Running head: Latent Problem Solving Analysis

Latent Problem-Solving Analysis: A computational theory of representation in experienced problem solving

José Quesada, Walter Kintsch

Institute of Cognitive Science

University of Colorado, Boulder

Muenzinger psychology building

Campus Box 344

Boulder, CO 80309-0344

[quesadaj, wkintsch]@psych.colorado.edu

Phone: 303 492 1522 Fax: 202 492 7177

Emilio Gomez

Department of Experimental Psychology

University of Granada

Campus Cartuja, S/N

Granada (Spain)

egomez@ugr.es

Phone: 011 34 958 246240 Fax 011 34 958 246239

Submitted to: Cognitive Science

Abstract

A theory of problem solving in complex dynamic environments is proposed in which problem spaces are generated using the statistical properties of the environment. Latent Problem Solving Analysis (LPSA) uses log files that record the state of the environment and the actions of problem solvers to create a metric multidimensional space that allows computing similarity measures between any problem solving episodes. These similarity measures are shown to correlate well with human similarity judgments in two complex problems-solving tasks, a dynamic microworld simulation in which participants must extinguish forest fires and a more realistic task where instructors rate landings performed by airline pilots in a high-fidelity simulator. We propose that the problem space generated by LPSA reproduces accurately the knowledge acquired by problem solvers in these situations.

Keywords: Problem Solving, Knowledge representation, Complex systems, Computer simulation, Human factors, Latent Semantic Analysis

Latent Problem-Solving Analysis: A computational theory of representation in experienced problem solving

Introduction

There seems to be a lack of clarity on the representational assumptions that theorists make in problem solving. For example Gordon (2003) states that “one of the hallmarks of artificial intelligence planning systems is the general absence of representational commitments” (p. 23). Newell and Simon’s (1972) theory was not centered on knowledge representation, but on processes of information transformation (called methods). A method is a collection of information processes that combine a series of means to attain an end. They acknowledged that representation was not a main objective: “the theory to be presented in this book has much more to say about methods and executive operations than about creating new representations or shifting from one representation to another” (Newell & Simon, 1972, p. 90). This paper focuses on representation in experienced problem solving.

Although the representational assumptions of the classic problem space view are not clear, it seems that most authors would agree in that the problem space can be represented by *finite directed graphs* (e.g., Newell & Simon, 1972; Richman, Gobet, Staszewski, & Simon, 1996)[1]. Examples of structures used that fit in the general description of directed graphs are list structures and attribute-value associations. However, most theories of problem solving do not define explicitly the particular representational format used, or do not propose mechanisms to generate the representations from the information that the problem solver has available.

The theory we present, Latent Problem Solving Analysis (LPSA) was designed to compensate for this lack of representational clarity, and to be applied to complex problem-solving tasks that change in real time, where the deliberate, time-consuming operations that the classical, rule-based approach to problem solving proposes are not always optimal. LPSA is based on Latent Semantic Analysis {LSA, \Landauer, 1997 #25. In LSA, the assumption is that people can learn the meaning of words by just experiencing them in different contexts. LSA defines a context as a passage of text, for example a paragraph or document. LSA is trained on a corpus (that is, a collection of documents). In LSA, each different word (type) is represented as a function of all the other word occurrences (tokens) in all the passages in the corpus. LSA creates a multidimensional space of 100-500 dimensions reducing the dimensionality of a matrix of contexts by types. Each type is then represented as a point in this space. LPSA proposes that this mechanism may be more general in cognition than simply word meaning, and that several other cognitive skills can be represented in a similar way.

LPSA's main assumption is that some problem spaces can be better described as metric multidimensional spaces and these spaces can be derived by the application of a self-organizing procedure to a huge set of raw trial log files. In that sense, the theory does not rely on verbal protocols, similarity judgments, expert knowledge elicitation techniques, or analytical descriptions of the task such as manuals to build the space. Instead, LPSA uses representative samples of people's interactions with their tasks.

One of the basics ideas of our proposal is that we should use the same information in quantity and quality that is available in the environment for the cognitive system. This type of approach proposes that the structure of the environment determines the design of a cognitive system that evolves in it. It was pioneered by Brunswik {, 1956 #78} and Gibson (1979), revived by Anderson (1991), and developed by the contributors in the book edited by Oaksford and Chater (1998) among others. The basic idea is that modelers should be concerned with the purposes of the cognitive system, because they can help reducing the degrees of freedom in the modeling process. Chater and Oaksford (1999) pointed out that cognitive science has been strongly dominated by mechanistic explanations (*how* the mind does things), but recently purposive explanations (*why* the mind does things) are gaining importance. With the advent of these *why* questions, modelers are increasingly looking at the environment for answers. The statistical properties of the environment are an object of study by themselves, because the cognitive system has to be adapted to them and this adds a good amount of cues about its functioning.

Step 2 in Anderson's rational analysis requires developing a formal model of the environment. In this aspect, cognitive science has not produced consistent integrative theories, but local theories that tend to be valid for a particular task or experiment only, and several theoreticians have raised criticisms against this practice (e.g., Burns & Vollmeyer, 2000; Newell, 1980). Many authors (e.g., Brunswik, 1956; Juslin, Olsson, & Winman, 1996; Simon, 1981; Vicente, 1999) have proposed that the environment is at least as important as the individual's cognitive system to explain behavior, but formal analyses about the structure of the environment and how it generates cognitive representations have lagged behind. In his parable of the ant and the beach, Simon's (1981) wondered why the path that an ant describes when walking is so complex, and how this complexity can be ascribed to the ant. However, he continued, it could be the case that the complexity is in the beach itself. In other words, the cognitive system's complexity can be a reflection of the structure of the environment. A formal description of the constraints of this environment can be very well the best approximation to the cognitive representation that the mind uses. LPSA is a computational representation of the beach obtained by analyzing the paths of thousands of ants.

In this paper, we propose that the similarity between events (states/actions) can be computed using contextual usage. We call this context-based similarity. Imagine that every task leaves a log file of the activities performed from the beginning to the end. This is a context in LPSA, or a path of one ant in Simon's beach. We can compute context-based similarity for any task in which we can define events and contexts. To define a context, we should consider how the events in a task are related. Since events depend on each other, the appropriate working definition of context should contemplate all possible dependencies. Most problem solving situations will contain long-term dependencies. For example, the opening move in chess will be related to all the other moves, even the last one (checkmate). Thus, a natural way of defining context would be to use the whole game in chess, or whole trials in other activities and experimental tasks. A trial ensures that even the long-term dependencies are captured, since there are no dependencies that cross the limit of a trial. To define events, we can concentrate on system states or participant actions. They tend to be complementary, so the decision to use one or another may be determined by the information available in the task. These are equivalent to words (types) in LSA.

LPSA uses the contextual information not only at the within-trial level, but at the corpus level. A corpus here is a collection of solutions to a wide set of problems. Each solution is a context, equivalent to a passage of text in LSA. LPSA starts with first-order co-occurrence relations between events and contexts, that is, it uses all the contexts in which a particular event has appeared. A dimension reduction step helps eliminating noise and improves the discriminability of each stimulus once placed in the multidimensional space. It is the global pattern of contexts what determines the constraints that are used to represent the environment, not only the local context. For example, imagine a universe in which 100 actions can appear in any of 100 contexts. Imagine that action A1 appears in contexts (C45, C10, C90), Action A2 appears in contexts (C45, C10, C15) and action A3 is present in context (C1, C2, C3). Then, actions A1 and A2 are more similar to each other than to A3, because they appear in more shared contexts.

The linear reduction to a multidimensional metric space can be considered the lowest complexity "minimum common denominator" of representational models. Metric spaces have been used as representing spaces in several areas of cognitive science, for example perception (e.g., Shepard, 1987), object recognition (e.g., Edelman, 1998, 1999) and semantics (e.g., Landauer & Dumais, 1997). However, to our knowledge no theory has proposed that representation in problem solving can be understood using the same formalism. A metric space of this type groups events (actions or states) according their function, which is what really matters in a problem-solving task.

We propose that this mechanism, the “lowest common denominator”, could be context-based similarity as implemented in LPSA. LPSA assumptions (e.g., linear relationship between components, non-structured representations due to the metric features of the space, etc.) may not be optimal for some tasks, but even an oversimplified representation such as a LPSA space can explain a good amount of phenomena of interest in complex, dynamic tasks. We think LPSA is a step in the direction of creating a model of representation in problem solving that is robust under a rich variety of circumstances. If a common formalism can account for very different tasks, and if it can be empirically validated without changing its basic assumptions, that would be a good source of evidence for the psychological plausibility of a theory.

The range of tasks that we used differs markedly from the kind of tasks that other theories of problem solving have tried in the past (see also Ehret, Gray, & Kirschenbaum, 2000; Gray, 2002; Quesada, Kintsch, & Gomez, submitted). We have chosen these tasks because we believe they are representative of a subset of environments that are important for real-life situations, but have been neglected in the literature. Many real-world decision making and problem solving situations are (1) *dynamic*, because early actions determine the environment in which subsequent decisions must be made, and features of the task environment may change independently of the solver's actions; (2) *time-dependent*, because decisions must be made at the correct moment in relation to environmental demands; and (3) *complex,* in the sense that most variables are not related to each other in a one-to-one manner. In these situations, the problem requires not one decision, but a long series, and these decisions are, in return, completely dependent on one another. For a task that is changing continuously, the same action can be definitive at moment *t1* and useless at moment *t2* (Brehmer, 1992; Edwards, 1962; Frensch & Funke, 1995a, 1995b). However, traditional, experimental problem solving research has focused largely on tasks such as anagrams, concept identification, puzzles, etc. that are not representative of the features described above. For some time now, researchers work on a set of computer-based, experimental tasks that are dynamic, time-dependent, and complex, called microworlds, and the area of thinking and reasoning that deals with them is called Complex Problem Solving (CPS, e.g., Frensch & Funke, 1995a). CPS has been plagued with methodological problems, and performance has been analyzed at a very shallow level (process measures were mostly discarded). The interest of this research paradigm, a hybrid between field studies and experimental ones, is tied to the success of methodological advances and theoretical progress. However, "despite 10 years of research in the area, there is neither a clearly formulated specific theory nor is there an agreement on how to proceed with respect to the research philosophy" (Funke, 1992, p. 23). The need for a theory that integrates explanations for these CPS tasks together with current knowledge representation approaches is patent. But any theory that tries to explain representation should be independent of the idiosyncrasies of the tasks, even when those are really complex. For that reason, we test the theory against human judgments in several different tasks. In this paper we report results from two very different complex, dynamic tasks: The microworld Firechief (Omodei & Wearing, 1993, 1995) and a high-fidelity flying simulator. In the Firechief experiment described here, participants saw replay videos of pairs of trials, and had to assign a value between 0 and 100 to capture how similar they considered the pair, and these values were correlate with LPSA measures. In the flying simulator situation, two instructors evaluated pilots' landings, one of them sitting in the copilot seat, and the other one watching plots of relevant variables in real time. The nearest neighbors in the LPSA space of any new landing were used to generate landing ratings. The ratings that the model emitted agreed with both humans as much as the two human graders agreed with each other. LPSA has also been tested by comparing predicted and actual future states of the thermodynamic system DURESS (Quesada, Kintsch, & Gomez, 2003; Quesada et al., submitted). In that situation, LPSA was able to explain the effects of amount of experience, amount of structure in the environment, and prediction of future states of the environment. We think this variety of tasks strengthen the argument in favor of LPSA. In every case, the tasks that people represent and compare fulfill the three criteria for CPS. The Firechief and flying Simulator tasks are similar in several aspects: (1) In both cases people have to make decision within the space of seconds, and the decisions are interdependent. A decision too late will not do any good, for example, sending firefighters to a burn-out area, or deciding to cancel the landing maneuver when the aircraft is too close to the ground. Any benefit derived from a decision decreases with the length of time it takes to make it. (2) Workload is not under the control of the problem solver. (3) There are exogenous events that make the system unpredictable; there might be a new fire starting any time, or an unexpected obstacle in the runway appearing suddenly. This multi-task testing approach has been used before. For example Joslyn and Hunt (1998) assumed that there is a common factor (skill) for rapid, dynamic decision making tasks, and tested the hypothesis using tasks as varied as air traffic control and public safety dispatching (911).

Also, the two studies reported here are complementary. In one, we use a controlled laboratory situation where the amount of practice is known, and equal for each participant, but reduced. In the other, we use a realistic simulation where the amount of practice per participant is unknown, probably different for different people, but very large since all participants were professionals. In Firechief, the LPSA model has more experience than any individual participant; in the flying simulator case, it is reasonable to assume that any professional has more experience than the 400 landings that we used to train the model.

