Forbus, K. 2000. Exploring analogy in the large. In Gentner, D., Holyoak, K. and Kokinov, B. (Eds) *Analogy: Perspectives from Cognitive Science*. Cambridge, MA: MIT Press.

Exploring analogy in the large

Kenneth D. Forbus
Institute for the Learning Sciences
Northwestern University
1890 Maple Avenue
Evanston, IL, 60201, USA
Email: forbus@ils.nwu.edu

Final Version

Forbus, K. 2000. Exploring analogy in the large. In Gentner, D., Holyoak, K. and Kokinov, B. (Eds) *Analogy: Perspectives from Cognitive Science*. Cambridge, MA: MIT Press.

1 Introduction

There has been solid progress in the scientific understanding of the phenomena of analogy and similarity, as the work discussed in this volume attests. One of the drivers of this progress has been research on cognitive simulations that model a variety of phenomena in analogy, similarity, and retrieval (Gentner & Holyoak, 1997). My goal here is to highlight what I believe is a promising new direction for cognitive simulation of analogy and similarity. To date, most models have focused on exploring the fundamental phenomena involved in matching, inference, and retrieval. While there is still much to be discovered about these processes, I believe that it is now critical to focus more on exploring analogy "in the large": simulating the roles comparison plays in larger-scale cognitive processes. Along with many of the researchers represented in this volume, I believe that analogy is one of the core processes of cognition. This makes it crucial to ensure that the processes that we propose can play their intended role in cognition. This new focus on what might be called *large-scale analogical processing* can provide new understanding of the fundamental processes as well as yield new insights into the cognitive processes that rely on them.

This paper begins with a brief review of SME and MAC/FAC, our simulations of matching and retrieval. Next I lay out several arguments for exploring analogy in the large, including why it is now very feasible and what we can learn by such explorations. A new constraint on cognitive simulations, the *Integration Constraint*, is proposed: A cognitive simulation of some aspect of analogical processing should be usable as a component in larger-scale cognitive simulations. I believe that the implications of this new constraint for cognitive simulation of analogy are far-reaching. After that, two explorations of larger-scale phenomena are described. First, I describe a theoretical framework in which we model common sense reasoning as an interplay of analogical and first-principles reasoning. Second, I describe how SME and MAC/FAC have been used in a case-based coach that is accessible to engineering thermodynamics students worldwide via electronic mail. These examples show that exploring analogy in the large can provide new insights and new challenges to our simulations. Finally, the broader implications of this approach are discussed.

2 SME and MAC/FAC: A brief review

Structure-mapping theory (Gentner, 1983, 1989) provides an account of comparison processes that is consistent with a growing body of psychological evidence (Gentner & Markman, 1997). These computations have been simulated with SME (Falkenhainer *et al* 1989; Forbus *et al* 1994), which in turn has been used as a module in other simulations and in performance systems. SME takes as inputs two structured propositional representations as inputs, the *base* (about which more is presumably known) and the *target*. These descriptions are internal symbolic representations, with predicate/argument structure. They can include *attributes*, which are unary predicates indicating features, *relations* that express connections between entities, and *higher-order relations* that express connections between relations.

Given a base and target, SME computes a *mapping* (or a handful of them). Each mapping contains a set of *correspondences* that align particular items in the base with items in the target, and *candidate inferences*, which are statements about the base that are hypothesized to hold in the target by virtue of these correspondences. SME operates in polynomial time (Forbus & Oblinger, 1990), and can also incrementally extend its mappings as more information is added to the base and/or target (Forbus *et al* 1994).

MAC/FAC (Forbus, Gentner, & Law 1995) models similarity-based retrieval. The MAC stage first uses a simple, non-structural matcher to filter out a few promising candidates from a (potentially immense) memory of structured descriptions. The FAC stage then evaluates these candidates more carefully, using SME to provide a structural match, and thus the correspondences and candidate inferences that indicate how the reminded information may be relevant to the current situation. Scalability comes from the simplicity of MAC: The MAC stage lends itself to implementation in parallel (including connectionist) hardware. MAC/FAC has been used successfully to model various aspects of similarity-based retrieval (Forbus, Gentner & Law, 1995).

The purpose of computational models is to generate insights, and these in turn should lead to new psychological work. This has indeed happened with SME and MAC/FAC. Some of the psychological research which has used ideas from SME and MAC/FAC includes

- Systematicity and structural consistency influences interpretation of analogies (Clement & Gentner, 1991)
- Structural consistency influences inference in analogical reasoning (Clement & Gentner, 1991; Keane, 1996; Spellman & Holyoak, 1992, 1996; Markman, 1997) and in category-based induction (Lassaline, 1996; Wu & Gentner, 1998)
- Systematicity influences inference in analogical reasoning and category-based induction (Clement & Gentner, 1991; Wu & Gentner, 1998)
- Ordinary similarity comparisons utilize processes of structural alignment and mapping, i.e., similarity is like analogy (Gentner, 1989; Gentner & Markman, 1995, 1997; Markman & Gentner, 1993, in press; Medin, Goldstone & Gentner, 1993)
- Similarity-based retrieval is surface-driven, but similarity-based reasoning is structurally driven (Gentner, Rattermann & Forbus, 1993; see also Holyoak & Koh, 1987; Ross, 1989)

A number of SME's psychological predictions have been confirmed by experiments with human subjects, including

- Online processing of similarity and analogy is influenced both by object richness and by relational depth (Gentner & Rattermann, 1991; Markman & Gentner, 1993; Rattermann & Gentner, 1998)
- *Relational Shift:* Early in development object matches win over relational matches because of inadequate relational knowledge (Gentner & Toupin, 1986; Gentner, 1988; Gentner & Rattermann, 1991; Rattermann & Gentner, in press; see Goswami (1992, this volume) and Halford (1987, 1993) for related positions)
- Learning higher-order domain relations enables children to perform relational mappings (Gentner & Rattermann, 1991; Kotovsky & Gentner, 1996; Rattermann & Gentner, 1998; Goswami & Brown, 1989)

There are several psychological phenomena that we have not tried to capture so far within SME. One of these is working memory limitations (which are being explored in

LISA (Hummel & Holyoak, 1997)). Another, which we are just beginning to investigate, is predictions about processing time within a mapping (which have been addressed by IAM (Keane, 1990, 1997) and SIAM (Goldstone, 1994; Goldstone & Medin, 1994)).

3 Arguments for large-scale analogical simulations

I am suggesting that we devote relatively more of our effort to large-scale analogical simulation over studying analogy and similarity in isolation. This may sound strange from the perspective of other areas of Cognitive Science, where isolating processes is both highly valued and notoriously difficult to achieve. In cognitive psychology, for instance, studying any process must be done in the context of multiple concurrent processes, most of whose character is still conjectural. Rarely can single processes be isolated with absolute certainty. Of all the methods in Cognitive Science, cognitive simulation has the most direct access to isolated processes, since one can, with appropriate assumptions, create simulations of very specific slices of hypothesized systems and particular processes. This is a signature strength of the method, and can provide clarity of a sort that is quite difficult for other methods to achieve. Why work against a method's strength?

This ability to zero in on a specific process is indeed a major strength of simulation studies. However, it can also be a major weakness: The assumptions one makes about the surrounding systems become critical. Too many simulations have only been tested in single tasks or toy domains, with no particular crosschecks as to their assumptions about the properties of representations and other processes posited. To avoid myopia, we must look up, to larger units of analysis encompassing a broader range of phenomena.

3.1 The Integration Constraint

The ubiquity of structural alignment in cognition, as suggested by recent results in Cognitive Science, lends support to the need for larger-scale simulation studies. It appears that the same component processes are used in a range of cognitive processes spanning from visual perception to problem solving to learning to conceptual change. Models of these component processes must be capable of the same ubiquity. This suggests what I call the *integration constraint:* A cognitive simulation of an analogical process, such as matching or retrieval, should be able to serve as a component in simulations of larger-scale cognitive processes.

The integration constraint represents a new, and very difficult, challenge. As discussed in Section 7, no model can yet be said to fully satisfy it, although some are coming closer than others. This very difficulty is exactly what makes it worthwhile: Ultimately, our models need to be capable of human-level performance.

3.2 Other components are coming together

The scale of phenomena that can be simulated is increasing. This is due to the confluence of several effects:

- Better understanding of how to build AI systems. The various components that are needed to build larger-scale simulations, such as reasoning systems, natural language processing, and vision, are becoming more available. There is also an explosion in technologies for connecting heterogeneous software components, which ultimately should simplify experimentation.

- Tremendous increases in available computing power and memory. It seems likely that Moore's law, which predicts a doubling of capacity every 18 months, will last until at least 2010.
- Increased interest in creating larger-scale systems that can tackle real problems. For example, computers are now widespread enough that there is significant interest in making them radically easier to use, by making software that understands its users and what they are doing better. This provides new settings for research.

