

Distributed Representations of Structure: A Theory of Analogical Access and Mapping

John E. Hummel and Keith J. Holyoak
University of California, Los Angeles

This article describes an integrated theory of analogical access and mapping, instantiated in a computational model called *LISA* (Learning and Inference with Schemas and Analogies). *LISA* represents predicates and objects as distributed patterns of activation that are dynamically bound into propositional structures, thereby achieving both the flexibility of a connectionist system and the structure sensitivity of a symbolic system. The model treats access and mapping as types of guided pattern classification, differing only in that mapping is augmented by a capacity to learn new correspondences. The resulting model simulates a wide range of empirical findings concerning human analogical access and mapping. *LISA* also has a number of inherent limitations, including capacity limits, that arise in human reasoning and suggests a specific computational account of these limitations. Extensions of this approach also account for analogical inference and schema induction.

A fundamental challenge for cognitive science is to understand the architecture that underlies human thinking. Two general properties of thinking jointly present extremely challenging design requirements. First, thinking is *structure sensitive*. Reasoning, problem solving, and learning (as well as language and vision) depend on a capacity to code and manipulate relational knowledge, with complex structures emerging from the systematic recombination of more primitive elements (Fodor & Pylyshyn, 1988). Second, thinking is *flexible* in the way in which knowledge is accessed and used. People apply old knowledge to new situations that are similar but by no means identical, somehow recognizing and exploiting useful partial matches. Both of these properties, structure sensitivity and flexibility, are apparent in the use of analogies (Gentner, 1983), schemas (Rumelhart, 1980), and rules (Anderson, 1983).

The first steps in analogical thinking are *access* and *mapping*. Access is the process of retrieving a familiar *source* analog (or schema, or rule) from memory given a novel *target* problem as a cue. Mapping is the process of discovering which elements in the target correspond to which in the source. For example, in the analogy between the atom and the solar system, the sun maps to the nucleus of the atom rather than to the electrons (Gentner, 1983). Once a source has been retrieved from memory and mapped onto the target, the former can be used to generate inferences about the latter; jointly, the two can be used to induce a more general schema that captures the essential properties

they have in common (Gick & Holyoak, 1983; Ross & Kennedy, 1990). Analogical access and mapping are both structure sensitive (guided by relational correspondences) and highly flexible (able to tolerate partial matches), although access is less structure sensitive than mapping (Gentner, Rattermann, & Forbus, 1993; Wharton et al., 1994).

This article presents a theory of analogical access and mapping motivated by the problem of simultaneously achieving both structure sensitivity and flexibility. Although our current focus is on problems surrounding analogical thought, our aim is to lay the groundwork for a more general theory of human thinking. Analogy provides a useful starting point for the development of general models of thinking for three reasons. First, analogy is representative of human thinking in that it is both structure sensitive and flexible. A model of analogy therefore must show how both of these key properties can simultaneously hold (Barnden, 1994). Second, the use of analogy is ubiquitous in human reasoning (Holyoak & Thagard, 1995) and provides a basis for the induction of complex relational knowledge such as schemas and rules (Holland, Holyoak, Nisbett, & Thagard, 1986). An understanding of analogy, therefore, may provide the foundation for a broader theory of human learning and inference. Third, a number of sophisticated computational models of analogy have been developed in recent years. Their strengths and weaknesses are particularly informative in the attempt to understand the relationship between structure and similarity in human thinking and provide a basis for evaluating any new theory of analogical reasoning.

John E. Hummel and Keith J. Holyoak, Department of Psychology,
University of California, Los Angeles.

Preparation of this article was supported by National Science Foundation Grant SBR-9511504.

We thank Graeme Halford, Art Markman, and an anonymous reviewer for their helpful comments on a previous draft.

Correspondence concerning this article should be addressed to John E. Hummel, Department of Psychology, University of California, Los Angeles, California 90095-1563. Electronic mail may be sent via Internet to jhummel@lifesci.ucla.edu.

Structure Sensitivity and Flexibility in Computational Models of Cognition

The twin design requirements of structure sensitivity and flexibility have figured prominently in discussions of the contrast between symbolic and connectionist approaches to modeling human cognition. These approaches have a strikingly complementary pattern of apparent strengths and weaknesses (Barn-

den, 1994; Holyoak, 1991; Norman, 1986). Roughly, symbolic systems readily model structure sensitivity but often fail to demonstrate humanlike flexibility, whereas connectionist systems exhibit flexibility in pattern matching and generalization but have great difficulty in forming or manipulating structured representations. One approach to capitalizing on the strengths of both symbolic and connectionist models has been to develop hybrid models. For example, Holyoak and Thagard (1989) proposed a hybrid model of analogical mapping. Most hybrids combine symbolic knowledge representations with connectionist-style constraint satisfaction. However, although hybrid models have had considerable success in simulating important aspects of analogical processing, their architectures seem to represent a "marriage of convenience" between their symbolic and connectionist components, lacking a natural interface between the two. An alternative line of theoretical effort has focused on the development of more sophisticated connectionist representations that can code and manipulate structured knowledge (e.g., Elman, 1990; Hummel & Biederman, 1992; Pollack, 1990; Shastri & Ajjanagadde, 1993; Smolensky, 1990; Touretzky & Hinton, 1988).

The theory introduced in this article is based on the latter approach and takes the form of a structure-sensitive connectionist model. The model, embodied in a computer simulation called *LISA* (*Learning and Inference with Schemas and Analogies*), represents propositions (predicates and their arguments) as distributed patterns of activation over units representing semantic primitives. These representations have the flexibility and automatic generalization capacities associated with connectionist models. However, LISA departs from traditional connectionist models in that it actively (i.e., dynamically) binds these representations into propositional structures. The result is a system with the structure sensitivity of a symbolic system and the flexibility of a connectionist system. These representations—and the processes that act on them—naturally capture much of the flexibility and structure sensitivity of human cognition. However, this combination of strengths comes with costs: LISA has a number of inherent limitations, including capacity limits, sensitivity to the manner in which a problem is represented, and sensitivity to strategic factors such as the order in which elements of a problem are processed. A key theoretical claim is that similar limitations arise in human reasoning. LISA thus provides a computational account of many strengths and weaknesses of the human cognitive architecture.

As the model's name implies, our long-term goal is to account for human use of both analogies and more abstract schemas and to model the inductive learning that allows people to acquire abstractions from experience with concrete examples (e.g., Gick & Holyoak, 1983; Novick & Holyoak, 1991; Ross & Kennedy, 1990). This article focuses on the processes of retrieving analogs and schemas from long-term memory (i.e., analogical access) and performing structure-sensitive comparisons between two analogs (or between a schema and an analog) in working memory (i.e., analogical mapping). Although there are numerous models of analogical access and mapping in the literature (reviewed shortly), LISA is the first to unify these two processes, and it is the first to account for some complex asymmetries between them (as elaborated in the Simulation Results section). In addition, we consider analogical access and map-

ping to be foundational for developing a theory of schema induction.

We first review what is known about analogical access and mapping and the strengths and limitations of existing computational models in this area. The empirical evidence concerning human analogical reasoning provides standards by which to assess LISA's adequacy as a psychological theory; the limitations of current models in part motivate the novel representation and processing assumptions embodied in LISA. Next, we briefly consider alternative connectionist architectures that exhibit structure sensitivity and describe the rationale for the particular architecture on which LISA is based. We then describe LISA and report various examples of its operation. Finally, we consider the prospects for applying the LISA architecture to a wider range of cognitive phenomena, particularly inference generation and schema induction.

Analogical Access and Mapping

A great deal is now known about how adults access and use analogies and how analogical abilities develop over the course of childhood (for reviews, see Gentner, 1989; Goswami, 1992; Holyoak & Thagard, 1995; Keane, 1988; Reeves & Weisberg, 1994). These findings have motivated the development of a number of theories and computational models of analogical access and mapping (e.g., Falkenhainer, Forbus, & Gentner, 1989; Forbus, Gentner, & Law, 1995; Halford et al., 1994; Hofstadter & Mitchell, 1994; Holyoak & Thagard, 1989; Keane, Ledgeway, & Duff, 1994; Kokinov, 1994; Thagard, Holyoak, Nelson, & Gochfeld, 1990). These models differ in important ways, but they have converged in positing a few basic constraints that guide human intuitions about natural correspondences between the elements of source and target analogs. We describe these constraints in the terminology of Holyoak and Thagard's (1989, 1995) *multiconstraint* theory of analogical mapping.

Three broad classes of constraints, which overlap with those identified by other theorists (e.g., Gentner, 1983, 1989), form the basis of the multiconstraint theory.

1. The structural constraint of *isomorphism* has two components:
 - (a) *structural consistency* implies that source and target elements that correspond in one context should do so in all others, and
 - (b) *one-to-one mapping* implies that each element of one analog should have a unique correspondent in the other.
2. The constraint of *semantic similarity* implies that elements with some prior semantic similarity (e.g., joint membership in a taxonomic category) should tend to map to each other.
3. *Pragmatic centrality* implies that mapping should give preference to elements that are deemed especially important to goal attainment and should try to maintain correspondences that can be presumed on the basis of prior knowledge.

Each of these constraints is inherent in the operation of LISA. However, the design of LISA differs from that of previous models based on the multiconstraint theory in that it also honors additional cognitive constraints on representation and processing.

Table 1
*Empirical Phenomena for Evaluation of Models
of Analogical Access and Mapping*

Access and its relationship to mapping

1. Semantic similarity has greater impact than in mapping
2. Isomorphism has less impact than in mapping
3. Close analog and schema easier to access than far analog
4. Access is competitive
5. Familiar analog accessed more readily

Analogical mapping

6. Isomorphism
7. Semantic similarity
8. Pragmatic centrality
9. Multiple possible mappings for one analogy
10. Correct initial correspondence facilitates finding subsequent mappings
11. Difficulty finding mapping for "unnatural" analogy problems^a
12. Possible to map predicates with different numbers of arguments^a

Phylogenetic and ontogenetic change

13. Limited analogy ability exhibited by "language"-trained chimpanzees
14. Ability to process deeper and more complex analogies increases over childhood

^aThese criteria are considered plausible but lack direct empirical evidence.

Table 1 summarizes 14 interrelated empirical phenomena concerning analogical access and mapping. For convenience we divide these phenomena into three broad classes, which concern the relationship between access and mapping, detailed aspects of mapping, and phylogenetic and ontogenetic change in mapping ability.

Access and Its Relationship to Mapping

In general, analogical access (the process of retrieving one analog, usually a source, from memory when given another analog as a cue) appears to be sensitive to the same basic constraints as analogical mapping (the process of discovering the specific correspondences between the source and target). However, semantic similarity appears to have relatively greater impact on access than on mapping (Phenomenon 1), whereas isomorphism has a greater impact on mapping than access (Phenomenon 2; Gentner et al., 1993; Holyoak & Koh, 1987; Ross, 1987, 1989). When analogs must be cued from long-term memory (rather than simply being stated as part of an explicit mapping problem), then cases from a domain similar to that of the cue are retrieved much more readily than cases from remote domains (Keane, 1986; Seifert, McKoon, Abelson, & Ratcliffe, 1986). For example, Keane (1986, Experiment 1) measured retrieval of a *convergence* analog to Duncker's (1945) radiation problem (for which the key solution is for a doctor to apply multiple low-intensity rays simultaneously to a stomach tumor from different directions). The source analog was studied 1–3 days before presentation of the target radiation problem. Keane found that 88% of participants retrieved a source analog from the same domain (a story about a surgeon treating a brain tumor), whereas only 12% retrieved a source from a remote domain (a story about a general capturing a fortress). This differ-

ence in ease of access was dissociable from the ease of post-access mapping and transfer, as the frequency of generating the convergence solution to the radiation problem once the source analog was cued was high and equal (about 86%) regardless of whether the source analog was from the same or a different domain.

Access is also facilitated by learning conditions that encourage induction of an abstract schema from remote analogs (Brown, Kane, & Echols, 1986; Catrambone & Holyoak, 1989; Gick & Holyoak, 1983). For example, Catrambone and Holyoak (1989, Experiment 5) had college students read three convergence problems drawn from three distinct domains and then answer questions that highlighted the abstract structural commonalities among them (e.g., use of multiple small forces from different directions to achieve the effect of a single large force). After a 1-week delay, 74% of participants spontaneously generated an analogous convergence solution to the radiation problem. The overall pattern of findings concerning analogical access suggests that close analogs and schemas are accessed relatively easily, whereas access is considerably more difficult for remote analogs (Phenomenon 3).

Factors other than similarity and isomorphism also influence access. It has been shown that analogical access is inherently competitive (Phenomenon 4). For any cue, people are more likely to retrieve a case from long-term memory if it is the best match available (based on both structural and semantic constraints) than if some other stored case provides a better match (Wharton, et al., 1994; Wharton, Holyoak, & Lange, 1996). In addition (Phenomenon 5), highly familiar cases tend to be preferentially retrieved (even when the familiar case is less similar to the cue than are some alternative stored cases). A particularly well-established example is the prevalent use of the person analog by children to make inferences about other animals and plants (Inagaki & Hatano, 1987). It has also been shown that people understand new individuals by spontaneously relating them to significant others, such as a parent or close friend (Andersen, Glassman, Chen, & Cole, 1995).

Mapping

The role of isomorphism in mapping (Phenomenon 6) is apparent when people are able to find sensible relational correspondences in the absence of substantial similarity between mapped objects (e.g., Gick & Holyoak, 1980; Gentner & Gentner, 1983), or even when relational correspondences conflict with object similarity (Gentner & Toupin, 1986). Recent work has established that human similarity judgments are also sensitive to isomorphism, in that perceived similarity is increased by consistent role correspondences (Goldstone, 1994; Goldstone, Medin, & Gentner, 1991; Markman & Gentner, 1993a, 1993b; Medin, Goldstone, & Gentner, 1993). For example, common features contribute more to the perceived similarity of two patterns when they participate in similar relations in the patterns ("matches in place") than when they participate in different relations across the patterns ("matches out of place"; Goldstone & Medin, 1994). Semantic similarity (Phenomenon 7) reveals its influence in the greater ease of mapping when similar objects fill parallel roles than when objects and roles are "cross mapped" (Gentner & Toupin, 1986; Ross, 1987, 1989). It has

also been shown that increasing predicate similarity (when it converges with structural parallels) decreases the latency of finding structurally consistent mappings (Keane et al., 1994). The impact of pragmatic centrality (Phenomenon 8) is evident when mappings are ambiguous on the basis of structural and semantic constraints. In such cases, people tend to preferentially map objects and relations they deem important to their goal (Spellman & Holyoak, 1996).

