN DECISION AIDS: A MODEL AND ITS TRAINING IMPLICATIONS[1]

Marvin S. Cohen, Raja Parasuraman, and Jared T. Freeman

1. THE PROBLEM OF TRUST

Decision aids have much in common with other types of automation. For example, they vary in the *level* of automation that they offer — from data integration, through expert systems that generate decision options, to associate systems that take action unless the user overrides. In principle, "autonomous" systems can be developed that evaluate, choose and act without the user's knowledge.

Automation of decision making has not advanced as far along this spectrum as automation in other fields. Explanations for this "failure" vary. According to some advocates of decision aids, the reason is a sort of irrational technophobia, evidenced by a lack of appropriate *trust* in the decisions that such aids make. According to skeptics, on the other hand, a good reason for rejecting decision aids altogether is *overtrust*: the tendency of users to rely on an aid as if it were infallible when they should instead rely on their own judgment. Strangely enough, both camps share a similar concept of *trust*. In both views, trust is a relatively enduring attitude that a user has toward an aid, akin to love, hate, or faith, rather than a more transient and situation-specific attitude, such as agreement or disagreement.

We will propose an alternative, more differentiated conception of trust. It includes the more enduring concept as a special case, but emphasizes instead the specific conditions under which an aid will and will not perform well. According to this alternative approach, the problem of decision aid acceptance is neither undertrust nor overtrust, but *inappropriate* trust: a failure to understand or properly evaluate the conditions affecting good and bad aid performance.

The issue of trust marks an important difference between decision aids and other types of automation. Decision aids are often intended to help users handle *uncertainty* about a domain. Yet, an obstacle to the effective use of decision aids is uncertainty about the decision aids themselves. Unlike other kinds of automation, therefore, decision aids may transform, but not eliminate, the human task that was to have been automated.

Existing training has neglected the issue of uncertainty. It typically focuses on what the user must do to make the aid work, i.e., inputs, outputs, and modes of operation. In doing so, it has inadvertently reinforced the misconception that trust must be a permanent stance, to accept or reject an aid as a whole. More often than not, however, to benefit from a decision aid, the user must learn, or be trained, to recognize and act on uncertainty about the quality of the aid's recommendations, and to understand how such uncertainty can change from situation to situation. In most cases, this task is not trivial. The domains in which decision aids are introduced tend to be complex; novel situations are likely to arise that were not anticipated by aid designers; the workload and task priorities of users may shift, along with the attention they can devote to the decision aid itself; and by the very nature of uncertainty, even the best decision may on occasion have a bad outcome, or a bad decision a good outcome. It is not easy in this context to acquire an understanding of the aid's decision making processes that will support effective exploitation of the aid. The challenge is perhaps not unlike the one we face in learning

[1] This work was supported by the Aviation Applied Technology Directorate, ATCOM, Fort Eustis, VA, Contract No. DAAJ02-97-C-0009 with Cognitive Technologies, Inc. We thank Keith Arthur, the project's technical monitor.

to work effectively with our fellow humans.

The research to be described had two goals: (1) To develop a systematic and general framework for understanding trust in decision aids, (2) to derive training implications from that framework, and (3) to apply the framework and test its feasibility by developing a training strategy for a specific decision aiding environment (the Rotorcraft Pilot's Associate). Section 2 of this paper briefly describes a part of the framework that has been developed, while Section 3 explores some of its training implications. Section 4 compares the framework to previous work on trust. A more complete description of the framework, as well as of its initial application to RPA, can be found in Cohen, Parasuraman, Serfaty, & Andes (1997).

2. A QUALITATIVE MODEL OF TRUST IN DECISION AIDS

The model of trust that we will describe depends on two key concepts: (1) The qualitative structure of trust is represented by a template for *arguments* of a certain kind. Such arguments marshal observations and prior beliefs to make predictions about the quality of system performance under specific conditions and over a specified period of time. (2) The quantitative aspect of trust highlights the uncertainty of these predictions, and can be conveniently represented by probability distributions over the appropriateness of system actions given features of the system and of the situation: e.g., *p(correct action | system, situation)*. Because of this duality, we refer to the theory as the *Argument-based Probabilistic Trust* (APT) model. This model builds on previous work by Muir and others (e.g., Muir, 1987.1988; Zuboff, 1988; Riley, 1994). However, interpreting trust in terms of arguments and uncertainty leads to a theory that is more general, more parsimonious, and more useful for training than some of the current formulations. The present paper will focus on the qualitative aspects of the model.

Trust as Context-Specific Arguments about System Performance

The qualitative aspect of trust is based on Toulmin's (1958) theory of argument. Toulmin's goal was to examine actual methods of reasoning in real-world domains such as law and medicine, rather than idealized forms of reasoning represented in logic. He provides us a convenient framework for reasoning about the expected quality of a decision aid's performance.

The basic structure of an argument, according to Toulmin is shown in Figure 1. A *claim* is any conclusion whose merits we are seeking to establish. The claim is supported by *grounds,* or evidence. The reason that this particular evidence supports this particular conclusion is the existence of a *warrant*, i.e., a belief in a general connection between this type of grounds and this type of claim. The *backing* provides an explanation of the warrant, i.e., a theoretical or empirical basis for the connection between ground and claim. Modal qualifiers (e.g., *probably, possibly, almost certainly*) weaken or strengthen the validity of the claim. Possible rebuttals are factors capable of deactivating the link between grounds and claim, by asserting conditions under which the warrant would be invalid.

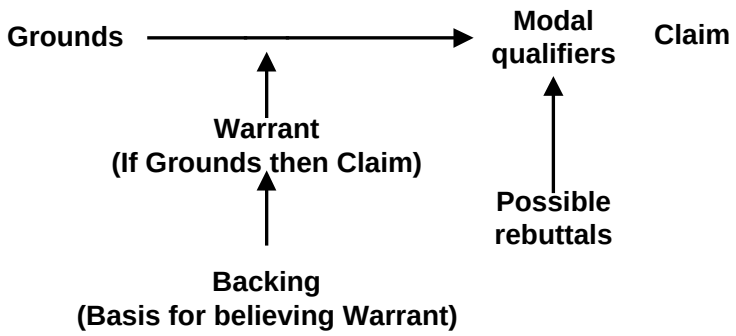

Figure 1. Toulmin's model of argument. The structure can be read: Grounds, so Qualified Claim, unless Rebuttal, since Warrant, on account of Backing.

Figure 1 shows how APT uses the components of Toulmin's model. We will describe each of its elements in turn:

Warrant. A Warrant says that certain conditions are associated with a certain quality of aid performance. The conditions, which the user believes to be correlated with aid performance, may be features or combinations of features of the system, situation, task, or even specific aid conclusions. The quality of performance may be described specifically (e.g., the system will be wrong under these conditions), probabilistically (e.g., the system is right about 3 times out of 4 under these conditions), or with more vague qualifiers (e.g., the system is highly reliable under these conditions).

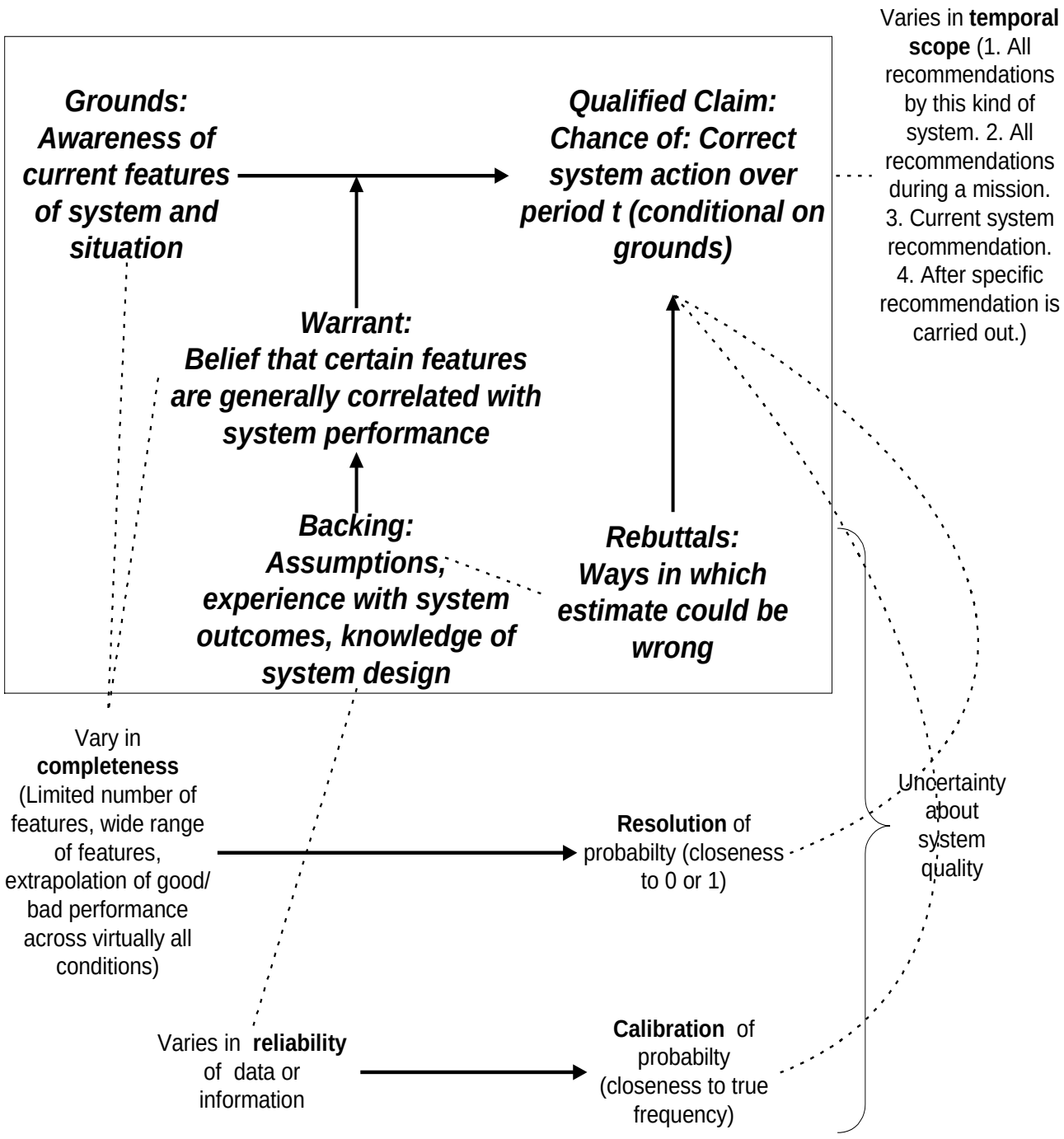

Figure 2. Main components of Argument-based Probabilistic Trust (APT) model. The argument structure is shown within the box. Parameters associated with these components are shown outside the box, linked to the relevant components by dotted lines.

Grounds. To play a role in an argument for (or against) trusting an aid, a feature or combination of features must also be observed on the particular occasion for which trust is being assessed (as reflected in the grounds). Grounds are simply the current or on-going observations that influence a user's judgment of the reliability of an aid.

Qualified Claim. The output of the trust model is a qualified claim. This Qualified Claim represents the degree of trust in the system under the conditions specified in the Grounds, according to the Warrant. As noted above, qualifications may be precise (e.g., 30%) or, as is more often the case, vague (e.g., very reliable).

Degree of trust in a system, then, is the probability (or more vaguely expressed uncertainty) that the system will produce correct actions over a given period of time, conditional on relevant features of the system, current situation, task, and/or conclusion.

Backing. The fourth component of the model is the origin of the user's predictions about system performance, i.e., how the Warrants were learned or inferred. Users can learn about a system, and develop an appreciation of factors that influence trust, in many different ways: by direct experience with the system, by learning about system design, by talking to more experienced users, or by making assumptions (e.g., best case or worst case).

Rebuttals. Rebuttals are possible exceptions to the rule expressed by the Warrant. They may lurk as implicit assumptions in the backing, for example, that one's past experience with the aid has been representative of present conditions. Sometimes, rebuttals reflect explicit assumptions, for example, a decision by the user to assume worst-case conditions for aid validity until he or she learns otherwise. Assumptions are natural and inevitable, since it is impossible to verify every condition that could potentially affect an aid's performance. As a result, any assessment of trust is subject to rebuttals, even assessments that are based on long experience with the aid or on thorough design knowledge. When events violate expectations, however, assumptions are worth ferreting out and re-examining through a process of critical thinking.