The scale of what can be accomplished has changed. For example, the SOAR cognitive architecture has been used to create "software pilots" that operate in shared virtual worlds with humans. In one recent large-scale exercise (Jones *et al* 1999), all of the U.S. tactical and logistics flights were carried out by SOAR pilots. These pilots communicated with their human comrades through speech, using simulated radio channels.

In what sense are these SOAR pilots cognitive simulations? No one compared their millisecond to millisecond performance against human pilots doing a similar task, as might be the goal in using a simulation of a single pilot to evaluate an interface. Here, the goal was a larger unit of analysis, the tactical behavior occurring second to second. These pilots had to operate in the simulated world in a manner that enabled them to be treated as human by their collaborators, in a wide variety of circumstances. That kind of robust capability, to be able to perform successfully over a range of complex situations in a human-like way, is a powerful measure of success for a large-scale simulation.

An argument often raised against larger-scale simulations is that the number of underconstrained assumptions required becomes unmanageable. It is true that, given the state of knowledge about cognitive processes in general, the number of underconstrained assumptions about any process or representation (including its very existence, as the debates on the existence of prototypes in category research illustrates!) remains high, and that adding more processes increases the number of assumptions involved. However, the need to work in concert with other processes invokes new constraints: If for example one process P1 exploits the results of another process P2, then P2’s output must be compatible with P1’s input. Thus studies of analogical encoding, for example, require studying the likely kinds of representations created by the human visual system (Ferguson, 1994; Ferguson *et al* 1996). Small-scale simulations still require assumptions about their inputs, but such assumptions are all too often made without any justification. Being used as a component in a larger simulation shines new light on those assumptions.

Every cognitive simulation has aspects that are intended to be models of human processing and aspects that are not. Large-scale cognitive simulations do not necessarily need to make more assumptions than small-scale simulations, if they limit the range of their claims. Even if, say, a natural language module is not something that the simulation authors would bet on as a psychological model, it still can be used beneficially in a larger simulation, if it decreases the reliance on hand-coded representations.

Ultimately, we will build larger-scale simulations because, without them, we cannot explore via simulation the same range of human cognition that other areas of Cognitive Science can explore. Unless we increase the size of our unit of analysis, we cannot use simulation to explore problem solving, learning, and conceptual change. This increase will not be without costs: For example, it requires substantially more work to build large systems than small systems. Moreover, looking at the macro puts the micro out of focus -- reaction times are unlikely to be a useful measure at this larger scale, and more emphasis will have to be placed on content-oriented measures, such as errors, patterns of misconceptions, and models learned. But what we gain from this shift is the ability to model human behavior at a grander scale, reducing the gap between what people do and what our simulations of them can do.

3.3 Large-scale analogical processing as a source of constraints

I have argued that looking at larger scale analogical processing can provide useful insights and constraints on the component processes involved. A concrete example is provided by Phineas, a system created by Brian Falkenhainer in the late 1980s. The experience in creating Phineas led to substantial changes in SME. Let us look more closely at this example, to see how large-scale analogical processing works.

Phineas (Falkenhainer, 1987, 1988, 1990a) learned physical domain theories by analogy with previously understood examples. In addition to SME, its design exploited several other modules that have themselves been used in other projects:

- QPE (Forbus, 1990), is a qualitative simulator using Qualitative Process theory (Forbus, 1984). Given a theory describing the qualitative laws of a domain and a specific situation, QPE produced predictions of the kinds of behaviors that could occur in that situation. This description, called an *envisionment*, describes behavior symbolically, in terms of states and transitions between them.
- DATMI (Decoste, 1990), is a measurement interpretation system. It takes as input an envisionment, as produced by QPE, and a specific behavior, described in terms of measurements. It figures out how the specific behavior can be explained in terms of the states of the envisionment.

The architecture of Phineas is illustrated in Figure 1.

The best way to illustrate how Phineas worked is by example. Phineas started with the description of the behavior of a physical system, described in qualitative terms. In one test, Phineas was given the description of the temperature changes that occur when a hot brick is immersed in cold water. Phineas first attempted to understand the described behavior in terms of its current physical theories, by using QPE to apply these theories to the new situation and qualitatively simulate the kinds of behaviors that can occur, and using DATMI to construct explanations of the observations in terms of the simulated possibilities. In this case, Phineas did not have a model of heat or heat flow, so it could not find any physical processes to explain the observed changes. In such circumstances Phineas turned to analogy to seek an explanation.

To derive an explanation, Phineas attempted to find an analogous behavior in its database of previously explained examples. These examples were indexed in an abstraction hierarchy by their observed behaviors. Based on global properties of the new instance's behavior, Phineas would select a potentially analogous example from this hierarchy. When evaluating a potential analog, Phineas used SME to compare the behaviors, generating a set of correspondences between different physical aspects of the situations. These correspondences were then used with SME to analogically infer an explanation for the new situation, based on the explanation for the previously understood situation. Returning to our immersed brick example, the most promising candidate explanation is a situation where liquid flow causes two pressures to equilibrate. To adapt this explanation for the original behavior Phineas created a new process, `PROCESS-1` (which we'll call heat-flow for simplicity after this), which is analogous to the liquid flow process, using the correspondences between aspects of the two behaviors. In this new physical process, the relationships that held for pressure in the liquid flow situation are hypothesized to hold for the corresponding temperature parameters in the new situation.

Generating the initial physical process hypothesis via analogical inference is only the first step. Next Phineas has to ensure that the hypothesis is specified in enough detail to actually reason with it. For instance, in this case it is not obvious what the analog to liquid is, nor what constitutes a flow path, in the new heat flow situation. It resolved these questions by a combination of reasoning with background knowledge about the physical world (e.g., that fluid paths are a form of connection, and that immersion in a liquid implies that the immersed object is in contact with the liquid) and by additional analogies. Falkenhainer calls this the *map/analyze cycle*. Candidate inferences were examined to see if they can be justified in terms of background knowledge, which may in turn lead to further matching to see if the newly applied background knowledge can be used to extend the analogy further. Eventually, Phineas would extend its candidate theory into a form that can be tested, and proceeded to do so by using the combination of QPE and DATMI to see if the newly extended theory can explain the original observation.

We believe that Phineas provides a model for the use of analogy in learning, and indeed for the role of analogy in abduction tasks more generally. The least psychologically plausible part of Phineas' operation was its retrieval component, in which a domain-specific indexing vocabulary was used to filter candidate experiences (although it might be a reasonable model of expert retrieval). On the other hand, Phineas' map/analyze cycle and its method of using analogy in explanation and learning were, we believe, plausible in their broad features as a psychological model.

In addition to providing insights about the use of analogy in learning and abduction, our experience with Phineas led to several changes in SME itself:

- The need to match non-identical relational predicates in physical models led to a new way to relax the identicality constraint[1]. *Minimal ascension* (Falkenhainer, 1988) allows non-identical relational predicate matches when they would support a larger relational overlap and they have a close common superordinate.
- The map/analyze cycle involves incrementally extending a match, when new items are added to the base or target. In Phineas this ability was provided by mechanisms external to SME; since then, SME (v3) itself has been made incremental (Forbus, Ferguson, & Gentner, 1994).
- When constructing a new theory by analogy, the correspondences between entities suggested by the mapping between similar behaviors must be used to carry over the explanatory theory from the base to the target, to provide the starting point for the theory of the new domain. This experience was one of the factors that led to the introduction of pragmatic constraints (Forbus & Oblinger, 1990; Forbus *et al* 1994), a

[1] The *tiered identicality constraint* of structure-mapping states that the default criterion for matching relational statements is that the predicates involved be identical. Non-identical functions are allowed to match, since they provide cross-dimensional mappings. We are currently exploring predicate decomposition as another method to achieve partial matches among relations.

simple mechanism for task information to influence the mapping process by requiring or excluding correspondences.

- Although the Phineas examples all ran fine with the original SME exhaustive merge algorithm, plans to move to larger examples led to replacing the exhaustive merge algorithm with a linear-time greedy algorithm (Forbus & Oblinger,1990) that produces at most a handful of mappings.

Each of these innovations has been used in subsequent systems for larger-scale tasks. For instance, the ability to express constraints on correspondences is useful in reasoning from historical precedents. When reasoning about what a specific country might do in a crisis, one wants to find a precedent that is similar to the current situation, but in which the country in question maps to itself. More distant precedents might be considered, but are less preferred in this task than precedents involving the same country.

4 Example: Mental Models

An active area of research in cognitive science is studying *mental models* (Gentner & Stevens 1983), the models people use in reasoning about the physical world. Understanding mental models is a central issue for cognitive science because they appear important in reasoning about complex physical systems, in making and articulating predictions about the world, and in discovering causal explanations for what happens around us. Mental models research also offers practical benefits. In an increasingly technological society, understanding the nature of mental models for complex physical systems and could help people learn better models, which could reduce accidents and improve productivity (Norman, 1988).