It has also been established that different people will produce different, internally consistent mappings for the same analogy (Phenomenon 9; Burns, 1996; Spellman & Holyoak, 1992, 1996). That is, it is often the case that there is not just one correct mapping between two analogs. Much like the two incompatible visual interpretations of a Necker cube, people typically arrive at one interpretation of an ambiguous analogy (although they may be able to shift from one interpretation to another).

Analogical mapping is sensitive to order of processing. When people are led to map analogs incrementally (e.g., by mapping as they read the two analogs), then the overall accuracy of their object mappings is influenced by the order in which mappings are made. Keane and colleagues (Keane, 1995; Keane et al., 1994) have shown that for purely structural analogies, mapping is more accurate when the order in which the analogs are processed encourages a correct initial mapping, which can then constrain subsequent mappings (Phenomenon 10).

Phenomena 11 and 12 have a less firm empirical basis than the other entries in Table 1; however, each can be supported by informal observations and "thought experiments." We discuss these possible phenomena later in the context of evaluating current computational models of mapping.

Phylogenetic and Ontogenetic Change

Another major class of phenomena that must be accounted for by a theory of analogical reasoning concerns variations across species and across human development. Chimpanzees that have received training in an artificial "language" based on manipulable tokens are able to solve simple relational analogies, such as "can opener is to can as key is to ?," where the answer is "lock" (Gillan, Premack, & Woodruff, 1981; Premack, 1983). However, chimpanzees are apparently unable to solve analogies at the level of adult human competence (Premack, 1988), suggesting that species with basic analogy ability nonetheless differ in the upper bound on the complexity of mappings they can solve (Phenomenon 13). A similar pattern is apparent within the course of human cognitive development. Sensitivity to abstract relational similarities increases with age (Smith, 1989), and children's ability to comprehend deeper and more complex analogies increases at least up until 12 years of age (Phenomenon 14; e.g., Gentner, 1988; Goswami, 1989; Johnson, & Pascual-Leone, 1989). This developmental pattern is, doubtless, in part due to increases in children's knowledge; however, there is also evidence that the shift is in part due to maturational increases in the capacity of working memory (Halford, 1992, 1993).

These are by no means the only phenomena that can be used to assess models of analogy, but they provide a challenging set. (See Markman, Gentner, & Wisniewski, 1995, for an overlapping set of empirical criteria for model assessment.)

Limitations of Current Analogy Models

Although it seems fair to say that no existing analogy model unambiguously captures all of the phenomena discussed thus far, it also seems that none of these phenomena is clearly beyond the reach of current models or reasonable augmentations of them. Why, then, are we taking the tack of developing a very different type of cognitive architecture for analogy? Although there is no definitive basis for rejecting current models, a strong case can be made for considering alternative approaches.

Consider the general mode of operation of representative current models. The most general models of analogical access and mapping are two pairs of systems: SME (mapping) coupled with MAC/FAC (access; Falkenhainer et al., 1989; Forbus et al., 1995), and ACME (mapping) coupled with ARCS (access; Holyoak & Thagard, 1989; Thagard et al., 1990). These systems are broadly similar in that they take symbolic, propositional representations as inputs and perform complex symbolic operations to generate plausible sets of candidate mappings. ACME/ARCS uses a connectionist constraint-satisfaction algorithm to compute mappings; however, the input to this algorithm is a set of units representing all syntactically legal mapping possibilities, constructed by symbolic procedures operating on the propositional representations provided to the program. MAC/FAC includes a vector-processing stage in its retrieval algorithm, but the vector elements are high-level concepts without a detailed semantic representation.

In broad terms, these models have, at most, localist representations of the meaning of concepts (e.g., a semantic network in the case of ARCS), and most of their processing is performed on propositional representations unaccompanied by any more detailed level of conceptual representation (e.g., neither ACME nor SME includes any representation of the meaning of concepts). Although these properties may be viewed as gaps in implementation rather than strong theoretical tenets, it is unclear how either approach could be readily adapted to the use of more detailed representations of semantic content.

These general properties of current analogy models give rise to four general classes of doubts about their potential for extension.

Psychologically Implausible Working-Memory Requirements

First, although the models are quite successful in accounting for many aspects of human analogical competence (i.e., predicting what analogies people will retrieve and what mappings they will identify), their algorithms are difficult to reconcile with established properties of cognitive architecture (see Keane et al., 1994, for similar reservations). In particular, working-memory limitations have generally been ignored (although the STAR model of Halford et al., 1994, addresses the role of working-memory capacity in human analogical mapping). ACME implements the multiconstraint theory by using an algorithm for parallel constraint satisfaction in which all possible matches between source and target elements, and all constraints relevant to the selection of those matches, are considered simultaneously. SME performs mapping by a process of graph matching, which requires explicit formation and manipulation of propositional

graph structures. Both ACME and SME form explicit representations of very large numbers of possible local matches between elements of source and target analogs, most of which are then discarded as mapping proceeds. Such processes do not seem compatible with the generally accepted limits of working memory. ARCS (and to a lesser degree MAC/FAC) performs similar constructive operations in the course of accessing cases in long-term memory (i.e., before the to-be-retrieved cases have even entered working memory), which seems especially unrealistic as a psychological model.

The working-memory requirements for mapping can be reduced by using incremental algorithms to serialize processing (Keane et al., 1994; Forbus, Ferguson, & Gentner, 1994). However, the algorithms themselves do not necessitate operating within a limited working-memory capacity: As algorithms, parallel constraint satisfaction and graph matching are just as happy to operate on a 10,000-node network (or tree) as on a 10-node network (or tree). As a result, these algorithms provide no principled basis for estimating the maximum working-memory capacity available for mapping; that decision is left to the intuitions of the modeler. Typically, the estimate is large: Even the incremental algorithms require explicit formation of multiple graph structures to determine mapping order. Although the number of possible matches explicitly considered is reduced relative to nonincremental algorithms, it can nonetheless be quite large relative to reasonable estimates of human working-memory capacity.

Failure to Integrate Access and Mapping

A second limitation of current models is that they do not provide a graceful integration of analogical access and mapping. For example, for the purposes of access, ARCS uses a semantic network that is not available to ACME for the purposes of mapping. Similarly, MAC/FAC uses a vector representation for access that is not available to SME for mapping. If one assumes that access and mapping reflect different operations on the same stored knowledge, then it would seem desirable to understand how both processes could operate on a single basic representation of that knowledge. Such an understanding should help account for the known similarities and differences between access and mapping (Table 1, Part I) in terms of differences between the processes available to long-term memory (access) and those available to working memory (mapping).

Over- and Under-Powerful Mapping Performance

A third limitation of current models is related to the fact that they are premised on unrealistic assumptions about working-memory capacity: In some respects, these models are too good at analogical mapping. For example, both ACME and SME can map analogies of virtually any size or complexity with roughly equal facility. Human reasoning is both more limited and more domain specific. A physician, for instance, might understand a complex analogy between different organ systems or diseases yet fail to understand an analogy between, say, different computer operating systems, even if the latter problem is, in some sense, formally simpler. Extant models of analogical mapping do not directly explain the domain specificity of mapping ability

(Novick, 1992). In a similar vein, current models provide no clear linkage between variations in available working memory and analogical performance. In other words, they do not offer specific predictions about the manner in which performance will improve or degrade as available working memory increases (e.g., by maturation) or is reduced (e.g., by concurrent tasks or brain damage).

The excessive mapping power of some current models is related to Phenomenon 11 in Table 1. Although based more on informal observations than on rigorous data, it seems that people find some analogy problems to be highly "unnatural" and difficult to map. Current analogy models have remarkably little difficulty in solving some such unnatural problems. Table 2 presents an analogical mapping problem that exemplifies how ACME in particular may be too powerful relative to humans. In this "boys-dogs" problem, the task is to state the correspondences between three boys described by a total of three unary predicates and three dogs described by three different and semantically unrelated predicates. This problem was constructed by Holyoak and Thagard (1989) to illustrate ACME's ability to map semantically and pragmatically meaningless analogs solely on the basis of the structural constraint of isomorphism. Indeed, ACME's parallel constraint algorithm finds the unique structural solution (see Table 2) to this mapping problem just as easily as it finds the solution to many semantically meaningful problems.

It is possible for people to solve this problem. Holyoak and Thagard reported that 6 of 8 college students produced the six correct mappings within 3 min. Keane et al. (1994) found that participants could solve the problem in roughly 3–6 min when given feedback whenever they produced an erroneous mapping. However, the boys-dogs problem is intuitively a very unnatural analogy, and informal observation suggests that few, if any, people solve the problem by parallel constraint satisfaction as ACME does. Rather, people tend to consciously re-represent the analogs (noticing, for example, that Steve and Fido each have just one property, rather than two like the remaining individuals) and then explicitly step through the possible correspondences, solving them like a puzzle ("Hmm . . . if Steve is Fido, then . . ."). Such augmented representations, coupled with serial hypothesis testing (and in Keane et al.'s procedure, error correction by the experimenter), can eventually yield the solution. But the human solution process for this problem seems qualitatively very different from that of ACME, which finds all mappings in parallel by simultaneous considering all structural connections

Table 2
Semantically Empty Boys-Dogs Mapping Problem Based on Unary Predicates (Holyoak & Thagard, 1989)

"Boys" analog	"Dogs" analog
Smart (Bill)	Hungry (Rover)
Tall (Bill)	Friendly (Rover)
Smart (Steve)	Hungry (Fido)
Timid (Tom)	Frisky (Blackie)
Tall (Tom)	Friendly (Blackie)

Isomorphic solution: Bill→Rover, Steve→Fido, Tom→Blackie,
smart→hungry, timid→frisky, tall→friendly

without re-representing the analogs or receiving error correction. The human solution to the boys-dogs problem also seems qualitatively different from the human solution to more natural problems (such as the fortress story and the radiation problem; Gick & Holyoak, 1980; or the Persian Gulf War and World War II; Spellman & Holyoak, 1992). People appear to use less labored methods to map such semantically rich analogies, even when they are structurally more complex than the artificial boys-dogs problem. Although the evidence is only suggestive, there is reason to suspect that people are sensitive to differences in the naturalness of analogies that are not reflected in the operation of current models based on massively parallel constraint satisfaction, such as ACME.

Even though current models are, arguably, overpowerful relative to people in some respects, they may be underpowerful in other respects. In particular, all current models share an inviolable structural constraint: A predicate with n arguments can only map to another predicate with n arguments. This " n -ary restriction" is basic to the operation of all current mapping models; without it, the number of syntactically legal mappings that would need to be explicitly represented would greatly increase, and the mechanisms the models use to enforce isomorphism would be compromised. Nevertheless, the n -ary restriction is psychologically questionable. For example, it precludes mapping the following analogs:

Target	Source
tall (Abe)	taller than (Chris Dean)
short (Bill)	

As a thought experiment, however, it is compelling that Abe maps to Chris and Bill maps to Dean. In a similar way, it has often been argued that people represent transitive orderings by mapping from relational premises to some sort of mental array (e.g., DeSoto, London, & Handel, 1965; Huttenlocher, 1968). Suppose, then, that a person attempts to map binary ordered pairs from a set of three objects onto a three-place ordered array. This mapping problem would have a form such as:

Target	Source
taller than (Abe Bill)	top-to-bottom (top middle bottom)
taller than (Bill Charles)	

Because of the n -ary restriction, no current model of analogical mapping could map the people into the array on the basis of height. One might attempt to account for the human ability to find such correspondences by assuming that one analog is represented before mapping to match the syntactic form of the other analog. It is unclear, however, how such a re-representation process would actually operate. If one provisionally accepts that mappings between predicates with unequal numbers of arguments are sometimes psychologically natural (Phenomenon 12 in Table 1), then it follows that current models are not only too profligate in explicitly generating all syntactically legal mappings but also too restrictive in that they define some sensible mappings as syntactically illegal.

We believe that current mapping models are overdependent on the n -ary restriction for a basic reason: They treat predicate-argument structures as passive data structures equivalent to lists, using list position as a simple and inviolable cue to possible

matches between fillers of argument slots (i.e., first argument to first argument, second argument to second argument, etc.). In contrast, LISA represents predicate-argument structures as collections of activation patterns distributed over units representing semantic features. The list position of an argument has no meaning in this type of representation, and symbolic list-matching operations are unavailable to LISA. As we will demonstrate, one consequence of this fundamentally different approach to representing and processing structure is that LISA is able to map examples that violate the n -ary restriction. Thus, although in some respects LISA's mapping ability is less powerful than that of current models (because of its limited working memory), in other respects LISA has greater mapping ability because of the flexibility afforded by its distributed representations of meaning.

Extendability to Schema Induction

The fourth general source of doubt about current analogy models concerns their extendibility to account for aspects of learning and creativity—capacities that are commonly linked to analogical reasoning (Gentner, 1989; Hofstadter & Mitchell, 1994; Holland, Holyoak, Nisbett, & Thagard, 1986; Holyoak & Thagard, 1995). On the face of it, current models have provided a key prerequisite for modeling complex relational schema induction. Structure-based correspondences between elements of the source and target analogs (the output of analogical mapping) provide information about what elements to generalize over and should therefore provide essential input to a generalization mechanism. However, the apparent connection between analogy and schema induction, although widely recognized, has generally not been computationally realized (however, see Falkenhainer, 1990, for an effort in this direction). Some models can use computed mappings to generate inferences specific to the target (e.g., Falkenhainer et al., 1989; Hofstadter & Mitchell, 1994; Holyoak, Novick, & Melz, 1994), but even these models cannot form new abstractions (either complex relational schemas or new predicates and concepts). For example, the Copycat model of Hofstadter and Mitchell (1994), formulated with the explicit goal of modeling creative aspects of analogy, in fact lacks any capacity to learn. After laboring to retrieve and map analogs, most models simply throw away the knowledge thereby acquired. As a result, the analogical episode has no consequences for future processing of related problems.