How Trust Varies

The most important use of APT is to chart how trust varies, from one user to another, from one decision aid to another, from one situation to another, and across phases of decision aid use. To track such changes, APT supplies a set of five interrelated parameters to describe any given assessment of trust. Two of these parameters, resolution and calibration, are most easily illustrated by reference to probabilities. However, other, more verbal assessments of uncertainty could be substituted. The quantitative aspect of the model is secondary to the qualitative aspects, and is of value primarily for the light it sheds on qualitative relationships These parameters, as shown in Figure 2, are:

Temporal scope. This is the duration of time that the assessment (i.e., the qualified claim) covers. As shown in Table 1, we will distinguish four principal phases in the use of a decision aid, corresponding to decreasing temporal scope: (1) trust in a system generally over all its potential uses, before a specific mission has been assigned or a specific task has been undertaken, (2) trust in the system's capability for a specific mission or task, (3) trust in a specific recommendation that the aid has made, before the recommendation has been verified or implemented, and (4) trust in a specific aid recommendation after it has been verified or implemented and its quality is known. These "phases" do not necessarily follow a strict sequence, and the boundaries may be blurred (e.g., missions are "replanned" in the course of execution). As shown in Table 1, the main point is to capture the relationship between different kinds of decisions about interaction with the decision aid, the different kinds of information they rely on, and the differences in temporal scope that usually attend such decisions.

Table 1.Temporal phases of trust.

	Phase of Aid Use			
	1: Pre-theater	**2: Mission / task planning**	**3: Task /mission execution**	**4: Task outcome**
Temporal scope of trust and reliance decision	Entire lifetime of aid	Aid during mission/task	Specific aid conclusion	Specific aid conclusion
Illustrative information used to predict trust (grounds)	Type of system, environments to which system is suited, functions it can perform.	Terrain, mission objectives, types of tasks.	Task goals, task situation. Type of aid conclusion; content of aid conclusion; aid's confidence level [conditional on user's choosing to monitor aid in Phase 2].	Quality of aid conclusion [conditional on user's deciding to verify or implement aid conclusion in Phase 3].
Illustrative reliance decisions made	*Managers*: Build aid or not; basic aid functionality. *Designers*: types of interaction & automation modes; degree of adaptability of aid by user, degree of automatic adaptiveness of aid to user). *Trainers*: scenarios & aid functions to focus on. *Users*: degree of acceptance of aid.	*Users*: Select automation mode (e.g., fully automatic; monitoring & possible verification by user; manual with monitoring by aid; fully manual); adjust aid parameters [conditional on designers' choosing an adaptable aid design in Phase 1]	*Users*: Accept, override, modify, or verify specific aid conclusion; use aid to verify user's conclusion [conditional on user's choice of automation modes in Phase 2].	

Completeness. Grounds and Warrant can vary in their coverage of the features that potentially affect system performance. Completeness thus reflects the degree to which the user understands the conditions that affect trust at any given temporal phase.

Resolution. The resolution of the qualified claim is the degree to which the user can

reduce uncertainty about aid performance, either for better or worse, by discriminating situations. If trust is measured as the probability of an acceptable system response given the situation, average resolution is increased by defining situations in more detail, so that this probability moves closer to either zero or one for each situation that might occur. Completeness obviously affects the resolution of the trust assessment. The more information that is used to predict an aid's performance, i.e., the more completeness in the grounds and warrant, the higher the average, or expected, resolution. For example, a decision aid user may believe that an aid tends to be correct in 80% of all the situations in which it is used. But trust assessments will have more resolution if the user can observationally identify specific types of situations where the performance of the aid is better (e.g., 95%) or worse (e.g., 60%) than the overall average.

These quantitative aspect of the model, i.e., probabilities, are secondary to the qualitative aspects, and are of value primarily for the light they shed on qualitative relationships.

Reliability. Reliability of the backing refers to the amount and quality of data or information that underlies the trust assessment. The more experience a user has with the aid and the situation, the more representative that experience is, the more detailed and accurate the design knowledge the user has, or the more robust the assumptions the user makes, the more reliable is the Backing for trust assessments. In probabilistic terms, a user who is highly experienced with a decision aid may be confident that the percentage of correct aid recommendations is between 80% and 90%, while a user who is less familiar with the aid may know only that the percentage is somewhere between 50% and 100%.

Calibration. Calibration of the qualified claim is the correspondence of trust to the true quality of aid performance, within the specified situation. In probabilistic terms, calibration is the relationship between the probability estimate and the true frequencies of correct system response given the conditions in the Grounds. Just as completeness is related to resolution, so reliability is related to calibration. The amount and quality of information in the Backing determines the calibration of trust. For example, suppose each of two users assesses as 85% the probability of an aid's selecting an appropriate response. One user, who is familiar with the aid, is unlikely to be off by more than 5%, while the other user, who is less familiar with the aid, may be off by as much as 25%.

Trust as an Evolving, Uncertain Prediction

A key feature of trust is that it evolves not only as the user gains experience with an aid, but also as the user moves through the various phases of a particular mission or task. In this section, we delve more deeply into this important aspect of APT. In so doing, we also explore and clarify the elements of the APT model.

In our discussion, we will draw on examples from a Battle Position Planner such as the one being developed for the Rotorcraft Pilot's Associate program. The Battle Position Planner evaluates potential sites from which an enemy, such as a moving tank column, can be engaged by attack helicopters. Sites are evaluated by the aid in terms of the concealment provided by the terrain, the presence of a backdrop to prevent the helicopter from being silhouetted, the distance of the battle position from the target relative to the helicopter's weapon range, the altitude of the site relative to the target, room to maneuver within the site, and others. The aid is designed to relieve the user from the workload of noting and weighing these factors.

However, there is a fly in the ointment: Other factors that are relevant to the evaluation of battle positions are *not* considered by the aid. One of these, for example, is rotorwash, which is a cloud of dust, leaves, snow or water that may be thrown up by the helicopter's blades, and which can give away the helicopter's position to the enemy. Another factor omitted by the aid is angle

of attack, i.e., whether the enemy will be attacked from the front, rear, or side. A flanking or rear attack is preferable to one that engages the enemy frontally. Omission of these (and other) factors can result in selection of an unacceptable battle position. If trust is regarded as a single, global assessment of the aid, the result may be total rejection by users. Clearly, a more differentiated conception of trust is required, which allows users to reap the aid's benefits where possible, while selectively allocating their attention to issues that the aid fails to address when they are important.

Once the aid recommends a battle position (phase 3), the user can determine the angle of attack by reference to maps and in the light of expected enemy avenues of approach. By contrast, it is usually not possible to determine whether rotorwash will be a factor from a map alone, without visually inspecting the prospective battle position (phase 4). However, the pilot may have some prior notion of the likelihood of rotorwash in the type of terrain where the mission will take place, and this knowledge may influence the pilot's degree of trust in an aid recommendation in earlier phases.

Suppose a user of the Battle Position Planner has studied the recommended battle position on a map, and determined that it represents a flanking angle of attack, which is acceptable. The user still does not know if the position will suffer from rotorwash. An illustrative argument regarding the reliability of the aid's recommendation in phase 3 might be the following:

Angle of attack is acceptable, and we are in desert terrain (Grounds). Thirty per cent of desert terrain is typically affected by rotorwash; the aid does not consider rotorwash; and rotorwash makes a battle position unacceptable (Warrant). So the chance the system will recommend an acceptable battle position is 70% (Qualified Claim) – unless the terrain has been changed in some way recently (e.g., it may have snowed or rained) and unless other factors besides rotorwash and angle of attack also affect the aid's accuracy (Rebuttals). The backing for this argument is derived from the pilot's long experience with desert terrain, and use of the aid in situations where its omission of rotorwash as a factor was apparent.

Event Trees for Trust in Decision Aids

With successive phases of aid use (Table 1), more and more information about the performance of an aid is available, and different types of decisions are made about the degree to which the user will rely on the aid:

This process of acquiring information over time, and updating trust, can be conveniently visualized as a progression along the branches of an event tree(Shafer, 1996). Such an event tree is a succession of observations or experiences, in each of which the decision-aid user learns something new that is relevant to predictions about aid performance. The event tree thus represents all the factors that are known to affect the aid's accuracy, organized in the sequence in which they are expected to be observed by an aid user. A full sequence of observations and experiences determines a particular path through the tree. Each such path is a possible story, or scenario, about decision aid use. The end or outcome of each story is either a successful or unsuccessful contribution by the decision aid to the user's task (e.g., a successful attack).The expectation that this final contribution will be acceptable, at each point in the event tree, is the user's trust in the aid at that point.

The components of our trust model, as well as their parameters, can all be defined with respect to such event trees. Note that the event tree itself is not equivalent to the user's *mental model* , or way of thinking about, the decision aid's performance. It is, however, a summary of the *effects* of the user's mental models on expectations about aid performance. It includes the

factors that play a role in such models in generating predictions of aid performance.

Figure 3 is an event tree which illustrates the kind of information that might become available to a user at each phase of use. In this simple example, this user is aware of two factors that bear on the accuracy of the Battle Position Planner: angle of attack and rotorwash. The user believes that terrain interacts with both of these factors. In particular, the user believes that rotorwash will be worse if he or she is assigned to a mission in the desert than to one in the mountains, and that a rear or flanking battle position is more likely to be selected by chance in the desert than in the mountains. The illustrative argument given above represents a user at point A in the event tree, having become familiar with the system as a whole during pre-theater training (phase 1), having been assigned to a desert mission in phase 2, and having evaluated the angle of attack of a recommended battle position in phase 3, but prior to visually inspecting it in Phase 4.

In Figure 3, we have assigned probabilities summing to 1.0 to the branches emerging directly from each node. These reflect the chance that the factor corresponding to each branch will in fact occur. For example, in the pre-mission stage (phase 1), the user believes there is a 60% chance of being assigned to a desert region, and a 40% chance of being assigned to a mountainous region. If the mission turns out to be in desert terrain, the user believes the chance is three out of four that the aid's recommended battle position will be a rear or flanking one rather than frontal. But if the mission is in mountainous terrain, the user believes the chances are even. In addition, if the mission is in the desert, the user believes that about 30% of the terrain will be subject to rotorwash, while if the mission is in the mountains, rotorwash will be a factor in only about 5% of the potential sites.

The 1's and 0's in the terminal nodes of this tree stand for the aid's selection of an acceptable or unacceptable battle position, respectively. They can also be interpreted as probabilities of a successful completion of the mission or task (in this case, a successful attack) *with respect to battle position*. For an attack to succeed, it is assumed that more than an acceptable battle position is required: other factors must also go the right way. Since this tree represents trust in the Battle Position Planner, it shows the chance that battle position will not be a cause of failure of the attack. Note that the 1's and 0's could be replaced by more graded assessments, e.g., if the user believes that rotorwash is not necessarily fatal to a successful attack.

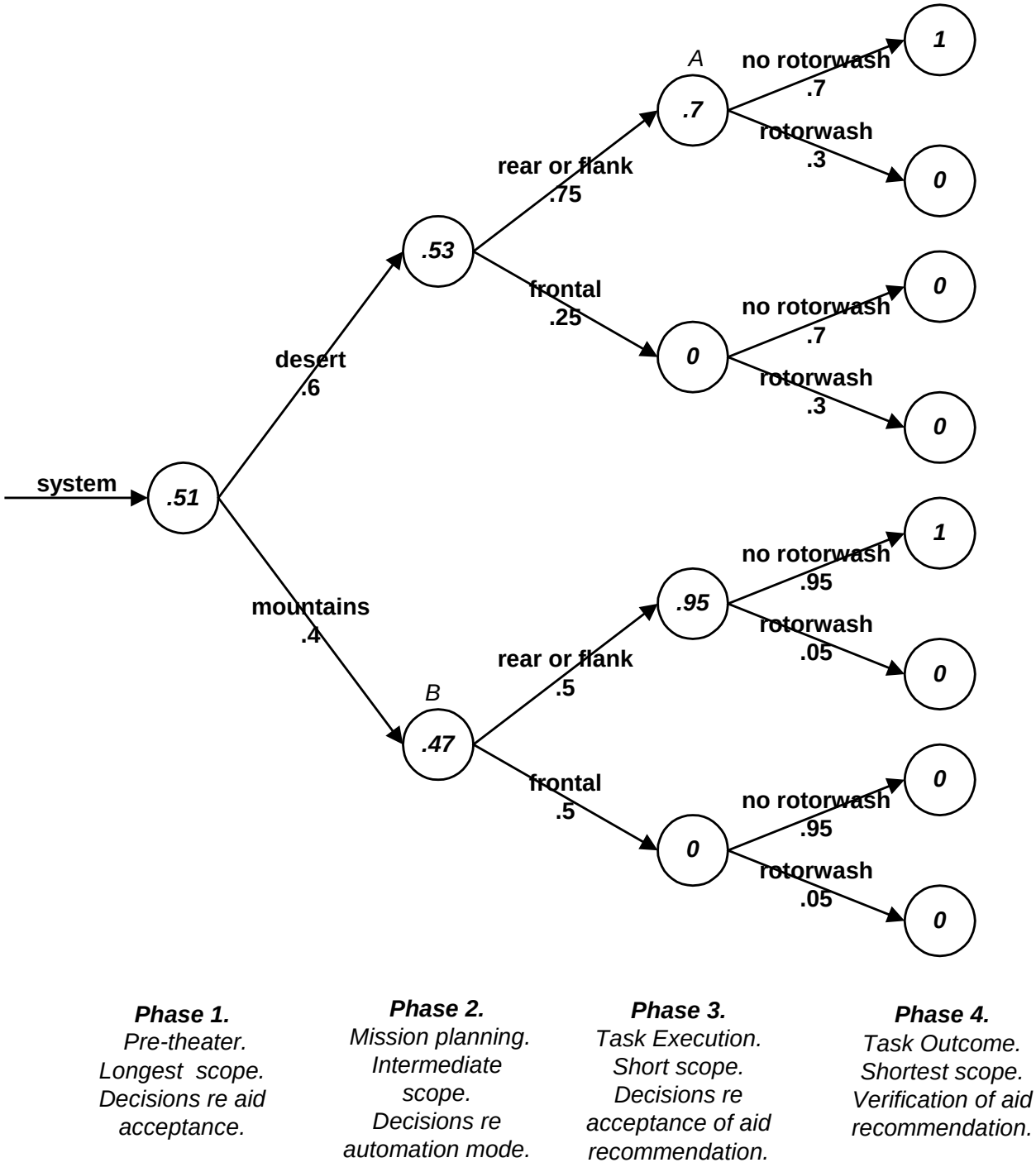

Figure 3. An event tree with illustrative probabilities for each branch.