A caution note about expertise is needed here. The results presented in this paper do not depend on the expertise level of our participants. Most of our participants would not fit in the definition of expert. In other words, this is not a paper about the acquisition of expertise. It is, however, a paper of how people represent tasks after they have experience with them. This is why we talk about “experienced” problem solving, not “expert” problem solving. For an LPSA explanation of expertise effects, see Quesada, Kintsch and Gomez (submitted).

The structure of the paper is as follows. We introduce the theory and relate it to previous proposals. Then, we present the two experiments and associated simulations to provide evidence for the theory. We conclude with a general discussion on the problems that LPSA solves and the questions that it raises.

LPSA: Representing Similarity Using Context

LPSA is inspired by Latent Semantic Analysis (LSA, Landauer & Dumais, 1997), a theory of representation that explains how meanings of words can be learned from the exposure to large amounts of experience. LPSA assumes that the mechanisms proposed by LSA to learn the meaning of words by its contextual usage might be more general than initially thought, and applicable to other types of cognition.

The best way to understand LPSA is to deal with LSA first. There are several papers in the literature that introduce LSA (e.g., Landauer & Dumais, 1997; Landauer, Foltz, & Laham, 1998), and the interested reader should consult these references for a better understanding of the technique. Recently Landauer (2002) has proposed a new formal way of understanding LSA. LSA is presented as a linear system of equations that is solved by Single Value Decomposition (SVD). From the theoretical point of view, the meaning of a passage is simply the sum of the meaning of its words, and this can be represented by the expression:

meaning of word$_1$ + meaning of word$_2$ + … + meaning of word$_n$ = meaning of passage.

The equation above represents one passage. If we have a corpus of text that contains, say, 10,000 passages, it can be represented by 10,000 linear equations like the one above, where the words would be variables that we want to solve. Since the passages will contain common words, there must necessarily be equations that constrain the value of the same variable. We can imagine the same situation for a corpus of 10,000 contexts, where each context is a set of stimuli. We can infer the "meaning" of the stimuli by the information that we have about the contexts in which they appear. For example, if we define "context" as a problem-solving episode (trial), and stimuli as every state that the system has traversed, we can derive the functional value of each state in a similar way that we derive the meaning of words: by solving the huge system of simultaneous linear equations. This is the approach that we have taken in LPSA. To quote Landauer (2002):

"(…) every passage of language that a learner observes to be an equation of this kind. Then a lifetime of language observation constitutes a very large system of simultaneous linear equations. This set of equations is certain to be highly 'ill-conditioned' in mathematical terminology, meaning that there will be too few equations to specify the value of many of the variables and some of the subsets of equations will imply different values for the same variable. As a model of natural language semantics, these deficiencies do not seem out of place; word and passage meanings are often vague or multiple. Mathematically, such complexities can be dealt with by abandoning the requirement of finding absolute values, settling for relations among the variables, and representing them in a richer manner than as real values on a number line (scalars). " (p. 49-50).

Instead of words, LPSA uses a different kind of type. Types are defined as all unique actions or states in a sample of human-system interaction known as the corpus. Each occurrence of a type is called a token. Instead of documents or passages of text, LPSA uses trials as context to define a matrix of types by trials, that is then decomposed using SVD and recomposed after dropping a significant number of dimensions.

The basic assumption is that the meaning of trials is a linear function of the meaning of the tokens contained in it and the influence of additional factors that can be encapsulated into the construct of a context. This basic equation is written as (after Landauer, 2002):

$$m(\text{trial}) = f\{m(sa_1), m(sa_2), \ldots.. m(sa_n), \text{context}\}$$

Where *m* is the meaning of a trial, expressed as a function of the meaning of tokens (*states* or *actions*).

LPSA adopts several simplifying assumptions to be able to generate a computable representation. LPSA assumes that the function is linear. Any context information that is not defined by the tokens themselves is represented by c_k. To simplify, the effect of c is considered to be negligible. A passage is a set of actions or states of any length, and a trial is a passage of length determined by the duration of the trial. Any passage representation can be calculated as the addition of the representations for each of the states or actions that form it.

The resulting system of equations for *k* trials would be:

$m(trial_1) = m(sa_{11}) + m(sa_{21}) +\ldots.. + m(sa_{n1}) + c_1$

$m(trial_2) = m(sa_{12}+ m(sa_{22}) +\ldots.. + m(sa_{n2}) + c_2$

$m(trial_k) = m(sa_{1k}) + m(sa_{2k}) +\ldots.. + m(sa_{nk}) + c_k$

The number of dimensions used is a free parameter that normally ranges between 100 and 500; this number is in agreement with the dimensionality observed in most natural languages. The resulting space can be considered a formal equivalent to the representation that people create when they interact with the task for an equivalent period of time. Since the corpus may contain performance from many different individuals, the resulting space is an abstraction over subjects; that is, it does not represent a particular person, but more of an ‘average’ participant on a task.

The singular value decomposition (SVD) is the technique of choice to approximate a solution for such a big system of linear equations. The linearity is an assumption of the method: non-linear systems would be much more difficult (even impossible) to solve[2]. The following two sections will concentrate on how LPSA spaces are created and tested against human data. They represent two very different domains (one laboratory task, and one naturalistic task) so the formalism is tested under very different conditions.

Predicting Similarity Judgments

In this section we use the microworld Firechief. The reasons for selecting this task are many. (1) It is dynamic, in the sense that the task changes with time (the fire advances) and as a consequence of the participant interventions. (2) There are factors that are not under the participant's control, for example changes in the direction of the wind. (3) There is time pressure in the decision-making process. (4) The task can be repeated systematically, and it is short enough to enable participants to complete 20-25 trials in two experimental sessions, and (5) there exists an available corpus of experience on Firechief. We describe the experimental task, comment the materials used to create a representative corpus, the corpus creation procedure, an experiment to gather human similarity judgments and the results of LPSA modeling those similarity judgments.

Corpus compilation and Space creation

Apparatus

In Firechief (Omodei & Wearing, 1993, 1995), participants are confronted with a simulation of a forest through which a fire is spreading. Their task is to extinguish the fire as soon as possible. In order to do so, they can use helicopters and trucks (each one with particular characteristics), which can be controlled by mouse movements and key presses. The different cells (see Fig. 1) have different ratings of flammability and value: houses are more valuable than tree cells, for example. The participant's mission is to save as much forest as possible, to preserve the most valuable cells, and to prevent the trucks from being burned. Helicopters move faster and drop more water than trucks, but the latter can make control fires. Trucks are unable to extinguish certain fires, and they will be destroyed if they are sent to them. The fire is more intense and spreads faster depending on the wind direction. Wind direction and strength are indicated in the top-right corner of the interface. Participants can see a window with their overall performance score at the end of a trial, which is calculated by adding every safe cell and subtracting the value of the burnt trucks. At the same time, it is possible to experimentally control features of the system, and to prepare experimental situations for testing a wide variety of hypotheses.

There are three commands that can be used to control the movement and functions of the appliances: (1) drop water (2) start a control fire (trucks only), and (3) move an appliance. Commands are given using a 'drag-and-drop' philosophy. At the end of each trial, the program saves the command sequence that the participant issued in that trial. These logs files were used to train LPSA. An example of those log files (only the fist 8 actions; a normal trial contains an average of 103 actions) is provided in Table 1. To create an LPSA space, data from experiments 1 and 2 described in Quesada et al. (2000) were used as the corpus, plus data from the experiment described in Cañas et al. (2003). The experiments were designed to test hypotheses about cognitive flexibility when facing new, changing conditions after training under constant conditions in the dynamic task Firechief. Thus, the experiments consisted of a long period of constant environmental conditions followed by a short period of variable conditions. The conditions manipulated were wind direction and fire extinguishing efficiency of the appliances. All trials were equal in terrain, number & type of units, pace, initial fire location and all other variables that determine a Firechief scenario. Since here we are interested in a sample of representative behavior when interacting with the system, the actual hypothesis and design of the former experiments are not relevant.

Corpus preprocessing

By putting together all the experience of all the participants in the experiments used in Quesada et al. and Cañas et al., we created a text corpus of 3441[3] log files. It is important to note that LPSA uses experience from more than one participant, being representative of one 'average' human. This is needed because it is very rare to find datasets where a single participant has been studied for long periods of time in the task. However, using these data when available, LPSA can be employed to model individual behavior.

Actions were coded by joining the information contained in one line of the log files. For example, the first line in Table 1 would be transformed into:

Move_44_100.00_4_Copter_11_4_12_9_Forest

using underscores to join all the variables into a single token. This token, that could be an action or a state of the system depending on what is the preferred option in the logging, is the basic unit of analysis in LPSA. It corresponds to the selection of 'words' in classical LSA.

It is important to note that this procedure does not necessarily require the experimenter to select part of the information. However this is sometimes useful. Since LPSA must be trained on a set of variables that should resemble closely the information that participants use to perform in the task, the experimenter's intuition can be used to aid in the removal of information that participants most probably do not use. In this case we dropped the information on (1) appliance number and (2) departure coordinates from the logs. This decision is based on findings in spatial attention when applied to multiple moving targets (e.g., Pylyshyn, 1994, 2001; Scholl, Pylyshyn, & Feldman, 2001). In these experiments, participants had to keep track of up to four items (targets) that moved randomly in a field of eight or more identical items (distractors). After all the items stopped moving, participants point out which ones were the targets. The assumption was that observers who had tracked the targets correctly also had kept track of their individual identities. Thus they should be able not only to identify the objects as members of the target set, but also to recall other identifying information initially associated with them such as names, colors or place of origin. This experimental situation very much resembles the one that our participants experienced in the human judgments experiment. Pylyshyn found that people can track the items, but not recall their identities; in our case, people should not be able to say whether one truck is truck 1, originally starting in cell (11, 9), or truck 2 which started in cell (4, 11). These variables were removed from the log files that LPSA used as input. For example, the first line in Table 1 would translate into Move_Copter_12_9_Forest. Thus, instead of 360199 actions in 3441 trials, the reduced corpus had 3627 different actions in 3441 trials.

After performing singular value decomposition on the actions by trials matrix, we kept 319 dimensions. Altering the number of dimensions in the range that LSA normally uses (100 – 1500) did not change the results significantly, so we will report the results for 319 dimensions only. Before we present the evidence for the theory, we want to illustrate in a graphical way what kinds of situations are considered similar by LPSA, and how these are visually and conceptually similar to the bare eye. Three example situations are shown in Fig. 2. They correspond to the first 8 actions of three trials that have been selected so that according to LPSA, two are very similar to each other, and the third one is dissimilar to both. The graphs show that this pattern is also evident to human intuition. Every trial starts with a fire in the center of the screen, and the wind is blowing to the east, where the houses are. The vectors indicate the appliance moves, the squares are protective fires (control fires) and the circles are ‘drop water’ commands. Examples 1 and 2 protect similar areas, near the houses, even though they do not share but one out of eight actions. The similarities for 1 and 3, and 2 and 3 are low because example 3 protects a far-away corner (upper-left), where the fire will not go with the current wind direction. The LPSA cosines for those pairs are 0.72, 0.05, and 0.07 respectively, capturing human intuition.

Correlations between LPSA and Human Judgments

If LPSA captures similarity between complex problem-solving performances in a meaningful way, then human similarity judgments on a sample of representative trials could be used as a validation. The problem is that, contrary to what happens when one uses LSA to model text comprehension, it is not easy to find experienced humans in the task at hand. Most normally educated adults are expert readers, but not everybody is an expert in controlling the particular dynamic system called Firechief. To test our assertions about LPSA, we recruited a group of participants and exposed them to the same amount of practice as the participants used to create the corpus, so they could learn the constraints of the task. After that, they were asked to make similarity judgments on a set of representative Firechief trials, and their judgments were compared to the LPSA cosines for those trials.

Method

Participants: 14 University of Colorado undergraduates participated in this study as part of a class requirement.

Procedure: After 24 practice trials, these participants were used to assess the external validity of LPSA similarities. We used an interface that offered Firechief trial videos and participants had to watch seven pairs of trials (at a pace eight times faster than normal) and express similarity judgments about these pairs. They were asked to express numerically (1-100) how similar each pair of trials was. The appendix presents the instructions.