The apparent difficulty in building abstractions using current analogy models is at least partly attributable to their lack of detailed semantic representations of concepts. It is not obvious, however, how to adapt current mapping algorithms to use distributed representations of concept meaning, especially given the role of structure in analogical mapping. Purely localist representations work well for the purposes of analogical mapping but are inadequate for generalization and building abstractions (basic components of schema induction). Thus the representational requirements of most current mapping models may be fundamentally incompatible with the representational requirements of schema induction (see Hummel & Holyoak, in press).

Summary: Justifying a New Approach

In summary, we do not claim it is impossible to extend or modify current analogy models to make them compatible with

human working-memory limits, to unify analogical access and mapping, to more closely match the limitations and strengths of human mapping, or to account for the learning of new abstractions. However, we do argue that the hurdles facing current models are sufficiently daunting as to justify serious consideration of alternative approaches. To the extent that these hurdles are related to one another—as we have argued they are—there is particular utility in seeking a new approach to modeling analogical processes. A unified approach that overcomes all of these limitations would be preferable to a traditional approach augmented with a patchwork of specific fixes.

Given the widely recognized links between analogy and other mental operations, a psychologically realistic model of analogical mapping might also be expected to bear a resemblance to models of other cognitive processes. As detailed in the Discussion, LISA bears more than a passing resemblance to some models of object perception and recognition (Hummel & Biederman, 1992; Hummel & Stankiewicz, 1996). We argue that these resemblances are not coincidental but rather reflect important properties of a general solution to the problem of representing and processing structure—properties, we argue, that are shared by the human cognitive apparatus.

Distributing Structure Over Time

Human analogical reasoning (as well as other types of thinking) depends critically on the capacity to manipulate structured representations, including predicates with multiple arguments and higher order predicates that take propositions as arguments (Barnden, 1994; Markman et al., 1995). Localist connectionist units (e.g., as in ACME) and symbolic tree structures (e.g., as in SME) are extremely well-suited to this task. At the same time, pattern matching (e.g., for analog access), pattern completion (e.g., for mapping and inference), and generalization (e.g., for schema induction) require representations that are more flexible than purely localist or symbolic representations. Distributed representations are well-suited to such problems, but traditional distributed representations are notoriously bad at capturing structure. The basic representational challenge for a model of analogical reasoning is to satisfy these joint requirements of flexibility and structure sensitivity, and the fundamental binding problem lies at the heart of this dilemma.

Representing a proposition entails binding the argument roles of the proposition to their fillers (Fodor & Pylyshyn, 1988). For example, to represent the statement “John loves Mary,” John must be bound to the role of lover while Mary is bound to the role of beloved. Traditional distributed representations are ill-suited to representing such structures because they do not make these bindings explicit: Simply jointly activating patterns representing “John,” “Mary,” and “loves” cannot distinguish “John loves Mary” from “Mary loves John” (or even from a description of a narcissistic hermaphrodite).

A number of distributed approaches to representing role bindings have been proposed (e.g., Elman, 1990; Pollack, 1990; Shastri & Ajjanagadde, 1993; Smolensky, 1990; Touretzky & Hinton, 1988). One general approach is to introduce units or vectors that code conjunctions of roles and fillers. Models based on tensor products (Smolensky, 1990) and holographic reduced representations (Plate, 1991) are particularly sophisticated vari-

ants of this conjunctive coding approach. The STAR model of analogical mapping (Halford et al., 1994) is based on multidimensional tensor products. Models based on conjunctive coding can provide some of the advantages of distributed representations while preserving structure, but they also have limitations (Hummel & Biederman, 1992). Units in a conjunctive code represent combinations of case roles (predicates) and fillers (objects) rather than individual roles and fillers, so the natural similarity structure of the individual predicates and objects is lost. For example, if separate units represent (a) John as lover, (b) John as beloved, (c) Mary as lover, and (d) Mary as beloved, then the proposition “John loves Mary,” represented by units a and d, would have absolutely no semantic overlap with “Mary loves John,” represented by c and b. Although we have illustrated this problem by using a strictly localist code (i.e., one unit per conjunction), the basic problem cannot be solved simply by postulating more distributed conjunctive codes. For example, Smolensky’s (1990) tensor products are distributed, but the representation of a given object bound to one case role will not necessarily overlap at all with the representation of the same object bound to a different case role. In general, the capacity of a distributed conjunctive code to represent binding information declines in proportion to its capacity to preserve similarity across different bindings: In a conjunctive code, these capacities are fundamentally in conflict (Hummel & Holyoak, 1993).

Dynamic Binding

An alternative to conjunctive coding is *dynamic binding*, in which units representing case roles are temporarily bound to units representing the fillers of those roles. Dynamic binding is difficult in neural networks because it requires an explicit tag with which units can represent group membership. This tag must be independent of a unit’s activation (because activation expresses information that is independent of binding; Hummel & Biederman, 1992). One possible dynamic binding tag is based on synchronized oscillations in activity, in which units fire in synchrony if they are bound together and fire out of synchrony if they are not (Milner, 1974; von der Malsburg, 1981, 1985; see Gray, 1994, for a review). Although controversial (see Tovee & Rolls, 1992), there is some neurophysiological evidence for binding by synchrony in visual perception (e.g., in striate cortex; Eckhorn et al., 1988; Gray & Singer, 1989; König & Engel, 1995) and in higher level processing dependent on frontal cortex (Desmedt & Tomberg, 1994; Vaadia et al., 1995).

Numerous connectionist models use synchrony for binding. This mechanism has been applied in models of perceptual grouping (e.g., Eckhorn, Reitboeck, Arndt, & Dicke, 1990; von der Malsburg & Buhmann, 1992), object recognition (Hummel & Biederman, 1990, 1992; Hummel & Saiki, 1993; Hummel & Stankiewicz, 1996), and rule-based reasoning (Shastri & Ajjanagadde, 1993). LISA (like its predecessor, the Indirect Mapping Model, IMM; Hummel, Burns, & Holyoak, 1994; Hummel & Holyoak, 1992; Hummel, Melz, Thompson, & Holyoak, 1994) uses synchrony to represent role-filler bindings in propositions. For example, “John loves Mary” would be represented by units for John firing in synchrony with units for the agent role of “loves,” whereas units for Mary fire in synchrony

with units for the patient role. The John-agent units must fire out of synchrony with the Mary-patient units.

Dynamic binding permits a small set of units to be reused in an unlimited number of specific bindings. This capacity to reuse units allows the representation of case roles and objects to be completely independent of one another, thereby preserving similarity across different bindings. For example, all propositions in which John serves as a role filler are similar by virtue of their sharing the units that represent John; likewise, all propositions involving the predicate "love" employ the same "love" units. Accordingly, the representation of two propositions overlap to the extent that their meanings overlap. Analogous benefits accrue from the use of dynamic binding to represent structure in models of perception (see Hummel & Biederman, 1992). The common role of dynamic binding in perception and reasoning, although not surprising, suggests additional linkages between perception and cognition: Because both require a capacity to represent structure, both are subject to the computational constraints imposed by the need for dynamic binding.

Although synchrony is a convenient (and perhaps neurally plausible) mechanism for dynamic binding, we are not committed to the use of synchrony per se. Rather, our theoretical claim is that dynamic binding (in one form or another) is a fundamental prerequisite to representing structure and that dynamic binding is therefore a problem that must be taken seriously in the attempt to model human reasoning. The critical properties of any scheme for dynamic binding are that (a) the binding tag must be independent of activation and (b) dynamic binding is necessarily capacity limited (i.e., there is a finite number of tag values; Halford et al., 1994; Hummel & Holyoak, 1993). These properties are apparent in binding by synchrony. Whether two units fire in or out of synchrony with one another (the binding tag) is independent of the magnitude of their firing (their activation). The capacity limit is apparent in the fact that only a finite number of groups of units can be active and mutually out of synchrony with one another (as elaborated shortly). Although synchrony is a neurally plausible basis for dynamic binding, it is by no means the only imaginable basis (e.g., Mozer, Zemel, Behrmann, & Williams, 1992, describe a network that uses imaginary numbers as a binding tag). Our argument is that independence is a necessary requirement for dynamic binding, and that any realistic scheme that achieves it also has capacity limits (Hummel & Biederman, 1992; Hummel & Holyoak, 1993; Hummel & Stankiewicz, 1996).

Constraints on Dynamic Binding in the Representation of Structure

Dynamic binding necessarily operates in working memory (if two units are not active, they cannot be dynamically bound) and, hence, is unsuitable as a basis for binding in LTM. For this purpose, static binding (e.g., by conjunctive units) is necessary.¹ Nonetheless, dynamic binding conveys two critical advantages as a representation of binding in working memory. First, it provides a mechanism for making rapid systematic inferences without incurring the computational cost of forming a long-term memory code. Second, it removes the need for a static code (e.g., as stored in LTM) to directly capture the structured content of a proposition. The sole requirement for the LTM repre-

sentation of a role-filler binding is that it be capable of reinstating the original dynamic binding when the proposition is reactivated. Once reactivated, the dynamic form of representation will make the structured content of the proposition explicit. This distinction between dynamic binding in active memory and static binding in LTM is consistent with the fact that dense anterograde amnesia does not result in catastrophic loss of reasoning abilities: It is possible to lose the capacity to store static bindings in LTM in a retrievable form without losing the capacity for dynamic binding in active memory.

Although it is useful, dynamic binding is nonetheless subject to important limitations. One is a capacity limit. Dynamic binding (by synchrony or any other means; see Hummel & Stankiewicz, 1996) does not permit unlimited parallel processing over distributed representations. Rather, the number of dynamic bindings that can be represented at any given time is limited to the number of distinct tags in the binding scheme. For example, in the case of synchrony, the number of binding tags is given by the number of groups of units that can be simultaneously active and mutually desynchronized. Let us refer to a collection of mutually synchronized units as a *group*, and to a collection of mutually desynchronized groups as a *phase set*. The capacity limit of binding by synchrony is equivalent to the size of the phase set: How many groups is it possible to have simultaneously active but mutually out of synchrony?² This number is necessarily limited, and its value is proportional to the length of time between successive peaks in a given group's output (the period of the oscillation) divided by the duration of each peak. Single unit recording studies with monkey and cat suggest that, at least in the visual systems of these animals, a reasonable estimate of the size of a phase set is between four and six phases (e.g., Gray & Singer, 1989; see also Hummel & Biederman, 1992).

A second and less obvious limitation of dynamic binding is the *one-level restriction*, which concerns the number of levels of embedding that can be represented simultaneously at the level of semantic primitives. Dynamic binding can only operate at one level of abstraction or hierarchy at a time (Hummel & Holyoak, 1993; Hummel et al., 1994). Consider the higher order proposition "Sam believes that John loves Mary," in which an entire proposition is bound to the patient role of the "believe" predicate. To represent the internal structure of "John loves Mary," the bindings "John-lover" and "Mary-beloved" must be desynchronized; however, to represent "John loves Mary" as the unitary filler of "what is believed," the representation of the entire proposition must be synchronized with the patient role of the "believes" predicate. Simply binding "what is believed" to both case roles of "John loves Mary" creates ambiguity at the level of semantic binding (see Hummel & Holyoak, 1993). Synchrony can only dynamically represent role bindings

¹ Note that it is not necessary to postulate the preexistence of all possible conjunctive units. Rather, a novel binding can first be represented dynamically (in active memory), with a conjunctive unit created only when it is necessary to store the binding in LTM.

² Although this discussion is couched in terms of phase, it is not necessary to assume that the units outputs are strictly phased-locked (i.e., oscillating with regular and equal periods). The analysis operates in the same way if one simply assumes that the units are synchronized.

at one hierarchical level at a time. For the same reason, it can only represent the structure of one proposition at a time. If John loves Mary and also is hated by Sam, then simultaneously representing "John-lover" and "John-hated" would (given distributed representations of predicate meaning) blend the "lover/hated" roles.

The synchrony-based model of Shastri and Ajjanagade (1993) can "stack" multiple roles of an object within a single phase, apparently violating the one-level restriction. However, this is possible only because the model uses localist representations of concepts and objects. In essence, their model uses the static binding implicit in a localist code to eliminate the need for dynamic binding at the lower levels of the hierarchy. The benefit of this approach is the ability to stack objects and predicates, thereby avoiding the one-level restriction; the cost is that, as in any localist code, the resulting representations do not capture the similarity relations among the entities they represent. If the model represented objects and predicates in a distributed fashion, then it would capture these similarity relations, but it would also be subject to one-level restriction.

LISA uses distributed representations of concepts and objects, thereby capturing similarity relations. However, as a consequence, it is subject to the one-level restriction. LISA can nonetheless map complex structures involving higher order relations and multiple role bindings. However, because LISA is founded on distributed representations of concepts and objects, it is forced to use various types of serial processing to maintain structure sensitivity when mapping hierarchical structures.

In summary, LISA translates the computational constraints imposed by the similarity-structure trade-off into an architecture that combines distributed representations with a capacity for dynamic binding. The benefit of this approach is that LISA can represent structure without sacrificing similarity. The cost is that, in contrast to models such as ACME and SME, LISA must operate within inherent capacity limits (given by the size of the phase set). As a result, LISA lacks the ability to solve analogical mappings by massively parallel constraint satisfaction. Rather, mapping in LISA requires systematic serial processing combined with limited parallel constraint satisfaction. The theoretical claim is that the human cognitive architecture represents a similar algorithmic solution to the same computational problem. The surprising result is that for a wide range of suitably structured problems, the serial mapping algorithm yields outputs that closely mimic those that would be produced by unlimited parallel constraint satisfaction. However, analogies that make excessive demands on working memory, or that have an "unnatural" structure, reveal LISA's limitations—which appear to be shared with people.

The Architecture and Operation of LISA

Overview

LISA is designed to represent propositional knowledge both dynamically in working memory and statically in LTM. Based on these representations, it performs structured comparisons during both access and mapping as a form of guided pattern matching. When a proposition becomes active, the units representing it in LTM generate distributed, synchronized patterns

of activation (one for each case role) on a collection of *semantic units*. These patterns, which serve as the model's working-memory representation of the proposition, capture both the semantic content of the proposition (by virtue of the distributed representation) and the structure of its case role-argument bindings (via dynamic binding by synchrony). The semantic primitive units are shared by all propositions in LTM, so the pattern generated by one proposition will tend to activate one or more similar propositions in other analogs. This process is a form of memory access. Mapping is performed by augmenting access with the capacity to learn which propositions responded to which patterns, that is, to learn the correspondence between the generating proposition and the responding (recipient) proposition or propositions. These correspondences, which are stored as connection weights, then serve to constrain subsequent memory access. Over the course of several propositions, the result is a representation of the correspondences between the elements of two or more analogs.