Event Trees and Components of Trust

Event trees with probabilities clarify APT's components and parameters:
Warrant. An event tree with probabilities (or some other, perhaps qualitative

representation of uncertainty) is a compendium of all the warrants a user might draw on for arguments about trust. We can use it to derive generalizations about the connection between potential observations (features of the situation, aid, task, or aid recommendation) and aid performance. For example, our illustrative user at the node labeled *A* in Figure 3 was assigned to the desert and received an aid recommendation that involves a rear or flanking attack. For this user, the major remaining uncertainty about aid performance is due to the possibility of rotorwash. Since the user believes that approximately 30% of this terrain will be affected by rotorwash and that there is a 0% chance of successful attack with rotorwash, the user's trust in the aid (probability of an acceptable recommendation) is (.70) (1.0) + (.30) (0) = .70. The event tree in Figure 3 thus embeds within itself the following warrant: "If this system is used in the desert and recommends a battle position that involves a rear or flanking attack, the chance of the recommendation's being acceptable is approximately 70%."

The same event tree implies a warrant for trust assessments at every vantage the point the user might encounter. Each such vantage point, represented by the circular nodes, involves a series of observations corresponding to the branches leading *to* it (grounds). And each such vantage point also corresponds to a specific prediction regarding the performance of the aid, which can be calculated from the branches leading *from* it. A warrant is simply a pairing of a set of potential observations on the path leading up to a node, and a prediction of system performance based on all the paths leading out of that node.

Grounds. The grounds for a trust assessment consist of the observations that were made by the user as he or she moved along the path to the currently occupied node. Each time the user advances along another branch of the event tree, the observation corresponding to that branch is added to the grounds for the next trust judgment. Thus, the grounds for a trust assessment in phase 1, as shown in the tree of Figure 3, is a knowledge of system features that are common to all its potential uses. By the time the user gets to phase 2, a mission has been assigned, and the grounds include system features plus a knowledge of the terrain (desert or mountain). In phase 3, the system has made some specific recommendation, and grounds include system, terrain, and specific features of the recommendation (i.e., that it involves either a rear or flank attack, or a frontal attack). Once the recommendation has been verified by observing the recommended site in phase 4, the grounds also include "no rotorwash" or "rotorwash."

Qualified Claim. At each node in the tree, the user can determine the probability of a correct aid conclusion conditional on the grounds. Trust is simply the *expected*, or probability-weighted average, aid performance as seen from that particular viewpoint, and it is determined by looking toward the possibilities (if any) branching from the node toward the right. Numbers within the circular nodes of the tree represent trust at that point in the event tree.

Backing. Backing refers to the sources of the knowledge that is summarized in an event tree. The knowledge required to generate predictions of system performance can come from many sources: the user's experience with the system in different environments and tasks, the user's experience with analogous systems, the user's respect for the designers or the design process, reports by other users of their experiences with the same or similar systems, projection of the user's own strengths and weaknesses into the aid, and/or inference from knowledge of system design. In fact, more than one of these sources might be available to a user simultaneously, e.g., a user who has both design knowledge and personal experience with an aid. Multiple confirming sources of knowledge will increase the *reliability* of the trust assessment, while conflicting sources will reduce it.

Although the event tree itself is not the user's mental model, it crisply summarizes the

user's knowledge of factors that are relevant to prediction of the aid's performance. To illustrate, suppose a different user never learned the importance of angle of attack or rotorwash for the aid's accuracy. However, in many simulation experiences with this particular system in mountain and desert terrain, this user developed a sense of its overall accuracy in each of these environments – even though unable to analyze the reasons for success or failure on particular occasions. The event tree for such a user might be represented by Figure 4, rather than by Figure 3. Since this user fails to note the significance of angle of attack and rotorwash, no further relevant information is acquired in phases 3 and 4, so the trust assessment remains the same.

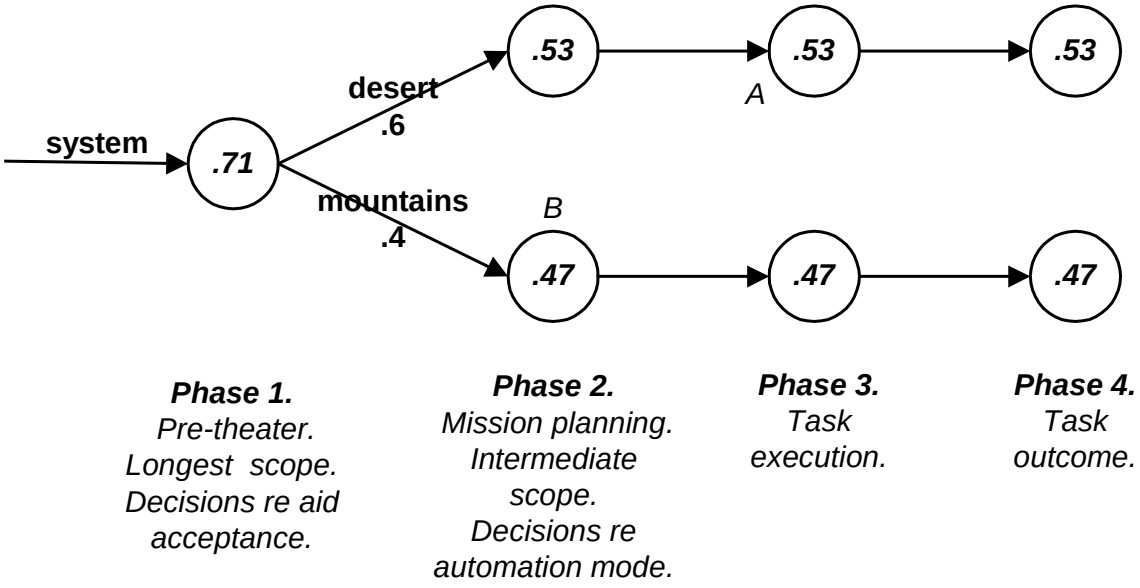

Figure 4. The user is unaware of the importance of either rotorwash or angle of attack, but has had extensive experience with the aid in both desert and mountainous terrain.

Rebuttals. Rebuttals are challenges to assumptions, either implicit or explicit, underlying an assessment of trust. Assumptions represent limitations of the user's mental model, and are open to challenge by new experience, design knowledge, or training. When an assumption is rejected, judgments of trust that depended on it may be dramatically changed.

Typically, assumptions arise because of incomplete knowledge of the domain, lack of experience with an aid, or poor understanding of the aid's design. For example, let us imagine yet another user of the Battle Position Planner, who recognizes the importance of rotorwash, but does not understand the importance of angle of attack in selecting a battle position. However, this user has been fortunate thus far to experience only situations where angle of attack was irrelevant, perhaps because the targets to be engaged did not represent a significant threat to the attackers. The event tree for this user may look like Figure 5.

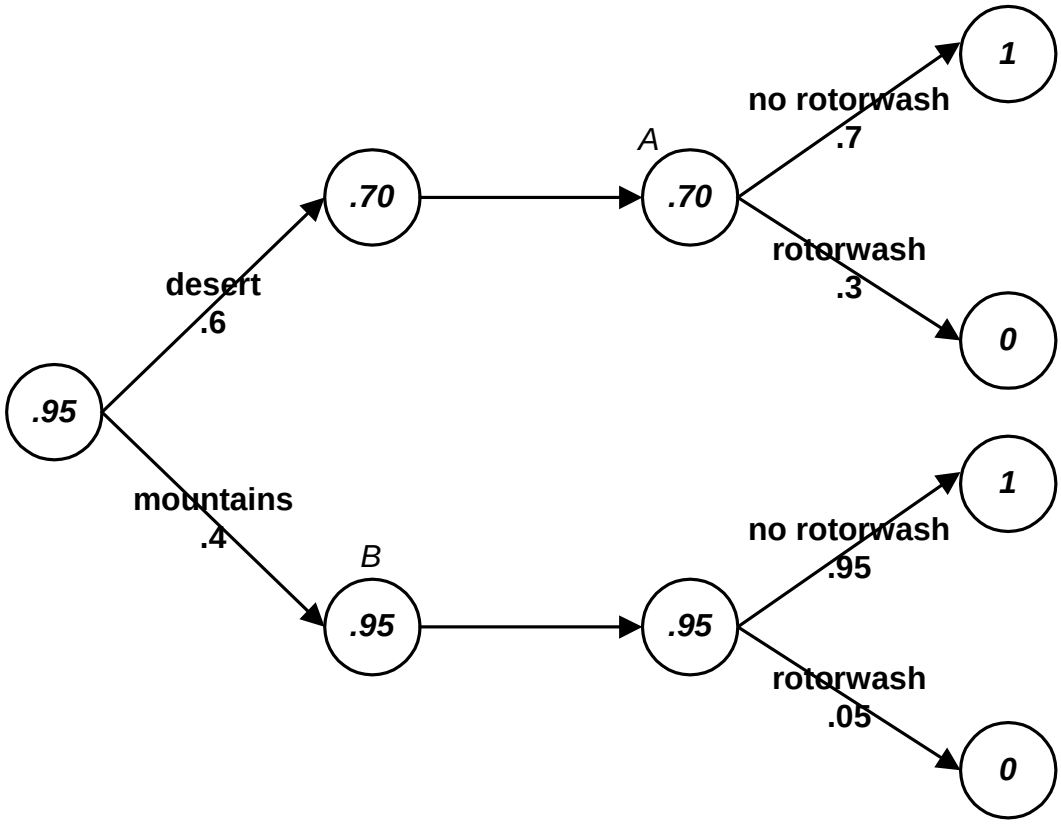

Figure 5 Event tree in which the user is unaware of the importance of angle of attack, and only experiences situations in which it is irrelevant. As a result, trust is higher than in the fuller event tree of Figure 3, and no updating occurs in Phase 3.

In this tree, the probability of correct system action, given that the aid is used in the desert, is 70%. Notice that 70% would be an accurate assessment of trust, according to Figure 3, if we expanded the grounds to include the additional condition that the recommended battle position is on the flank or rear. Similarly, 95% would be an accurate assessment of trust in the mountains, if we knew that angle of attack were ideal. The belief that angle of attack is ideal thus functions as an implicit assumption in this event tree. As a result of it, the user's trust in the aid will on many (but not all) occasions be higher than it should be.

Suppose that the user's subsequent experience in mountain and desert exercises is not so fortunate, and as a result the user learns that his or her assessment of trust in phase 2 was overly optimistic. In particular, these new experiences eventually lead the user to learn a probability of correct system response of 53% in desert terrain, and to learn a probability of correct response of 47% in mountain terrain. Figure 4 summarizes this new experience-based mental model, in

which the user has accurate trust, but no coherent way to break it down into causal factors such as rotorwash and angle of attack. It conflicts with a more detailed mental model that this user also possesses, which is summarized by the event tree of Figure 5.

One option for resolving conflict, though not a very good one, is to average the competing estimates. A problem with this approach is that it offers no explanation of the conflict; the user learns nothing new about the system's performance. Another, more fruitful approach is to use the conflict as a symptom that something is wrong in one's thinking about the aid and/or the situation. The solution is to probe deeper for causes of the conflict, by looking for mistaken assumptions underlying one or the other of the conflicting assessments (Cohen, 1986). In this example, both of the competing event trees involve assumptions: trust assessments based on Figure 4 assume that the user's experience has been representative; trust assessments based on Figure 5 assume that the user has explicitly recognized and sampled important factors that can degrade aid performance. the user might find the first assumption more plausible, and surmise that the overly optimistic event tree in Figure 5 must be incomplete. There is some factor in addition to rotorwash degrading the system's performance. This realization may initiate a process of more careful monitoring of the aid, which eventually leads both to a more accurate assessment of trust in the present case and to a more accurate mental model for use in the future.