Participants watched a randomly ordered series of trials, in a different order for each participant. The trials were paired and selected sampled uniformly from the distribution of cosines (pairs A, B, C, D, E, F, with cosines 0.75, 0.90, 0.53, 0.60, 0.12 and 0.06 respectively[4]).. That is, the cosines selected are evenly distributed in the continuum of similarity (0-1) that LPSA offers. This stimulus set is small. We asked participants to watch only six pairs of trials due to two main reasons: (1) so they could replay several times every pair if they wanted to fine-tune their judgment. Our pilot studies showed that people tend to assign initial numbers in a first pass, and fine-tune the similarities afterwards by replaying the pairs that are important for their decision. (2) The time necessary to perform this task increases exponentially with the number of items to classify. During the replay, participants have to pay attention and find commonalities and differences. The task is very demanding and tiring, and most people spent more than half an hour to finish it. We found six pairs to be a practical maximum that could be asked without deteriorating judgment performance.

Apparatus: Participants controlled a computer program in which they could click a button to watch any video replay of any trial. Buttons were distributed in pairs and near each pair there was a text box where they were asked to fill a similarity value from 0 to 100. The buttons changed their ordering randomly with each participant, to control for order of presentation effects. When a button was clicked, a full screen video was played at a speed eight times faster than the standard one. The faster pace of presentation helped participants to review a particular trial several times if needed, without using up too much time. The pace was still appropriate to understand the intentions of the author of the trial, and no participant expressed any difficulty with the speed of the replays. They were allowed to replay trials as many times as they wished, and change the similarity judgments to reflect new insights. The instructions were present on the screen at all times.

Results

People's judgment ability. To calculate inter-judge agreement, the Intra Class (ICC) Correlation coefficient was selected (for a recent review, see McGraw & Wong, 1996; for a classic reference, see Shorut & Fleiss, 1979). To measure the relation of variables representing different measurement classes, the most commonly interclass correlation coefficient is the Pearson r. However, when the variables of interest are from a common class (i.e., share their metric and their variance), intraclass correlation coefficients are a better alternative (McGraw & Wong, 1996). That is the case in judge-agreement situations such as the one that occupies us. We can write our ICC coefficient as ICC(A, 14) = .9091, highly significant, F(5,65) = 11.00, p<.001[3]. The high value indicates that the average of the human ratings has very high internal consistency.

Model prediction of human similarity judgments. LPSA cosines predicted human similarity judgments very well indeed. To see how well LPSA cosines mimicked human judgments we averaged the human ratings and correlated[4] this (reliable) composite measure with the LPSA cosines for the six items. This correlation is extremely high (r = .949, df = 4, p<.001). This value indicates a very good agreement between LPSA and the average human rating. Fig. 3 shows the relationship between observed and predicted average judgments. Although the overall fit is very good, there is an important difference: participants tend to be reluctant to rate items as "very dissimilar" or "low similarity", as the intercept of 35.39 shows. That is, where LPSA would predict a similarity of zero, a human participant is expected to say 35.39. This is, *per se*, an interesting result.

Discussion

As far as we know, no other model of representation during complex problem solving has been validated against human judgments of similarity after watching replays of trials. The results presented are encouraging in the sense that they mimic human intuitions with a very simple formalism. LPSA can explain a good amount of variance without proposing elaborated constructs such as mental models or structured representations of causal dependencies in the system observed.

However, the task that we used is a laboratory one and the results may be affected by the idiosyncrasies of the experimental situation. For example, giving a number between 0 and 100 to express similarity may not be seen as natural as performing some other daily-life task that implies similarity judgments, for example grading performance. Having real professionals instead of psychology undergraduates as participants will also help to clarify if LPSA is accuracy capturing human judgments is kept when the amount of experience and complexity of the system is higher. In the second study, we will use a more realistic task with human raters evaluating 100% naturalistic situations: plane landings.

Predicting Landing Performance Judgments

We would like to use LPSA applied to a real world task to emphasize the generalizability and power of the theory. A second objective is to present a new technique of knowledge elicitation that we employed, called rater-model triangulation, designed to work in complex environments. We use two raters to select the relevant variables to model: one has complete-information about every aspect of the system (complete-information rater) and the other has to choose a small set of variables to do the task (reduced-information rater). We use in our model the variables selected by the reduced-information rater, and try to optimize our model so it correlates maximally with both raters.

There is currently no methodology to automatically assess landing technique in a commercial aircraft or a flying simulator. Instructors are a significant cost for training and evaluation of pilots, and the use of instructors also incorporates a subjective component that may vary from pilot to pilot. An automatic method of evaluating landing performance would be very advantageous, but the complexity of the landing task has discouraged researchers and modelers.

Selecting the set of variables that should be used to train the model is not a trivial task. Is the visual information to be considered? In which way? Which variables are relevant? Our approach was to develop a methodology based on two key ideas: (1) Rater-model triangulation: while an rater was able to monitor almost every single variable relevant (cockpit rater), another one was limited to watching a real-time plot of a very limited set of variables chosen by him (reduced-information rater). If the judgments of these two rater are highly correlated, the variables selected by the reduced-information rater have sufficient explanatory power to perform the evaluation. (2) Modeling of the landing evaluation task using LPSA with the variables selected by the reduced-information rater. The resulting system was able to evaluate landing performance automatically.

This second section of the paper describes an experiment conducted to implement these ideas. The purpose of this experiment was twofold: First, we would like to expand the area of application of LPSA, and connect the results to previous theories knowledge representation. Second, we would like to introduce a system that can assess landing technique. The applied value of the system is in replacing or complementing the instructor needed to evaluate landings, plus increasing the objectivity of the judgments.

We work under the assumption that a rater in this situation uses her past knowledge to emit landing ratings by comparing the current situation to the past ones, and generates an expanded representation of the environment by composing the past situations that are most similar to the current one. As pointed by Landauer (unpublished), most people use conscious logic only to narrow realms where they also possess large volumes of hidden intuitive knowledge. Experts are supposed to be attuned to the constraints of their environments (e.g., Ericsson & Lehmann, 1996; Vicente & Wang, 1998) in a way that presumes automaticity and not necessarily conscious processing. Our proposal does not deny that people also employ some other more analytical method. However, we want to explore how well a memory-based model could do in the absence of analytic, logical processes.

The landing task

The variability in the requirements of the landing task is immense with landing conditions such as wind, gust, and visibility. However, our data collection experiment was designed to be very simple with the idea of minimizing the variability due to uncontrolled factors.

The manufacturer of the aircraft normally provides charts with the preferred value of a variable (e.g., Glideslope) given some possible values of other variables (e.g., the air speed). In other cases, it is the Government who provides the charts. That way, practically all known situations are covered and the pilots normally look up the recommended values or they just know them by heart after flying the same aircraft several times.

The landing is usually divided into approach, flare, touchdown and after-landing roll. A graphical, simplified description of these four concepts is shown in Fig. 4.

To evaluate the landing technique, we selected a set of five criteria consulting several landing technique instructors and simulator specialists. The list included: (1) Flare initiation height: The flare has to be initiated at a particular height; this height is not rigid as lower flares can be compensated by a higher pitch rate for example. (2) Thrust Reduction: The reduction should be progressive, and does not last too long. It has to be started in a particular moment in time. (3) Pitch rate. The pitch was evaluated using five discrete levels, from too high to too low. (4) Overall landing score. This is a general rating that expressed how good the landing was, from one to five. In a sense, it is not a summary of the former measures, as it adds new information. Some landings can have, for example, an incorrect Flare initiation height, but end up getting a five out of five, because it was compensated with other means. The possible ways in which these different grades can be observed and their interaction enables a complex set of data to model.

The problem of variable selection and complexity reduction

In some circumstances, the modeler has to be very knowledgeable about the task to be able to create a successful model (for example, chess modelers tend to be good chess players). Although this point may seem redundant and obvious, it is very important, since it is not always the case that the modeler can invest the long time required to master the task to model. The alternative approach to task modeling is to ask the experts what information they use, and what procedures they have developed to perform the task. In this line, expert knowledge elicitation techniques have been developed to try to ‘extract’ the knowledge from human experts and ‘insert’ it into the system. Thus, many expert systems are rule-based systems. The approach that we have taken here is different. A basic idea is that people are able to confront complex tasks because they have managed to reduce the dimensionality of the respective problem spaces of their jobs. There is a need to translate this dimensionality reduction to the system that is going to perform in their same environment.

Two reductions in the dimensionality of the task are performed to represent the variability in the environment in an efficient way: (1) Selection of variables suggested by the rater-model triangulation methodology, and (2) the one performed by LPSA's SVD when the lower-dimensional space is created. They are explained in the following two sections.

The rater-model triangulation

In this section, we present a possible solution to the problem of variable selection. It uses a configuration of two raters, who perform the task in two very different conditions.

The question is: How do we know which variables a model should pay attention to? It is hard to imagine that the information-processing system keeps track of every dimension that could possibly be registered. For example, a high-fidelity flying simulator can log up to 10000 variables, each with a precision (sampling ratio) of 1/100 seconds. Since it is recorded in the log files, we can assume that this amount of information is available to the pilot and copilot in the commercial aircraft simulated. Of course, in a particular temporal moment *t*, the human components of the system (pilots and ATCs), are aware of a very small proportion of these variables, and the focus of attention is changing from *t*1 to *t*2… to *t*n. It is computationally unfeasible for a cognitive system (either human or artificial) to work in such a high dimensional space.

As a step towards solving this problem, we present the rater-model triangulation method. It is very simple and susceptible of being applied on a variety of areas. The basic idea is that if we cannot model an domain because of its complexity we can use two raters with different access to the information available to discriminate the importance of each variable in the task they perform. A first approach, quite used in modeling work, is the effort to model directly the expert behavior using as many as possible of the variables he can access in his normal, daily performance. Let us call this person ‘the complete-information rater’. However, when the task is complex, trying to model the whole situation often proves itself to be an excessively difficult task. Some theories do propose ways of selecting the relevant parts to model. This selection is a priori, that is, the assertion ‘The person is using variable X but not Y’ is part of the theory. What we propose is to use a second person to do the variable selection in a non-theory-driven way. The second rater will have limited access to the variables in the system (for example, he can only plot a limited number). For that reason, this second rater is called ‘the reduced-information rater’, and is forced to select a small set of variables. The model will be created to reproduce the behavior of this rater, and this is often a key step since the modeling task can change from being intractable to being tractable. Note that the theory does not have a priori assumptions about what are the task’s most important variables: the rater does.

A schema of the rater-model triangulation method is presented in Fig 5.

Construction of the reduced-dimensional space

The second way of reducing the complexity of the problem is performing dimension reduction. The dimension-reduction step and its properties to explain learning and generalization are important in several cognitive theories (e.g., Edelman & Intrator, 1997; Rumelhart, Smolensky, McClelland, & Hinton, 1986). The algorithm used in LPSA is the Singular Value Decomposition (SVD) of the frequency matrix of states by landings, and the reduction in the number of singular values. As a result, we obtain a representation of both states and landings in the same space. Any new landing that is not in the space can be represented as a linear combination of the vector of its states. We can predict the ratings of any new landing by averaging the ratings of the *k* known nearest neighbors of the vector representing the landing in this space. To construct the space we used the variables that the reduced-information rater was using, as suggested by the triangulation technique.

Method

10 pilots performed 40 landings each. We manipulated wind direction and intensity at 6 levels, using incomplete counterbalanced design to control for order effects. The levels selected were 30, 20, 10, 0, and -10 (tail wind) knots.

The two raters used a set of criteria to rate the landings, described in ‘the landing task’ section. Both raters rated all landings. The reduced-information rater was allowed to select and plot as many variables as he wanted, with the limitation that they should fit in his 20” computer screen. He plotted only the following five variables: Vertical acceleration, Radio altitude, Pitch rate, Rate of descent, Pitch angle. There was still some space left, which implies that the rater considered that he did not need to plot any more variables. It turned out that rate of descent and pitch angle were barely used, as they were never referred to when the expert explained his ratings to the experimenters. Thus, rate of descent and pitch angle were omitted from the analysis. The rater in the copilot seat (complete-information rater) used a huge array of information, since he was exposed to the same environment as the pilot, being able to see the runway approach and feel the movements of the aircraft when the wheels touched the ground, for example. The basic idea was to calculate the agreement between the two human graders. If the agreement was very low, the judgments are too subjective, and a possible automated method of assessing landing technique is hard to validate. Then, the same agreement would be calculated for each human expert and LPSA.

To do the model selection part where we tried to find the right parameter set, the criterion used was the average correlation between the model and the human ratings of both human graders.