Architecture and Representation of Propositions

The core of LISA's architecture is a system for representing dynamic role-filler bindings and encoding those bindings in LTM. In working memory, role-filler bindings are represented as activation patterns distributed over the semantic units. These semantic units are linked to *structure units* that store, recreate, and respond to patterns on the semantic units. Structure units serve the purely structural function of encoding the binding relations among the components of a proposition into LTM. This function is enhanced by their strictly localist implementation (Hummel & Holyoak, 1993).

Every proposition is encoded in LTM by a hierarchy of three types of structure units (see Figure 1). At the bottom of the hierarchy are *predicate* and *object* units. (Predicate and object units are functionally equivalent; we distinguish them only for conceptual clarity.) Each predicate unit locally codes the semantic primitives of one case role of one predicate. For example, the predicate unit *loves1* represents the first (agent) role of the predicate "loves" and has bidirectional excitatory connections to all of the semantic units representing that role (e.g., *actor*, *emotion1*, *positive1*, *strong1*, etc.); *loves2* represents the second role of "loves" and is connected to the semantic units representing that role (e.g., *patient*, *emotion2*, *positive2*, *strong2*, etc.). Different instantiations of a predicate may have different shades of meaning (e.g., *loves* in "John loves Mary" differs from *loves* in "John loves winning an argument"), but much of this difference can be captured in the semantics attached to the arguments filling the roles. In the current implementation, shades of meaning that cannot be captured in this fashion are captured by coding the different shades as distinct semantic units. We generally distinguish semantic primitives for predicates by place: *Emotion1*, representing emotion in the first role of an emotion predicate, and *emotion2*, representing emotion in the second role, are separate units. However, this separation is not strictly necessary, and different roles of the same predicate may share semantic units. Similar predicates will tend to share units in corresponding roles (e.g., the predicate units *loves1* and *likes1* will be connected to many of the same semantic units). However, some predicates will share units in different roles

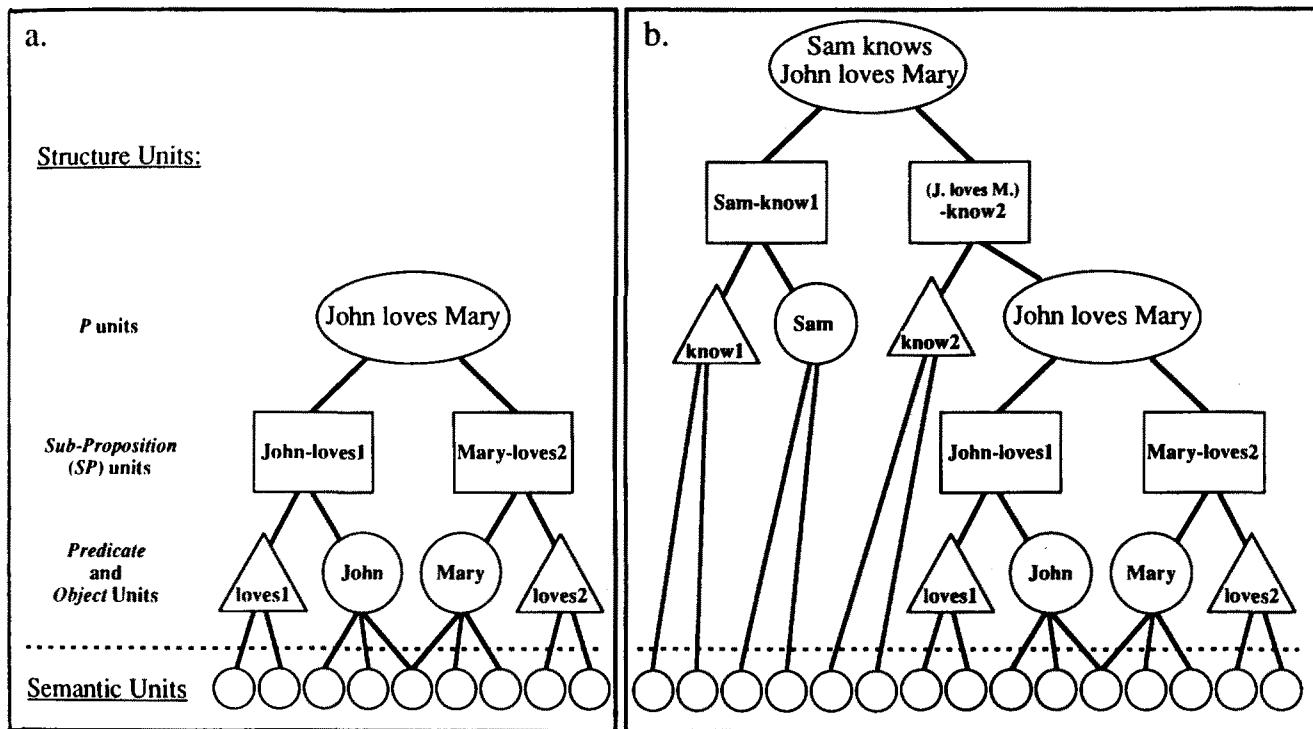


Figure 1. (a) Illustration of the LISA representation of the proposition *loves (John Mary)*. (b) LISA representation of the hierarchical proposition *knows (Sam knows (John loves Mary))*. J. = John; M. = Mary.

(e.g., converses such as *parent of* and *child of*, which have reversed roles). In this way, the structural and semantic similarity of different predicates is made explicit.

Object units are just like predicate units except that they are connected to semantic units describing things rather than roles. For example, the object unit *Mary* might be connected to semantic units such as *human*, *adult*, *female*, and so on, whereas *John* might be connected to *human*, *adult*, and *male*.

Subproposition (SP) units are structure units that bind case roles to their fillers (objects or propositions) in LTM. For example, “John loves Mary” would be represented by two SP units: one representing John as agent of loving, and the other representing Mary as patient of loving. The John-agent SP would share bidirectional excitatory connections with *John* and *loves1*, and the Mary-patient SP would share connections with *Mary* and *loves2*.

Proposition (P) units reside at the top of the hierarchy. Each P units shares bidirectional excitatory connections with the corresponding SP units. P units serve a dual role for the purposes of representing hierarchical structures, such as “Sam knows John loves Mary” (see Figure 1b). In their role as binding units for a proposition, they are connected to SPs representing their constituent role-filler bindings (Figure 1a). When a proposition serves as the filler of a role in another proposition, the lower-level P unit serves in the place of an object unit under the appropriate SP. For example, the P unit for “John loves Mary” is connected (in the place of an object) to the second SP of the proposition “Sam knows [X].” In this way, the binding of

“John loves Mary” to the patient role of “knows” is made explicit in LTM.

Every analog is represented by a hierarchy of this type for each proposition it contains. Object and predicate units are not repeated within analogs. For example, if John serves as an argument to multiple propositions in a given analog, the same *John* unit will be connected to multiple SP units. Each type of structure unit plays an important role in LISA’s operation. Having separate units for roles, objects, and propositions allows the model to treat each such entity as an entity, a capacity that is critical for mapping analog elements onto one another. At the same time, it is important to emphasize that the structure units do not directly encode meaning. Rather, they work together to impose (and respond to) particular patterns of synchrony on the semantic units; it is only the latter that encode meaning.

The final component of LISA’s architecture is a set of *mapping connections* between structure units of the same type in different analogs. Every P unit in one analog shares a mapping connection with every P unit in every other analog; likewise, SPs share connections across analogs, as do object and predicate units. These connections are assumed to reside in working memory and to be established when the analogs are called into working memory for mapping. At the beginning of a simulation run, the weights on these connections are initialized to zero. A mapping weight grows larger (taking larger positive values) whenever the units it links are active simultaneously and grows more negative whenever one unit is active and the other is inactive. In this way, LISA keeps track of what-corresponds-to-what

across the analogs. By the end of a simulation run, corresponding units will have large positive weights on their mapping connections, and noncorresponding units will have large negative weights.

The mapping connections play an important role in the model's operation. LISA treats analogical mapping as a form of learning. By allowing structure units in one analog to directly activate structure units in other analogs, mapping connections permit learned mappings to constrain future mappings. We will see that by treating mapping as a kind of learning, LISA provides a unified account of analogical access and mapping with a single assumption: Mapping connections can only be learned when the analogs reside in working memory.

Operation

For clarity, the model's operation is described here only in broad strokes. The fine points of the algorithm, including equations and parameters, are detailed in Appendix A.

For the purposes of mapping, analogs are divided into two mutually exclusive sets: a *driver*, and one or more *recipients*. The driver and all recipients are assumed to reside in working memory. For the purposes of memory retrieval, there is a third class of *dormant* analogs, which are assumed to reside in LTM but not working memory. Dormant analogs are candidates for retrieval from memory but cannot participate in analogical mapping (i.e., no mapping connections are established for dormant analogs). The only differences between the operation of recipient and dormant analogs are that (a) the former, but not the latter, update their mapping connections with the driver; and (b) the activations of units in dormant analogs decay faster than the activations of units in recipient analogs (Equation 14, Appendix A). The theoretical assumption motivating (a) is that the mapping connections reside in working memory; the assumption motivating (b) is that activation consumes attentional resources, so unattended (dormant) elements decay more rapidly than attended (recipient) elements. As in systems such as ARCS and MAC/FAC, retrieval in LISA is an inexpensive process that consumes fewer working memory resources than mapping. However, in contrast to ARCS and MAC/FAC, access in LISA operates in the same fundamental way—and on the same knowledge representations—as analogical mapping. As a result, access in LISA is sensitive to role-arguments bindings. Except when it is necessary to distinguish mapping from memory retrieval, we refer to both active recipient analogs and dormant analogs as recipients.

There is a strong asymmetry between the activity of driver and recipient analogs, in that mapping and retrieval are controlled by the driver. Note that there is no necessary linkage between the driver–recipient distinction and the more familiar source–target distinction. However, a canonical flow of control would involve initially using a target analog (the unfamiliar, novel situation) as the driver to access a source analog stored in long-term memory. Once a source is in working memory, mapping can be performed in either direction (including successive switches of driver assignment from one analog to the other). After a stable mapping is established, the source will be used to drive inference generation and schema induction in the target (processes beyond

the scope of this article; however, see Hummel & Holyoak, 1996, in press).

Driver operation. As a default, propositions in the driver are selected one at a time to become active in the phase set. (We discuss the possible selection of multiple propositions shortly.) The protocol specifying the order in which propositions are selected is given as input. We assume that the order in which propositions are selected is based, in part, on factors that determine text coherence, such as argument overlap (Kintsch & van Dijk, 1978) and causal connections (Keenan, Baillet, & Brown, 1984; Trabasso & van den Broek, 1985). That is, successive propositions entering the phase set will tend to overlap in the objects that fill their slots or will themselves be related by a higher order proposition expressing a causal or other functional dependency. Although we have yet to implement a specific algorithm by which LISA would determine its own selection order, it is possible to take advantage of the fact that normal text is ordered so as to maintain coherence. Therefore, as a general default for verbal analogies, we simply select propositions in the order in which they appear in a sequential propositional representation of the text describing the driver analog. In addition, we assume that propositions that are deemed to be especially important (e.g., because of their apparent relevance for achieving a goal) will tend to be selected earlier and more frequently than less important propositions. Such factors (e.g., goal relevance and causal-chain status) also serve as top-down constraints on strategic processing (e.g., allocation of attention) in human text and event comprehension (Fletcher, 1986; Fletcher & Bloom, 1988; van den Broek, 1988). For example, readers allocate more attention to statements following causal antecedents than to statements following causal consequents (Fletcher, Hummel, & Marsalek, 1990). As we will demonstrate, variations in selection priority allow LISA to simulate the influence of pragmatic centrality on mapping.

A proposition is selected by setting the activation of its P unit to 1.0 and allowing that unit to excite the SP units below itself. SPs under the same P unit represent the separate role-filler bindings of that proposition, so they must fire out of synchrony with one another. To this end, SPs inhibit one another, competing to respond to input from the P unit. Due to random noise in their excitatory inputs, one SP will win the initial competition, becoming highly active and inhibiting all others to inactivity. Active SPs excite the predicate and object units under them, which in turn excite the semantic units to which they are connected. The result is a pattern of activity on the semantic units representing the semantic primitives of the object and case role connected to the winning SP (see Figure 2a). SPs "take turns" firing because of the operation of an inhibitory unit (von der Malsburg & Buhmann, 1992) associated with each SP. As detailed in Appendix A, an SP's inhibitor allows the SP to remain active for a short time and then temporarily inhibits it to inactivity. The result is that, with a constant input from a P unit, an SP's activation will oscillate between zero and one. In combination with SP-to-SP inhibition (and SP-to-predicate and SP-to-object excitation), this arrangement causes SPs, predicates, and objects to oscillate in SP-based groups. For example, when "John loves Mary" is selected, the John-as-lover group fires out of synchrony with the Mary-as-beloved group, alternat-

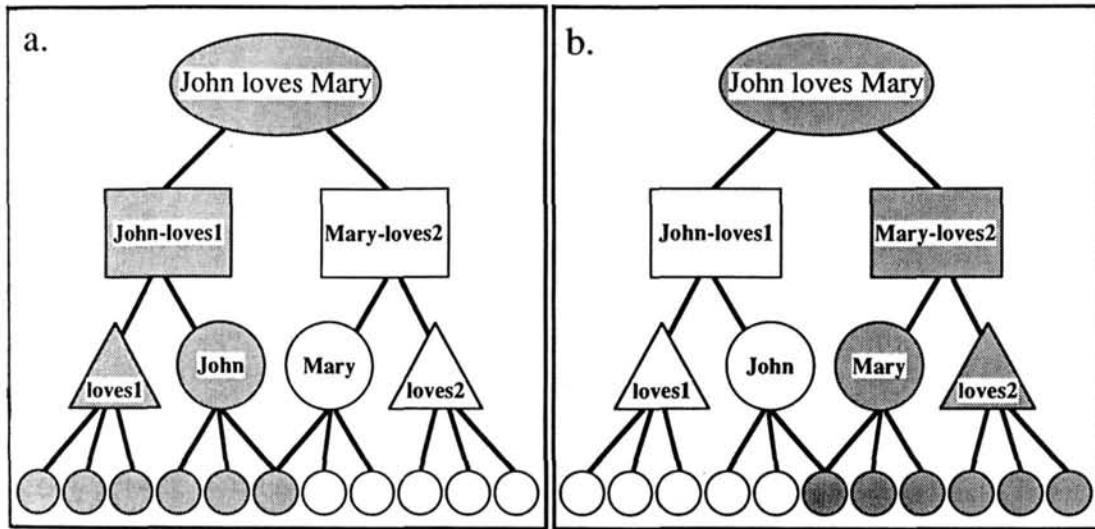


Figure 2. Proposition (P) units impose the working-memory representation of a proposition onto the semantic units via subproposition (SP), predicate, and object units. (Case roles of a proposition fire out of synchrony with one another.) Shaded areas depict active units, and unshaded (white) areas depict inactive units.

ing between the patterns depicted in Figures 2a and 2b, respectively.