A key intuition often associated with mental models is that they are *runnable*, i.e., there is a sense of deriving answers via mental simulation rather than logical reasoning. An appealing explanation for runnability is that mental model reasoning is like watching a movie of a physical system with your mind's eye. It does seem likely that spatial mental models rely in part on visual computations (e.g., Schwartz, 1999). However, we know of no evidence suggesting that the data needed for full quantitative simulation is available in common sense reasoning tasks, nor do we know of any evidence that people have a mental simulation facility capable of using such information. Consider predicting the pattern of liquid that will appear on a rug if a half-full cup of coffee is knocked off a table. Our visual apparatus is powerful enough to describe the shape of the patterns that result. However, we are not capable of predicting what specific shapes will result in advance. Solving this problem to a high degree of accuracy involves computational fluid dynamics; it seems quite unlikely that we are capable of performing such a prodigious feat mentally.

If we aren't watching a high-fidelity simulation, then what are we doing? One possibility is that we are doing some kind of simulation, but a *qualitative* simulation (de Kleer & Brown 1984; Forbus, 1984; Bredeweg & Schut 1991, White & Frederiksen 1990). Qualitative physics research has developed a variety of techniques for reasoning from first-principles about physical situations, even with very little information. The results of qualitative simulation involve high-level, conceptual descriptions of physical values and their changes, typically involving sign information (e.g., that a parameter is increasing, decreasing, or constant) and ordinal relationships between parameters (i.e., relative rates of flows, temperatures relative to phase transition boundaries). The systems of qualitative mathematics developed in this research succinctly capture the kinds of conceptual relationships people often use when describing interactions between continuous parameters (e.g., "heat rising causes temperature to rise").

While the representations of qualitative physics are appealing as models of the contents of human mental representations of commonsense physics, the qualitative simulation algorithms developed to date are problematic as models of human reasoning. Current qualitative simulation algorithms operate via first-principles reasoning over general-purpose axiomatic knowledge. They often produce a huge number of possible behaviors (hundreds or even thousands) even for relatively simple situations (Kuipers, 1994). The reason for this is that qualitative simulation, because of the decreased resolution of information about a state, tends to be ambiguous. In a quantitative simulation there is a unique next state. But in qualitative simulations there can be several next states, corresponding to different transitions that are logically consistent with the resolution of the qualitative state information. Each of these potential next states has several next states in turn, and so the number of simulation states in general grows exponentially. To be sure, there are some industrial applications where exploring every possible behavior - what is called *envisioning* (de Kleer, 1979) - is necessary. Unfortunately, this exponential behavior makes such algorithms seem psychologically implausible, given how easily people reason about everyday physical situations.

A second problem with first-principles qualitative simulation algorithms as models of human common sense reasoning is that their predictions tend to include a large number of spurious behaviors (Kuipers 1994), behaviors that logically follow from the low-resolution qualitative descriptions that they use as input, but are not in fact physically possible. In engineering applications, such behaviors are generally pruned by using more detailed knowledge (e.g., specific equations or numerical values). But that is not a viable option for modeling the common sense of the person on the street, who is capable of making reasonable predictions even without such detailed information.

We (Forbus & Gentner, 1997) suggest that the solution to this puzzle lies in our use of within-domain analogies (e.g., literal similarity) in common sense reasoning. We claim that a psychological account of qualitative reasoning should rely heavily on analogical reasoning in addition to reasoning from first principles. Qualitative predictions of behavior can be generated via analogical inference from prior observed behaviors, described qualitatively. Predictions based on experience reduce the problems of purely first-principles qualitative reasoning, because they are limited to what one has seen. The set of observed behaviors, barring misinterpretations, does not include physically or logically impossible occurrences. Predicted behaviors that are physically impossible might still be generated, since an experience might be applied to a situation containing differences that make it irrelevant, but there would still be many fewer of them than would be generated by a first-principles algorithm. Moreover, predictions from experience have the advantage of being more likely, since they have actually occurred, rather than simply being logically possible, which greatly reduces the number of predicted behaviors.

The fact that people store and remember behaviors of physical systems is uncontroversial. But how far can literal similarity go in explaining physical reasoning is still an open question. A major issue is generativity: How flexibly can past experiences be used to make new predictions, and especially predictions about novel systems and/or configurations?

We believe there are three factors that make memory-based reasoning more generative than some might otherwise expect. First, qualitative representations reduce differences. Assuming people store and use qualitative representations of situations and behavior, then two situations that vary only in quantitative details will look identical with respect to the qualitative aspect of their behavior. Second, analogical reasoning can generate predictions for novel situations. For common sense reasoning, within-domain analogies (i.e., predicting what will happen when pouring coffee into a cup based on previous experiences pouring coffee into a different cup) should provide a reliable guide to action. Third, multiple analogies can be used to piece together models for complex systems (Spiro *et al* 1989).

4.1 The hybrid similarity model of common sense reasoning

There is psychological evidence that the same comparison processes used for cross-domain analogical thinking are also used for within-domain comparisons, in tasks ranging from visual perception to conceptual change (Gentner & Markman, 1997). It would be surprising if such processes were not used in common sense physical reasoning. However, memory-based reasoning alone is insufficient to explain our ability to use general-purpose, domain-independent physical knowledge - something that we undeniably do, even if there is disagreement over how much of it people do routinely and under what circumstances. Consequently, we suspect that common sense reasoning arises from the interplay of analogical and first-principles reasoning.

The model we are creating differs from our previous model (Forbus & Gentner, 1986), in that we now suspect that the kinds of knowledge and processes that we previously divided into stages are actually tightly interwoven. Specifically, we now believe that comparison processes play a central role throughout the span of expertise. Our assumptions include

- The representational constructs of qualitative physics are psychologically plausible as part of the constituents of human mental models.
- People encode varying amounts of detailed information about the values of continuous properties, in addition to qualitative properties.
- People sometimes use domain-independent principles of qualitative reasoning and situation-independent general knowledge of particular domains.
- Much of people's knowledge is highly context-specific, i.e., that many principles of qualitative reasoning people use are domain specific, and that much of their knowledge about a domain is tied to situations or classes of situations within that domain.

This view is very different than the standard view in qualitative physics, where domain knowledge is completely generic. The best way to see the implications of this view is to consider a simple example: Pouring too much coffee into a cup, leading to it overflowing. Consider this sequence of states of knowledge about that kind of situation:

1. A remembered behavior concerning a specific cup at a specific time, e.g., more coffee pouring into your favorite cup leading to it flowing over the top and spilling on your desk. The behavior's description probably includes many concrete details, such as visual descriptions of the objects and their behaviors.

2. A remembered behavior concerning a specific cup at a specific time, including a causal attribution relating different factors or events, e.g., the overflow was caused by continuing to pour coffee once the cup was full. This attribution might come about by someone explaining the situation to you, or by analogy with an explanation given for another situation, or by the application of a more general principle. Additional qualitative relations might be included, such as blaming the overflow event on pouring a liquid, with the rate of overflow depending on the rate of pouring.
3. A generalization that coffee cups can overflow if you keep filling them up with liquid. This generalization might be formed by successive comparisons of very concrete situations, conservatively stripping away details that are not common across otherwise similar situations. Visual properties may be gone, but many aspects of the descriptions are still very concrete - coffee cups instead of containers, for instance, or even coffee instead of any liquid. More qualitative relationships may be included.
4. A generic domain theory of containers, liquids, and flow that supports limit analysis (Forbus, 1984), e.g., the coffee cup is a container, the coffee in it is a contained liquid, therefore one limit point in the quantity space for the level of the contained liquid is the height of the cup's top, and that a qualitative transition in behavior will occur when the level (which is rising due to being the destination of a liquid flow, which is the only thing happening that is affecting the amount of coffee in the cup) reaches the height of the top of the cup.

All of these states of knowledge can be used to make predictions. The first state of knowledge represents pure memory. Making a prediction with this kind of knowledge involves using the candidate inferences from a literal similarity match. The last state of knowledge represents the sort of explanation that would be generated by first-principles qualitative simulators. This state of knowledge supports reasoning purely from first principles, but this knowledge can also be used to explain new situations by analogical abduction, based on previous explanations.

The knowledge states in this sequence are samples from a continuum of knowledge about the physical world. The states in between represent what we suspect what might be very common in human mental models: intermediate levels of generalization and explanation, where partial explanations have been constructed in a conservative fashion (e.g., generalizing across liquids but still restricted to coffee cups). They are examples of what we could call *situated rules*, pieces of knowledge that are partially abstracted but still partially contextualized.