Corpus creation

The states in each landing were stripped off of all the variables except for the reduced information that the rater was actually using: flare initiation height, thrust reduction and pitch rate. Since the average duration of a landing when the starting point is 500 feet was about 15 seconds, and the sampling ratio was 10 samples a second, the average number of states per landing was 150.

The flare initiation height, expressed as feet, was transformed (rounded) to be multiples of ten (e.g., 112 feet would be 110, 89 would be 90, etc) and the vertical acceleration and thrust reduction were rounded to the nearest integer (e.g., a vertical acceleration of -9.8 would be -10, and a thrust value of 3.2 would be 3). This rounding is necessary because LPSA assumes that a landing is a sequence of states, and the continuous flow of these values has to be discretized. Since decimal values are not relevant, and humans would consider that, for example, an altitude of 45 feet is the same as 46 or 47 feet for most purposes, we applied the rounding in our model.

The original sampling ratio was 10 times a second. That made a total number of 75571 unique states in 400 landings. Although LSA has been applied to text corpora with the same number of types, and even several orders of magnitude more, the limited number of landings imposes a severe restriction. Most known learning mechanisms, including LSA, need several repetitions of the units to learn them. That is, LPSA learns better when a good proportion of the states can be found in more than one context. The transformations and rounding that we performed were serving the purpose of decreasing the number of different states in the corpus, and rendered a total of 569 unique states. When the states are described using continuous variables, and these variables are sampled at a fast rate, a non-rounded corpus would have as many unique states as the total number of states (that is, each state would appear in the corpus only once, leaving little room for learning). The variables were joined with underscores to make them a single token, and use space as token separator. This way, a state in the system was represented as a token as follows: "flare initiation height_thrust Reduction_pitch rate". This token is the equivalent to a word in standard LSA. The matrix of states by landings was created, and an SVD was performed on it. After the decomposition, the biggest N (where N is a free parameter) dimensions were kept. The parameter manipulation is explained in the results section (model selection). A web interface to the 400 landings graphs (mimicking the reduced-information rater's display), experimental conditions and ratings used in this paper can be visited at http://lsa.colorado.edu/ ~quesadaj/adriVisor.cgi. The complete corpus is also available upon request.

Apparatus

The Netherlands' National Aerospace Laboratory (NLR) National Simulation Facility (NSF) simulator was used. A Boeing 747 cockpit was installed consisting of a side-by-side full glass airliner cockpit with a layout, equipped with six programmable CRTs. The facility featured a four-degrees-of-freedom platform.

Participants

10 commercial pilots were recruited to land the simulator, and 10 other commercial pilots and instructors were hired to act as the 'complete-information rater'. The ideal situation would be to have a unique 'complete-information rater' rating all the landings, as we had for the 'reduced-information rater'. However, this arrangement was not possible, so from now on we

will refer to the ‘complete-information rater’ as one single person even though this role was played by 10 different instructors. Of those 10 instructors, most of them had experience as B747 instructors, and some had publications on the topic. Since it was very difficult to find such a large number of instructors, two NLR professional pilots (highly experienced on the simulator) doubled as instructors in two of the evaluations.

Design

We paired randomly instructors and pilots who were going to perform the landings. The reduced-information rater was situated in a control room, with no direct information about the landing taking place in the simulator other than the plots of the variables he selected beforehand.

Since wind conditions influence the landing procedure, the ideal experimental design would be to select a representative sample of all the possible wind directions during landing. We selected a very restrictive set of wind conditions, and manipulated wind direction and intensity at 5 levels, using incomplete counterbalanced design to control for order effects. The wind levels selected were 30, 20, 10, 0, and -10 (tail wind) knots. That is, all conditions had head wind, but one. The wind condition was announced before the trial started. As a side note, we were interested in how performance changes with "mental set" (repetition trials vs. change trials), so pilots experienced different randomized sequences that contained both repeated and shift trials, (e.g., 30, 30, 20, 20, 10, 10, 0, 0, -10, -10). That gave us a total of 5 different sequences, so each sequence was experienced by two pilots. This way, pilots experienced five different wind conditions twice for each block in an alternate pattern of two repetitions.

Each pilot performed 4 blocks of 10 trials. The 4 blocks contained the exact same sequence of wind conditions. Overall, each pilot experienced each wind condition 8 times.

For each landing, data recording started when the aircraft reached 500ft, and stopped when the three wheels were touching the ground. They performed 10 landings in an hour, plus a little rest between sections of about 15 minutes. They walked out of the simulator and were engaged in some unrelated activities during their rest.

Results

Significance tests

The polychoric correlation was selected because of its suitability for analyzing judgment data on ordinal scale. To test the hypothesis of the correlation being significantly different from zero, we used resampling methods, concretely a randomization test. That is, we used a Monte Carlo approach to estimate the probability of our results (correlations) being obtained due to a bias in the computation. For example, imagine that both the rater and the model say simply 'correct' all the time. The bias is 'say always correct'. The correlation human-model would be 1. As well, if we randomly rearrange the values of the model or the rater, so that they do not line up with each other (for example, the model rating for landing 1 would be matched to the rater judgment for landing 67, and so on), the correlation would still be one. In this extreme case of bias, having a high correlation between the model and the rater does not mean any merit for the model, since any random rearrangement of the data would obtain the same correlation. The randomization tests performed were conducted resampling 500 times.

Model selection

We created several corpora modifying the number of dimensions (100, 150, 200, 250, 300, 350, and maximum dimensionality, 400) and the number of nearest neighbors used to estimate the landing ratings (from 1 to 10). Another manipulation was the inclusion or exclusion of a time tag, and the type of weighting scheme used (log entropy vs. none). This way, the possible combinations of levels were (7 x 10 x 2 x 2) = 240. For each of these combinations of levels, we used leave-one-out to calculate the ratings for the landing excluded. The estimated ratings for each of the 400 landings were then correlated with the real ratings. The combination of levels that best correlated with both humans was selected, and that was: Corpus with 200 dimensions, 5 Nearest Neighbors, no weighting, no time tagging).

Model fitness

The first thing to observe is that the average agreement between human raters was not very high (polychoric correlation .48, see boxed bars in Fig 6). To our knowledge, there are no studies that report statistics on specifically landing technique raters, so we will use general expertise for comparison. Shanteau (2001, p. 237, table 13.2), presents data on consensus (agreement) between experts in different domains. The landing technique raters reported in this study had an inter-rater reliability better than Clinical Psychologists (.40), Stockbrokers (<.32), polygraphers (.32). Their agreement is in line with Livestock Judges (.50), Pathologists (.55), and Grain Inspectors (.60). However, it is lower than the ones reported for Weather forecasters (.95), and, Auditors (.76). Since we do not have any objective measure (no objective landing score exists) to justify that our raters are actually reliable, we should not appeal to their expertise level to validate the model. As a matter of fact, we do not need them to be highly-reliable experts at doing this task. Our argument for LPSA stands as strong as before if we consider this a sample of human behavior in a complex situation (landing evaluation), without starting a discussion about their degree of expertise.

The average correlation between the model, and the reduced-information rater was about the same as the correlation between the two humans (.48 vs. .46, boxed bars in Fig. 6). Note that the ceiling for the model is the correlation between two humans doing the task; a model that correlates with one human better than two humans correlate with each other is under suspicion. It seems that the judgment on thrust reduction is particularly difficult for the two raters to agree (human-human correlation of only .27).

One of the LPSA assumptions is that experienced humans perform dimension reduction to represent their environments. The equivalent model (5 nearest neighbors, no weighting, no time stamping) without performing dimension reduction (that is, using 400 dimensions, which is the shortest dimension of the matrix) correlates with humans (on average for all criteria) only .26, which can be interpreted as evidence for dimension reduction in the representation.

All of the agreements between human judges were highly significant: Flare initiation height (.52, p = .002[7]), thrust reduction (0.27, p .002), pitch rate (.46, p= .002) and overall landing performance (.61, p = .002).

The equivalent model without dimensionality reduction (400 dimensions, 5 neighbors, no weighting, no timestamp) produced .37, .08, .57, .50 correlations for the above used criteria respectively.

In our design, we tried to mimic the reduced-information rater (the one that had access to only a few selected variables) since the model used the variables this rater utilized. However, having a good correlation with the complete-information rater (located in the copilot seat) is desirable too, so in the process of finding the right parameters, the models were selected for their correlation with both humans. Fig. 6 presents the correlations obtained for the complete-information rater. Note that the only criterion where the model correlates with any of the raters more than they correlate to each other is thrust reduction. Thrust reduction seems to be a very difficult feature to judge, since the agreement between humans is the lowest (.27) and also it is the one in which the reduced-information rater obtains the lowest test-retest reliability (0.538, see test-retest measures).

All the polychoric correlations between the reduced-information rater and the model were significant (p = .002). So were the correlations between complete-information rater and model.

Test –retest measures

One common method to assess how reliable human raters are is the test-retest correlation. It simply consists in having the same rater grade twice the same item in two different temporal moments, preferably distant in time. It is well known that humans have imperfect test-retest reliability. In our study, we asked the reduced-information rater to reevaluate a random sample of 100 plots displayed in the same way he experienced during the experiment. The plot contained wind information, but all other information that could identify the landing (pilot name, landing number, ratings etc.) was removed from the graph. The reassessment took place about one 8 months after the end of the experiment. The reliabilities were .64, .53, .84, and .72 for flare, thrust, pitch and overall score respectively.

To our knowledge, there are no studies that report statistics on specifically landing technique raters, so we will use other domains of expertise to figure out how our reduced-information rater stands. The average test-retest reliability (0.69) is better than some other studies of reliability of expert judgments reviewed in Shanteau (2001, p. 237, table 13.3), concretely better than for Clinical Psychologists (.41), Stockbrokers (<.40), Grain Inspectors (.62) and Pathologists (.50). His test-retest reliability is however lower than the one reported in the same work for Weather forecasters (.98), Livestock Judges (.96), Auditors (.90) and polygraphers (.91). It is worth noticing that a computational model such as LPSA has a test-retest reliability of 1, and that could be viewed by the trainees as a good feature.

Application of the model in a non-structured corpus

A cognitive system (human or machine) exposed to expert-level amounts of experience in a non-structured environment will show a very poor performance, similar to those of novices. Product theories of expertise (e.g., Vicente & Wang, 1998) propose that the amount of environment structure is the main explanatory factor for the expertise advantage, and LPSA should be able to reflect this fact. To test this hypothesis we ran exactly the same simulations on an artificial corpus with 400 landings where the states for each landing were randomly sampled from the original corpus. This random corpus contained landings where all the variables changed randomly for the (average) 15 seconds that a landing lasts. In the hypothetical case of having a human exposed to a domain similar to such a non-structured environment, the amount of learning obtained by the human after the long-term experience would be very little. This poor learning would be reflected by a poor ability to predict future states, and the landing rating case, a poor rating skill, and the LPSA model reflects that in Fig. 7.

Discussion

The evaluation of landing technique is a complex task. It takes several years to learn the basics to be able to land a plane, and even longer to be able to evaluate the quality of a landing in a consistent way and give advice on how to improve it. LPSA requires a lot of experience before it can do this retrieval-based rating, as do humans. An important criticism can be raised in that we are not giving the model the same amount of practice as the rater has. We have presented data that have been generated using only 400 landings in a very limited set of environmental conditions (6 different wind strengths, no changes in direction) on only one runway. Ideally, the system would be trained with the particular circumstances of relevance (several runways, wind conditions, aircraft types, etc.). We do not want to argue that the current data and results presented are a complete model of landing technique evaluation, or that it can substitute instructors in their task. It must be demonstrated that the model as it is developed here can render similar performance in a wider set of conditions. However, there are reasons to believe that, in practice, the system will scale up well. LPSA has been applied to corpora far larger than the one used here. For example, in Quesada, Kintsch and Gomez (2003) the corpus used contained the equivalent of three years of daily practice. When the same ideas on knowledge representation are applied to semantics and text comprehension, the corpus used represents the exposure to printed text that an average human may have by the time she reaches college level, and this is several years of practice.

Another concern that could be pointed out is the nature of the dataset. In the expert systems literature, the systems are often trained with cases that lie in the borders of categories – cases that are difficult to classify. It is assumed that the generalization to easy cases would be straightforward. In our experiment, all the landings could be considered easy, because that is what the model would be doing most of the time in standard training and evaluation conditions. If we could gather a bigger corpus of landing performance, we think that a key factor is to keep it representative of the natural distribution of landings, that is, easy landings should be sampled more often than difficult ones, as they are more often experienced by the pilot and instructor we model.