It is always possible for the aid to behave in unexpected ways in new situations, because of some overlooked factor. It is neither possible nor worthwhile to try to enumerate and test all the assumptions underlying a particular assessment of trust. Hidden assumptions are worth ferreting out, however, in situations where the current mental model does proves inadequate, and the aid behaves differently than expected. In this situation, rebuttals are a reminder that a user's mental model of the aid is never quite finished or perfect. A critical cue for the need to consider rebuttals is conflict among different sources of information about the aid's performance.

The lesson for training is important: Users must learn to monitor not simply for the specific features that signal degraded aid performance (as in the event trees we have considered), but should also monitor for more subtle signs of trouble with the event tree itself, such as conflicting assessments of trust.

Event Trees and Parameters of Trust

Event trees help clarify the parameters of the APT model and their interactions.

Temporal scope. The event tree representation makes clear the sense in which larger temporal scope (Table 1) corresponds to more general judgments of trust, i.e., judgments that cover more possible cases of decision aid use. As the user moves along the phases of aid use from left to right, the temporal scope of the assessments decreases, from a consideration of the entire event tree at the extreme left node (prior to assignment to a theater), to consideration of more and more restricted sets of possibilities represented by smaller and smaller subtrees, as the user moves into a mission, and from there into a specific task. A problem with overly global conceptions of trust is that they do not track the more differentiated assessments that occur in smaller temporal windows.

Completeness. Completeness in a judgment of trust refers to the richness of the event tree, i.e., the number of factors relevant to the prediction of aid performance (each represented by a node with branches emerging from it). Completeness involves the ability to dynamically update *situation awareness* regarding the trustworthiness of the decision aid. For example, the user in Figure 4 starts out in phase 2 with the same level of trust as the user in Figure 3. In subsequent phases, however, the user with the less complete tree (Figure 4) is unable to update the assessment of trust based on new information about angle of attack and rotorwash.

The introduction of a decision aid changes the requirements for situation awareness. Situation awareness must be shaped to de-emphasize factors that are effectively handled by the decision aid, and to *include* factors that are predictive of decision aid success or failure. We can refer to this important implication of a differentiated trust model as *decision-aid driven situation awareness*.

Resolution. Resolution, in conjunction with probabilities, provides a mathematical way to measure the effect of completeness on trust. A more complete event tree obviously provides more alternative paths from one temporal phase to another. For example, the user at B in phase 2 of Figure 5 has only one path to phase 3, since no new information will be acquired. On the other hand, the user at B in phase 2 of the more complete tree in Figure 3 has two pathways that could take him or her to phase 3, one via the observation of a frontal angle of attack and the other via an observation of a rear or flanking angle of attack.

The more alternative paths provided by an event tree from an earlier temporal stage to a later one, the higher the resolution is *expected* to be at the later time, from the point of view of the earlier time.

It is both interesting and important to realize that resolution has *nothing* to do with whether the numbers that someone assesses (e.g., for trust in aid performance) behave like real probabilities. First, the numbers need not correspond to true relative frequencies of events (e.g., whenever the user assesses 95% probability, the event might only occur 35% of the time). Second, the assessments need not be logically related to one another like probabilities (e.g., the probability of two independent events need not equal the product of their respective probabilities). In fact, the "numbers" need not be numbers at all. Instead of saying that a decision aid recommendation is 95% likely to be successful, the user could say "George" instead, or, more plausibly, that success is "highly likely." What matters for resolution is that truly different situations are in fact discriminated from one another, no matter what numbers (or other expressions) happen to be used to do the discriminating, and that the situations have a real chance of occurring. Measuring the resolution of a trust assessment therefore does not require that the decision aid user actually assess trust as a probability.

Consider again the users at point B in phase 2. It might seem that the .95 trust assessment in Figure 5 represents higher resolution than the .47 assessment in Figure 3. In fact, however, the resolution of the two assessments is exactly the same. Resolution is based on the proximity of the *true* probability to 0 or 1, not the assessed probability, given the discriminations made by the assessor. The true probability must be the same for these two individuals, since they have traversed the same event tree, and thus discriminated the same situations, *up to* point B (both have taken note of system properties and the mountainous terrain). We can thus infer that the resolutions are equal, even if we don't know what the true probability of a correct recommendation by this system in mountainous terrain is.

The event trees are, of course, different after point B. Thus, the *expected* resolution in the future is different (at point B) for the two users. The user at B in Figure 3 can be expected, in phase 3, to have discriminated situations in which the chance of a successful aid recommendation is truly different (and, if the probability assessments are accurate, close to either .95 or to 0). By contrast, the user at B in Figure 5 will have made no additional discriminations in phase 3 that were not already made in phase 2. Expected resolution is thus higher for the user in Figure 3 than for the user in Figure 5.[2]

[2] The resolution of a trust assessment is usually measured negatively. Thus, to maximize

Calibration. Calibration, unlike resolution, depends very much on the assessments themselves. Calibration is the correspondence of the trust assessments to real-world frequencies of successful system performance. Thus, while resolution depends on making discriminations, calibration is a matter of the correct *numerical labeling* of the situations that have been discriminated. Having distinguished a set of truly possible situations in which the probability of an event truly does vary (resolution), has the decision maker correctly estimated the probabilities associated with those situations (calibration)?

Miscalibration may be caused by unreliable backing, e.g., faulty design knowledge or non-representative experiences. For example, in the event tree of Figure 5 a lucky but non-representative set of experiences with the aid led to the incorrect assumption that angle of attack is irrelevant. As a result, the probabilities are miscalibrated (relative to the tree in Figure 3). Miscalibration could also be due to lack of skill in assessing probabilities.

Miscalibration, however, is not caused merely by incomplete knowledge, i.e., a sparse event tree. Whether an assessment of trust is correctly calibrated or not always depends on the context in which the *user* intended it, i.e., on the user's decision tree and current location within it. Users with different knowledge regarding features that affect aid performance, hence, different event trees, will make different assessments of trust. But they *may* all be well calibrated within their respective *intended* contexts, i.e., given the situations that they respectively discriminate.

For example, the user of the incomplete, but accurate tree in Figure 4 obtains only one new piece of information after Phase 1, viz., the nature of the terrain (phase 2). Despite this poor resolution, the probabilities in Figure 4 are correct, if understood relative to the information actually relied on (the grounds of that user's assessment). This user's assessments of trust in phases 3 and 4 continue to be conditioned on the same grounds, i.e., the system and the terrain, as in phase 2. The user ignores the newly available information about angle of attack and rotorwash, and continues to report (correctly!) the likelihood of a successful system response *across all desert situations*.

As the user advances through phases 3 and 4, more specific assessments would in fact be far more useful. The large scope of the user's Phase 2 assessment becomes less and less relevant to the decision-making requirements of later phases. (This user's assessment of trust, if given in phase 3 after the aid has actually made a recommendation, might naturally be mistaken by others for an assessment of the likely appropriateness *of that recommendation* — rather than a generalization about how the aid performs in the desert!) What users need is a probability of correct performance in the detailed circumstances that arise. To get this, users must increase the *completeness* of their event trees and make the observations required to advance along branches

---

resolution, we minimize a *resolution penalty*. The resolution penalty is derived by calculating the product of the true probability corresponding to a discriminated situation times its complement. We then multiply each of these products by the true probability of the discriminated situation, and sum.

Assuming that the probabilities in Figure 3 approximate the true relative frequencies, the resolution penalty for both users is (.43)(.57) = .117 at point B. The expected resolution penalty for phase 3 for the user at B in Figure 3 goes down, to (.5)(.95)(.05)+(.5)(0)(1) = .02375. The expected resolution penalty for phase 3 for the user at B in Figure 5, on the other hand, is unchanged from phase 2, at (1)(.47)(.53) = .117.

of the richer tree. In short, they must drop their relatively global and undifferentiated approach to trust, and make distinctions. Resolution, not calibration, is designed to capture this differentiated aspect of trust.

The primary function of probabilities, or any other scheme for representing uncertainty, is not necessarily to come up with "correct" numbers, but to promote discriminations among situations that vary significantly in their implications for performance. Resolution is more important than calibration.

Trust in User-Decision Aid Interaction

Trust in an aid may evolve over time not only because of new observations, but because of active decisions by the user. As users gain understanding of the aid's strengths and weaknesses, they also learn how to interact more effectively with the aid, compensating for the weaknesses and exploiting the strengths. As a result of their own active participation, users' trust in an acceptable outcome is likely to increase. Trust must now be considered a product of the *interaction* between the decision aid and the user.

Decisions regarding the interaction between user and aid (i.e., *reliance* decisions) vary with temporal phase, as indicated by Table 1. For example, in phase 2, users may select automation modes and adjust aid parameters (if permitted by the aid's design). In phase 3, they may choose to accept, reject, or verify a specific aid conclusion (if permitted by the chosen automation mode). Our framework lends itself nicely to the way an evolving assessment of trust supports these different kinds of decisions at different times.

Figure 6 shows some variables that could influence a user's reliance on an aid, at each phase. These variables include trust in the aid, trust in one's own performance, anticipated or actual workload or time stress, and the stakes of the decision (see Lee and Moray,1994, and Riley, 1989, for empirical support). Decisions in each phase, of course, draw on knowledge relevant to that phase (e.g., regarding the entire domain, a specific mission and situation, or a particular task and aid conclusion), is influenced by estimates of trust, stakes, and workload over a different temporal period, and has effects that span different periods of time. Choices made in the longer decision cycles are revisited less frequently, and determine the options that are available to users at the shorter cycles. For example, a user will not have the opportunity to verify an aid conclusion in phase 3 if that user did not decide to monitor aid conclusions in phase 2.

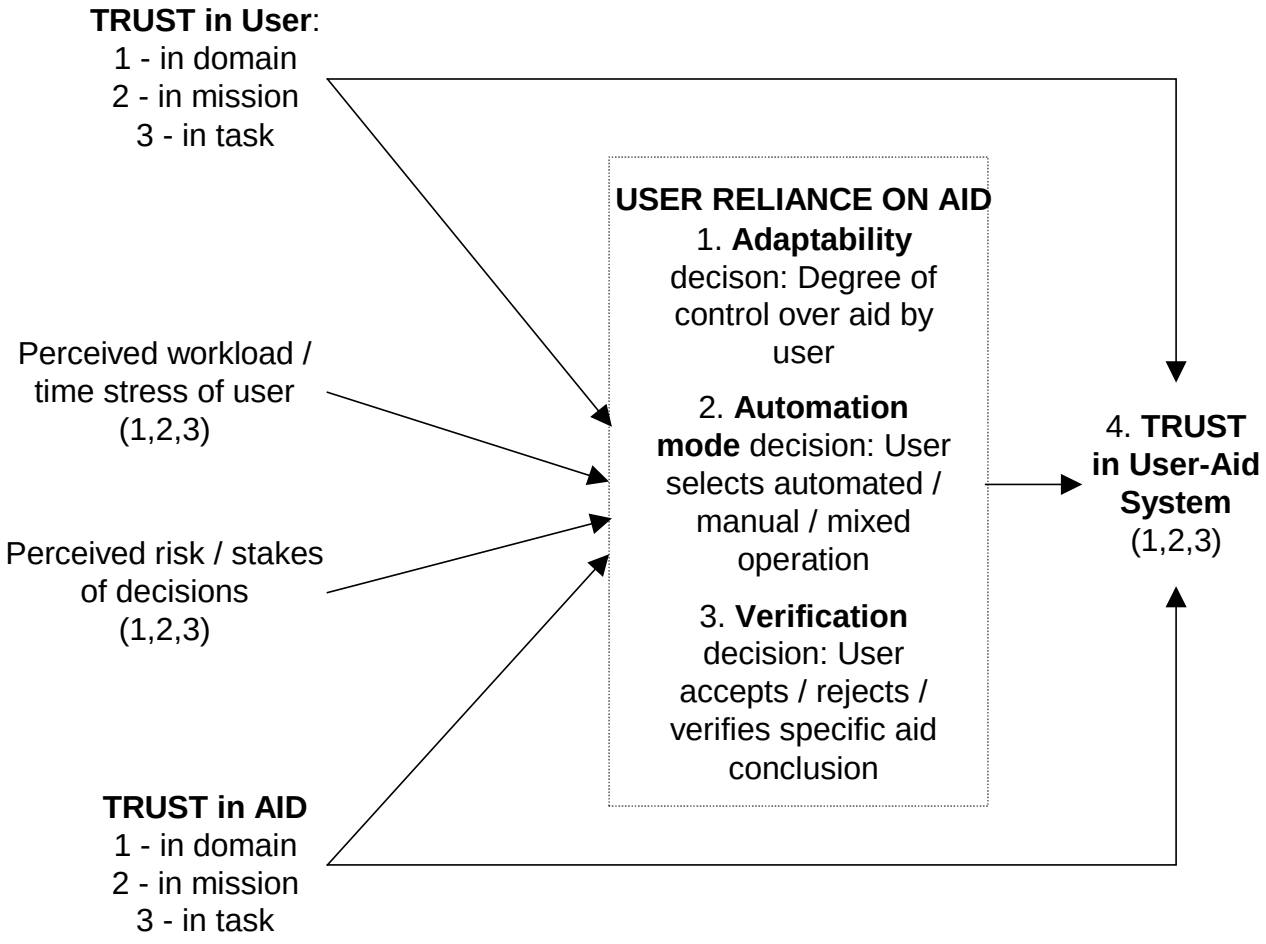

Figure 6. How trust and other factors might influence user reliance decisions. Numbers represent decisions at different temporal phases, and the factors that affect them in the corresponding phase.