From an applications perspective, situated rules are the bane of good knowledge engineering practice. When engineering a domain theory, one strives for generality and broad coverage. In that context, the use of partially abstracted, partially contextualized knowledge represents a failure of analysis[2]. But the situations faced by knowledge engineers and by human learners are very different. Human learning is often initially conservative (Forbus & Gentner, 1987; Gentner & Medina, 1998; Medin & Ross, 1989). Situated rules provide an intermediate form of knowledge between concrete or slightly schematized descriptions of behaviors and the mechanism-based ontologies of standard qualitative physics.

We conjecture that situated rules are used to express principles of qualitative physics as well as knowledge about particular domains. That is, it seems likely that there is a range of knowledge about physical reasoning, varying from concrete, situated rules applicable to a small class of situations to the kinds of overarching, general principles encoded in performance-oriented qualitative reasoning systems. English-speakers commonly use the phrase “what goes up must come down”, and other language communities have similar expressions. How many of those speakers know that, assuming classical continuity, this statement implies the existence of an instant of time between going up and going down where the vertical velocity is zero? There is a large terrain between knowing nothing and having a broad-coverage general theory, and that terrain is not empty.

While we have not yet created a cognitive simulation of our hybrid analogical/first-principles account of qualitative reasoning, most of the components required have already been created. Let us see how these pieces might fit together to make predictions:

Let the input be a (partial) description of a physical situation. An augmented version of *generate and test* could be used to make predictions as follows:

1. Retrieve similar behaviors (using MAC/FAC). The candidate inferences from mapping these remembered behaviors onto the observed behavior provide additional expectations about the current situation, and hypotheses about the states to follow, based on what happened in the remembered behavior. The state transitions hypothesized in the candidate inferences form the initial set of predictions.
2. If qualitative simulation rules or procedures are available for generating new behaviors (either by association with this type of task or because they are retrieved by MAC/FAC along with the behaviors used in the previous step), use them to expand the set of predictions.
3. If qualitative simulation rules or procedures are available for evaluating the consistency of possible transitions (from the same sources as the previous step), use them to filter the set of predictions.
4. If there are multiple predictions remaining, estimate their relative likelihood. Return the best, or several, if others are close to the best.

The first step provides quick recognition of familiar behaviors. If the overlap with the current situation is high and the behavior predicted unique, processing may stop at this point, depending on task demands. The second step augments this recognition by domain-specific or first-principles consequences. The third step provides an opportunity

[2] See for example the *No function in structure* principle of de Kleer and Brown (1984), which is clearly violated by situated rules.

for applying exceptions and caveats ("if it were overflowing, you would see coffee coming down the outside of the cup" and "strong acid dissolves coffee cups"). In the fourth step, we suspect that a variety of methods are used to estimate relative likelihood, ranging from domain-specific knowledge ("filling a wax paper cup with hot coffee usually causes it to leak") to estimates of relative frequency based on accessibility in memory ("I've never seen a ceramic coffee cup shatter when it was filled").

Although much work lies ahead in exploring this hybrid qualitative simulation approach, it already illustrates how analogical processing ideas can be used to tackle larger-scale phenomena. The decomposition of analogical processing into units, and the existence of concrete models for those units, provides a vocabulary for analyzing more complex psychological phenomena.

5 Example: Case-based coaching

A second benefit of exploring analogy in the large is that it can lead to interesting new kinds of applications. These applications are both useful in their own right, and help establish the capabilities of the simulation software in real-world tasks. Such systems, as a whole, are typically not cognitive simulations: Performance, rather than fidelity, is their goal. However, such systems can still provide interesting insights for cognitive modeling in two ways. First, they can provide insights about the information processing requirements of a class of tasks, which in turn provides bounds on the required capabilities for people to carry out those tasks. Second, systems that produce explanations as part of their tasks must produce explanations that are considered plausible by their human users. This provides an information-level constraint on the knowledge that is used in a task.

One important application of analogy is *case-based coaching*. Good instructors typically use analogies and examples in helping learners master material. When a learner needs help in solving a problem, a coach might suggest that a specific principle or technique from an example might be relevant, and show the learner how it might be applied in their situation. We have used SME and MAC/FAC to create such a case-based coach, to help students who are learning engineering thermodynamics.

Engineering thermodynamics involves transformations of heat and work. It is the science underlying engines, power plants, refrigerators, and cryogenic systems. It is typically taught to college sophomores, and is one of the toughest courses in the engineering curriculum. We have developed software intended to help students learn engineering thermodynamics, by providing an *articulate virtual laboratory* (Forbus & Whalley, 1994) where students can safely tackle design projects in a scaffolded environment that provides explanations. The program, called CyclePad (Forbus, 1996; Forbus *et al* 1998), presents its users with a schematic interface on which they design their cycles by "wiring up" components (e.g., turbines, pumps) and making assumptions about numerical values (e.g., pressures, temperatures, flow rates) and properties of the components and the working fluid involved (e.g., choosing water versus some other substance, modeling a turbine as adiabatic, etc.). CyclePad uses AI techniques to derive the consequences of a student's assumptions, detect physically impossible combinations of assumptions, and provide explanations for its derivations on demand.

CyclePad is used by a large number of students at a variety of geographically distributed sites. We use a *distributed coach*, where part of the coaching software is embedded in CyclePad itself and part of the coaching services are provided via electronic mail. The "onboard" facilities are lightweight and well tuned, providing rapid feedback for students. The email facility, called the *CyclePad Guru*, enables us to experiment with different coaching techniques without requiring our users to download new software. This is especially important for case-based coaching, since cases often include a variety of media that can be centralized on a server rather than distributed *en masse* to each student.

The email-based coach works like this: An email dialog in CyclePad offers students several kinds of queries. For our purposes, the only relevant query is a request for design help, which invokes the case-based coach. Design help concerns how to improve the student's design, i.e., how one might increase the efficiency of a cycle or lower its operating costs. The student's query is sent via email to an agent colony, RoboTA (Forbus & Kuehne, 98), at Northwestern. The coach's response is sent back via email. Assuming the student's question made sense, this email contains up to two suggestions about how they might change their design to improve it. (If the student hasn't finished analyzing their design yet, or it is contradictory, this is pointed out to them instead.) These suggestions are based on MAC/FAC retrieving plausible entries from a library of design transformations, and using SME's candidate inference mechanism to generate specific advice about how to apply the transformation used in the case to the student's design. Let us see how this works.

The structural and teleological[3] aspects the student's design is used as a probe to MAC/FAC. (We found that numerical aspects of the student's design were irrelevant and hence we filter them out.) The cases in the MAC/FAC memory pool include a description of a design, a problem with that design, and a transformation that modifies the original design in a way that solves this problem. Each reminding includes the SME match created by the FAC stage for that reminding, which is the analogical match between that case and the student's design. A reminding might give rise to several suggestions, since SME can generate more than one mapping for a comparison. Each mapping potentially contains a suggestion. For example, a plan to improve turbine efficiency by increasing turbine inlet temperature is applicable in three different ways to a design that has three turbines. Recall that a candidate inference of a mapping is a statement in the base (here, the case) that is suggested by the correspondences of the mapping as possibly holding in the target (here, the student's design). Candidate inferences are the source of advice. Figure 2 shows the candidate inferences when the coach is reminded of reheat given a Rankine Cycle.

Suggestions are filtered in two ways to ensure that they are relevant. First, the candidate inferences must include the case's transformation - otherwise, there is no advice to give. Second, the candidate inferences must include a statement of the form

```
(implies <structural/functional properties of cycle>
        (applicable <plan of case>))
```

[3] Teleology refers to how a design achieves its intended function. CyclePad automatically derives teleological information from the structure of a student's design, ascribing a role to each component in the cycle (i.e., "this mixer is acting as a jet ejector") using a Bayesian recognition architecture (Everett, 1995, 1999.)

Each case is guaranteed to include a statement of this form due to the way cases are created, outlined below. The antecedents are exactly those properties that must be true for the case's transformation to make sense. For example, neither of the suggestions in Figure 2 would make sense for a cycle that didn't have heater. All of the antecedents for the implication must have correspondences in the suggestion's mapping, without skolems[4], in order for the case to be applicable to the student's design.

The suggestions generated by MAC/FAC are prioritized according to the complexity of the transformation they suggest, with simpler transformations being preferred, and on the structural quality of the candidate inference (Forbus et al 1997). At most two suggestions are selected to be used for generating advice, to avoid overloading the students. If MAC/FAC did not produce at least two suggestions, the case that it was reminded of is removed from the pool and MAC/FAC is run again, using the student's design as a probe like before. This iterative process continues at most *N* times (currently *N*=2), until the requisite number of suggestions are generated.