The rater-model triangulation, although not an integral part of LPSA, is an additional theoretical contribution. It is a simple methodology that can be applied in a variety of expertise domains, helping modelers to generate computational theories with a minimum of a-priori, ill-motivated assumptions (other than the ones that the reduced-information rater has in his domain of expertise). We believe that this methodological step could help the development of models of really complex tasks. Most real-world tasks are so complex that the experimenter simply cannot invest the long time required to master the task to model. Using the triangulation technique, experimenters do not necessarily have to be knowledgeable in the task modeled. In this sense the rater-model triangulation could expand the range of psychological tasks that a modeler can afford to tackle.

General Discussion and Conclusions

LPSA learns to create a representation of the constraints of a complex environment by putting them together in a system of simultaneous equations. This system is solved by a mechanism involving dimension reduction, and the result of it is a multidimensional space where every event and context is represented as a vector. We propose that the creation of this space is what characterizes learning in complex, dynamic tasks.

Problem solving, mental models, and reasoning are the explanatory concepts employed in cognitive science to account for performance in complex task. LPSA shows that simple ideas such as similarity-based processing and pattern matching could have a role even in cognitively complex tasks. LPSA is a simple computational model based on the analysis of massive amounts of knowledge. It assumes that representation takes place in a space that has fewer dimensions than the external (distal stimuli) space represented. It also assumes that humans retrieve the most similar past experiences to the current one automatically. The response to the current situation (for example, in grading a landing) occurs partially because the ratings of past landing which are similar to this one "come to mind", and the response is a composite of those ratings.

LPSA and the concept problem space

LPSA is a new way of representing a problem space. Moreover, the theory is explicit about how an agent could proceed to *generate the representations* from a corpus of experience.

Newell and Simon posed an interesting question: How can we find an objective representation of the task, or at least, one that most participants share? They proposed that a different problem space could be constructed for two tasks that were formally equivalent. The idea of generating an objective representation, something that is shared by any participant, was considered impossible. In their example they compared two tasks - *number scrabble*[8] and *tic-tac-toe* - where the first was represented as strings of numbers, and the second as the typical grid containing Os and Xs. It is not clear which representation should be chosen to represent the problem space. A player could be using either representation, or some other one when playing either game.

They proposed two solutions to this problem: (1) to omit the description of an objective problem space in the theory, and (2) to construct a hypothetical problem space that is objective only in the sense that "all of the representations that human subjects in fact use can be embedded on as specializations in this larger space" (Newell & Simon, 1972, p.64). This second option is what they followed in their research.

LPSA proposes a different solution. The answer is related to the solution that researchers such as Landauer (2002) and Edelman (1999) have proposed to the classic epistemological problem of the commonality in representation. The problem has often been illustrated with the following question: "What is common in minds – or the nervous systems - of two different persons who see a cat on a mat?" This is a deep problem of representation, since most theories do not explain how there is a common representation in two minds, or how the representation has been generated. The "Landauer-Edelman" answer to this problem for semantics and object recognition respectively is that the two minds are exposed to huge amounts of experience coming from the same environment, and statistically even if the samples of experiences are only partially shared, the representations of the environment in two minds tend to be similar.

Provided that (1) the mechanism used in the two observers is the same (whatever the SVD and dimension reduction approximate in LSA and LPSA), (2) their experience (size of the sample) is extensive, and (3) the environment has a determined structure, the two representations in the heads of the two subjects must be approximately the same. In the field of perception and object recognition, the Chorus of prototypes model (Edelman, 1999) has this same propriety, and we expect that LPSA models of huge domains trained with ample experience may exhibit it too. In that sense, LPSA proposes that an objective problem space can be determined. It is, of course, true that each person that masters a domain (e.g., pilots, nuclear power plant operators, or managers) has a different representation of it. But it is also the case that, if the three conditions described above are true for a particular domain and pair of subjects, their representations of the domain must be very similar. These representations can be captured and de-individualized by a model that does something like LPSA does, and that formalization can be considered the closest thing to an objective problem-solving representation that we can propose.

Even though similarity plays a fundamental role in theories of knowledge representation in psychology, the symbolic approach used in Newell and Simon (1972) and perpetuated in the rule-based unified theories of cognition (e.g., Anderson & Lebiere, 1998; Newell, 1990) does not represent similarity between symbols as a structural characteristic. For example, one of the problem-solving tasks that Newell and Simon used, Cryptarithmetic, has states defined by letters and numbers. For this task, they explicitly said that there is absolutely no representation of similarity (our emphasis): "An important consequence of taking letters as primitive symbols is that we cannot then speak of one pair of letters as more closely *resembling* each other than another pair. We cannot say that *s* resembles *z* more closely that it resembles *t*. Letters, as primitive symbols, can only be tokens of the *same* type or *different* type. There is no notion of degree of difference among primitive symbols" (Newell & Simon, 1972, p. 26). In this sense, the otherwise central construct of similarity is ignored. This is the case not only for letters, but for many different representations used in problem solving in the classical tradition.

LPSA represents the problem space using similarities between types. The similarities are derived empirically from a corpus of contexts where each type appears. Our assumption is that types appearing in the same contexts tend to fulfill similar roles and functions in problem solving. When the vocabulary of a task increases, the probability of two different actions having the same functionality is higher. There are no known natural languages that do not exhibit polysemy. There are virtually no complex problem-solving tasks in which the same goal cannot be achieved by different series of actions.

There is another epistemological problem that LPSA solves: the origin of problem spaces. The range of tasks that we used differs markedly from the kind of tasks that other theories of problem solving have tried in the past. For the very early attempts of the general problem solver (GPS, Newell & Simon, 1963) to the more contemporary approaches to expert and novice comparisons, the tasks selected have had a remarkable importance in the shape theories take to explain them. Particularly distressing was the fact that each task generated an independent theory, often poorly specified by a flow diagram, and thus different results could barely be integrated: ”Theorists seem simply to write a different theory for each task" (Newell, 1980, p. 694). Newell proposed that the integrative metatheory could be built around the problem space hypothesis: ”the fundamental organizational unit of all human goal-oriented symbolic activity is the *problem space*“ (Newell, 1980, p.696).

The concept of problem space has been central theories of problem solving.[9] This concept has been used widely, and different researchers have interpreted it in different ways. Even their own creators have changed it in successive publications (see Quesada, unpublished, appendix J for a review of the changes in the definition). For example, in the book ‘universal subgoaling and chunking’ (Laird, Rosenbloom, & Newell, 1986), we find some evidence that the definition of problem space is not considered a closed, finished one. Newell mentions ‘a list of some forty key questions that needed to be answered to define problem spaces adequately’ (p. xiv) that was never published, implying that a univocal definition of the concept was still under construction.

It is natural that a useful concept experiences many incarnations in psychology. LPSA retakes and extends the central idea of the problem space, but has reworked it once more. In this sense, LPSA is more a continuation of the theory building efforts from the past than a completely new start. There are, however, several differences between the classical idea of problem space and the one that LPSA proposes. We find the following differences worth mentioning:

(1) In LPSA, problem spaces are metric spaces; this can be considered a significant departure from the original proposal, in which problem spaces were characterized as finite directed graphs, and often as simple search trees. LPSA takes advantage of the mathematical apparatus defined on vector representations. This approach is also comparatively simpler than alternative representations, such as propositions and productions.

(2) Problem spaces are derived empirically from experience. In most theories of representation in problem solving, the problem spaces are hypothesized by experimenters, not derived automatically by an unsupervised procedure. Newell (1980) pointed out that one of the major items on the agenda of cognitive psychology is to banish the homunculus. The homunculus is present in the classical problem-solving theory at least in one place: the intelligent creation of the problem space. The proposal goes like this: the generation of problem spaces is a symbolic cognitive task, and then it must be performed by the subject by means of a problem

space. LPSA provides a solution to break this recursive explanation: the problem spaces are derived automatically from experience as described in this paper, there is no need for an intelligent agent in their creation.

Two systems of reasoning

Since LPSA needs a corpus of experience, and does not propose mechanisms to act when there is no experience, we need to assume that there are two modes of reasoning, one for situations in which we know very little and another for those situations where we already have a knowledge base. In this sense, LPSA is a theory of experienced problem solving, but not a theory for all kinds of problem solving. A logical step is to propose that there is more than one system of reasoning, and that LPSA is reflecting one of several systems.

The question, ”Do humans use two modes of [cognition, representation, processing, learning, etc] or just one?“ is central to cognitive science and has been discussed by several theorists (e.g., Hahn & Chater, 1998; Pothos, in press; Shanks & St. John, 1994; Sloman, 1996). In the domain of expert chess, the question has been formulated as ”How much of expert problem-solving behavior is explained by real-time search through the task problem space and how much is explained by pattern recognition?“ (Gobet, 1997, p. 291). Evans and Over (1996, chapter 7) proposed that almost all reasoning tasks show evidence of a logical and non-logical component of performance. They argued that the two modes of performing are interactive rather than competing. Their distinction is closely related to the literature on implicit versus explicit learning.

Stanovich (1991) reviews a collection of theories in which theorists have proposed two different systems of reasoning. He grouped them as system1 and system2. System1 is characterized as automatic, unconscious, and relatively undemanding of computational capacity.

System2 is controlled, reflected in IQ tests, rule-based, and involved in analytical cognition. The tasks that pertain to system1 are highly contextualized, whereas system2 tends to decontextualize (abstract) problems. The two systems can produce contrary solutions to the same situation. Stanovich (1991) seems to be very concerned with the evolutionary interpretation of the two systems and its relevance to set the arguments about rational behavior.

Sloman (1996) tagged the two components “associative system” and “rule-based system.” He used similarity-based thought and temporal similarity relations to draw inferences and make predictions that resemble those of a sophisticated statistician: ”Rather than trying to reason on the basis of an underlying causal or mechanical structure, it constructs estimates based on underlying statistic structure. Lacking highly predictive causal models, this is the preferred mode for many forecasters, such as weather and economic ones“ (Sloman, 1996, p. 4). In pilots, for example, the underlying causal or mechanical structure can be present, but, after extensive experience, it can be easier for them to operate using the statistical structure that they have extracted during practice.

We consider LPSA to correspond to the associative system, ”system1.” The system2 or rule-based system could be made responsible for all the effects that LPSA does not cover: knowledge-lean tasks, learning from being told and instruction, deductive reasoning, and planning.

Related approaches and future directions

LPSA is a similarity-based theory and it uses a spatial metaphor in the Shepard (1987) tradition. The level of specification of LPSA is more detailed than any other (partial) theory of CPS. To our knowledge, there are no theories of CPS that actually spell out the representations and processes that are supposed to be functioning when people interact with different complex tasks. Below, we compare LPSA to four related approaches:

(1) Case-based reasoning. The idea of using similarity-based theories for problem solving is not new, it is explicitly presented in case-based reasoning (CBR), as well as in the work on analogical reasoning (e.g., Holyoak & Thagard, 1989), where similar elements in two situations have to be mapped in order to solve the problem. In CBR, a solution to a new problem is created by adapting past solutions to problems that resemble the new one. Both LPSA and CBR share certain characteristics: for example both may use nearest neighbors algorithms to select the closer represented cases to the current situation. Also, both tend to use flat (non-hierarchical) representations. However, the CBR expert systems are not the result of automatic learning procedures, but rather based on “knowledge engineering”, where humans compile the relevant domain knowledge into a computer accessible form (Hahn & Chater, 1998). Another difference is that LPSA does not keep a different representation for different repetitions of the same event, but a single abstraction that combines the knowledge that it has of that particular event. In this sense, LPSA is closer to abstraction-based models of representation (e.g., Smith & Medin, 1981) than to exemplar-based models (e.g., Nosofsky, 1988).

(2) In Case-based decision theory (Gilboa & Schmeidler, 1995) the memory is also considered to be a matrix of problems by actions, like in LPSA. In contrast with LPSA, there can be only one solution per problem, no repeated problems, and problems are not defined as collections of actions. There is also no dimension-reduction step. These are all reasonable assumptions for discrete, one-shot decisions. LPSA assumptions are better suited to multi-stage decision situations. In these situations it is not easy to map actions and results (important in CBDT to calculate utilities of actions) because they may be separated by long intervals. This is

known as the credit assignment problem and LPSA bypasses it by not using utility to create representations (as CBDT), but context-based similarity.