So far, we have illustrated the driver's operation with a simple, nonhierarchical proposition. Hierarchical propositions work in the same way except that they are subject to two additional constraints (dictated by the one-level restriction). The first is that only one level of hierarchy is selected to be active at a time. For example, consider the hierarchical proposition "Sam knows John loves Mary" (Figure 3). This proposition will be activated (selected) in two sets: the parent set, "Sam knows [X]" (Figures 3a and 3b), and the daughter set, [X] = "John loves Mary" (Figures 3c and 3d). When "John loves Mary" is selected, LISA operates exactly as described previously. When "Sam knows [X]" is selected, LISA also runs as described previously: Its SPs and their associated role and filler units fire out of synchrony. However, now, one filler is a P unit rather than an object unit. The second constraint on hierarchical propositions is that P units activated as fillers (i.e., via SPs above themselves, in the way that "John loves Mary" is activated when "Sam knows [X]" is selected) are not allowed to excite their constituent SPs. When the SP for "[X]-as-known" fires (Figure 3b), the P unit for "John loves Mary" will become active, but it will not activate its own SPs (John-as-lover and Mary-as-beloved). We refer to these two modes of operation as the "parent" and "daughter" modes, respectively. Selected P units enter parent mode, exciting their constituent SPs ("Sam knows [X]" is selected in Figures 3a and 3b; "John loves Mary" is selected in Figures 3c and 3d); however, when a P unit is activated by an SP above itself, it enters daughter mode and does not excite its constituent SPs (e.g., "John loves Mary" in Figure 3b). As elaborated in Appendix A, this mode distinction is straightforward to implement on the basis of signals that are completely local to the P units. Parent propositions and their daughters are selected at separate times, just like any other pair

of propositions, but they will typically be selected in close temporal proximity (usually with the daughter immediately following the parent).

Recipient operation and learning a driver–recipient mapping. Distributed patterns of activation are produced on the semantic units in response to the sequential selection of propositions in the driver. These patterns are arranged hierarchically in time. On a coarse time scale, patterns correspond to propositions: One proposition is selected to fire, followed by another. At a finer time scale, patterns correspond to role-filler bindings: The SPs under a common P unit fire out of synchrony with one another.

The job of a recipient analog is to respond to these patterns. Object, predicate, and SP units in the recipient compete to respond to patterns varying over the fine time scale, and P units compete to respond to patterns varying over the coarse time scale. That is, the recipient analog treats patterns on the semantic units as inputs to be classified—a task for which connectionist networks are extremely well-suited. Important to note is that the recipient is initially completely blind to the driver producing the patterns; its only source of information is the patterns themselves. Thus, unlike virtually all current analogy models in which mapping is treated as an explicit comparison between analogs (based on symbols representing hypothesized correspondences), mapping in LISA is a much more implicit process in which the recipient reacts to semantic patterns created by the driver.

In the recipient, units of the same type are competitive (mutually inhibiting one another) and units within a proposition are cooperative (mutually exciting one another). (The competitive rule takes precedence such that SPs in the same proposition inhibit one another.) Consider mapping "John loves Mary" in the driver onto "Bill likes Susan" versus "Peter fears Beth" in the recipient (see Figure 4). When the SP for John-as-lover

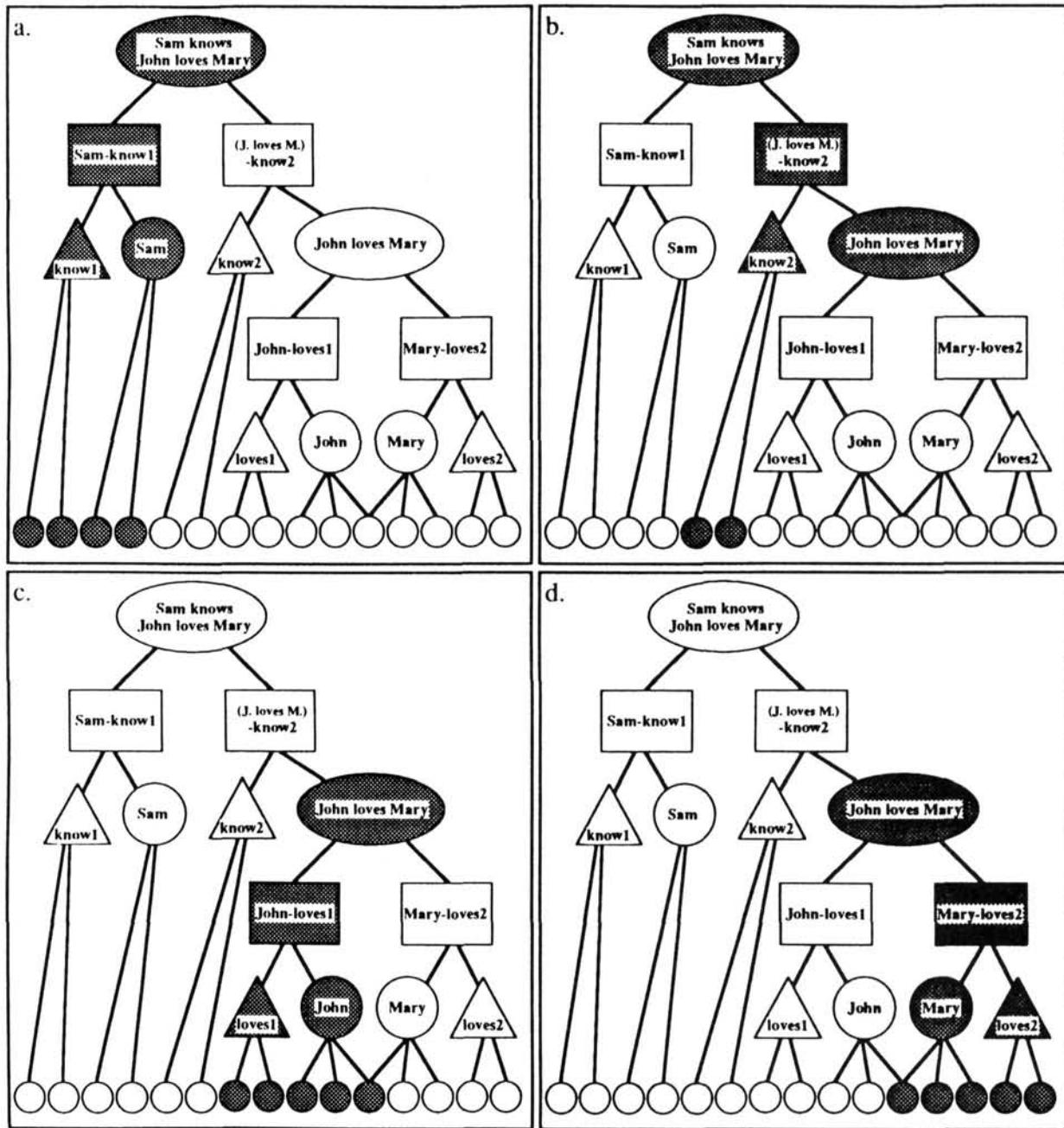


Figure 3. Illustration of parent and daughter propositions. Parent propositions and their daughters (argument propositions) are constrained by the one-level restriction to be selected (active) at different times. Shaded areas depict active units, and unshaded (white) areas depict inactive units. (a) The higher-level proposition (*P*) *know* (*Sam X*) is selected and its first subproposition (SP), *Sam-know1* is active. *J. = John*; *M. = Mary*. (b) The higher-level proposition *X-know2 know* (*Sam X*) is selected and its second SP, is active. When it receives input from *X-know2* the *P* unit for *loves* (*John Mary*) goes into daughter mode and passes no excitation to its constituent SPs. (c) The lower-level proposition is selected (and therefore in parent mode) and its first SP, *John-loves1* is active. (d) The lower-level proposition is selected and its second SP, *Mary-loves2* is active.

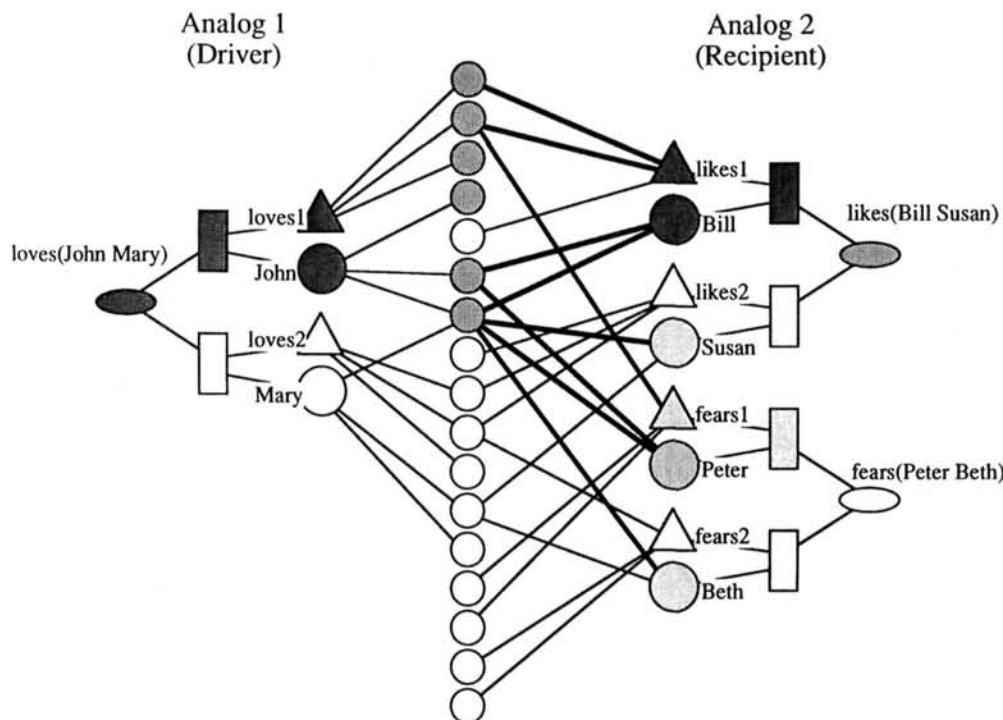


Figure 4. Illustration of mapping *loves(John Mary)* (Analog 1) onto *likes(Bill Susan)* versus *fears(Peter Beth)* (Analog 2). The subproposition (SP) encoding the binding of *John* to *loves1* (the agent role of "love") is active in the driver (Analog 1). For both the driver and the recipient, black units are very active and white units are inactive. In the recipient, moderately active units are depicted in gray. *Loves1* shares more semantic primitives with *likes1* (the agent role of "like") than with *fears1* (the agent role of "fear"), so the pattern generated by *loves1* will activate *likes1* rather than *fears1*. As a result, *John* will map to *Bill* rather than *Peter*, even though *John* shares just as much semantic content with *Peter* as with *Bill*.

(*John loves1*) fires in the driver, it will activate *John* and *loves1*, which will activate their semantic units (e.g., *human*, *male*, *adult*, and *emotion1*, *positive1*, *strong1*). This pattern will excite object and predicate units in the recipient, which will compete to become active. *Human*, *male*, and *adult* will excite *Bill* and *Peter*; *human* and *adult* will excite *Susan* and *Beth*. In this competition, *Bill* and *Peter* will become equally active, inhibiting *Susan* and *Beth*. Based on their semantic overlap alone, LISA begins to act as if *John* corresponds to either *Bill* or *Peter*. At the same time, *emotion1*, and *positive1* will excite the predicate unit *likes1*, but only *emotion1* will excite *fears1*. *Likes1* will inhibit *fears1*: LISA begins to act as if *loves1* corresponds to *likes1*. Because *likes1* is more active than *fears1*, the SP *Bill-likes1* will receive more bottom-up input—and therefore become more active—than the SP *Peter-fears1*. SPs excite the P units to which they belong, so the unit for "Bill likes Susan" will become more active than the unit for "Peter fears Beth." Hence, LISA concludes that "John loves Mary" corresponds to "Bill likes Susan" rather than "Peter fears Beth." The SP mappings also allow LISA to resolve the semantically ambiguous *John-to-Bill* versus *John-to-Peter* mappings. SPs feed activation back to their predicate and object units, giving *Bill* an edge over *Peter*. Now, LISA concludes that *John* corresponds to *Bill* rather than to *Peter*. Analogous operations will cause LISA to conclude that *Mary* corresponds to *Susan* rather than

to *Beth*, and that *loves2* corresponds to *likes2*. Selected P units in the driver remain active for a fixed number of iterations. At the end of this time, the activations of all units in the recipient are initialized to zero in preparation for the next driver proposition.

As these operations run, the cross-analog mapping connections keep track of which structure units are coactive across the analogs (as detailed in Appendix A). When structure unit *i* is active at the same time as structure unit *j* (where *i* and *j* are units of the same type in different analogs), a *buffer* on the connection from *j* to *i* is incremented; when *i* is active while *j* is inactive, the buffer from *j* to *i* is decremented. After all propositions in the phase set have been selected (and run), the values accumulated on the mapping connection buffers are used to set the weights on those connections: A connection weight is incremented or decremented in proportion to the value of the corresponding buffer, with the result that structure units develop positive mapping weights to the extent that they are active simultaneously, and negative weights to the extent that one is active while the other is inactive. The mapping weights are then normalized to enforce the constraint of one-to-one mapping: Whenever the weight from *j* to *i* increases by some value, $\Delta w_{ij} > 0$, all other mapping weights leading *into* unit *i* are decremented by $\Delta w_{ij}/n_i$, (where n_i is the number of other weights leading *into* *i*), and all other weights leading *out* of unit *j* are decremented by $\Delta w_{ij}/n_j$, (where n_j is the number of other weights leading *out* of *j*). In the preceding example, the

mapping weights from *John* to *Bill*, *Mary* to *Susan*, *loves1* to *likes1*, and *loves2* to *likes2* (as well as the associated SP and P weights) will all develop positive values. All other mapping weights will develop negative values. The resulting weights serve as LISA's representation of the mapping, with positive weights between corresponding elements.