Advice is generated from each suggestion by a module that translates CyclePad's internal representations into understandable English. Figure 3 shows the advice generated from the candidate inferences in Figure 2. The advice generator treats structural transformations differently than other suggestions. This is because structural transformations must be followed in order to implement the suggestion, whereas the other assumptions may or may not be relevant to the student's particular situation. We believe that forcing the student to think carefully about which assumptions make sense is good exercise for them. In Figure 3, for example, it is suggested that the new components be made of molybdenum, which a student should recognize as unusual and expensive. If the suggestion is about modifying parameter values rather than structural changes, qualitative descriptions of relative changes in value are used in the advice. The advice also includes a URL describing the case in detail, using hypertext supplied by the case's author, encouraging the student to learn more by following up in the web-based design library.

New cases in the coach's design library are automatically generated by a *case compiler*. To add a case, instructors provide two snapshots of a CyclePad design, one before and one after their transformation. They also specify the goals of the transformation, in terms of changes in parameter values (i.e., what parameters must have increased or decreased), some strings to be used in templates, and a URL pointing to a detailed rationale for that case. While we insist that the web page for the case include an explanation of the case, this explanation is in natural language: case authors only need to be thermodynamics experts, not AI experts. The case compiler uses CyclePad to analyze the before and after design snapshot. It uses a record of user actions stored internally with each dumped design to construct the description of the transformation that leads from one to the other, and augments the case description with this plan, the problems it is intended to solve, and the applicability condition described above. (It also checks to ensure that the transformation actually achieves the claimed goals, since even experts can make mistakes.) Adding the new case to the MAC/FAC memory is simple, since no indexing is required: The structured representations needed to support reasoning also suffice for

[4] Skolem constants are introduced in candidate inferences when the projected base fact contains an entity that is not part of the mapping (Falkenhainer *et al* 1986, 1989). Each such entity gives rise to a skolem, indicating that some entity in the target domain must be identified (or postulated) to play its role before the candidate inference can be used.

retrieval, since the content vectors needed by the MAC stage are automatically computed from them.

Our case library currently consists of 14 cases, averaging 74 expressions involving 23 entities each. Retrieval and advice generation is very quick: less than five seconds on the average, with no more than six seconds at most, on a 200 MHz Pentium Pro. This performance comes from two factors. First, the MAC stage provides significant filtering, with only two or three cases proposed for processing by SME each time. Second, SME now uses a polynomial-time greedy algorithm in its merge step, making its overall complexity quadratic in the size of descriptions compared (Forbus et al 1994). The coach has been continuously available to CyclePad users since February 1998.

Our use of SME and MAC/FAC provides two significant advantages over the state of the art in case-based coaching. Most case-based coaching systems use very weak machine-understandable representations - often list of features - complemented by student-friendly media, such as video. This requires hand-encoding and indexing of every new case, a labor-intensive process that is a major cost in creating and maintaining such systems. Furthermore, the lack of rich formal representations (e.g., proofs or causal arguments) makes it impossible for a software coach to show a student just how the principle illustrated by the case might apply to their situation. Our use of MAC/FAC means that domain experts can use CyclePad and a case compiler to extend the design library without any knowledge of how CyclePad's representations operate nor any knowledge of how the retrieval software operates. Our use of SME enables us to ensure that the advice can actually be applied to the student's problem, and generate step-by-step instructions showing them how to do so. Thus MAC/FAC functions as a black-box retriever and SME functions as an explanation and inference generator.

While the CyclePad Guru is an application of analogical processing, it is interesting to see how its use of these models connects to psychological considerations. First, by using a psychologically motivated model of retrieval, we avoid the by-hand indexing that plagues most CBR systems, as noted earlier. Second, it may at first glance seem paradoxical that MAC/FAC produces good remindings, given the evidence on the difficulty of analogical retrievals. However, its behavior actually fits the psychological data quite well: The cases are in the same domain as the student's problems, so it is retrieving literal similarities, rather than cross-domain analogies. Moreover, its encoding processes for cases correspond more to experts than novices, which again is a factor that facilitates retrieval.

6 Scaling up: How Structure-mapping fares

These examples suggest that structure-mapping can go the distance: The ability to handle large, complex representations that are automatically generated to help with real problems is encouraging. While our work on hybrid qualitative simulation is still in progress, the clarity that can be achieved by casting larger-scale process models in terms of preexisting component process models suggests that this perspective is a valuable theoretical tool. The case-based coach suggests that structure-mapping can be effectively applied to complex, real-world problems.

Two other application efforts in progress are worthy of mention, because they give some indication of the scale that is involved in some of the tasks that people find straightforward. Under the DARPA High Performance Knowledge Base[5] program, we have been using SME and MAC/FAC to tackle two kinds of human reasoning: Reasoning from historical precedents about international crises, and critiquing battlefield courses of action by analogy. These are tasks that experts do daily, and that novices can understand the results of easily (even if they have trouble solving the problem by themselves).

These complex real-world tasks provide interesting data points concerning the size of descriptions that need to be handled for scaling up to handle the breadth of human analogical reasoning. In crisis management, we worked with the two competing teams, one led by SAIC and the other by Teknowledge/Cycorp. Each team had its own large knowledge base, and did all of the encoding of historical case information and associated background knowledge about countries. Cases tended to be hundreds of propositions in size, with some being well over 2,000 propositions. In the battlefield critiques, the knowledge base was built by a collaborative team, with the contents of cases being created by drafting tools and natural language parsing tools that produced representations, which were on the order of 1,000 propositions each. While these descriptions were rich enough to (mostly) handle the problems thrown at them, we suspect that people would be able to reason with far more knowledge than these systems would, given the same inputs. In crisis management, where a formal evaluation was performed, our average score was 2.3 out of 3, which was quite respectable given the difficulty of the problems[6].

Tackling problems of this magnitude led us to develop some extensions to SME. The most interesting of these are *automatic case extraction* and *dynamic case expansion*. With automatic case extraction, the propositions that constitute a case are drawn from a larger knowledge base according to properties of the task, rather than being pre-packaged as a combined experience. While our particular implementation of this operation is not intended to be psychologically plausible (i.e., pattern-matching and backward-chaining reasoning), how dynamically determined are the descriptions used in analogical reasoning, and by what means, is an interesting issue worth exploring. With dynamic case expansion, incremental additions are made to the contents of a case based on the progress of the match so far. This is useful in exploring complex analogies. For example, the SAIC Year Two scenario description consisted of 2,341 propositions and the SAIC Operation Desert Shield description consisted of 2,159 propositions. Directly matching both of these descriptions leads to creating hundreds of thousands of match hypotheses. But by starting with a “small” initial subset (778 versus 663 propositions), the match could be grown to include the relevant subset of the information (ultimately 1,513 by 1,605 propositions) in a tractable manner.

The Phineas experience, our first large-scale analogical processing simulation, led to substantial changes in SME. The HPKB experience, aside from performance tuning, has led to fewer internal changes in SME, but significantly expanded the ways it can interact with large knowledge bases. We find this relative stability encouraging: It suggests that SME can evolve to go the distance.

7 The Integration Constraint Revisited

[5] http://projects.teknowledge.com/HPKB/

[6] The specifications for the challenge problems can be found at http://www.iet.com/Projects/HPKB/.

Now that we have seen some examples of large-scale analogical processing let us return to the integration constraint and consider what lessons we might learn, in order to improve simulation models of analogical processing.

Processes need to operate in a range of domains. Modeling human performance in a domain naturally requires assumptions about how that domain is represented. Domain-specific representation assumptions are often core components of a theory, as in the assumptions about qualitative representations made in Section 4. But models of similarity processes, to capture their ability to operate across multiple domains, need to avoid domain-specific assumptions. For example, CopyCat (Hofstadter & Mitchell, 1994) and TableTop (French, 1994) each provide suggestive models of similarity for a specific domain and task. Unfortunately, their use of hand-coded domain-specific correspondences and matching algorithms means that for each new domain a new system must be created (Forbus *et al* 1997). This makes it hard to see how these principles might generalize to other arenas.

Systems must be able to scale up. As Sections 5 and 6 indicate, real-world analogical processing tasks often involve descriptions that are much larger than the traditional test cases for analogy simulations. Scaling up to larger descriptions and larger memories is a serious challenge. ARCS (Thagard *et al* 1990), for example, worked with small examples, but fell apart on larger memories[7] (Forbus, Gentner, & Law, 1994). IAM's (Keane, 1990, 1997) purely serial mapping strategy can be led astray by syntactic garden paths and miss larger potential matches, a problem that will likely be more severe as the size of descriptions grows.

Candidate inferences must be structurally sound. To produce coherent advice, for example, in suggesting to a student how to improve their design, the candidate inferences of an analogy need to follow the 1:1 constraint and parallel connectivity constraint. ACME's ability to produce many-to-one mappings, while intuitively appealing in some ways, leads to candidate inferences that are psychologically implausible (Markman, 1997).