In phase 4, the outcomes of system and user performance become known. Trust in the user-system interaction, at any time prior to that, is simply the user's expectation of the performance that will be observed in phase 4, averaged over the relevant temporal interval. Trust in this new sense, i.e., trust in overall user-system interaction, is a function of the user's self-trust, the user's trust in the aid by itself, and the way performance by the user and aid are mixed in the decision making process by reliance decisions that affect the relevant temporal interval.

Trust and the Verification Decision

In the verification decision, the user decides whether to accept, reject, or consider further a specific aid conclusion. We will use this decision to illustrate the way reliance decisions can be modeled within the APT framework. Each reliance decision (e.g., to monitor the aid in phase 2, or to verify a specific conclusion in phase 3) is based on all the information the user has collected along the path in the event tree up to that point. The verification decision, therefore, is based on information about the system, domain, and situation obtained prior to the current aid

recommendation. In addition, the user will have some information about the *aid recommendation itself* simply by virtue of having decided in phase 2 to monitor the aid: (1) First, the user knows the *type* of recommendation that is involved. Many decision aids support more than one type of decision, which the user might decide to monitor. For example, an attack planning aid may recommend a battle position as well as a route to that position. A target identification aid might classify contacts and also prioritize them for engagement. The user may trust the aid more on some of these matters than on others, and this trust will influence decisions about verifying the conclusions. (2) Second, the user may be aware of the *content* of the aid conclusion or recommendation, and this too can influence verification decisions. For example, the user may trust identifications of contacts as friends (since they are based on reliable Identification-Friend-or-Foe (IFF) procedures), but not trust identifications of contacts as foes (since friends may stray from designated areas or turn off their IFF transponders). (3) Finally, the user may be aware of such supporting information as the aid's own reported *confidence* level, or its *explanation* of its reasoning in arriving at the conclusion. These reports, too, (if the user trusts them!) may influence decisions about whether or not to verify the conclusion.

Verification includes a number of different activities, such as checking the aid's reasoning, examining the aid's conclusion against evidence known to the user but not to the aid (e.g., angle of attack or rotorwash), or attempting to find (or create) a better alternative. Verification is not usually a once-and-for-all decision. More typically, it is an iterative process. If the user does decide to verify an aid recommendation by collecting more information, that new information will then influence trust, and thus shape subsequent decisions to continue or not to continue verifying. If the user chooses to continue verifying, the user may consult more of the available evidence or try out a different verification strategy. The process should end when the uncertainty is resolved, the priority of the issue decreases, or the cost of delay grows unacceptable.

Decision Trees for Verification

The process of discovering new information or insights during verification can be pictured as an event tree, in which the user's trust in the aid evolves as new observations are made. It is illuminating to incorporate the decision whether or not to verify within such a tree as an event under the control of the user. A tree that includes both chance events and decisions is, of course, called a decision tree (Raiffa, 1968; Shafer, 1997).

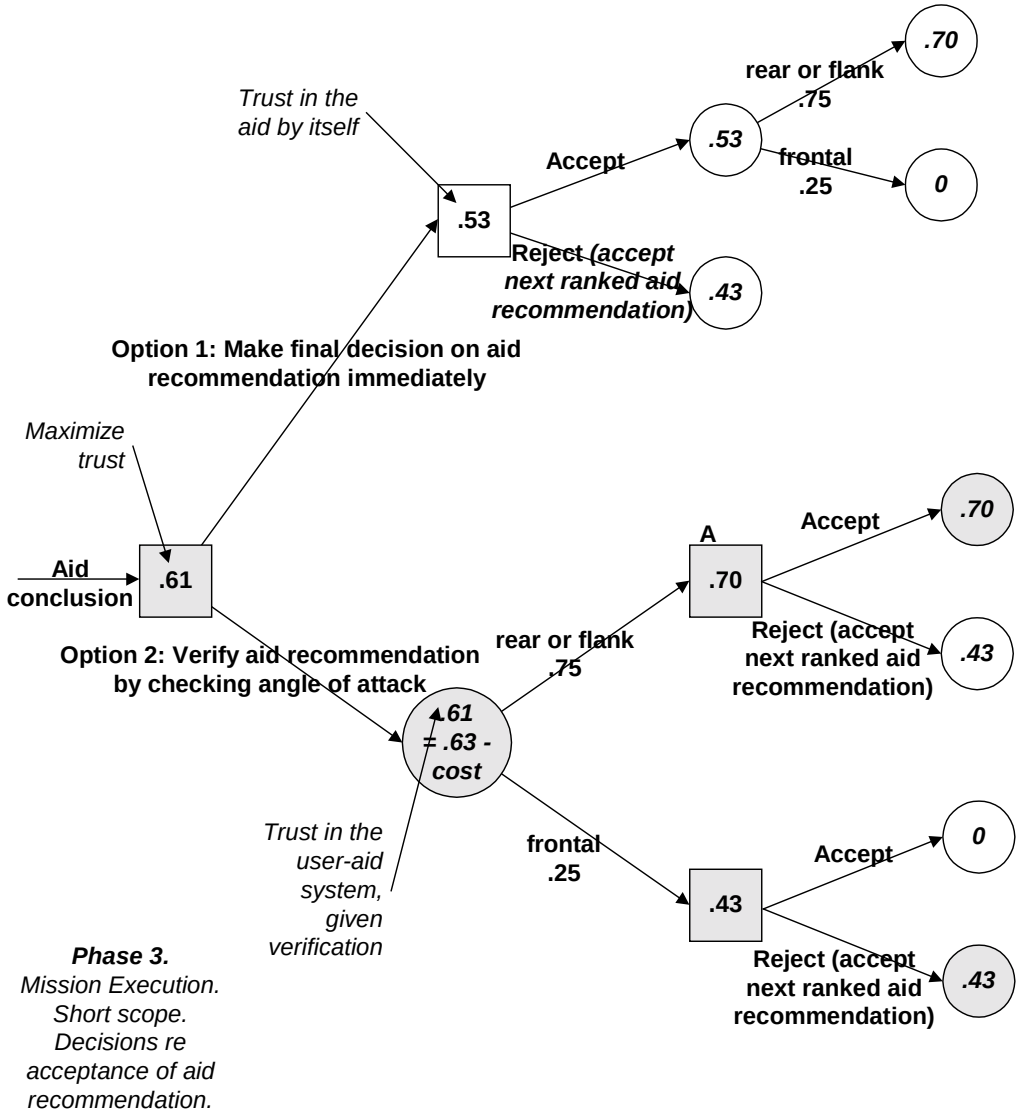

Figure 7. A decision tree showing a verification decision and the subsequent decision to accept, continue to verify or reject the aid's recommendation. Shading indicates the part of the tree that the user may traverse, depending on chance events (circular nodes) and decisions (square nodes).

Figure 7 is a decision tree with a simple verification decision, based on Figure 3. A user of the Battle Position Planner has been assigned a desert attack mission, and at the beginning of Phase 3 the aid has recommended a specific combat battle position. The user must first decide whether to accept this recommendation at once, reject it at once, or verify it by collecting more information.

This verification decision will determine the completeness and resolution of the subtree that the user will traverse during the remainder of phase 3. In particular, if users decide to verify the conclusion, they will traverse a subtree that contains branches for different angles of attack (like the tree in Figure 3). Angle of attack information will then be included in the grounds of subsequent judgments of trust, and the subsequent accept/reject decision will be based on this information. On the other hand, if users choose not to verify the aid's conclusion, they face a subtree that is missing branches for angle of attack (like the incomplete tree shown in Figure 4),

grounds for trust will not include angle of attack information, and the information will not be available for the accept/reject decision (although, like rotorwash, angle of attack may become known *later*, after the recommendation is executed in Phase 4). When the user chooses not to verify, the accept/reject decision will be based on *average* trust, aggregated across the various angle of attack possibilities, rather than on specific knowledge. One important goal of reliance decisions is to improve the *resolution* of the judgments upon which subsequent decisions are based.

The numbers at each node in Figure 7 do not represent the user's trust in the decision aid alone. In a decision tree, they represent the expected, or average, chance of successful collaborative user-aid performance in the future, conditional on any such collaboration that may have already taken place in the past.

Is it worthwhile for the user to verify the aid's recommendation? The answer is surprisingly simple, and involves a comparison of what is, in effect, *trust in the aid by itself* with *trust in the overall user-aid system given that the user will verify*. The user chooses the path through the event tree that gives the best chance of ending up with successful attack. We know from Figure 3 that the user's trust in the aid by itself before learning angle of attack is .53. This number reappears in Figure 7 as trust in the overall user-aid system given that the user chooses *not* to verify. In Figure 7, trust in the user-aid system given that the user *does* verify is .66 minus the costs of verification. Such costs may include heightened risk, for example, of being targeted by the enemy, or loss of opportunity to perform other important tasks. Suppose the user estimates these risks as no greater than a 2% reduction in chance of successful attack. Since the expected success of the user-aid system is greater if the user verifies the conclusion (.66 - .02 = .64) than if the user does not (.53), the user should verify. *Users can make reliance choices by maximizing trust.*

Value of Verification Information

The user in Figure 7 appears to benefit significantly by verifying the aid's recommendation. It is illuminating to consider why this benefit occurs. There are three basic requirements:

- *Significant uncertainty, i.e., chance of correcting an error*: The information to be collected by verification must be capable of discriminating among situations in which a subsequent decision by the user will be different. In Figure 7, a user who does not verify will *accept* the aid's top-ranked recommendation (chance of success = .53) rather than take a chance with the next ranked aid recommendation (chance of success = .43). A user who verifies angle of attack, on the other hand, may end up changing his or her mind, and *rejecting* the top-ranked recommendation. There is a .25 chance that verification will reveal a frontal angle of attack, and cause this change of mind. If the user does not verify, there is the same .25 chance that the decision to accept will turn out to be an error, due to poor angle of attack. By increasing resolution (i.e., reducing uncertainty), verification helps avoid mistakes.

- *Stakes*: The change in a subsequent decision (such as accepting versus rejecting a battle position) must make a difference to whatever the user values For example, it must change the chance of successful attack. The higher the cost of an error, the more valuable verification becomes. In our example, verification might help the user avoid the error of accepting a frontal angle of attack. If this happens, trust grows from 0 (if the user were to mistakenly adopt a recommended frontal battle position) to .43 (the overall chance that the next top ranking aid recommendation will be acceptable). This represents an improvement of .43 in

the chance of successful battle position selection.

- *Time*: The cost of verifying must be outweighed by the benefits, as represented by the first two factors. The expected benefit due to verification in this situation is the product of the first two factors, i.e., the chance of an observation that changes behavior (.25) and the improvement in chance of successful attack that results (.43): (.25)(.43) = .11. This is clearly greater than the cost in time, which is.02 chance of a successful attack. By deciding to verify —even before collecting the information about angle of attack — the user manages to increase the expected success of joint user-aid battle position selection (i.e., trust) by 9%.

A convenient tool for putting these ideas together, and for building benchmark models of reliance decisions, is the decision theoretic concept of *value of information* (VOI) (Cohen & Freeling, 1981; Raiffa & Schlaifer, 1961; LaValle, 1968). A simple formula for value of information, as applied to verification decisions by aid users, is the following:

Value of verification information = Sum over all observational outcomes that could change the user's subsequent decision
*[probability of the observational outcome * change in trust due to the change in decision – cost of time spent making the observation]*

The user should verify the aid's recommendation if this value is greater than zero. Value of information is a significant improvement over other information measures, such as entropy reduction, which measure the sheer quantity of information without taking into account the reason why information may be of value, i.e., its actual role to support decision making. And it is better motivated and simpler than the large number of rather vague measures typically used in information management system research, such as *completeness, precision, accuracy, relevance, timeliness, clarity, and readability* (see Cohen & Freeling, 1981, for discussion).

Benchmark Model for Verifying A Specific Decision Aid Conclusion

Despite their generality, measures based on the value of information have limitations. A major problem is that the information to be collected must be specified in advance (Cohen & Freeling, 1981). This is not unreasonable for the purpose of training quick recognition of standard patterns and associated responses (for example, "when in the desert, verify rotorwash"). However, the advantage of interactive over automated systems may be the human ability to handle novel and unexpected situations. In these cases, the possible results of a human intervention (such as verification) may not be known ahead of time. There are several, closely related problems:

1. *Visual recognition.* The verification process may be very straightforward in some cases, yet the potential observations cannot be anticipated. For example, the user of a target identification aid can verify identification of an image as a hostile tank simply by looking at the image, yet it might be very difficult to specify in advance all the relevant details that the user might see. (For an application of the model in that domain, see Cohen, Thompson, & Freeman, 1997.)