These mapping weights serve to enforce structural consistency, both with hierarchical propositions and with multiple non-hierarchical propositions. First consider the nonhierarchical case. Imagine that LISA learns to map *John* to *Bill* and *Mary* to *Susan*, and let us add some additional propositions to the driver and recipient analogs: "Mary gave John flowers" in the driver, and "Susan gave Bill candy" and "Beth gave Peter a watch" in the recipient. If we assume that *flowers* shares just as much semantic content with *watch* as it does with *candy*, then in isolation the "give" mapping would be ambiguous: "Mary gave John flowers" would map equally well to both "Susan gave Bill candy" and "Beth gave Peter a watch." However, if "John loves Mary" has already mapped to "Bill loves Susan," then "Mary gave John flowers" will tend to map to "Susan gave Bill candy" because of the positive weights from *Mary* to *Susan* and from *John* to *Bill*. As a result, LISA will also map *flowers* to *candy* rather than to *watch*. This kind of biasing works with analogies of any size, serving both to implement structure sensitivity in mapping and to produce an interesting order sensitivity: LISA can resolve the mapping of *flowers* to *candy* in the context of the *gave* propositions only after it has mapped *Mary* to *Susan* in the context of the *loves*-*likes* propositions. Additional implications of this order sensitivity are discussed in the report of simulation results.

The same biasing allows LISA to map hierarchical propositions in spite of the one-level restriction. Let the driver consist of the proposition "Sam knows John loves Mary," and let the recipient consist of "Joe knows Bill likes Susan," and "Robert likes Beth." At the lowest level of hierarchy, the mapping of *John* to *Bill* versus to *Robert* is completely ambiguous: "John loves Mary" maps equally well to both "Bill likes Susan" and "Robert likes Beth." However, in the context of the higher-level propositions, "Sam knows [X]" and "Joe knows [Y]", the mapping is unambiguous. "John loves Mary" and "Bill likes Susan" serve as arguments of corresponding higher level propositions and therefore correspond to one another. Hence, the correct mapping places *John* into correspondence with *Bill* rather than with *Robert*. LISA discovers this mapping as follows. When "Sam knows [X]" is selected in the driver, "Joe knows [Y]" will become active in the recipient, establishing the correspondence between *Sam* and *Joe* and—more importantly—between [X] ("John loves Mary") and [Y] ("Bill likes Susan"). This latter mapping will be stored as a positive weight on the connection from the P unit for "John loves Mary" to the P unit for "Bill likes Susan." When "John loves Mary" is selected, this positive weight will give "Bill likes Susan" an advantage over "Robert likes Beth." As a result, *John* will map to *Bill* rather than to *Robert*.

Constraints on Working Memory

It is important to relate LISA's operation to our theoretical claims about the processes underlying human reasoning, especially the limits of working memory. LISA's working memory

can be subdivided into three levels, operating at increasingly fine-grained time scales. *Active memory* can be viewed as the largest subset of long-term memory that is currently the focus of attention. Active memory is assumed to be relatively stable over a time range of several seconds to minutes, and the contents of active memory are assumed to be held in a state in which they can be rapidly shifted in and out of *working memory* (which lasts on the order of a few seconds, as described shortly). We assume that the cross-analog connection buffers representing correspondences between driver and recipient reside in active memory. It follows that analogical access and mapping differ in a fundamental way: Mapping can be guided by the emergence of new structural connections (i.e., altered weights on mapping connections), but access must rely on preexisting connections. In addition, access, by its nature, involves a competition among numerous stored cases in LTM, whereas mapping (typically) involves only a single case in active memory as recipient. LISA (like the ARCS model of Thagard et al., 1990) treats access as a competitive process in which lateral inhibition between stored cases creates a retrieval bottleneck, such that only a small number of stored analogs can become highly active after a single retrieval attempt (see Appendix A).

Within active memory, a very small number of propositions in one analog can enter a dynamic state of phase-locked activity that actively represents their variable bindings. This working memory, which is viewed as the most significant bottleneck in the system, corresponds to the *phase set*, consisting of the set of mutually desynchronized role-argument bindings (SPs). As stated previously, we assume that adult humans can maintain an upper bound of about 4–6 nonoverlapping phases. The phase set, along with the collection of mappings involving the elements of that set (i.e., the mapping connection buffers referring to those elements), is assumed to be relatively stable over a range of a few seconds and intuitively corresponds to the system's "current thought."

For clarity, the previous examples treated individual propositions as the residents of each phase set. For example, the proposition "John loves Mary" requires two role-argument bindings and would therefore fill a phase set of size 2. "Mary gave John flowers" has three role-argument bindings and would fill a set of size 3. However, in general, we assume that the phase set (working memory) may contain more than one proposition. Specifically, we associate the size of the set with the number of independent SPs (role-filler bindings) that can be activated before updating the weights on the mapping connections. For example, if LISA had a working memory (phase set) of size 5, then it could select "John loves Mary" (occupying 2 phases) and then select "Mary gave John flowers" (occupying 3), and then update the mapping connections, converting the buffer values to weight values. By contrast, if LISA had a maximum set size of 3, then it would have to update the connection weights between selecting the propositions (i.e., select the *loves* proposition, then update the weights, then select the *gave* proposition, then update the weights again). With a set size of 2, it could not reliably represent the three-argument *gives* proposition at all.

The theoretical claims motivating the above convention are the following: (a) Working memory (the phase set) corresponds to the current contents of "awareness." (b) The maximum size

of the phase set is given by the number of role-filler bindings that can be explicitly entertained simultaneously (i.e., be kept simultaneously active but out of synchrony). (c) The buffers on the mapping connections reside in working memory (whereas the weights do not), so the mapping hypotheses generated in working memory (i.e., during a given phase set, as embodied in the buffers) must be consolidated (i.e., converted to weight values so the buffers may be "flushed") before moving on to the next phase set.

At the most microscopic level of operation (on the order of about 25 ms; Gray et al., 1989), each individual group in the phase set becomes active serially.³ Following the one-level restriction, the content of a group is restricted to a single role-filler binding (i.e., one SP-based group). It follows that the maximum amount of knowledge that can coexist in working memory is a number of propositions that do not jointly involve more than 4–6 role bindings.

In the operation of LISA, each subdivision of active memory is theoretically linked to a particular aspect of the structure-sensitive comparison processes. Comparisons that are internal to active memory (i.e., mapping) can result in learning of new mapping connections that code systematic correspondences between the two analogs. The general principle (LISA's explanation for Phenomena 1 and 2 in Table 1) is that access must rely on preexisting connections, whereas mapping can create new ones.

The division of LISA's mapping operations into a hierarchy of temporal scales (with individual SPs at the fastest scale, combining into propositions, which combine into phase sets, which finally compose a complete mapping session) has important consequences for LISA's performance as a mapping engine. Competition (and therefore mapping) at the level of individual SPs operates on the basis of parallel constraint satisfaction: An SP in the driver generates a pattern of activation on the semantic units, and all units in the recipient compete in parallel to respond to that pattern. It is only at this very small scale that LISA's mapping algorithm has the power of full parallel constraint satisfaction. At the level of whole propositions, the algorithm only weakly approximates parallel constraint satisfaction. The activation of a P unit persists across multiple SPs, and P units excite the SPs to which they are connected. Therefore, mappings established by one SP can bias the mappings established for other SPs in the same proposition.

The main computational significance of the phase set for LISA is that mappings for different propositions in the same set do not constrain one another at all. Mappings established by one proposition are stored as weights that serve to bias subsequent access and mapping. Mapping weights are updated between phase sets (i.e., after one phase set ends and before the next one begins), so the biasing that results from the mapping connections operates only between separate phase sets: The ordering of propositions within a phase set has no impact on mapping. Effectively, propositions in the same phase set are considered in parallel. Accordingly, the size of the available phase set (i.e., the size of working memory) places an upper bound on the formal complexity of the analogies that LISA can map. Variations in this aspect of working-memory capacity may provide a partial explanation of some phylogenetic and ontogenetic differences in analogical competence.

Simulation Results

We now describe a series of simulations of analogical access and mapping using the LISA model. The program used in these simulations is written in TurboPascal (Version 7, Borland International, Inc., Scotts Valley, CA) and runs on an IBM-compatible 486 microcomputer. Our aim in this article is not to test the capacity of the model to scale up to large examples⁴ but rather to demonstrate its ability to account for the qualitative phenomena listed in Table 1. Accordingly, the simulations are based on small examples that illuminate specific phenomena. Most of the examples are based on simplified versions of materials used in empirical studies of human processing of analogies. Unlike most previous analogy models (with the exception of the Copycat system of Hofstadter & Mitchell, 1994), the performance of LISA is stochastic, potentially yielding different outputs for multiple runs on the same inputs. In addition, LISA is sensitive to the order in which propositions in the driver enter the phase set. Each simulation we report is based on a number of runs sufficient to establish a stable pattern of results. Unless otherwise indicated, simulations involved placing one proposition (with from one to three arguments) at a time in the phase set, so that the mapping weights were updated after each proposition.

It is necessary to specify dependent measures that can be used to relate the performance of LISA to that of people. For mapping, the most common dependent measure used in human studies is some form of accuracy measure. For LISA, the accuracy of specific mappings can be identified with the learned weights on the relevant mapping connections. Accordingly, we use the mean advantage of the learned weight for the correct correspondence relative to the strongest competitor as an index of LISA's mapping accuracy. To model experiments in which latency is the primary dependent measure, we use the number of cycles and/or propositions required for LISA to generate the correct set of correspondences.

To provide a measure of analogical access, we use a retrieval index based on the fit between the driver (i.e., retrieval cue) and each analog in LTM. Two retrieval indices—a raw index and a Luce index—are computed for each recipient or dormant analog. The raw retrieval index, R_i , for analog i in LTM is computed as a Weber function of the sum of maximum P unit activations, j , in i :

$$R_i = 100 \left(\frac{\sum_{j=1}^{N_i} \alpha_j}{(0.5 + N_i)} \right)^2, \quad (1)$$

where α_j is the maximum activation obtained (during the run) by P unit j in analog i , and N_i is the number of propositions in analog i . (The scaling constant, 100, was chosen simply to keep

³ Each phase lasts only $\frac{1}{40}$ th of a second and there are only 4–6 phases in a set, yet we assume that each phase set is maintained for a few seconds. The reason is that, for the sake of stability, each phase is assumed to become active several times during the duration of the phase set.

⁴ Our microcomputer is running under Microsoft DOS (Version 5.0, Microsoft Corp., USA, Redmond, WA), the memory limitations of which severely limit the size of the simulations we can run.

retrieval indices roughly in the range of 0.01 to 2.00.) R_i is proportional to the number of propositions in i that become active in response to the driver and the degree to which each becomes active. The Weber fraction (Marshall, 1995) allows analogs that have few propositions to compete on an equal footing with analogs that have many; it is superior in this regard both to a raw sum of activations (which would favor retrieval of large analogs over small ones) and to a raw proportion (which would favor small analogs over large ones). The squaring serves to enhance small differences between the Weber fractions of different analogs. This retrieval index can be computed even if only a single analog at a time is stored in LISA's LTM and is directly proportional to the degree of fit between the driver and the recipient analog.

We assume that during retrieval, analogs compete through shunting (divisive) inhibition. The effect of such inhibition is captured by the Luce retrieval index, L :

$$L_i = \frac{R_i}{\sum_j R_j}, \quad (2)$$

which is mathematically equivalent to subjecting each R_i to Luce's (1959) choice axiom. Wharton, Holyoak, Downing, Lange, & Wickens (1991) have shown that human analog retrieval fits well to the predictions of the Luce choice model.

Relationship Between Analogical Access and Mapping

We begin by reporting simulations of phenomena that bear on the relationship between analogical access and mapping (Table 1).

Similarity and role binding in access and mapping. Ross (1987, 1989) has investigated the influence of two types of similarity—general domain similarity (i.e., similarity of objects and predicates), and consistency of role bindings (i.e., whether or not similar objects played similar roles)—on both analogical access and mapping. His results highlight both similarities and differences between access and mapping. The pattern Ross observed is complex and, to the best of our knowledge, has never been simulated by any computational model of analogy. This set of simulations therefore provides a challenging test of the psychological adequacy of LISA, particularly with respect to our claim that the model provides a unified account of access and mapping.

Ross gave college students word problems illustrating the use of equations to solve probability problems (e.g., permutation and combination problems). He then gave them new word problems that differed in their semantic and structural relationships to the studied source problem. For example, the source might involve assigning cars to mechanics at the IBM motor pool (e.g., for repair). In the illustrated equation for permutations, the number of cars instantiated a variable n representing the total number of items from which an ordered subset of a fixed size is selected. The subsequent target problem varied whether the basic story line was similar (e.g., another car assignment problem at the motor pool vs. a formally equivalent problem involving assignment of students at a high school). A second dimension of variation involved bindings of objects into roles,

that is, which objects (humans or artifacts) served as n in the equation. Recall that in the source problem the two major types of objects were humans (mechanics) and artifacts (cars), and the artifacts filled the role of n in the permutation equation. The test problem could either be consistent with the source (humans and artifacts, with the artifacts as n), inconsistent (cross mapped) with the source (humans and artifacts, with the humans as n), or neutral (two categories of humans, one of which served as n).

Ross tested both access (participants' ability to retrieve the source problem given the cue) and mapping (their ability to use the source as a basis for solving the target) as a function of the semantic and structural relations between source and target. Table 3 (adapted from Table 3 in Ross, 1989, p. 464) summarizes his major qualitative findings. Following the notation used by Ross, each condition is denoted by an expression in the form $\{+, 0\} / \{+, 0, -\}$, with one symbol selected from within each set of brackets. The first symbol represents similarity of story line (+ for close, 0 for far). The second symbol represents the similarity of objects in matching roles (+ for similar, 0 for neutral, - for dissimilar). For example, the condition $+/-$ represents a target problem with a similar story line as the source problem (e.g., both involving car repairs at a motor pool) but with similar objects playing dissimilar roles (e.g., mechanics instead of cars as n).