Processes must have realistic resource requirements. For example, SME originally produced an exhaustive enumeration of mappings. Although convenient for studying the possible outcomes of an analogy, the idea of judging among dozens of mappings is psychologically highly implausible, and the factorial worst-case computational complexity even more so (Falkenhainer *et al* 1989). This caused us to replace the exhaustive merge algorithm with a greedy algorithm that produces only a few intepretations (Forbus & Oblinger, 1990). A good example in this regard is LISA (Hummel & Holyoak, 1997), an evolving model which places a priority on operating under strong working memory constraints – a design choice that makes scaling up more of a challenge.

Tailorability in processing should be avoided. Hand-tuning an algorithm for specific examples reduces its explanatory power and compromises its ability to be integrated into larger-scale models. For example, central to LISA is its use of serial activation – the pattern of which is currently set by hand.

---

[7] This was due in part to the normalization algorithm used in the winner-take-all network valuing absolute size over relative fit (Holyoak, personal communication). As (Forbus, Gentner and Law, 1994) report, the ARCS algorithm could also be improved by replacing ARCS' semantic similarity constraint with structure-mapping's tiered identicality constraint.

Constraints can emerge from the interaction of processes. Gentner & Clement (1988) argued that the best way to capture the influence of goals and task-specific constraints on analogy is in terms of their role in setting up the inputs and evaluating the outputs of a fixed analogy engine. The use of automatic case extraction in crisis management and the use evaluation of candidate inferences in the case-based coach provide supporting evidence for this argument[8]. By contrast, ACME's mingling of semantic similarity, pragmatic relevance and structural consistency within the mapping engine itself led to the problematic many-to-one mapping described above. On the other hand, we have found that very limited forms of "advice" to the mapping engine, based on task constraints (e.g., requiring or excluding correspondences (Forbus & Oblinger, 1990)) can be valuable in real problems.

These lessons are sobering: Clearly we are a long way from achieving models that fully satisfy the Integration Constraint. However, it is equally clear that the field is making significant progress, and there is every reason to believe that this progress will continue.

8 Discussion

My suggestion in this paper has been that the basic mechanisms of comparison and retrieval are understood well enough that more of the field's energy should be focused on what might be called *large-scale analogical processing*, that is, exploring the roles analogical processing plays in other cognitive processes. This has always been a central concern of cognitive psychologists, of course. My argument is that now it should become a central concern for cognitive simulation as well. Creating larger-scale simulations of cognitive phenomena is an exciting new frontier, opened up by the combination of progress in Cognitive Science and the Moore's Law expansion of available computational power. Exploring this frontier leaves behind the stultifying world of microworlds that provide no general insights and wind-up toy models that cannot scale up to realistic phenomena. This challenge only makes sense now because research on cognitive simulation of analogy has climbed up past the foothills of the phenomena and, from the ridge we are standing on, can now see new peaks, representing new phenomena to explore.

There is of course a second frontier opening up: Progress in neuroscience has lead to the intriguing possibility of modeling analogical processing on the microscale, so to speak. This complementary direction has its own promises and perils that others can speak of more knowledgeably than I. Happily, the solid consensus that has been achieved at the information and algorithm levels of understanding (in the sense of Marr (1982)) provides a canvas that explorers in both directions can contribute results to.

New frontiers are always difficult, and inertia is a powerful force. However, for progress to continue, I believe that scaling up is crucial. The most serious danger in foregoing investigations of analogy in the large is intellectual myopia. All too often, researchers are tempted to keep tuning the microstructure of a simulation and arguing over fine details, ignoring the fact that their simulation cannot possibly scale to handle kinds of cognitive processing that human beings clearly do. Exploring the microstructure

[8] The crisis management system also involved the evaluation of candidate inferences, and task constraints are built into the case compiler used with the case-based coach, but these were not mentioned here for brevity.

of the processes of mapping and retrieval are still important enterprises, to be sure, but they cannot continue to dominate analogy research. Arguably, they will not even be the major source of constraints on models in the foreseeable future: Given what we have learned so far about how analogy and similarity work, I believe that we will learn more by exploring their roles in other processes than by exclusively focusing on them in isolation. Simulating the use of mapping in discourse (c.f., Boronat & Gentner, in preparation), the roles of analogy in argumentation (c.f. Spellman & Holyoak, 1992), and the use of analogy in conceptual change and scientific discovery (c.f. Gentner *et al* 1997) are all examples of simulation challenges that will help us discover much more about the nature of analogy and similarity. There is a mountain range of phenomena to be explored, and it will not be explored by playing "King of the Foothill".

See you in the mountains.

9 Acknowledgements

This work has been carried out in collaboration with Dedre Gentner. The case-based coaching work described here has been done in collaboration with John Everett, Mike Brokowski, Leo Ureel, Sven Kuehne, and Julie Baher. This research is supported by the Cognitive Science and Artificial Intelligence Programs of the Office of Naval Research. Research on CyclePad was supported by the Applications of Advanced Technology program of the National Science Foundation. Valuable discussions with Dedre Gentner, Ron Ferguson, Sven Kuehne, Art Markman, Keith Holyoak, and John Hummel helped shape this paper.

10 References

Boronat, C., & Gentner, D. (in preparation). Metaphors are (sometimes) processed as generative domain mappings.

Bredeweg, B., and Schut, C. 1991. Cognitive plausibility of a conceptual framework for modeling problem solving expertise. *Proceedings of the 13th Annual Conference of the Cognitive Science Society*, 473-479. Hillsdale, New Jersey: Lawrence Erlbaum.

Clement, C. A., & Gentner, D. (1991). Systematicity as a selection constraint in analogical mapping. *Cognitive Science, 15,* 89-132.

de Kleer, J. (1979). The origin and resolution of ambiguities in causal arguments. *Proceedings of the Sixth International Joint Conference on Artificial Intelligence*, 197-203.

de Kleer, J. and Brown, J.S. 1984. A qualitative physics based on confluences. *Artificial Intelligence*, 24: 7-83.

Decoste, D. 1991. Dynamic across-time measurement interpretation. *Artificial Intelligence*, **51**(1-3), pp 273-341.

Everett, J. O. 1995. A Theory of Mapping from Structure to Function Applied to Engineering Domains. 14th International Joint Conference on Artificial Intelligence, Montreal, Morgan Kaufmann.

Everett, J.O. (1999). Topological Inference of Teleology: Deriving Function from Structure via Evidential Reasoning. *Artificial Intelligence*.

Falkenhainer, B., Forbus, K., and Gentner, D. (1986, August) The Structure-Mapping Engine. *Proceedings of AAAI-86*, Philadelphia, PA

Falkenhainer, B., Forbus, K., Gentner, D. (1989) The Structure-Mapping Engine: Algorithm and examples. *Artificial Intelligence*, 41, pp 1-63.

Ferguson, R. W. (1994). MAGI: Analogy-based encoding using symmetry and regularity. In A. Ram & K. Eiselt (Eds.), *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society* (pp. 283-288). Atlanta, GA: Lawrence Erlbaum Associates.

Ferguson, R. W., Aminoff, A., & Gentner, D. (1996). Modeling qualitative differences in symmetry judgments, *Proceedings of the Eighteenth Annual Conference of the Cognitive Science Society* . Hillsdale, NJ: Lawrence Erlbaum Associates.

Forbus, K. D. (1984). Qualitative process theory. *Artificial Intelligence*, 24(1), 85-168.

Forbus, K. D. (1990). The qualitative process engine. In D. S. Weld and J. de Kleer (eds.), Readings in Qualitative Reasoning about Physical Systems. San Mateo, California: Morgan Kaufmann.

Forbus, K. Using qualitative physics to create articulate educational software. *IEEE Expert*, **12**(3), May/June 1997.

Forbus, K., Everett, J.O., Ureel, L., Brokowski, M., Baher, J., and Kuehne, S. (1998) Distributed Coaching for an Intelligent Learning Environment. *Proceedings of the Twelfth International Workshop on Qualitative Reasoning*. Cape Cod, Mass. (Proceedings available as a AAAI Technical Report)

Forbus, K., Ferguson, R. and Gentner, D. (1994) Incremental structure-mapping. *Proceedings of the Cognitive Science Society,* August.

Forbus, K. and Gentner, D. 1986. Learning Physical Domains: Towards a Theoretical Framework. in Michalski, R., Carbonell, J. and Mitchell, T. (Eds.) *Machine Learning: An Artificial Intelligence Approach, Volume 2*, Morgan-Kauffmann.

Forbus, K., & Gentner, D. (1997). Qualitative mental models: Simulations or memories? *Proceedings of the Eleventh International Workshop on Qualitative Reasoning*, Cortona, Italy.