2. *Critical thinking.* The verification process itself may be less straightforward in some situations. For example, conflict between an aid's recommendation and their own or others' conclusions, may prompt a process of critical thinking, in which users look for an explanation of the differing recommendations. Resolution of the conflict may take the form of discovering unreliable assumptions that were implicit in the user's conclusions or the aid's. It is virtually impossible to make all assumptions explicit in

advance in an event tree. Key assumptions may come into focus only when they lead to problems, such as conflicting recommendations (Cohen, Freeman, & Thompson, 1997).

3. *Novel situations.* More generally, new issues to investigate may spring up as a result of unique or unusual circumstances, or due to the pattern of ongoing verification results. Just as novel situations may not be anticipated by the designer of a decision aid, so they may not be anticipated by the training designer.

Fortunately, these (and other) difficulties can be surmounted without giving up the essence of the value of information approach. We will describe a simple framework for deriving benchmark models of verification performance, without specifying all possible observations, in situations where previously learned or explicitly identified patterns may be insufficient to guide decisions about user-aid interaction. This framework will apply even when verification involves visual recognition of unanticipated patterns, critical thinking that ferrets out hidden assumptions, and creative problem solving in novel situations.

The solution is to derive necessary conditions, or *constraints*, that must be satisfied if any verification at all is to be of value. If the situation does not satisfy these constraints, verification cannot be worthwhile, regardless of the number of unmodeled potential observations and insights. These constraints need not be static, but may change dynamically as a the situation itself evolves.

The constraints are derived based on the principle that if perfect information would not be worth the cost of collecting it, then no other information is worth the cost either. Perfect information is defined, in this context, as information that discriminates all situations for which different actions would be appropriate. If verification that produced perfect information is not worthwhile, then it cannot be worthwhile under more limited conditions. It turns out that these constraints can be expressed simply, in terms of current trust in the aid, the cost of verification, and the potential costs of errors that might be avoided by verification (see Cohen et al., 1997, Appendix A, for a derivation). In particular, users should accept an aid recommendation without verification if:

trust > 1 - cost of verification / the cost of incorrectly accepting the aid recommendation

If the aid's conclusion is binary (e.g., classification of a contact as friend or foe), we get two constraints on the user's verification decision: an upper bound on trust (above which users should simply accept the recommendation) and an lower bound on trust (below which the users should simply accept the *negation* of the aid recommendation). It may be appropriate to verify the conclusion if neither of the two constraints is satisfied, i.e.:

1 - cost of verification / the cost of incorrectly accepting the aid's recommendation > trust >
cost of verification / the cost of incorrectly rejecting the aid's recommendation

Figure 8 represents a benchmark model for a binary decision based on these constraints. At any point in time, the vertical dimension, representing trust, is divided into two or three regions. If trust in the aid's conclusion falls in the upper region, the user should simply accept the conclusion (e.g., engage the target), without taking further time for verification. If trust in the aid's conclusion falls in the lower region, the user should reject the aid's conclusion without taking further time. (For example, a target identification aid concludes that a vehicle is an enemy tank, but the user is reasonably sure based on visual identification that the target is a friendly.) If trust is neither high nor low, but falls in the intermediate region, then it may be worthwhile for

the user to take more time to decide what to do.

In Figure 8, the dashed line shows that trust in the aid begins relatively low and warrants further verification of the aid's conclusion. After a while, however, the user's confidence in the aid increases as more information is collected. For example, the angle of attack of a recommended battle position might be discovered to be flanking, it is then observed that there is lots of room for other aircraft in the recommended position, and so on. Trust soon becomes high enough to enter the upper region, where the aid conclusion should be accepted. At this point, the user should stop thinking and act.

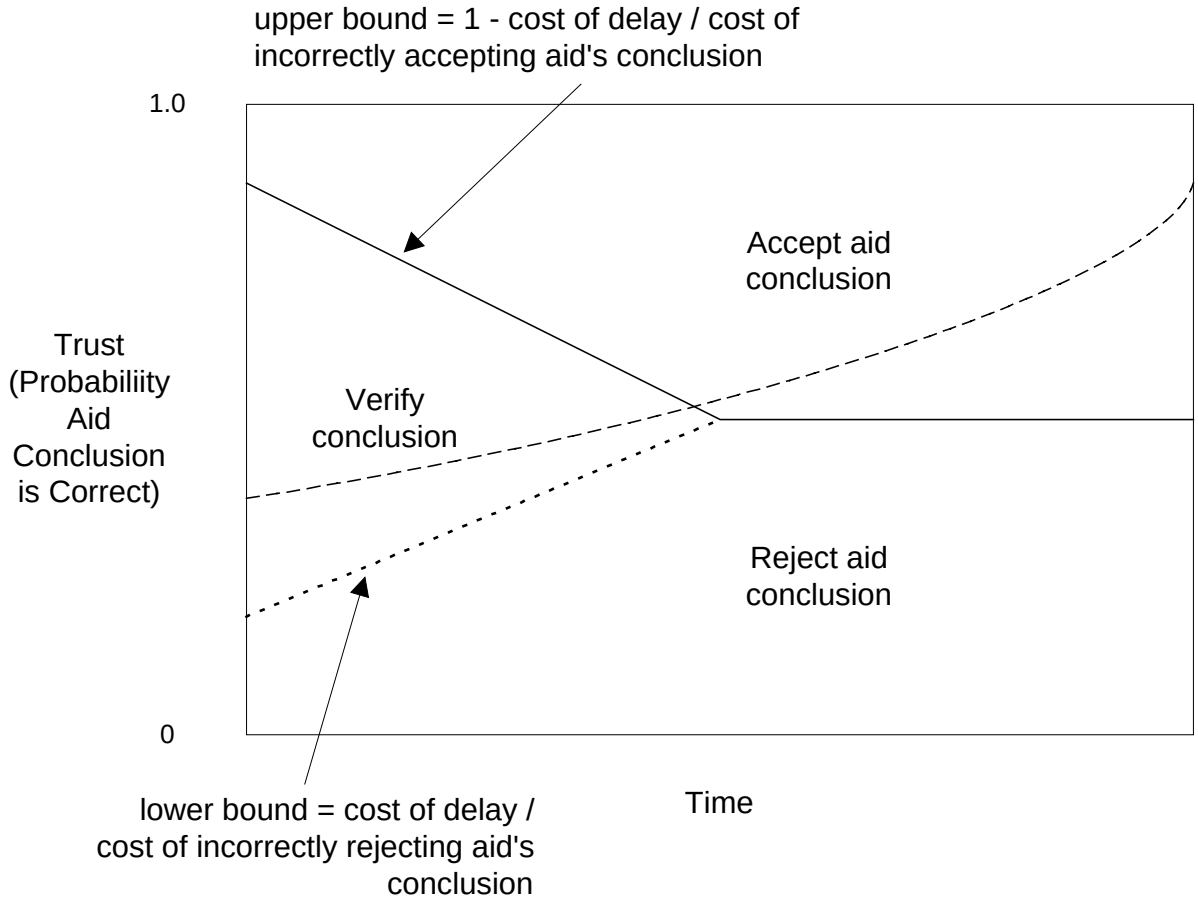

Figure 8. Benchmark model for deciding when to accept, reject, or take time to verify a decision aid's conclusion. Trust is represented by the dashed line.

What determines reliance decisions in this model? Like any value of information model, this surprisingly simple representation has only three key variables: uncertainty, time stress, and stakes.

1. *Uncertainty* pertains principally to the *resolution* of the trust assessment, i.e., the proximity to zero or one of the probabilities discriminated by the user. The less resolution in the user's assessment of trust, the more likely that an assessment will fall in the middle region of Figure 8, and the user will tend to utilize more time before making a decision. As we have noted, the average resolution of a trust assessment is influenced by the *completeness* of the user's knowledge of conditions that affect system performance. The more complete the knowledge of

relevant features of the domain, situation, task, and system, and the more reliably these features are observed on a given occasion, the closer the calibrated trust assessments will come to zero or one. We now see an important implication of the connection between completeness and resolution. *Training that improves a user's knowledge (prior to phase 3) of features that predict aid performance will reduce the amount of time the user needs to spend verifying the system (in phase 3).* Informed users will be able to assess the value of aid recommendations more quickly.

2. Time stress is represented by the cost-of-delay parameter in the equations determining the upper and lower bounds. When the cost of delay is great, action is more imperative, even with high uncertainty about trust. The cost of delay need not be constant, but may itself be a function of time. As time stress increases, the upper and lower boundaries move toward each other, squeezing out the region in which verification is appropriate. In Figure 8 the cost of each further moment of delay is higher than the one before, until the upper and lower bounds meet, and the user must act, regardless of the level of trust.

3. Stakes. Time stress affects the upper and lower bounds symmetrically. By contrast, there are two different kinds of stakes, corresponding to the costs of mistakenly accepting or rejecting the aid's conclusion, respectively, which affect the two bounds independently. To think about stakes, the user simply asks, regarding whatever action he or she is about to take, *what are the consequences if I am wrong?* The more severe the consequences of a mistake, the more difficult it is to clear threshold for taking the corresponding action. For example, suppose that a target identification aid recommends engagement of a contact, and the user considers accepting this recommendation. The cost of an error is the difference in the expected value between engaging an inappropriate target and not engaging it. Engaging an inappropriate target is likely to be more costly, the higher the proportion of friendlies among the non-targets. As a result, increasing the number of friendlies in the area will raise the upper bound, setting a higher requirement for trust before acting on the aid's recommendation to engage.

In the same way, as the cost of incorrectly rejecting the aid's conclusion increases, the user's distrust must be greater (or trust lower) to justify doing so. The cost of failing to engage an appropriate target is higher as the target becomes more threatening to one's own platform or to other friendly assets.

Similar benchmark models have been developed for other reliance decisions at all phases of decision aid use. These models address such issues as: deciding whether or not to verify an aid conclusion when there are more than two responses (e.g., classification of a target as tank, apc, truck, jeep, etc.); the verification decision when there is an open-ended set of possible conclusions (e.g., discovering a good battle position or developing an attack plan); automation mode decisions (e.g., assigning primary responsibility for a task to the user or to the aid, choosing whether or not to monitor aid conclusions, choosing whether or not to let aid to monitor the user's performance); and modification of decision aid parameters.

3. A FRAMEWORK FOR TRAINING DECISION AID USERS

The most important implications of the APT framework are for training development. Various aspects of the model lend support to different elements of a comprehensive training strategy for decision aid users. Salas & Cannon-Bowers (1977) have proposed a framework to describe training. According to them, a training *strategy* orchestrates *methods* (such as instruction and practice) and *tools* (such as simulation, feedback, and performance measures) to convey a *content*. Table 2 summarizes the elements of a training strategy based on APT.

It is worth pointing out the distinction between the APT framework, which attempts to

model trust judgments and reliance decisions, and training based on that framework. We do not advocate teaching APT to decision aid users, nor do we propose that they learn argument structures, event trees, or decision trees, or that they assess their trust in terms of probabilities. However, the elements of APT provide resources for generating training content, methods, and tools, as shown in Table 2.

We discuss some of these elements in the remainder of this section.

Table 2. Outline of a training strategy for decision aid users based on APT.

Content to be trained	**Training Methods**	**Training Tools**
Task-organized understanding of decision aid performance Dynamic situation awareness of factors affecting aid performance at each phase of use Critical thinking strategies for novel situations Strategies for interacting with decision aids based on trust at each phase of use Methods for choosing among different interactive strategies	Introduction to concepts via brief lecture and discussion Guided simulation-based practice, modeling of desired responses, feedback	Interviews with designers, domain experts, and users to identify factors affecting aid performance and plausible user strategies Simulation scenarios based on event and decision tree models Feedback based on event tree and decision tree models Performance measures based on APT parameters

Training Content

Task-organized understanding of aid. Event trees provide a rich framework for summarizing the kind of knowledge required in effective decision aid use. These structures spell out the features that are predictive of successful decision aid responses and that are required for user decisions about reliance on the aid at each phase of use. They indicate when such information becomes available, and provide a clear mechanism for relating the information to assessments of trust.

Training in decision aid use should be derive its organization from the structure of the task, rather than from the structure of the decision aid. For example, training in use of the Battle Position Planner, and RPA in general, addresses the role of the aid, and possible user interaction strategies, at each stage of a mission. Task-organized training is likely to be better integrated by pilots, and better recalled in the real world, than training that is organized according to the architecture of the aid. The temporal structure of event trees lends itself to this kind of

organization.