Table 3 shows four sets of ordinal comparisons between conditions, for both access and mapping. The data provided by Ross (1989) for the first comparison, $(+/+) - (+/-)$, reveal that when the overall story line is similar, assigning similar objects to dissimilar rather than similar roles impairs both access and mapping. That is, the difference $(+/+) - (+/-)$ evaluates to a positive number. The second comparison, $(0/+)$ - $(0/-)$,

Table 3
Associations and Dissociations Between Access and Mapping as Observed in Humans and Simulated by LISA

Condition ^a	Access		Mapping	
	Humans ^b	LISA ^b	Humans ^c	LISA ^d
$(+/+) - (+/-)$.22	.58	.19	1.98
$(0/+)$ - $(0/-)$.04	.17	.14	1.82
$(0/0) - (0/-)$.01	.09	.11	0.94
$(0/0) - (+/-)$	-.22	-1.3	.16	0.94

Note. Human data from Ross (1989, Table 3). Adapted from "Distinguishing Types of Superficial Similarities: Different Effects on the Access and Use of Earlier Problems," by B. Ross, 1989, *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 15, p. 464. Copyright 1989 by the American Psychological Association. Adapted with permission of the author.

^a Human access score based on proportion of cases in which appropriate formula was recalled.

^b LISA's score based on the raw retrieval index, R .

^c Human mapping score based on proportion of cases in which variables were correctly instantiated in formula.

^d LISA's score based on value of correct mapping minus value of highest incorrect mapping.

^e All scores are based on differences between the two indicated conditions, expressed as story line similarity-object correspondence similarity.

indicates that when the story line is dissimilar, assigning similar objects to dissimilar roles impairs mapping but has little impact on access. The third comparison, (0/0) – (0/–), shows a similar dissociation between mapping and access when the cross-mapped condition is compared to the neutral condition. Finally, the fourth comparison, (0/0) – (+/–), pits the two factors in opposition to each other: a neutral assignment of objects to roles in the absence of a similar story line (0/0) versus a similar story line with a cross-mapping of objects to roles (+/–). This comparison yielded a dramatic reversal of difficulty between access and mapping, with the similar, cross-mapped condition (+/–) yielding easier access but less accurate mapping than the dissimilar, neutral condition (0/0).

Table 4 shows the representations of simplified versions of the problems in one of Ross's item sets that were provided to LISA. These consist of (a) a list of propositions for each analog; and (b) for each object and predicate used in the propositions, a list of semantic features. The features are given descriptive

labels to aid readability; however, these labels have no import to LISA (nor do the labels for predicates, objects, and propositions). Each predicate is defined to have a certain number of arguments slots; by default, each slot is represented by a distinct set of semantic features. The critical properties of these representations for the purposes of simulating Ross's results are the following:

1. The predicates and objects in the source and target problems with similar story lines have greater semantic overlap than do those in problems with dissimilar story lines, and
2. The object filling the first role of "assigned to" plays the role of n in the relevant equation to be instantiated.⁵

These representations were used to simulate the four comparisons summarized in Table 3, for both access and mapping. In each simulation, the source analog (which was constant across all runs) was the recipient, and the target was the driver. Propositions in the driver entered the phase set one at a time, in the order listed in Table 4. The "assigned to" proposition in the target was selected twice as often as any other proposition, reflecting its greater pragmatic importance. The sole difference between LISA's operation in simulating access versus mapping was that learning of mapping connections was disabled during access runs. We report the raw retrieval index, R (Equation 1), as a measure of access. (There was only one analog—namely, the source—in LTM during any simulation, so the Luce fit was always 1.0.) As a measure of mapping, we recorded the mapping weights from the driver (target) objects to the critical source (recipient) object, "cars" ("cars" in the source analog corresponds to n in the equation). We treated a mapping as correct if it would yield the correct answer in the target problem: Does the participant (or LISA) treat the correct object as n in the equation? The specific measure of mapping accuracy (in this and all subsequent simulation reports) was taken as the weight of the mapping connection for the correct mapping minus the weight of the strongest connection for an incorrect mapping (with the latter truncated to zero if it was negative). For both access and mapping, difference scores were calculated for each of the four comparisons of conditions provided by Ross (1989).

The dependent measures derived from the simulations with LISA (mean values over three runs) are presented in Table 3, along with the comparable data for human problem solvers as reported by Ross (1989). LISA captures the major qualitative associations and dissociations between the influence of the two similarity factors on access and mapping. When story lines are similar, assigning similar objects to dissimilar roles impairs both access and mapping (comparison 1). However, when story lines are dissimilar, assigning similar objects to dissimilar roles continues to impair mapping but has little or no impact on access

Table 4
LISA Representations of Problems Used to Simulate Access and Mapping Results of Ross (1989)

A. Propositional representations of analogs	
Source analog ("cars assigned to mechanics"; cars→ n)	
P1 (work-at mechanics motor-pool)	
P2 (belong cars executives)	
P3 (assigned-to cars mechanics)	
Target analogs	
+/+ ("cars assigned to mechanics"; cars→ n)	
P1 (work-at mechanics motor-pool)	
P2 (belong cars salespeople)	
P3 (assigned-to cars mechanics)	
+/- ("mechanics assigned to cars"; mechanics→ n)	
P1 (work-at mechanics motor-pool)	
P2 (belong cars salespeople)	
P3 (assigned-to mechanics cars)	
0/+ ("computers assigned to students"; computers→ n)	
P1 (enrolled-at students high-school)	
P2 (assigned-to computers students)	
0/0 ("students assigned to counselors"; students→ n)	
P1 (work-at counselors high-school)	
P2 (enrolled-at students high-school)	
P3 (assigned-to students counselors)	
0/– ("students assigned to computers"; students→ n)	
P1 (enrolled-at students high-school)	
P2 (assigned-to students computers)	
B. Semantic features	
Objects	
motor-pool: location company workplace IBM	
executives: animate person job business exec1	
salespeople: animate person job business sales1	
cars: inanimate machine transport cars1	
mechanics: animate person job mechanic1	
high-school: location school workplace high1	
students: animate person unpaid student1	
counselors: animate person job teaching counsell1	
computers: inanimate machine info device computer1	
Predicates	
work-at: locative activity work1	
enrolled-at: locative state enroll1	
belong: state posses belong1	
assigned-to: trans passive alter-poss assign1	

⁵ The predicate *assign* was used in all of the representations because it allows reversals of object roles (i.e., whether people or artifacts serve as the assigned set versus the assignee set). Bassok, Wu, and Olseth (1995) have shown that mapping performance on transfer problems similar to those studied by Ross (1989) is also guided by people's knowledge about the schematic roles that the specified objects tend to play with respect to each other (e.g., prizes are typically assigned to students rather than vice versa). Our simulations do not address such interpretive effects. We thank Miriam Bassok for advice on the representations used in these simulations.

(comparisons 2 and 3). Moreover, the relative impact of the two types of similarity is reversed for access and mapping: For access, it is better to have a similar story line even if objects are cross-mapped, but for mapping it is better to have a dissimilar story line with a neutral assignment of objects to roles than to have a similar story line with cross-mapped objects (comparison 4).

LISA's ability to simulate the patterns of access and mapping observed by Ross (1989) follows directly from its basic principles of operation. During both access and mapping, structured comparisons are initially driven by semantic features (of both predicates and objects) shared by the driver and recipient. The only major difference between access and mapping is that the latter process, which operates on information in active memory, is able to exploit the mapping connections, thereby encouraging subsequent mappings that are structurally consistent with initial mappings. (Recall that the only other difference is that activation decays faster for access than for mapping.) Prior to access, however, the recipient or recipients are in a passive state in LTM, and hence mapping connections cannot be updated. Access must depend solely on preexisting connections (i.e., shared semantic features), whereas mapping can create new structural connections. Similarity of story line (which LISA models as overlapping of semantic features) is therefore more critical to access than to mapping.

The benefit of having similar objects in similar roles (i.e., the benefit of consistent mappings over cross-mappings) follows from the fact that SP units integrate inputs from both predicates and objects. If both the object and predicate components of a driver SP pattern converge in activating a single SP in the recipient, then that SP (and the P unit to which it is connected) will clearly win the inhibitory competition in the recipient: The SP and associated P units will become highly active. Conversely, if the object pattern activates one SP while the predicate pattern activates another, then the lateral inhibition between SP and P units will result in no SP or P units becoming highly active. Together, these tendencies produce overadditivity for recipient SPs that match on both the object and predicate patterns relative to SPs that match on only one or the other. This tendency operates in both recipient and dormant analogs (i.e., in both working and long-term memory), but its effects are compounded in working memory as mapping connections are learned. If the driver and recipient have little semantic overlap in the first place (i.e., when the story lines are dissimilar), then the influence of similarity at the level of role bindings will be minimal due to the low baseline activation of the recipient.

The simulations of the results of Ross (1989) demonstrate that LISA can stimulate Phenomena 1 and 2 as well as Phenomena 6 and 7 (see Table 1).

Access and mapping for close analogs, far analogs, and schemas. Phenomenon 3 in Table 1 involves the ordering of ease of access: Close analogs and schemas are accessed relatively easily, whereas far analogs are much more difficult to access. We applied LISA to simulate this finding by using a set of materials based on convergence problems, which have been used in several relevant empirical studies (Catrambone & Holyoak, 1989; Gick & Holyoak, 1980, 1983; Keane, 1986). The results of these studies reveal differences in ease of access and also suggest a dissociation between access and mapping. Once a hint

to use the source problem or problems is given, the frequency with which college students generate the convergence solution to the radiation problem is high and roughly comparable (generally over 80%), regardless of whether the source is a close analog, a far analog, or a schema induced from comparisons between multiple analogs.

Appendix B provides the propositional representations for the four convergence problems: the radiation problem (always used as the target), a close analog ("The Surgeon"), a far analog ("The General"), and a schema for the class of convergence problems. The semantic features for each object and predicate are also listed in Appendix B. The general notational conventions used are the same as for the representations used to simulate the results of Ross (1989; see Table 4). However, the representations used for the convergence problems are considerably larger and more complex. The schema representation was constructed to capture the major common aspects of the specific analogs. It therefore has fewer propositions and more abstract objects and predicates (i.e., each object and predicate has relatively few semantic features). The representations of the source analogs are simplified by omitting solution information, as access and mapping are necessarily driven by information that overlaps with the target (which lacks a solution).

Three sets of runs were performed, one for each source analog. In each run, the radiation problem served as driver, with its propositions entering into the phase set in the order in which they are listed in Appendix B. The retrieval runs (with learning of mapping connections disabled) yielded mean Luce retrieval indices of .43 for the close analog, .34 for the schema, and .23 for the far analog, matching the empirical ordering. The mapping runs yielded roughly comparable results for all three sets. The representation of the target problem includes seven objects (the doctor, the tumor, the stomach, the tissue, the ray source, the high-intensity rays, and the low-intensity rays). For both the close analog and the schema, all seven objects were mapped correctly (weights on correct mapping connections ranging from .65 to 1.00, with negative weights on all incorrect mapping connections); for the far analog, six of the seven objects mapped correctly (weights on correct mapping connections ranging from .51 to 1.00, with negative weights on all but one incorrect mapping connection).

In terms of LISA's operation, the greater ease of accessing the close than the far analog is a direct consequence of the greater number of semantic features that a close analog shares with the target. The ease of accessing the schema relative to the far analog has a more subtle basis. Like *R*, the raw analog retrieval index, LISA's predicate and object units use a Weber-law input rule (see Appendix A), favoring recipients that have a greater proportion of their units activated by the driver.⁶ Relative to objects and predicates in a far analog, objects and predi-

⁶ The use of a Weber-fraction input rule (Marshall, 1995) as in LISA eliminates various retrieval biases. For example, ARCS (Thagard et al., 1990), which does not normalize analog activation in proportion to analog size, tends to be biased toward retrieving large rather than small analogs, even when the larger analogs are less isomorphic (Forbus et al., 1995). This bias is removed when the ARCS algorithm is modified by adding the Weber-fraction computation (Eric Melz, personal communication, June 1995).

cates in a schema share about the same number of semantic features with objects and predicates in the target, but they include fewer unshared features. (Objects and predicates in a schema represent general classes rather than specific instances.) A target analog will therefore tend to activate a small abstraction based on a subset of its features (i.e., a schema) much more readily than it will activate a remote analog that includes the same subset of the target's features embedded among many additional mismatching features (as the remote analog will have larger denominator terms in the Weber law input rule). In response to the objects and predicates in any given target, the objects and predicates in a schema will tend to become more active than the objects and predicates in any distant analog. This same property of a schema will tend to extend to the level of propositions: A distant analog is likely to contain propositions that do not fit (i.e., are not structurally consistent with) a target, whereas a schema, which is an abstraction containing only the relevant propositions, will tend to contain fewer such non-matching propositions. This difference will also tend to result in a higher retrieval index from any particular target to a schema than to a distant analog. At the same time, however, a near analog will tend to become more active than a schema: Due to the non-zero constant (0.5) in the denominator of the Weber law, additional propositions in an analog only "penalize" that analog when they do not fit with the driver (target). A near analog with, say, ten propositions, all of which fit the driver, will become somewhat more active than a schema with only five propositions, all of which fit the driver.

Competition and role binding in access. Wharton et al. (1994, in press) have demonstrated that analogical access is inherently competitive (Phenomenon 4). These investigators showed both that retrieval is sensitive to role binding (greater access to analogs in which similar objects play similar roles as in the retrieval cue; cf. Ross, 1989) and that access to any individual analog is decreased by the presence of a strong competing case stored in LTM. LISA was used to simulate a simplified version of Experiment 2 as reported by Wharton et al. (1994). Each analog was a single proposition. A sample set consisted of three interrelated analogs, such as:

1. The judge was alerted by the rescue worker. (cue)
2. The lawyer was warned by the paramedic. (consistent role assignments)
3. The attorney cautioned the firefighter. (inconsistent role assignments)

In the original experiment, participants first studied a series of such sentences. After a brief delay, they saw a series of cue sentences and were asked to write down any sentence of which they were reminded. Wharton et al. varied two factors. First, relative to the cue (e.g., sentence 1), a studied sentence either had consistent role assignments (similar objects in similar roles, as in sentence 2), or inconsistent assignments (cross-mapped, as in sentence 3). Second, for a given cue, the study set included either one related sentence (e.g., either 2 or 3), or both.