(3) Instance-based learning theory (IBLT, Gonzalez, Lerch, & Lebiere, 2003) is also an abstraction-based model. It has the added advantage that it proposes a general heuristic for the situations when the cognitive system has no experience (or the reasoning mode system1). LPSA and IBLT share some assumptions, try to explain similar phenomena (learning in dynamic decision making situations) and have very different strong points (representational clarity and generality for LPSA versus detailed performance process specified in IBLT).

(4) Minerva-DM (Dougherty, Gettys, & Ogden, 1999) Minerva-DM is an exemplar model of memory. It proposes that humans retrieve traces from memory every time they find a judgment situation (called probe). The response evoked by the probe is computed using the average of the responses for the memory traces that are most similar to the probe. The vectors for the traces and probe are concatenations of smaller vectors (minivectors) representing features. The mapping between the feature represented and the representation is arbitrary: any random vector will do, as long as it is different from the one representing a different feature. That is, representations are voided of semantics. This is a disadvantage when compared with the vectors that LPSA generates, which are all ‘grounded’ on a particular environment.

LPSA is a newcomer among theories of problem solving. We have tried to situate it by comparing and contrasting it with more familiar approaches, and we have stressed its limited scope. There is clearly more to problem solving than what LPSA addresses. Nevertheless, LPSA offers the cognitive theorist some powerful new methods. We have only begun to explore its promise, however. Future work must proceed in two directions. First, and most obviously, the ability of LPSA to account for a broader range of issues in the study of problem solving needs to be explored. In addition, however, alternative approaches to similarity must be considered. LPSA uses metric spaces, but other models of representation, such as the information-distance approach of Chater & Vitanyi (2003) and the Bayesian model of generalization proposed by Tenenbaum & Griffiths (2001). Similarity is a central concept in cognitive science, and cognitive science is about to find ways of dealing with it. What we propose is to use large, naturalistic corpora of problem solving activity to generate problem space representations. The particular representation LPSA generates is a metric space via dimension reduction. The great advantage of LPSA is that it can deal with truly complex problem solving tasks, and with large corpora that provide realistic estimates of human problem solving behavior.

References

Anderson, J. R. (1991). Is human cognition adaptive? *Behavioral & Brain Sciences, 14*(3), 471-517.

Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought*. Mahwah, New Jersey: Lawrence Erlbaum associates.

Brehmer, B. (1992). Dynamic Decision-Making - Human Control of Complex-Systems. *Acta Psychologica, 81*(3), 211-241.

Brunswik, E. (1956). Perception and the representative design of psychological experiments. Berkeley, CA: University of California Press.

Burns, B. D., & Vollmeyer, R. (2000). Problem Solving: Phenomena in search for a thesis. In L. R. Gleitman & A. K. Joshi (Eds.), *Proceedings of the cognitive science society meeting* (pp. 627-632). NY: Lawrence Erlbaum Associated.

Cañas, J. J., Quesada, J. F., Antolí, A., & Fajardo, I. (2003). Cognitive flexibility and adaptability to environmental changes in dynamic complex problem solving tasks. *Ergonomics, 46*(5), 482-501.

Chater, N., & Oaksford, M. (1999). Ten years of the rational analysis of cognition. *Trends in Cognitive Sciences, 3*(2), 57-65.

Dougherty, M. R. P., Gettys, C. F., & Ogden, E. E. (1999). MINERVA-DM: A memory processes model for judgments of likelihood. *Psychological Review, 106*(1), 180-209.

Edelman, S. (1998). representation is representation of similarities. *Behavioral and Brain Sciences, 21*(4), 449-498.

Edelman, S. (1999). *representation and recognition in vision*. Cambridge, Massachusetts: MIT Press.

Edelman, S., & Intrator, N. (1997). Learning as extraction of low-dimensional representations. In D. Medin & R. Goldstone & P. Schyns (Eds.), *Psychology of Learning and Motivation* (Vol. 36): Elsevier Science.

Edwards, W. (1962). Dynamic decision theory and probabilistic information processing. *Human Factors, 4*, 59-73.

Ehret, B. D., Gray, W. D., & Kirschenbaum, S. S. (2000). Contending with complexity: Developing and using a scaled world in applied cognitive research. *Human Factors, 42*(1), 8-23.

Ericsson, K. A., & Lehmann, A. C. (1996). Expert and exceptional performance: Evidence of maximal adaptation to task constraints. *Annual Review of Psychology, 47*, 273-305.

Evans, J. S. B. T., & Over, D. E. (1996). *Rationality and reasoning*. Hove, England: Psychology Press.

Frensch, P., & Funke, J. (1995a). *Complex Problem Solving: The European Perspective*. Hillsdale, NJ: Lawrence Erlbaum.

Frensch, P., & Funke, J. (1995b). Definitions, traditions and a general framework for understanding Complex Problem Solving. In P. A. Frensch & J. Funke (Eds.), *Complex Problem Solving: The European Perspective* (pp. 3-25). Hillsdale, NJ: Lawrence Erlbaum.

Funke, J. (1992). Dealing with dynamic systems: Research strategy, diagnostic approach and experimental results. *German Journal of Psychology, 16*, 24-43.

Gibson, J. J. (1979). *The Ecological Approach to Visual Perception*. Boston, MA: Houghton Mifflin.

Gilboa, I., & Schmeidler, D. (1995). Case-Based Decision-Theory. *Quarterly Journal of Economics, 110*(3), 605-639.

Gobet, F. (1997). A Pattern-recognition Theory of Search in Expert Problem Solving. *Thinking & Reasoning, 3*(4), 291-333.

Gonzalez, C., Lerch, F. J., & Lebiere, C. (2003). Instance-based learning in dynamic decision making. *Cognitive Science, 27*, 591-635.

Gordon, A. S. (2003). Strategy Representation: An Analysis of Planning Knowledge: Lawrence Erlbaum Associates.

Gray, W. D. (2002). Simulated task environments: The role of high fidelity simulators, scaled worlds, synthetic environments, and laboratory tasks in basic and applied cognitive research. *Cognitive Science Quarterly, 2*(2).

Hahn, U., & Chater, N. (1998). Similarity and rules: distinct? exhaustive? empirically distinguishable? *Cognition, 65*(2-3), 197-230.

Holyoak, K. J., & Thagard, P. (1989). Analogical Mapping by Constraint Satisfaction. *Cognitive Science, 13*(3), 295-355.

Joslyn, S., & Hunt, E. (1998). Evaluating individual differences in response to time-pressure situations. *Journal of Experimental Psychology-Applied, 4*(1), 16-43.

Juslin, P., Olsson, N., & Winman, A. (1996). Calibration and diagnosticity of confidence in eyewitness identification: Comments on what can be inferred from the low confidence-accuracy correlation. *Journal of Experimental Psychology-Learning Memory and Cognition, 22*(5), 1304-1316.

Laird, J., Rosenbloom, P., & Newell, A. (1986). *Universal subgoaling and chunking*. Boston: Kluwer.

Landauer, T. K. (2002). On the computational basis of learning and cognition: Arguments from LSA. In N. Ross (Ed.), *The psychology of learning and motivation* (Vol. 41, pp. 43 - 84): Academic Press.

Landauer, T. K. (unpublished). Some remarks on consciousness by a somewhat maverick cognitive scientist.

Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review, 104*, 211-240.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to Latent Semantic Analysis. *Discourse Processes, 25*, 259-284.

McGraw, K. O., & Wong, S. P. (1996). Forming Inferences About Some Intraclass Correlation Coefficients. *Psychological Methods, 1*(1), 30-46.

Newell, A. (1980). Reasoning, Problem Solving, and decision processes: the problem space as a fundamental category. In R. Nickerson (Ed.), *Attention and Performance VII* (pp. 693-718). Cambridge, MA: Harvard.

Newell, A. (1990). *The unified theories of cognition*: Harvard University Press.

Newell, A., & Simon, H. A. (1963). GPS: A program that simulates human thought. In E. A. Feigenbaum & J. Feldman (Eds.), *computers and thought* (pp. 279-293). Oldernbourg KG.

Newell, A., & Simon, H. A. (1972). *Human Problem Solving*. Englewood Cliffs, New Jersey: Prentice-Hall, Inc.

Nosofsky, R. M. (1988). Exemplar-Based Accounts of Relations between Classification, Recognition, and Typicality. *Journal of Experimental Psychology-Learning Memory and Cognition, 14*(4), 700-708.

Oaksford, M., & Chater, N. (Eds.). (1998). *Rational models of cognition*: Oxford University Press.

Omodei, M. M., & Wearing, A. J. (1993). Fire Chief (Version 2.3): University of Melbourne.

Omodei, M. M., & Wearing, A. J. (1995). The Fire Chief microworld generating program: An illustration of computer-simulated microworlds as an experimental paradigm for studying complex decision-making behavior. *Behavior Research Methods, Instruments & Computers, 27*, 303-316.

Pothos, E. (in press). The Rules versus Similarity Distinction. *Behavioral & Brain Sciences*.

Pylyshyn, Z. W. (1994). Some primitive mechanisms of spatial attention. *Cognition, 50*, 363-384.

Pylyshyn, Z. W. (2001). Tracking without track: Loss of item identity in multi-element tracking. *Meeting of the psychonomic society, (41ST annual meeting), 5*, 34.

Quesada, J. F. (unpublished). Latent Problem Solving Analysis (LPSA): A computational theory of representation in complex, dynamic problem solving tasks. Unpublished Phd. dissertation, Granada (Spain).

Quesada, J. F., Cañas, J. J., & Antoli, A. (2000). An explanation of human errors based on environmental changes and problem solving strategies. In C. P. Warren (Ed.), *ECCE-10: Confronting Reality*. Sweden: EACE.

Quesada, J. F., Kintsch, W., & Gomez, E. (2003). Latent Problem Solving Analysis as an explanation of expertise effects in a complex, dynamic task, *Proceedings of the 25th Annual Conference of the Cognitive Science Society*. Boston, MA.

Quesada, J. F., Kintsch, W., & Gomez, E. (submitted). Expertise as the creation of multidimensional spaces. *Psychonomic Bulletin & Review*.

Richman, H. B., Gobet, F., Staszewski, J., & Simon, H. (1996). Perceptual and memory processes in the acquisition of Expert performance: the EPAM model. In K. A. Ericsson (Ed.), *The road to excellence: The acquisition of expert performance in the arts and sciences, sports, and games* (pp. 167-187). Mahwah, NJ: Erlbaum.

Rumelhart, D. E., Smolensky, P., McClelland, J. L., & Hinton, E. (1986). Schemata and sequential thought processes in PDP models. In D. E. Rumelhart & J. L. McClelland (Eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. Cambridge, MA: MIT Press.

Scholl, B. J., Pylyshyn, Z. W., & Feldman, J. (2001). What is a visual object? Evidence from target merging in multiple object tracking. *Cognition, 80*(1-2), 159-177.

Shanks, D. R., & St. John, M. F. (1994). Characteristics of Dissociable Human Learning-Systems. *Behavioral and Brain Sciences, 17*(3), 367-395.

Shanteau, J. (2001). What does it mean when experts disagree? In E. Salas & G. Klein (Eds.), *Linking expertise and naturalistic decision making*. Mahwah, NJ: Lawrence Erlbaum Associates.

Shepard, R. N. (1987). Toward a universal law of generalization for psychological science. *Science, 237*, 1317-1323.

Shorut, P. E., & Fleiss, J. L. (1979). Intraclass Correlations: Uses in Assessing Rater Reliability. *Psychological Bulletin, 86*(2), 420-428.

Simon, H. A. (1981). *The sciences of the artificial*. Cambridge, MA: MIT Press.

Sloman, S. A. (1996). The empirical case for two systems of reasoning. *Psychological Bulletin, 119*(1), 3-22.

Smith, E. E., & Medin, D. (Eds.). (1981). *Categories and Concepts*. Cambridge, MA.: Harvard university press.

Stanovich, E., & Cunningham, A. E. (1991). Reading as constrained reasoning. In J. R. Sternberg & P. A. Frensch (Eds.), *Complex problem Solving: principles and mechanisms* (pp. 3-61). Hillsdale, NJ: Lawrence Erlbaum Associates.

Tenenbaum, J. B., de Silva, V., & Langford, J. C. (2000). A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science, 290*(5500), 2319.

Vicente, K. J. (1999). *Cognitive Work Analysis*. Mahwah, NY: LEA.

Vicente, K. J., & Wang, J. H. (1998). An ecological theory of expertise effects in memory recall. *Psychological Review, 105*, 33-57.

Author Note

José Quesada, Institute of Cognitive Science, University of Colorado, Boulder,

Campus Box 344, Boulder, CO 80309-0344,quesadaj@psych.colorado.edu.