Critical thinking about decision aid performance. In addition to the substantive knowledge about the aid represented in event trees, skilled users will also be adept in handling novel or unanticipated situations. Novel conditions may occur that were specifically anticipated neither by aid designers nor by user training. Critical thinking skills can help users learn to handle surprises effectively when they occur. For example, the following two critical thinking strategies may supplement the knowledge embedded in mental models:

1. *Detecting and handling conflict.* Conflict among different sources of information about trust can be a symptom of erroneous assumptions in a user's understanding of the decision aid, or in the user's understanding of the situation. For example, observations of actual aid performance under various conditions may violate the expectations generated by an event tree, or there may be a surprising difference between an aid's conclusion and the user's independent judgment. Situation awareness must be expanded to include such symptoms of trouble. Users can be trained to be alert to such conflicts and to use them as opportunities to learn more about the situation and the system (Cohen, Freeman, & Thompson, 1997).

2. *Devil's advocate.* When stakes are high and time is available, devil's advocate strategies can be effectively employed for uncovering hidden assumptions and generating alternative interpretations of events. In such a strategy, users try to generate arguments against a favored conclusion. For example, users may imagine that a conclusion of their own or of the aid is false, and to explain how that could be so. Such strategies have been found to be an effective countermeasure against overconfidence (Koriat, Lichtenstein, & Fischhoff, 1980) and to be successfully trainable in realistic operational settings (Cohen, Freeman, & Thompson, 1997).

Dynamic decision-aid driven situation awareness. A corollary to the development of adequate mental models of aid performance is the development of situation awareness required for applying those models. Another training requirement, therefore, is to help users develop the monitoring and observational skills necessary to track events that are diagnostic of decision aid reliability.

Such situation awareness is relevant at every phase of decision aid use. For example, at phase 2, users may need to monitor for information that could suggest the need for a change in automation mode. If conditions occur that are correlated with relatively poor aid performance, users may need to monitor the aid's conclusions more closely or even switch to manual mode. Similarly, at phase 3, users need to monitor for information signaling that more thorough verification of an aid conclusion is appropriate.

Event trees help define the features that should be monitored for at each of these phases. In addition, however, users may need to monitor for situations that call for critical thinking, for example, when the degree of novelty or uncertainty in the situation suddenly increases. By definition, such situations cannot be anticipated ahead of time in an event tree.

Interaction strategies. The concept of differentiated, context-specific trust implies the relevance of a wide range of interaction strategies other than simply accepting or rejecting the decision aid as a whole. Such strategies represent different ways of blending the strengths of the aid and the user. For example, the following are among the strategies for which users of the Battle Position Planner have been trained:

- Monitor the aid's conclusions / verify in detail those which seem suspect.

- Constrain the aid's conclusions in advance, e.g., by ruling out certain geographical areas.
- Monitor for conditions in which different factors become more or less important, and adjust the corresponding weights or thresholds that the aid used to arrive at its conclusions.
- Generate your own solution and let the aid evaluate it.

Choosing among interaction strategies. Decisions about which interaction strategy to adopt must be made quickly. Otherwise, users will incur the risks of delay without any of the benefits. For example, thinking in phase 3 about whether or not to verify an aid's conclusion can take time away from actually doing so. Or the time spent thinking in phase 2 about whether to select a more automated aiding mode may rob users of the advantage of automation. The premium on speed increases as the temporal scope of the decision decreases. Thus, the verification decision at phase 3 must be made more quickly than an automation mode decision at phase 2.

The goal of training is to sensitize users to patterns of cues that can be quickly and intuitively recognized. Such patterns (rather than decision tree models) are the real content of training. Benchmark models (such as Figure 8) are a valuable tool for identifying the elements of the patterns to which users should be sensitive, and the manner in which they should respond to them. We saw in the previous sections that such patterns can be simply characterized in terms of uncertainty, time stress, and stakes.

Training Tools

Scenarios and feedback. Event tree representations are useful in the construction of training scenarios and the design of feedback. The sequence of significant observations regarding aid performance that is represented in the event tree can serve as the basis for the design of scenarios that vary the features of the system, mission, task, and/or aid conclusion. In conjunction with probabilities, such event trees can be used to generate a population of scenarios with controlled statistical properties, e.g., concerning the chance of success with the aid under various conditions.

Such scenarios afford an opportunity to observe the effects of significant events on a users' assessments of trust and users' interaction decisions (such as selection of automation mode or acceptance/rejection of an aid conclusion). Debriefings can use event tree representations to provide specific feedback regarding trainee's performance with respect to features of the event tree that they failed to respond to or may have responded to inappropriately.

Benchmark models can be used to set up a series of training scenarios in which different reliance decisions are appropriate. Such scenarios can then be used to provide practice and feedback to decision aid users in making appropriate reliance decisions.

Figure 9 through Figure 12 show how a set of training scenarios for phase 3 verification decisions might be generated by systematically manipulating two of the three key variables — time stress and stakes. For this example, we have kept trust constant, at .4 chance that the aid's recommendation is correct. The aid has recommended that a contact be engaged. Stakes are varied for the upper bound, by manipulating the mix of friendlies and enemy non-targets, thus affecting the expected cost of a mistaken engagement. Time stress is varied by manipulating the rate of increase in the danger of being targeted as the user spends more time unmasked.

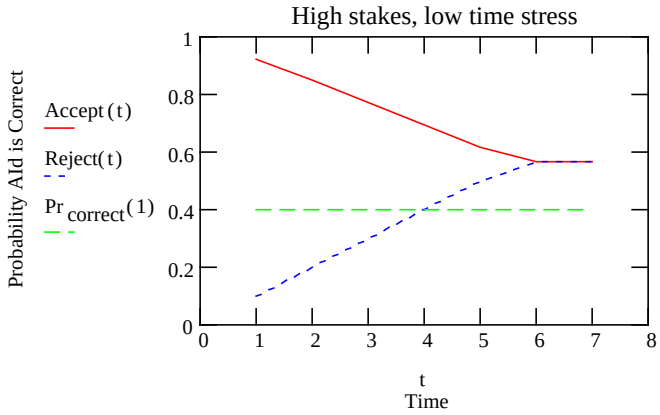

Figure 9. Scenario in which there is a large proportion of friendlies relative to enemy non-targets, producing high stakes of incorrectly accepting the aid's recommendation to engage. The probability of being targeted by enemy platforms is low, but increases with time. Trust is highly uncertain, at .4. The result is a significant amount of time (from time 1 to time 4) spent verifying the aid's recommendation to engage. Finally, the cost of remaining unmasked leads to a decision (in this case, not to engage).

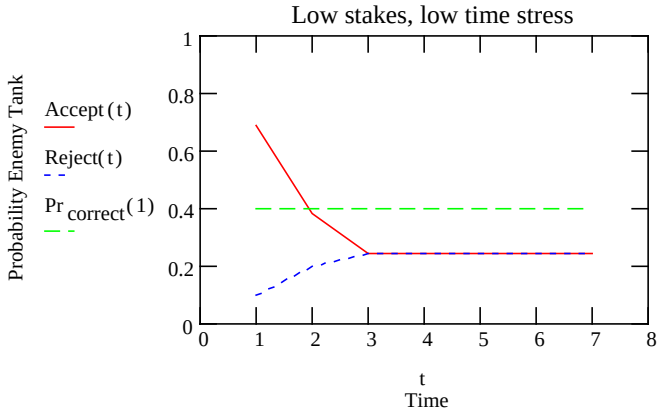

Figure 10. Scenario in which the low proportion of friendlies relative to enemy non-targets leads to a low threshold for engagement. Even though time stress is low (as in the previous example), less time is spent verifying the aid's recommendation (from time 1 to time 2) because of the low cost of an error. A relatively quick decision is made to engage.

In these scenarios, the user (or trainee) must decide not only what to do — i.e., whether to engage or not to engage a contact — but how long to wait before doing it. In two of the scenarios (Figure 10 and Figure 12), the appropriate action is to accept the aid's recommendation and engage, while in the other two (Figure 9 and Figure 11), the appropriate action is to reject the aid's recommendation and not to engage. The appropriate time spent verifying the aid's recommendation varies from 3 units (in Figure 9) to 1 unit (in Figure 10 and Figure 11) to 0 units (in Figure 12). Trainees can be evaluated and given feedback on both dimensions, the engagement decision and the time taken to make it. Exercises of this kind can help maintain skills in the primary task, while enhancing the ability to interact effectively with a decision aid.

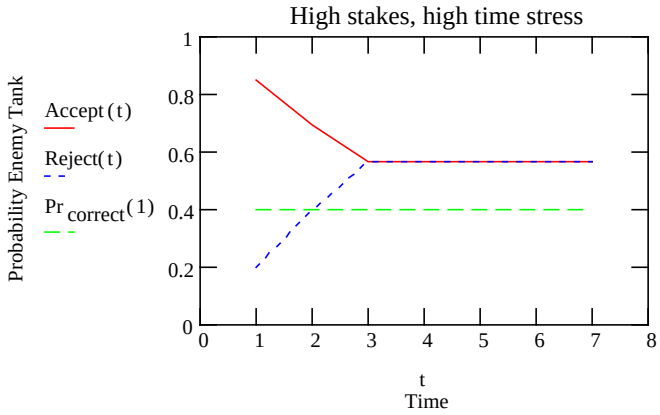

Figure 11. Scenario in which the cost of a mistaken engagement is high, due to a high proportion of friendlies. However, time stress is also high, due to a rapid increase in the chance of being targeted with time spent unmasked. This results in a relatively early decision, in this case not to engage.

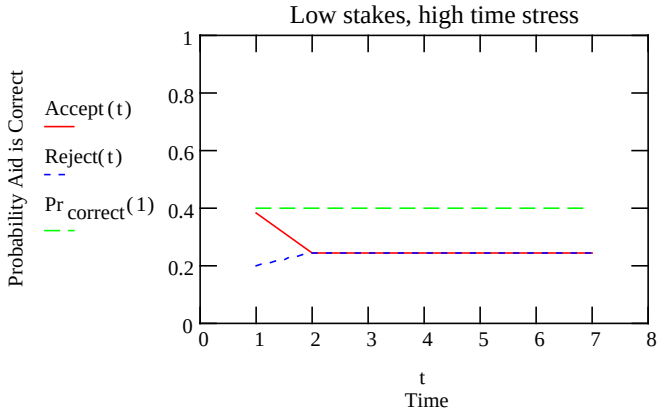

Figure 12. Scenario in which the cost of a mistaken engagement is low (due to low proportion of friendlies) and time stress is high (due to rapidly increasing chance of being targeted). The result is no time spent verifying aid's recommendation, and an immediate decision to accept the recommendation to engage.

In the above examples, the upper and lower bounds were independent of trust in the aid's conclusion, and trust remained constant. As Figure 13 illustrates, however, neither of these conditions is necessary. In this example, trust again starts at .4. However, in verifying the aid's recommendation to engage, the user finds evidence that supports the aid's identification of the contact as hostile. As the user becomes increasingly convinced that the contact is hostile, there is also a rise in the chance of being targeted. In short, time stress increases along with trust. The result is a somewhat earlier decision to engage the target, as compared with Figure 9.

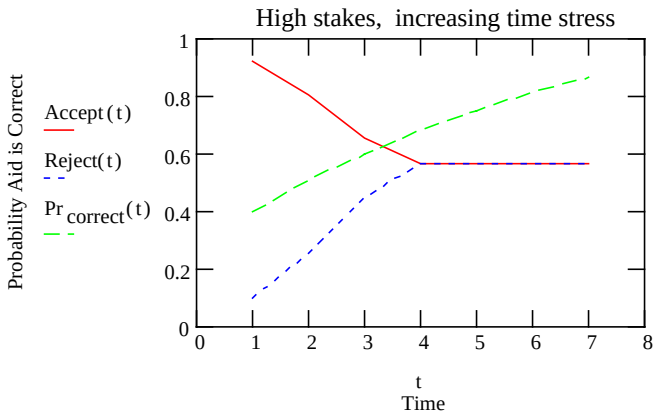

Figure 13. Scenario in which trust in the aid's identification of the contact as hostile increases, bringing with it an increase in time stress due to the expectation of being targeted. The result is a somewhat earlier decision to engage than in Figure 9, which is otherwise based on the same underlying parameters.

4. DISCUSSION AND RELATIONSHIP TO PREVIOUS WORK ON TRUST

Recent research by Muir, Moray, Lee and others has pioneered the application of the concept of trust to the human use of automation. Muir (1987, 1994) introduced a multi-dimensional definition of trust. Muir and Moray (1987) described a non-obtrusive method for eliciting subjective assessments of trust from users. Lee and Moray (1994) and Muir and Moray (1996) showed that such assessments of trust could be correlated with subjects' use of automation. Nevertheless, this work, as it stands, has shortcomings both in clarity and in the completeness with which it makes distinctions that are required for effective application to training. APT was developed to address these shortcomings.

Muir's (1987, 1994) definition of trust in automation borrows from and integrates two models of trust among humans. The first dimension of trust (Barber, 1983; shown in the first column of Table 3) specifies three kinds of expectation: *persistence* of physical, biological, and moral regularities, *technical competence* at skill-based, rule-based, and/or knowledge-based levels (Rasmussen, 1983), and *fiduciary responsibility*, or the expectation that designers' motives are reliable.