Forbus, K., Gentner, D. and Law, K. (1995) MAC/FAC: A model of Similarity-based Retrieval. *Cognitive Science*, 19(2), April-June, pp 141-205.

Forbus, K. and Kuehne, S. 1998. RoboTA: An agent colony architecture for supporting education. *Proceedings of Agents 98*.

Forbus, K. and Whalley, P. (1994) Using qualitative physics to build articulate software for thermodynamics education. *Proceedings of AAAI-94*, Seattle.

French, R. M. (1995). The subtlety of similarity. Cambridge, MA: The MIT Press.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. Cognitive Science, 7, 155-170.

Gentner, D. (1988). Metaphor as structure mapping: The relational shift. *Child Development, 59,* 47-59.

Gentner, D. (1989). The mechanisms of analogical learning. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning* (pp. 199-241). London: Cambridge University Press. (Reprinted in *Knowledge acquisition and learning*, 1993, 673-694.)

Gentner, D., Brem, S., Ferguson, R. W., Wolff, P., Markman, A. B., & Forbus, K. D. (1997). Analogy and creativity in the works of Johannes Kepler. In T. B. Ward, S. M. Smith, & J. Vaid (Eds.), *Creative thought: An investigation of conceptual structures and processes* (pp. 403-459). Washington, DC: American Psychological Association.

Gentner, D., & Forbus, K. D. (1991, August). MAC/FAC: A model of similarity-based access and mapping. *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, Chicago, IL.

Gentner, D., & Holyoak, K. J. (1997). Reasoning and learning by analogy: Introduction. *American Psychologist, 52,* 32-34.

Gentner, D., & Markman, A. B. (1995). Similarity is like analogy: Structural alignment in comparison. In C. Cacciari (Ed.), *Similarity in language, thought and perception* (pp. 111-147). Brussels: BREPOLS.

Gentner, D., & Markman, A. B. (1997). Structure mapping in analogy and similarity. *American Psychologist, 52,* 45-56. (To be reprinted in *Mind readings: Introductory selections on cognitive science,* by P. Thagard, Ed., MIT Press)

Gentner, D., & Rattermann, M. J. (1991). Language and the career of similarity. In S. A. Gelman & J. P. Brynes (Eds.), *Perspectives on thought and language: Interrelations in development* (pp. 225-277). London: Cambridge University Press.

Gentner, D., & Rattermann, M. J. (in press). Deep thinking in children: The case for knowledge change in analogical development. *Behavioral and Brain Sciences.*

Gentner, D., Rattermann, M. J., & Forbus, K. D. (1993). The roles of similarity in transfer: Separating retrieval from inferential soundness. *Cognitive Psychology, 25*, 524-575.

Gentner, D., & Toupin, C. (1986). Systematicity and surface similarity in the development of analogy. *Cognitive Science, 10,* 277-300.

Goldstone, R. L. (1994). Influences of categorization on perceptual discrimination. *Journal of Experimental Psychology: General,* 123(2), 178-200.

Goldstone, R. L., & Medin, D. L. (1994). Time course of comparison. *Journal of Experimental Psychology: Learning, Memory and Cognition, 20*(1), 29-50.

Goswami, U. (1992). Analogical reasoning in children. Hillsdale, NJ: Erlbaum.

Goswami, U., & Brown, A. L. (1989). Melting chocolate and melting snowmen: Analogical reasoning and causal relations. *Cognition, 35,* 69-95.

Halford, G. S. (1987). A structure-mapping approach to cognitive development. The neo-Piagetian theories of cognitive development: Toward an interpretation [Special Issue]. *International Journal of Psychology, 22*(5-6)*,* 609-642.

Halford, G. S. (1993). *Children's understanding: The development of mental models.* Hillsdale, NJ: Erlbaum.

Hofstadter, D. R., & Mitchell, M. (1994). The Copycat project: A model of mental fluidity and analogy-making. In K. J. Holyoak & J. A. Barnden (Eds.), *Advances in connectionist and neural computation theory: Vol. 2. Analogical connections* (pp. 31-112). Norwood, NJ: Ablex.

Holyoak, K. J., & Koh, K. (1987). Surface and structural similarity in analogical transfer. *Memory & Cognition, 15,* 332-340.

Holyoak, K. J., & Thagard, P. (1989). Analogical mapping by constraint satisfaction. Cognitive Science, 13(3), 295-355.

Hummel, J. E., & Holyoak, K. J. (in press). LISA: A computational model of analogical inference and schema induction. *Psychological Review.*

Jones, R.M., Laird, J.E., Nielsen, P.E., Coulter, K.J., Kenny, P. and Koss, F.V. 1999. Automated intelligent pilots for combat flight simulation. *AI Magazine*, 20(1), Spring. Menlo Park, CA: American Association for Artificial Intelligence.

Keane, M. T. (1990). Incremental analogising: Theory & model. In K. J. Gilhooly, M. T. G. Keane, R. H. Logie, & G. Erdos (Eds.), *Lines of thinking* (Vol. 1, pp. XX). Chichester, England: Wiley.

Keane, M. T. (1997). What makes an analogy difficult?: The effects of order and causal structure on analogical mapping. *Journal of Experimental Psychology: Learning, Memory and Cognition, 23*(4), 946-967.

Keane, M. T. (1996). On adaptation in analogy: Tests of pragmatic importance and adaptability in analogical problem solving. *The Quarterly Journal of Experimental Psychology, 49/A*(4), 1062-1085.

Kotovsky, L., & Gentner, D. (1996). Comparison and categorization in the development of relational similarity. *Child Development, 67,* 2797-2822.

Kuipers, B. 1994. *Qualitative Reasoning: Modeling and simulation with incomplete knowledge*. Cambridge, Mass.: MIT Press.

Lassaline, M. E. (1996). Structural alignment in induction and similarity. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 22*(3), 754-770.

Markman, A. B. (1997). Constraints on analogical inference. *Cognitive Science, 21*(4), 373-418.

Markman, A. B., & Gentner, D. (1993). Structural alignment during similarity comparisons. *Cognitive Psychology, 25*, 431-467.

Marr, D. (1982). Vision: A computational investigation into the human representation and processing of visual information. San Francisco, CA: Freeman.

Medin, D. L., Goldstone, R.L., and Gentner, D. (1993) Respects for similarity. *Psychological Review, 100*(2), 254-278.

Schwartz, D.L. (1999) Physical imagery: Kinematic versus dynamic models. *Cognitive Psychology, 38,* 433-464.

Spellman B. A., & Holyoak, K. J. (1992). If Saddam is Hitler then who is George Bush?: Analogical mapping between systems of social roles. *Journal of Personality and Social Psychology, 62*(6)*,* 913-933.

Rattermann, M. J., & Gentner, D. (1998). The effect of language on similarity: The use of relational labels improves young children's performance in a mapping task. In K. Holyoak, D. Gentner, & B. Kokinov (Eds.), *Advances in analogy research: Integration of theory & data from the cognitive, computational, and neural sciences* (pp. 274–282). Sophia: New Bulgarian University.

Ross, B. H. (1989). Distinguishing types of superficial similarities: Different effects on the access and use of earlier problems. Journal of Experimental Psychology: Learning, Memory and Cognition, 15(3), 456-468

Spellman, B.A, & Holyoak, K.J. (1996). Pragmatics in analogical mapping*.* *Cognitive Psychology, 31*, 307-346.

Spiro, R.J., Feltovich, P.J., Coulson, R.L., & Anderson, D.K. 1989. Multiple analogies for complex concepts: Antidotes for analogy-induced misconception in advanced knowledge acquisition. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning* (pp 498-531). Cambridge, MA: Cambridge University Press.

Thagard, P., Holyoak, K. J., Nelson, G., & Gochfeld, D. (1990). Analog retrieval by constraint satisfaction. Artificial Intelligence, 46, 259-310.

White, B. & Frederiksen, J. 1990. Causal model progressions as a foundation for intelligent learning environments. *Artificial Intelligence*, **42**, 99-157.

Forbus, K. 2000. Exploring analogy in the large. In Gentner, D., Holyoak, K. and Kokinov, B. (Eds) *Analogy: Perspectives from Cognitive Science*. Cambridge, MA: MIT Press.

Wu, M., & Gentner, D. (1998). Structure in category-based induction. Proceedings of the Twentieth Annual Conference of the Cognitive Science Society, 1154-1158.