The simulator and expert time was possible thanks to a grant supported by the European Community - Access to Research Infrastructure action of the Improving Human

Potential Program under contract number HPRI-CT-1999-00105 with the National Aerospace Laboratory, NLR This research was also supported by Grant EIA – 0121201 from the National Science Foundation, and by the grant BSO 2002-02166 from the Ministerio de Ciencia y tecnología, Dirección General de Investigación.

Our acknowledgements to Tom Landauer, Simon Dennis and Bill Oliver, for proposing interesting theoretical issues. We are grateful to Kim Vicente and John Hajdukiewicz for sharing experimental data and insightful discussions. Alex Rutten and Adri Marsman were of great help not only in their respective roles as simulator experts and coordinators of the project, but also generating hypothesis and discussing the results.

Footnotes

[1] Although it is pretty common to see problem spaces described as trees, we used the less exigent description of problem spaces as directed graphs because some problem spaces of classical tasks in the literature contain loops, for example the task 'missionaries and cannibals'. The tree structure specifically defines that there is only one path that connect two elements (leaves), so any structure that displays loop is not a tree.

[2] However, recent literature (e.g., Tenenbaum, de Silva, & Langford, 2000) has proposed methods for nonlinear dimensionality reduction.

[3] The estimated total is 69 participants x 20 trials (in Quesada et al.) + 84 participants x 25 trials (in Cañas et al.) = 3480. There were about 30 trials that were not logged correctly.

[4] This method was selected so the cosines were very distinctive and easy to compare. A subspace where a few items (instead of pairs) have very distinctive cosines could be found too, but this approach was used because the sample of stimuli was easier to obtain.

[5] Although the F test is not particularly informative, since it is testing the uninteresting hypothesis of the ICC being 0 versus the alternative of ICC being greater than 0, as McGraw and Wong (1996) noted.

[6] This time the required correlation coefficient is Pearson, since human similarity judgments and cosines are not in the same scale, and do not necessarily have a comparable variance, so intraclass correlation is ruled out.

[7]A p value of .002 indicates that none of the polychoric correlations for the randomizations was higher than the observed one, being the proportion 1/501 = .002

[8] Number scrabble consists on 9 cards on a table, with 9 digits, each participant draws one, in turns, and the one who collects 3 cards whose digits sum 15 (e.g., 2 + 4 + 9 = 15) wins. If

all the pieces are drawn without any player adding 15, the game is a draw. It is formally equivalent to tic-tac-toe.

9 However, the problem space is a surprisingly ill-defined concept that has been changed and reworked in successive papers by their own proponents, and by others. Since most researchers in problem solving use it, it has been stretched and adapted in different ways to cover new situations, and some authors (e.g., Burns & Vollmeyer, 2000) have issued warnings about this.

Table 1

Extract from a Firechief log file

No.	Command	Time	Performance Score	Appliance Code	Appliance Type	Position	Destination	Landscape
1	Move	44	100.00	4	Copter	(11, 4)	(12, 9)	Forest
2	Move	62	99.77	4	Copter	(12, 9)	(11, 9)	Forest
3	Drop Water	78	99.42	4	Copter	(11, 9)		Forest
4	Move	137	98.95	3	Copter	(8, 6)	(12, 10)	Clearing
5	Drop water	147	98.95	3	Copter	(12, 10)		Clearing
6	Move	178	98.71	2	Truck	(4, 11)	(13, 8)	Forest
7	Move	247	98.48	1	Truck	(4, 14)	(14, 10)	Forest
8	Control Fire	255	98.48	2	Truck	(13, 8)		Forest

Figure Captions

Figure 1: Annotated screenshot of an ongoing Firechief trial. Each cell type and appliance has different properties such as flammability, speed, etc.

Figure 2: First 8 movements in 3 slices randomly sampled from the Firechief experiments described in Quesada et al. (2000) and Cañas et al. (Cañas et al., 2003). When an action is shared by two extracts, it is marked as a shaded cell. The actions that participants performed are depicted as arrows for movements, squares for control fires, and circles for drop-water actions.

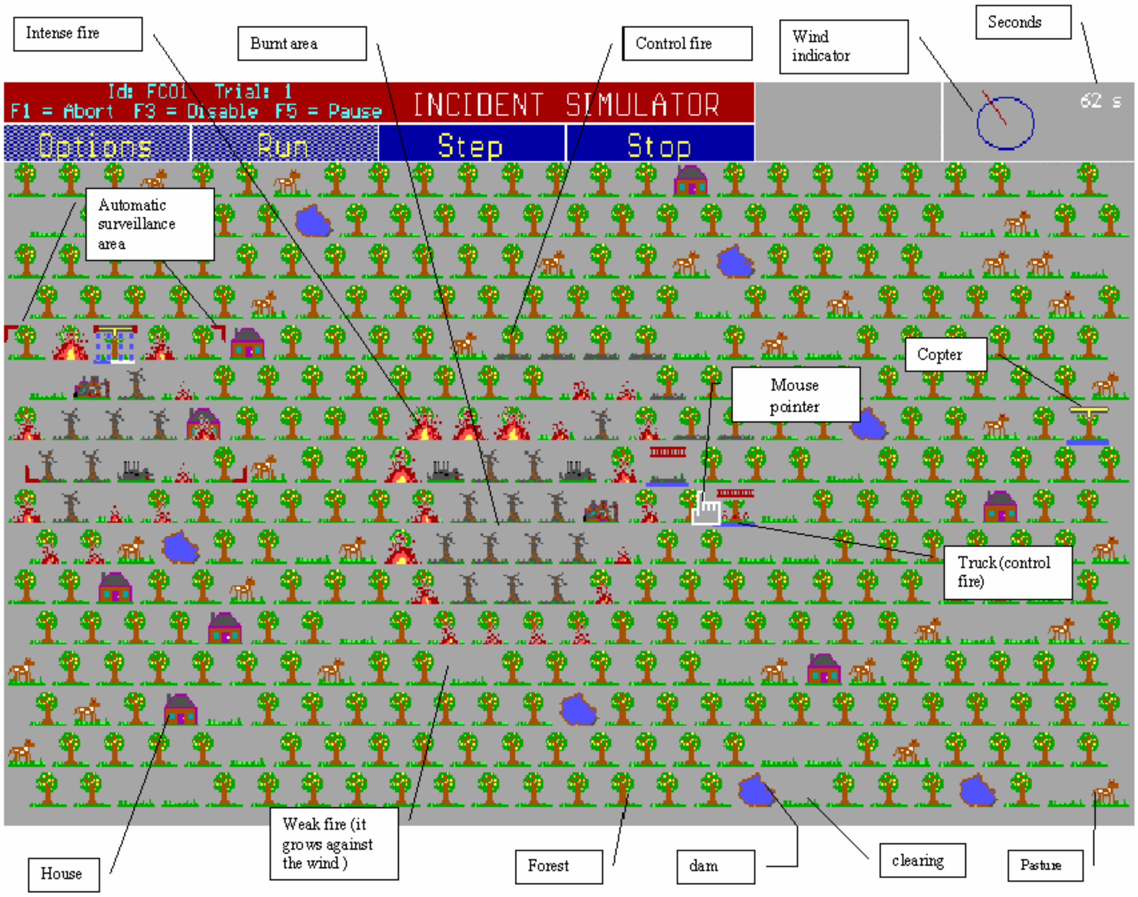
Figure 3: Observed versus predicted average human judgments for six pairs of Firechief trials.

Figure 4: Basic scheme of a landing. (1) Glideslope. (2) Flare initiation height. (3) Pitch rate (4) Glideslope predicted intercept point. (5) Touchdown point. (6) Reversal in the direction of the vertical acceleration at touchdown point.

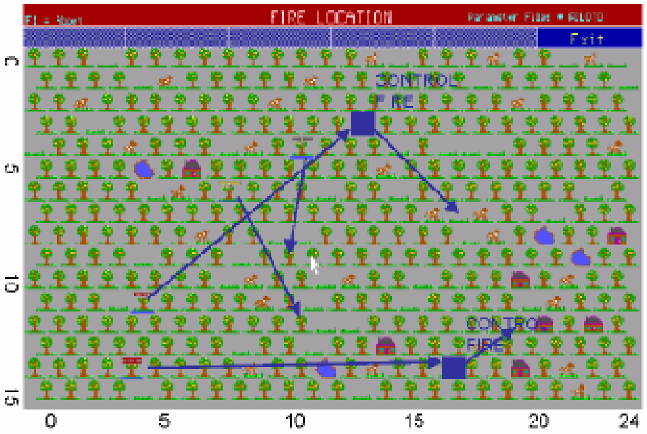
Figure 5: Schema of the proposed "triangulation of expertise" approach

Figure 6: Agreement between the model and the reduced-information expert for each of the rating criteria

Figure 7: Agreement between the model and the reduced-information expert for each of the rating criteria when the model has been trained on a corpus where the environment changes randomly.

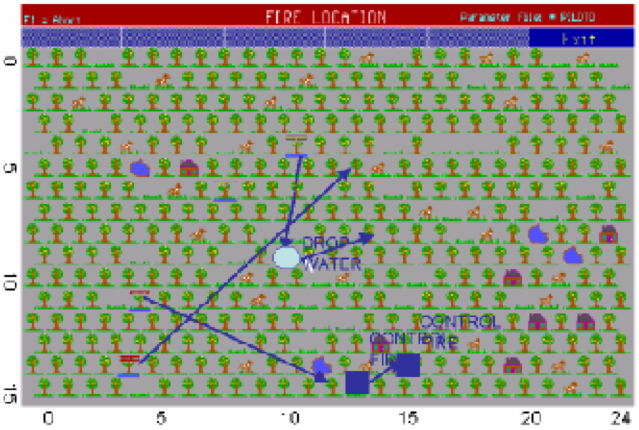

Intense fire
Burnt area
Control fire
Wind indicator
Seconds
Id: FC01 Trial: 1
F1 = Abort F3 = Disable F5 = Pause
INCIDENT SIMULATOR
62 s
Options
Run
Step
Stop
Automatic surveillance area
Copter
Mouse pointer
Truck (control fire)
House
Weak fire (it grows against the wind)
Forest
dam
clearing
Pasture

Example 1



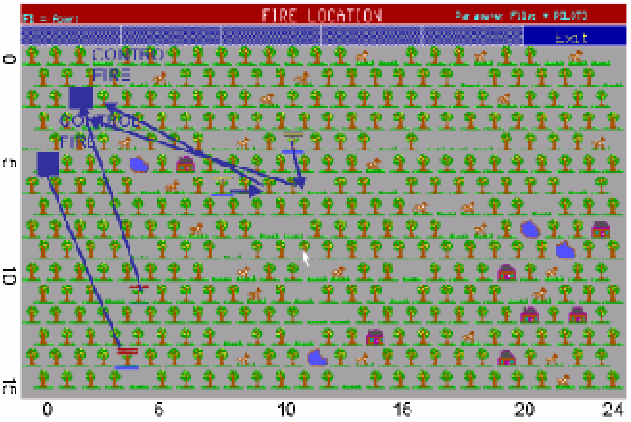
move_2_truck_4_11_13_3_forest
move_1_truck_4_14_16_14_forest
move_3_copter_8_6_11_12_forest
move_4_copter_11_4_11_9_forest
control_fire_2_truck_13_3_forest
control_fire_1_truck_16_14_forest
move_2_truck_13_3_17_7_clearing
move_1_truck_16_14_20_12_forest