The second dimension of Muir's theory (from Rempel, Holmes, & Zanna, 1985; shown in the second column of Table 3) is meant to be orthogonal to the first and describes the evolution of trust with experience. Trust evolves from *predictability* of the machine's behavior, to *dependability* of the machine's enduring dispositions, and finally to *faith*, or the conviction that the machine will behave as expected in unknown situations.

Lee and Moray (1992) argued that the two dimensions in Muir's theory were complementary rather than orthogonal. The arrows in Table 3 show components that they equate to one another. In particular, they regard faith and fiduciary responsibility as variants of the same concept. Both refer to the basis for trust in situations where the user has little experience with the automation and must fall back on expectations of underlying motives and intentions of the designer. Similarly, Lee and Moray merge the concepts of predictability and technical competence, claiming that each refers to "stable and desirable behavior or performance." Finally, Lee and Moray map Muir's conceptualization onto a classification of aspects of trust by Zuboff (1988; shown in the third column of Table 3). Zuboff's *trial-and-error experience* is equated to predictability and technical competence. Zuboff's *understanding* is equated to dependability.

Zuboff's *leap of faith* is equated to faith, and hence, to fiduciary responsibility.

Confusion regarding the distinctiveness of these concepts is a symptom, we think, of their lack of clarity and the absence of unifying principles in Muir's framework. Moreover, there are other, highly significant distinctions that Muir neglects or fails to make at all. These additional distinctions are crucial if trust is to be differentiated by context and by temporal scope, and thus support a variety of user interaction decisions at different phases of aid use. APT is intended to capture this more differentiated conception of trust in a systematic and clear way.

Table 3. The first two columns represent Muir's two-dimensional model of trust. Arrows link concepts in Muir's and Zuboff's models that Lee & Moray believe to be equivalent.

Types of expectation (Barber)	**Basis of expectation (Rempel et al.)**	**Aspects of trust (Zuboff)**
Persistence Physical Biological Social	Predictability (of acts)	Trial & error experience
Competence Skill-based Rule-based Knowledge-based	Dependability (of dispositions)	Understanding
Fiduciary responsibility	Faith	Leap of faith

Table 4 compares the first dimension in Muir's framework to corresponding elements of APT. It is not at all clear in what sense persistence, competence, and fiduciary responsibility are supposed to be different "types of expectation," much less a complete classification of all possible types of expectation. As indicated in Table 4 (second column), persistence refers to one kind of temporal scope of a trust assessment, competence to one sort of grounds for a trust assessment, and fiduciary responsibility to one kind of backing for a trust assessment. Rather than forming a single dimension, each "type of expectation" is thus better regarded as part of a separate dimension, or set of distinctions, pertaining to trust.

More importantly, these distinctions are not adequately explored in Muir's model. In particular, as shown in the third column of Table 4, a more differentiated conception of trust requires judgments over shorter temporal scopes than persistence for all time, requires finer discriminations than skill-based, rule-based, and knowledge-based competencies, and relies on other sources of knowledge than assumptions of fiduciary responsibility.

In her second dimension, Muir represents the evolution of trust with experience as a progression from *predictability* to *dependability* to *faith*. This progression seems to presuppose

only good news about the performance of the aid. The value-laden terminology (dependability, faith) precludes the possibility that *distrust* might also evolve as a user acquires experience with a system. More importantly, it does not allow for the possibility that conditions of trust and distrust might become better differentiated by a user with experience. If the user is fortunate, the aid will perform well under most or even all conditions, but this is not necessarily so. If not, the user may have to learn conditions of both good and bad performance, i.e., increase the resolution of his or her trust judgments.

As indicated in Table 5, Muir's second dimension is a hybrid of completeness and trust. It thus confounds consistency and desirability in system behavior. APT has three basic metrics for trust, rather than one: Trust itself (i.e., the context-specific chance of successful system performance), completeness of the grounds and backing of the trust assessment, and the reliability of the trust assessment. APT allows for completeness and reliability to increase with experience, without begging the question of whether trust itself increases or decreases, or simply becomes more differentiated by context.

Note that APT clarifies some of the distinctions originally made by Muir that Lee and Moray found obscure. For example, fiduciary responsibility and faith both involves assumptions, but are otherwise quite different. Fiduciary responsibility (Table 4) is a specific sort of backing for an argument about trust, involving assumptions (about the designer's motives) in the absence of other knowledge. Faith (Table 5) appears to be a high level of trust combined with completeness of backing for the assessment, based on extrapolation from a base of extensive knowledge about system performance.

The three aspects of trust in Zuboff's model map onto different sources of knowledge, or backing, for a trust assessment: trial and error experience with a decision aid, understanding of the aid's design, and "leap of faith" (i.e., making assumptions to fill gaps in experience and design knowledge). Other sorts of backing, and finer discriminations among sorts of backing, might sometimes be important, however, such as talking to other users, basing expectations on analogies with more familiar types of systems, and different types of assumptions (e.g., projecting one's own traits into the decision aid, or making worst case rather than best case assumptions about designers).

In sum, APT is rich enough to capture all the distinctions made by Muir and Zuboff, as well as many others they did not make. However, APT is simple and clear enough to incorporate all these distinctions within a single integrated framework, based on arguments about expected system performance.

Table 4. The first dimension of Muir's framework, interpreted within APT.

Types of expectation (Barber)	**Corresponding element of APT**	**Distinctions omitted by Muir's model**
Persistence Physical Biological Social	Persistence represents a prior bias regarding the trustworthiness of very broad classes of systems (physical, biological, and social). Thus, it involves the longest possible *temporal scope* of trust judgments, and corresponds to a phase of decision aid use prior to knowing anything about a decision aid other than that it is a physical system used by a biological system within a social organization.	More differentiated trust assessments involve more limited temporal scope, and appear to have more relevance to decision aid use than highly generalized biases. For example, Muir's model omits trust in a particular aid over the span of its existence, trust in a particular aid during a particular type of mission or task, and trust in a specific aid conclusion.
Competence Skill-based Rule-based Knowledge-based	Judgments of competence simply mean that the type of task undertaken by a decision aid can form part of the *grounds* for assessing trust. For example, a particular system may have a better chance of successful performance in rule-based tasks than in knowledge-based tasks.	This is only one of many variables that can affect predictions of system performance. Far more differentiated judgments are possible. For example, a medical expert system might be better for diagnosing infectious diseases than pulmonary disorders; a planning aid might be less trustworthy when a particular factor, e.g., rotorwash, is important; and so on.
Fiduciary responsibility	Fiduciary responsibility involves making assumptions about the good motives of system designers. As such, it is a sort of *backing*, or source, for trust judgments. In the absence of more direct experience with an aid, users might have to fall back on such assumptions.	Fiduciary responsibility is only one sort of assumption users might make (e.g., they might assume the worst regarding the designers' motives or competence). In addition, assumptions are only one kind of backing for a trust assessment. Other sorts of backing include direct experience with the aid, talking with other users, analogies to other kinds of aids, and design knowledge.

Table 5. The second dimension of Muir's model, interpreted within APT.

Basis of expectation (Rempel et al.)	**Corresponding element of APT**	**Distinctions omitted by Muir's model**
Predictability (of acts)	Low to moderate completeness, high trust: The aid has been observed or is understood in a limited range of conditions and has been found to perform well.	As completeness increases, more and more conditions of performance are observed, but overall trust may either increase or decrease. The resolution of trust assessments, however, will increase as the user differentiates conditions of good and bad performance and makes more specialized assessments. Reliability and calibration may also increase with experience, independently of whether trust increases or decreases.
Dependability (of dispositions)	Moderate to high completeness, high trust. The aid has been observed or is understood in a wider range of potentially degrading conditions and has been found to perform well.	
Faith	Highest completeness, high trust. The aid has been observed or is understood in so many conditions and found to perform well that it is inferred / assumed to perform well everywhere.	

One further issue worth noting is the relationship between trust in a system and predictions of system performance. In Muir's framework (e.g., Muir, 1994, Figure 3), there is a one-to-one relationship between trust and expected automation performance. Because of this one-to-one relationship, there is no benefit in regarding trust as a separate intervening variable, distinct from expected automation performance itself. Simplicity and clarity are furthered by regarding them as one and the same.

Moreover, by treating trust as a measure of uncertainty regarding system performance, we gain the systematic advantages of probability theory over an ad hoc measure of trust. As we have seen, we can use the probabilistic aspect of trust to generate training scenarios based on event trees, to explicate qualitative considerations in choosing an interaction strategy, and to develop feedback.

Most importantly, APT provides a single, unified framework in which user assessments of trust, and user decisions about interaction with an aid, can be studied and trained at any stage of decision aid use, and under a wide variety of different conditions.

5. REFERENCES

Barber, B. 1983. *The logic and limits of trust*. New Brunswick, NJ: Rutgers University Press.

Cohen, M.S. 1986. An expert system framework for non-monotonic reasoning about probabilistic assumptions. In *Uncertainty in artificial intelligence,* ed. J.F. Lemmer and L.N.

Kanal. Amsterdam: North Holland Publishing Co.

Cohen, M.S., and Freeling, A.N.S. 1981. *The impact of information on decisions: Command and control system evaluation* (Technical Report 81-1). Falls Church, VA: Decision Science Consortium, Inc.

Cohen, M.S., Freeman, J.T., and Thompson, B.B. 1997.Critical thinking skills in tactical decision making: A model and a training strategy. In *Decision making under stress: Implications for training and simulation,* ed. A. Cannon-Bowers and E. Salas. Washington, DC: American Psychological Association.

Cohen, M.S., Thompson, B.B., & Freeman, J.T. 1997. *Cognitive aspects of automated target recognition interface design: An experimental analysis.* Arlington, VA: Cognitive Technologies, Inc.

LaValle, I.H. 1968. On cash equivalents and information evaluation in decisions under uncertainty. *American Statistical Association Journal, 63*:252-290.

Lee, J.D., and Moray, N. 1994. Trust, self-confidence, and operators' adaptation to automation. *International Journal of Human-Computer Studies, 40:*153-184.

Lee, J.D., and Moray, N. 1992. Trust, control strategies, and allocation of function in human-machine systems. *Ergonomics, 35*(10):1243-1270.

Mosier, K.L., and Skitka, L.J. 1996. Human decision makers and automated decision aids: Made for each other. In *Automation and human performance: Theory and applications,* ed. R. Parasuraman and M. Mouloua, 201-220.Mahwah, NJ: Lawrence Erlbaum Associates.

Muir, B.M. 1987. Trust between humans and machines, and the design of decision aids. *International Journal of Man-Machine Studies, 27*:527-539.

Muir, B.M. 1988. Trust between humans and machines, and the design of decision aides. In *Cognitive engineering in complex dynamic worlds* , ed. E. Holnagel, G. Mancini, and D.D. Woods. London: Academic Press.

Muir, B.M. 1994. Trust in automation. Part I. Theoretical issues in the study of trust and human intervention in automated systems. *Ergonomics, 37*(11):1905-1922.

Muir, B., and Moray, N. 1987. Operator's trust in relation to system faults*. IEEE International Conference on Systems, Man, and Cybernetics*, 258-263. Alexandria, VA.

Muir, B., and Moray, N. 1996. Trust in automation. Part II: Experimental studies of trust and human intervention in a process control simulation. *Ergonomics, 39*(3):429-460.

Parasuraman, R., and Mouloua, M. 1996. *Automation and human performance: Theory and applications*. Hillsdale, NJ: Erlbaum.

Parasuraman, R., and Riley, V. 1997. Humans and automation: Use, misuse, disuse, abuse. *Human Factors*.

Parasuraman, R., Molloy, R., and Singh, I.L. 1993. Performance consequences of automation-induced "complacency*." The International Journal of Aviation Psychology, 3*:1-23.

Parasuraman, R., Mouloua, M., and Molloy, R. 1996. Effects of adaptive task allocation on monitoring of automated systems. *Human Factors, 38*:665-679.

Raiffa, H. & Schlaifer, R. 1961. *Applied statistical decision theory.* Cambridge, MA: The M.I.T. Press.

Rasmussen, J. 1983. Skills, rules, and knowledge: Signals, signs, and symbols, and other distinctions in human performance models. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-13(*3):257-266.

Rempel, J. K., Holmes, J.G., and Zanna, M.P. 1985. Trust in close relationships. *Journal of Personality and Social Psychology, 49*:95-112.

Riley, V. 1989. A general model of mixed-initiative human-machine systems. *Proceedings of the Human Factors Society 33rd Annual Meeting—1989,* 124-128. Minneapolis, MN.

Riley, V. 1994. A theory of operator reliance on automation. In *Human performance in automated systems: Current research and trends,* ed. M. Mouloua and R. Parasuraman, 8-14. Hillsdale, NJ: Erlbaum.

Shafer, G. 1996. *The art of causal conjecture*. Cambridge, MA: The MIT Press.

Toulmin, S. 1958. *The uses of argument*. Cambridge, UK: Cambridge University Press.

Zuboff, S. 1988. *In the age of the smart machine: The future of work and power*. NY: Basic Books.