11 Figures

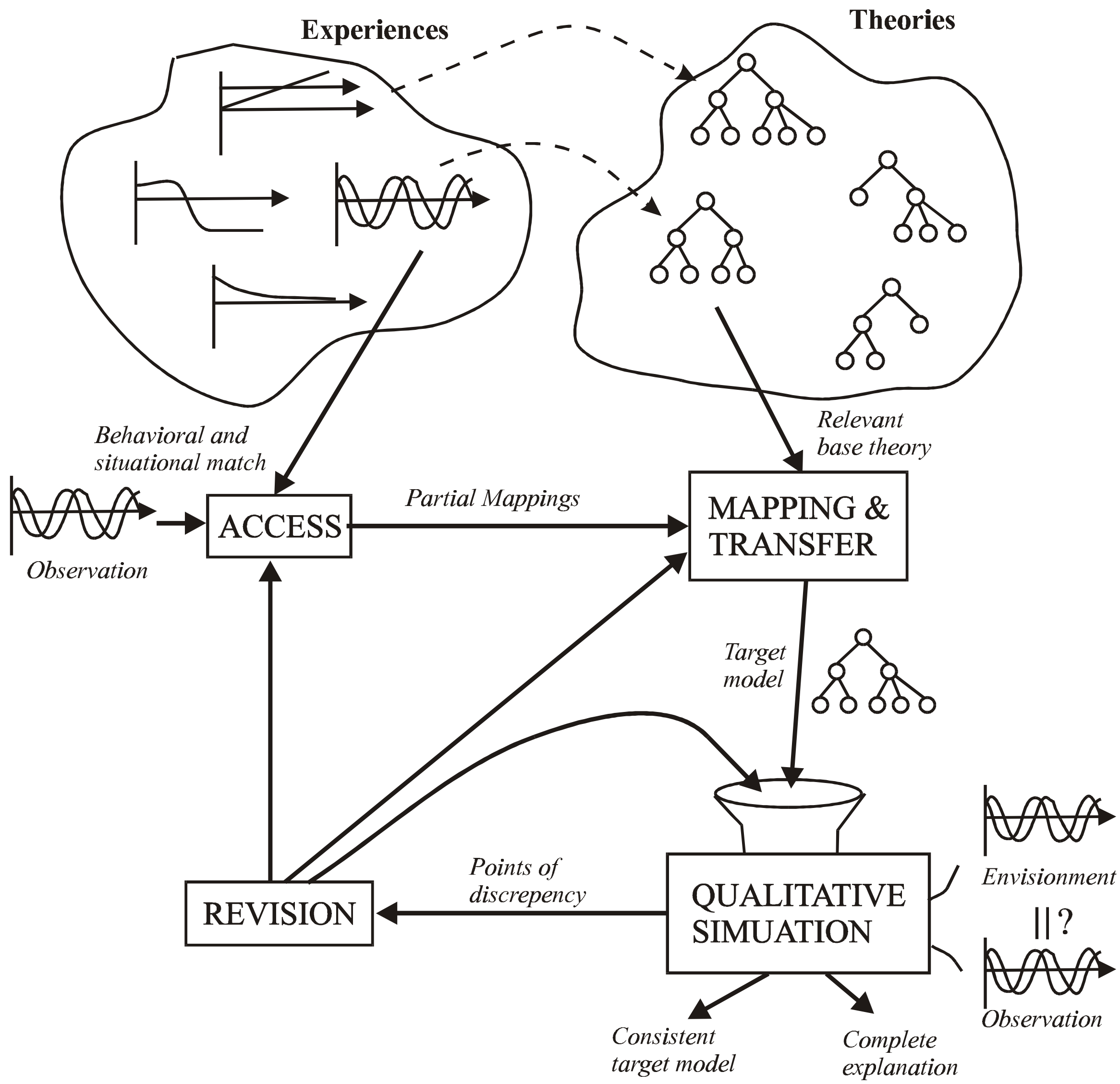

Figure 1: The architecture of Phineas, a system that used structure-mapping to learn qualitative mental models of physical domains.

```
Suggestions for <Desc WM of Rankine Cycle>:
Suggestion <Use INCREASE-RANKINE-BOILER-T>:
   1 step
   support=.085 extrapolation = 0.66
   normalized = 0.45 overlap = .408
   combined = .944
<Mapping 153 Candidate Inferences>
(BOILER htr1)
(CONDENSER clr1)
(IMPLIES (AND (TURBINE tur1 s2 s3)
              (HEATER htr1 s1 s2))
   (APPLICABLE (:SKOLEM :dsn-tr)))
(TRANSFORMATION-OF (:SKOLEM :dsn-tr)
      (STEPS (ASSIGN (T s2) (:SKOLEM :+))))
Suggestion <Use REHEAT-RANKINE-CYCLE>:
   16 steps
   support=0.03 extrapolation = .846
   normalized = .404 overlap = .134
   combined = .567
<Mapping 172 Candidate Inferences>
(BOILER htr1)
(CONDENSER clr1)
(IMPLIES (AND (TURBINE tur1 s2 s3)
              (COOLER clr1 s3 s4))
   (APPLICABLE (:SKOLEM :dsn-tr)))
(TRANSFORMATION-OF (:SKOLEM :dsn-tr)
   (STEPS (DISCONNECT (OUT tur1) (IN clr1) s3)
      (INSERT-DEVICE (:SKOLEM heater)
                     (:SKOLEM htr2))
      (CONNECT (OUT tur1) (IN (:SKOLEM htr2))
               (:SKOLEM s5))
      (INSERT-DEVICE (:SKOLEM turbine)
                     (:SKOLEM tur2))
      (CONNECT (OUT (:SKOLEM htr2))
               (IN (:SKOLEM tur2))
               (:SKOLEM s6))
      (CONNECT (OUT (:SKOLEM tur2)) (IN clr1)
               (:SKOLEM s7))
      (INVOKE-ASN (SATURATED (:SKOLEM s5)))
      (ASSIGN (DRYNESS (:SKOLEM s5))
              (:SKOLEM 1.0))
      (INVOKE-ASN (REHEATER (:SKOLEM htr2)))
      (INVOKE-ASN (ISOBARIC (:SKOLEM htr2)))
      (INVOKE-ASN (MATERIAL-OF (:SKOLEM htr2)
                  (:SKOLEM molybdenum)))
      (INVOKE-ASN (FUEL-OF (:SKOLEM htr2)
                  (:SKOLEM natural-gas)))
      (INVOKE-ASN (ISENTROPIC (:SKOLEM tur2)))
      (INVOKE-ASN (MATERIAL-OF (:SKOLEM tur2)
                  (:SKOLEM molybdenum)))
      (INVOKE-ASN (SATURATED (:SKOLEM s7)))
      (ASSIGN (DRYNESS (:SKOLEM s7)) (:SKOLEM 1))))
```

Figure 2: The suggestions generated by the CyclePad Guru, using MAC/FAC to retrieve cases that contain suggested design improvements. Cases are generated automatically by domain experts, using the learning environment and a case compiler.

```
   I have 2 suggestions.
   =============================================
   Suggestion #1
   Your problem reminds me of a method: increasing boiler temperature in a Rankine cycle.
Increasing the boiler temperature increases the efficiency of the cycle.
   You can find out more about this at
   http://www.qrg.ils.nwu.edu/thermo/design-library/turank.htm.
   Here is how you might apply this to your design:
   1. Increase T(S2).
   =============================================
   Suggestion #2
   Your problem reminds me of a method: reheat in a Rankine cycle.  Reheat adds another
heater and another turbine.  The second heater, a reheater, heats up the working fluid at
the turbine
   outlet, and the second turbine extracts yet more work from that.  This increases
efficiency because more heat is being added when the steam is still at a reasonably high
temperature.
   You can find out more about this at
   http://www.qrg.ils.nwu.edu/thermo/design-library/reheat.htm.

   Here is how you might do this with your design:

   1. Disconnect the outlet of TUR1 from the inlet of CLR1.
   2. Create a new heater, which we'll call HTR2.
   3. Connect the outlet of TUR1 to the inlet of HTR2.  Let's refer to the properties of
the working fluid there as S5.
   4. Create a new turbine, which we'll call TUR2.
   5. Connect the outlet of HTR2 to the inlet of TUR2.  Let's refer to the properties of
the working fluid there as S6.
   6. Connect the outlet of TUR2 to the inlet of CLR1.  Let's refer to the properties of
the working fluid there as S7.

   You might find the following assumptions relevant or useful:

   1. Assume that the working fluid at S5 is saturated.
   2. Assume quality(S5) = 1.0000[0-1]
   3. Assume that HTR2 is a reheater.
   4. Assume that HTR2 works isobarically.
   5. Assume that HTR2 is made of molybdenum.
   6. Assume that HTR2 burns natural-gas.
   7. Assume that TUR2 works isentropically.
   8. Assume that TUR2 is made of molybdenum.
   9. Assume that the working fluid at S7 is saturated.
   10. Assume quality(S7) = 1.0000[0-1]
   ===============================================
```

Figure 3: Advice for student generated from the suggestions of Figure 2. The step-by-step instructions on carrying out the transformation are generated from the candidate inferences generated during retrieval.