Example 2

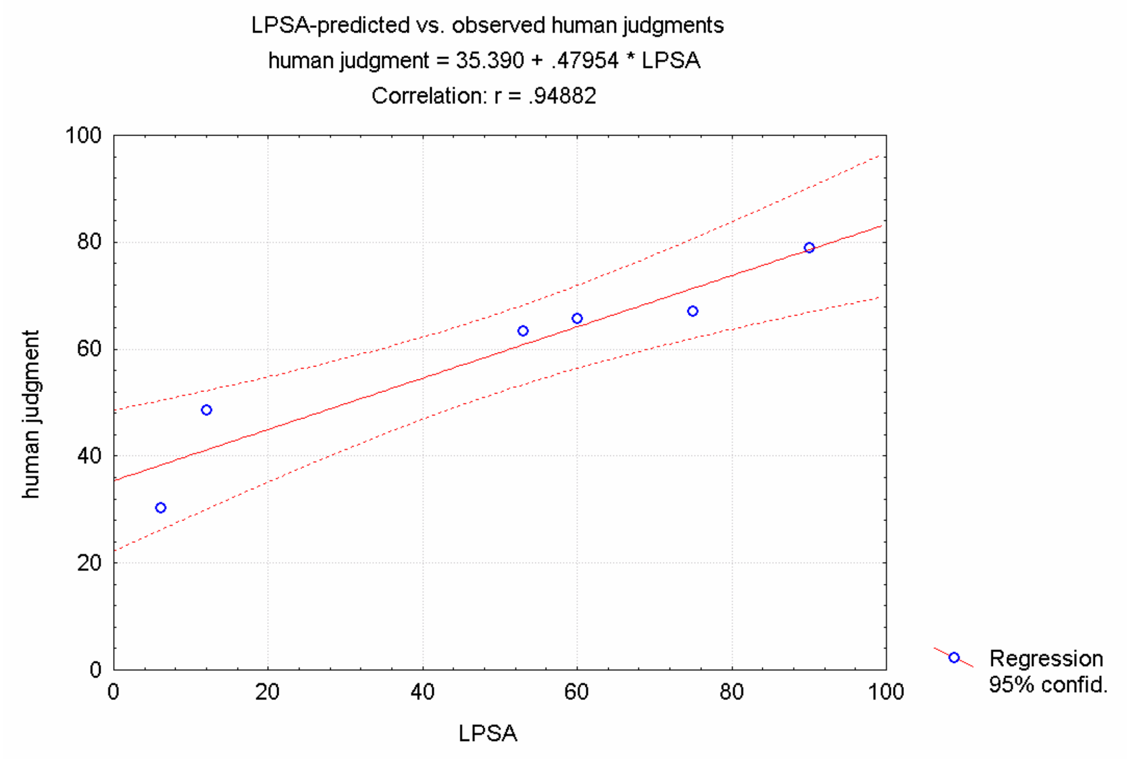


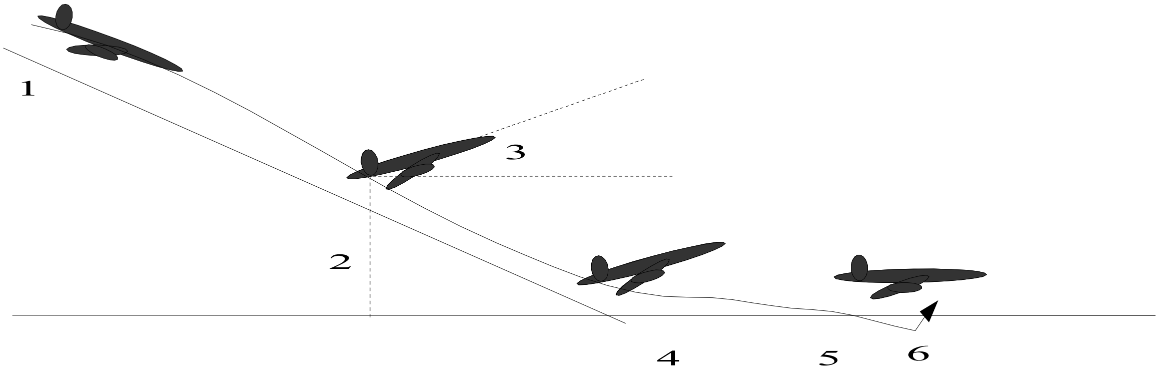
move_2_truck_4_11_12_15_forest
move_1_truck_4_14_13_5_forest
move_4_copter_11_4_11_9_forest
drop_water_4_copter_11_9_forest
move_4_copter_11_9_13_8_forest
control_fire_2_truck_12_15_forest
move_2_truck_12_15_13_14_forest
control_fire_2_truck_13_14_forest

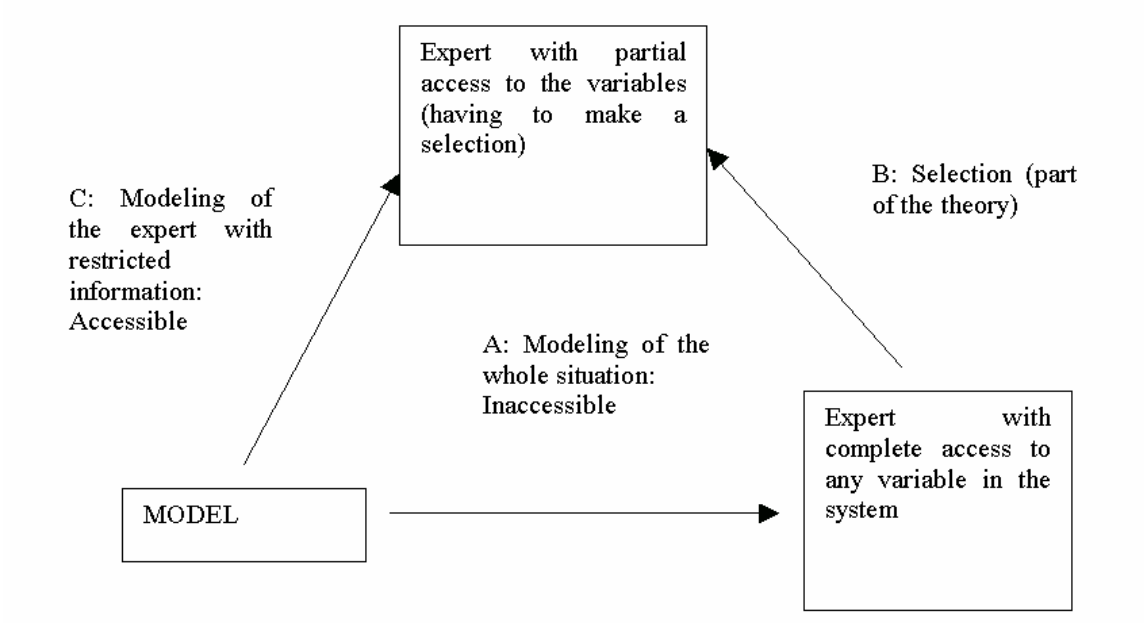
Example 3

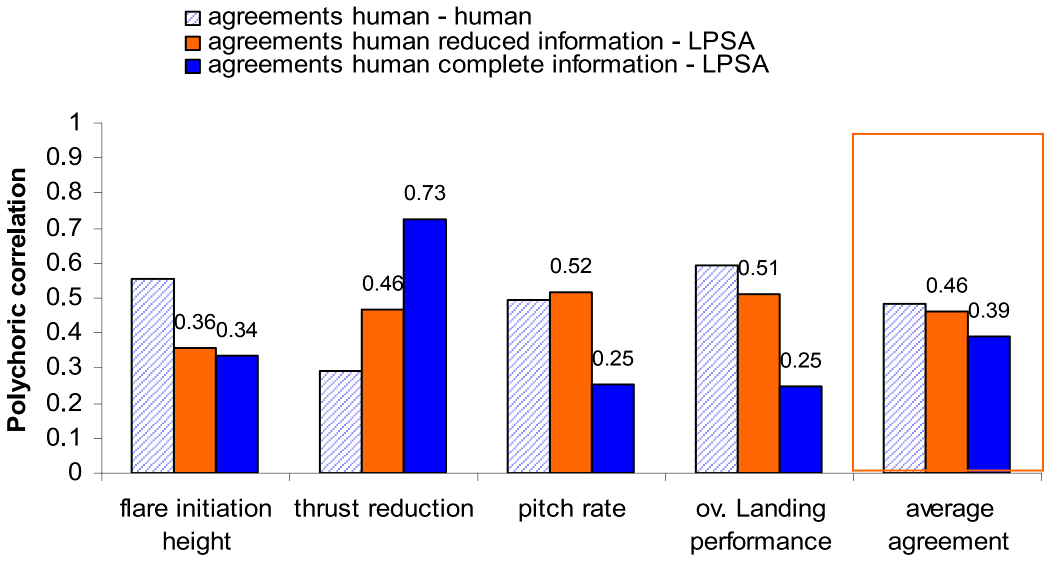


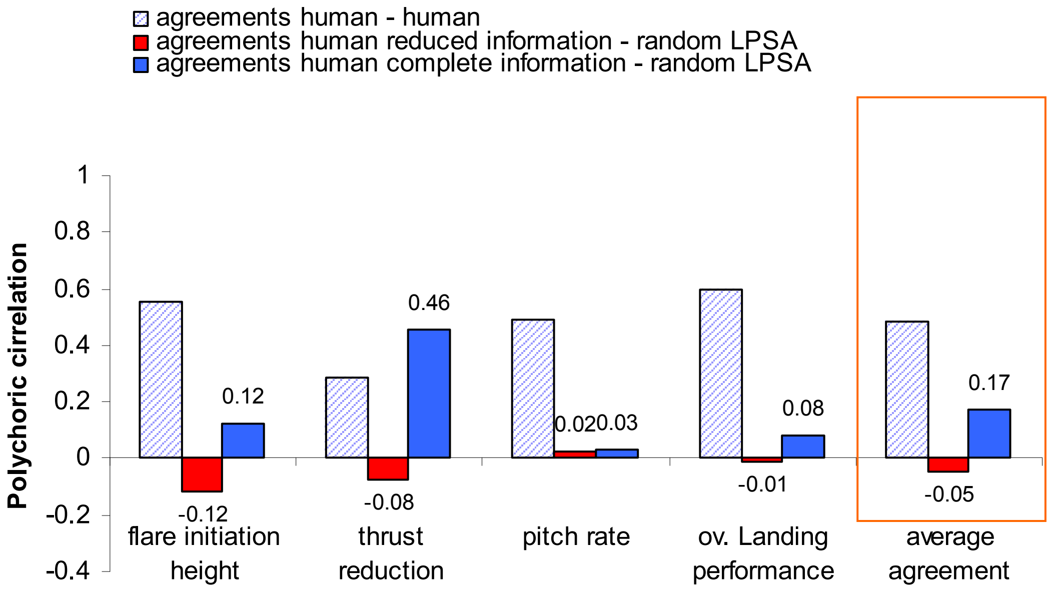
move_2_truck_4_11_2_2_pasture
move_1_truck_4_14_0_5_forest
move_4_copter_8_6_8_4_clearing
move_3_copter_8_6_8_10_clearing
control_fire_2_truck_2_2_pasture
control_fire_1_truck_0_5_forest
move_4_copter_8_4_4_2_forest
move_3_copter_8_10_2_3_clearing


LPSA-predicted vs. observed human judgments
human judgment = 35.390 + .47954 * LPSA
Correlation: r = .94882
human judgment
100
80
60
40
20
0
0
20
40
60
80
100
LPSA
Regression
95% confid.


1
2
3
4
5
6


Expert with partial access to the variables (having to make a selection)
C: Modeling of the expert with restricted information: Accessible
B: Selection (part of the theory)
A: Modeling of the whole situation: Inaccessible
Expert with complete access to any variable in the system
MODEL


▨ agreements human - human
🟧 agreements human reduced information - LPSA
🟦 agreements human complete information - LPSA
Polychoric correlation
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.36
0.34
0.46
0.73
0.52
0.25
0.51
0.25
0.46
0.39
flare initiation height
thrust reduction
pitch rate
ov. Landing performance
average agreement


▨ agreements human - human
🟥 agreements human reduced information - random LPSA
🟦 agreements human complete information - random LPSA
Polychoric cirrelation
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
0.12
-0.12
0.46
-0.08
0.02
0.03
0.08
-0.01
0.17
-0.05
flare initiation height
thrust reduction
pitch rate
ov. Landing performance
average agreement

Appendix: Instructions

The following test will measure how good you are and how much you have learnt. The buttons trigger videos of some replay trials from firechief. The buttons are grouped in seven groups of two. The speed is eight times faster than what you have experienced, but you should still be able to understand what the player is doing.

In this last part of the experiment your task will be to compare these seven pairs of trials and emit a similarity judgment from 0 to 100 in the text box. You will give it a 0 it they have absolute nothing in common, and a 100 if they are exactly the same or really similar in what they are doing.

You can change your mind -and your ratings- at any time. For example, imagine that you gave the previous pair of trials a 100, but in the current pair the trials are even more similar to each other. Then, you can go back, give the previous pair a 90 (or 80) and then rate the current pair as 100.

You can replay pairs to remember them better. For example, if you doubt which pair (example, A and C) should get a higher rating of similarity, just go ahead and play them again. It is very important to evaluate the similarity correctly. There are important differences in the pairs selected. You may want to know that they are selected from a continuum of similarity (that is, they are in different 'steps' of a 'ladder' of similarity). One pair is extremely related, one is very unrelated, and the rest lie in between.