Table 5 summarizes the results from Wharton et al. (1994), which revealed that access was more likely when (a) the role bindings were consistent rather than inconsistent, and (b) only one related sentence (singleton condition) rather than two (competition condition) had been studied. To simulate this pattern of results, we gave LISA the eight sentences used by Wharton et

Table 5
LISA's Luce Retrieval Index (Nonitalic) and Observed Retrieval Proportions (Italic)

Source analog	Cue condition		
	Singleton	Competition	
Consistent	.37	.80	.31
Inconsistent	.28	.67	.20

Note. Observed retrieval proportions (data) are from Wharton et al. (1994, Experiment 2).

al. to serve as the contents of LTM, including sentences 2 and 3 above. Sentence 1 always served as the retrieval cue (i.e., the driver). Three runs were performed: (a) singleton consistent condition, in which sentence 2 but not 3 was in LTM; (b) singleton inconsistent condition, in which sentence 3 but not 2 was in LTM (with the missing sentence being replaced in each of the above by an unrelated filler item); and (c) competitor condition, in which both sentences 2 and 3 were in LTM. Table 5 presents the Luce retrieval index in each condition. (Recall that the Luce index, L in Equation 2, is an index of competition.) The simulation results capture the major ordinal relations of the human recall data. In particular, in both the singleton and competition conditions, the Luce index is higher for consistent than inconsistent sentences; moreover, for both consistent and inconsistent sentences, the index is higher in the singleton than in the competition condition. For LISA, the advantage for consistent over inconsistent sentences is slightly greater in the competition condition than in the singleton condition. Although such an interaction was not observed in the particular experiment we simulated, it has been obtained in three similar experiments (Wharton et al., 1994, 1996).

The access advantage that LISA exhibits for the consistent over the inconsistent cases has the same basis as the model's fit to the comparable finding from Ross (1989; the advantage of the +/+ over the +/- conditions). The competition effect arises directly from LISA's use of lateral inhibition between stored cases (as reflected in the Luce retrieval index).

Familiarity and access. The final access phenomenon in Table 1, Phenomenon 5, is that highly familiar source analogs (e.g., people as an analog for other living things, or the Vietnam War as an analog of a new foreign crisis facing the United States) are especially likely to be retrieved (more likely than can be explained on the basis of similarity to the target). In its current state, LISA does not simulate this finding, but it is straightforward to understand how such an effect might obtain in terms of LISA's operation and architecture. Consider how LISA would learn a new analog. Encoding an analog into memory would entail establishing a new set of within-analog connections (e.g., from semantic units to object and predicate units, and from objects and predicates to SPs, etc.) to encode the analog's propositional content (see Hummel & Holyoak, 1996, in press). If this learning is assumed to be incremental (like virtually all learning in connectionist systems), then highly familiar analogs (i.e., analogs that have been encountered many times) would be expected to have higher within-analog connection weights than unfamiliar analogs. These higher weights

would manifest themselves in higher activation values for structure units (most important, proposition units) in familiar analogs than in less familiar analogs, so the former would be more readily retrieved from memory than the latter (see Equation 1).

Analogical Mapping

We now present a series of simulations that focus specifically on phenomena involving analogical mapping (Table 1). The impact of isomorphism (Phenomenon 6) and semantic similarity (Phenomenon 7), as well as the manner in which they interact, have already been illustrated in the simulations of the mapping results of Ross (1989), described earlier. We will now consider pragmatic centrality (Phenomenon 8) and multiple mappings (Phenomenon 9).

Pragmatic centrality and the resolution of ambiguous mappings. Spellman and Holyoak (1996) reported a series of experiments that investigated the impact of processing goals on the mappings generated for inherently ambiguous analogies. In Experiment 2, participants read two science-fiction stories about countries on two planets. These countries were interrelated by various economic and military alliances. Participants first made judgments about individual countries based on either economic or military relationships and were then asked mapping questions about which countries on one planet corresponded to which on the other. Schematically, Planet 1 included three countries, such that "Afflu" was economically richer than "Barebrute," whereas the latter was militarily stronger than "Compak." Planet 2 included four countries, with "Grainwell" being richer than "Hungerall" and "Millpower" being stronger than "Mightless." These relationships can be summarized in simple two-proposition representations, which were used as input to LISA:

Planet 1

- P1 (richer Afflu Barebrute)
- P2 (stronger Barebrute Compak)

Planet 2

- P1 (richer Grainwell Hungerall)
- P2 (stronger Millpower Mightless).

The critical aspect of this analogy problem is that Barebrute (Planet 1) is both economically weak (like Hungerall on Planet 2) and militarily strong (like Millpower) and therefore has two competing mappings that are equally supported by structural and similarity constraints. Spellman and Holyoak found that participants whose processing goal led them to focus on economic relationships tended to map Barebrute to Hungerall rather than to Millpower (43% vs. 35%, respectively), whereas those whose processing goal led them to focus on military relationships had the opposite preferred mapping (29% versus 65%). The variation in pragmatic centrality of the information thus served to decide between the competing mappings.

Spellman and Holyoak (1996) simulated the impact of pragmatic centrality using the ACME model, under the assumption that information about the analogs that is relatively less important to the active processing goal is inhibited. LISA's architecture provides a similar mechanism that makes mapping sensitive to the relative importance of information in active memory.

As noted earlier, we assume that people tend to think about information they regard as important more often and earlier than information they regard as unimportant. We simulate this in LISA by selecting important propositions in the driver earlier and more often than less important propositions. We tested the impact of pragmatic centrality in LISA using the above propositional representations of the two analogs, with semantic representations in which all planets were equally similar in terms of feature overlap. Planet 1 served as the driver. Five levels of pragmatic focus were defined. In the Econ-hi condition, the economic proposition P1 was selected three times, after which the military proposition P2 was selected once; in Econ-lo, P1 was selected once followed by P2 once. Two conditions favoring the military proposition (P2) over the economic proposition (P1) to varying degrees (Mil-hi, Mil-lo) were defined by reversing the above biases. In all four of these conditions, the mapping connections were updated after each proposition was selected (i.e., there was always only one proposition in the phase set), so that earlier propositions were able to influence the mapping weights before the latter propositions were selected. In the Neutral condition, P1 and P2 were both selected once in a single phase set, thus eliminating any variation in either frequency or order of selection.

Table 6 presents the mean weights on the two competing connections for the ambiguous country, Barebrute, based on two sets of runs for the seven conditions. These simulations reveal an orderly shift in the difference between the strength of the economic-based mapping (Hungerall) and the military-based mapping (Millpower) with the frequency and order of selection of the two driver propositions. These results show both that LISA is sensitive to pragmatic centrality and that it can resolve an inherently ambiguous mapping.

We also simulated the schematic design of Spellman and Holyoak's (1996) Experiment 3, which is presented in Table 7. In this experiment, participants read plot summaries for two elaborate soap operas. Each plot involved various characters interconnected by professional relations (one person was the boss of another), romantic relations (one person loved another), and cheating relations (one person cheated another out of an

Table 6
LISA's Mappings for Ambiguous Country ("Barebrute") as a Function of Emphasis on Economic Versus Military Propositions

Emphasis	Weight on mapping connection ^a		
	"Hungerall" (economic)	"Millpower" (military)	Difference
Econ-hi	.63	.05	.58
Econ-lo	.79	.71	.08
Neutral	.68	.64	.04
Mil-lo	.74	.77	-.03
Mil-hi	.08	.65	-.57

Note. Econ-hi = high emphasis on the economic relation; Econ-lo = low emphasis on the economic relation; Neutral = emphasis on neither the economic nor the military relation; Mil-lo = low emphasis on the military relation; Mil-hi = high emphasis on the military relation.

^a Means over two simulation runs.

Table 7
LISA's Schematic Propositional Representations of the Relations Between Characters in the "Soap Opera" Experiment of Spellman and Holyoak (1996)

Plot 1	Plot 2
Bosses (Peter, Mary)	Bosses (Nancy, John)
Loves (Peter, Mary)	Loves (John, Nancy)
Cheats (Peter, Bill)	Cheats (Nancy, David) Cheats (Lisa, John)

inheritance). Each of the three relations appears once in Plot 1 and twice in Plot 2. Based solely on the *bosses* and *loves* relations, Peter and Mary in Plot 1 are four-ways ambiguous in their mapping to characters in Plot 2: Peter could be mapped to any of Nancy, John, David, or Lisa; Mary could be mapped to any of those same characters. If the *bosses* propositions are made more important than the *loves* propositions (i.e., if a processing goal provided a strong reason to map *bosses* to *bosses* and hence the characters in the *bosses* propositions to each other), then Peter should map to either Nancy or David and Mary would correspondingly map to either John or Lisa. There would be no way to choose between these two alternative mappings (ignoring gender, which in the experiment was controlled by counterbalancing). However, if the *cheats* propositions are considered, then they provide a basis for selecting unique mappings for Peter and Mary: Peter maps to Nancy so Mary maps to John. In a similar manner, if the *loves* propositions are made important (absent *cheats* propositions), then Peter should map to John or Lisa and Mary would correspondingly map to Nancy or David; if the *cheats* propositions are also considered, then Peter should map to Lisa and, hence, Mary to David.

Spellman and Holyoak (1996) manipulated participants' processing goals by having them produce an analogical extension of Plot 2 based on Plot 1, where the extension involved either professional or romantic relations; the cheating relations were always irrelevant. Participants' preferred mappings were revealed on this plot-extension task by their choice of Plot 2 characters to play roles analogous to those played by Peter and Mary in Plot 1. Afterwards, participants were asked directly to provide mappings of Plot 1 characters onto those in Plot 1.

The results are presented in Figure 5A, which depicts the percentage of mappings for the pair Peter–Mary that were consistent with respect to the processing goal (collapsing across focus on professional vs. romantic relations), inconsistent with the processing goal, or some other response (e.g., a mixed mapping). The goal-consistent and goal-inconsistent mappings are further divided into those consistent versus inconsistent with the irrelevant cheating relation. The plot-extension and mapping tasks both proved sensitive to the processing goal, as participants' preferred mappings in both tasks tended to be consistent with the goal-relevant relation; however, this effect was more pronounced in the plot-extension than the mapping task. In contrast, consistency with the cheating relation was more pronounced in the mapping than the plot-extension task. Spellman and Holyoak (1996) modeled this qualitative pattern using ACME by assuming that goal-irrelevant relations were inhibited to a greater degree in the plot-extension task (where the goal focused directly on one type of relation, either professional or

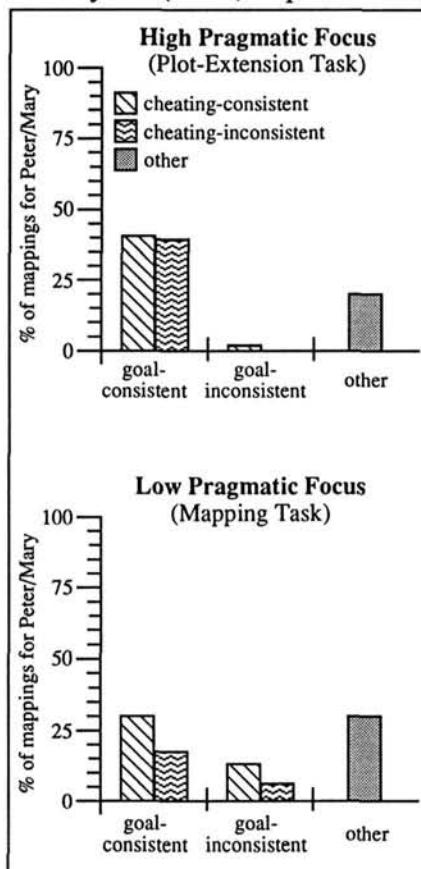
romantic) than in the mapping task (which was only indirectly influenced by the goal in the preceding plot-extension task).

To simulate the results of Spellman and Holyoak's (1996) Experiment 3, LISA was given schematic representations of the two plots on the basis of the propositions in Table 7. Plot 1 always served as driver. Four levels of pragmatic focus were generated. To counterbalance the order in which the low-focus propositions were selected, we ran two different selection orders at each level of focus. For clarity, we denote the proposition (in the driver) stating the professional relationship (the proposition listed first in Plot 1; see Table 7) as P, the proposition stating the romantic relationship as R, and the proposition stating the cheating relation as C. High focus on the professional relationships (P-hi) was simulated by selecting P four times for every one time R and C were selected. The orders used were [P P P R P C P] and [P P C P R P]. Low focus on the professional relationships (P-lo) was simulated by selecting P three times for every two that R or C were selected, giving the orders [P R C P C R P] and [P C R P R C P]. Note that, in addition to being selected more often, P (the proposition stating the professional relationship, which was the relationship of focus) was also the first and last to be selected in both the P-hi and P-lo orders. High and low focus on the romantic relationships (R-hi and R-lo) were defined by switching the selection of R and P in the P-hi and P-lo conditions, respectively; thus R-hi was [R R P R C R] and [R R C R P R], and R-lo was [R P C R C P R] and [R C P R P C R]. Each order was run 10 times, for a total of 20 runs in each condition (P-hi, P-lo, R-hi, and R-lo), and a total of 40 high-focus and 40 low-focus runs. On each run, we recorded whether the dominant mappings (as given by the strongest mapping weights for Peter and Mary in Plot 1) were jointly consistent with the professional relationship, the romantic relationship, or neither, and whether or not they were also consistent with the cheating relation.

The results of the LISA simulations are summarized in Figure 5B, using a format that matches the data from Spellman and Holyoak's (1996) Experiment 3 (presented in Figure 5A). The simulations were based on the assumption that low pragmatic focus corresponded to Spellman and Holyoak's mapping task and that high focus corresponded to their plot-extension task. The simulations thus simplified the situation posed to participants in Spellman and Holyoak's experiment by replacing an inference task (plot extension) with a mapping task. However, given that analogical inference necessarily depends on the correspondences computed in the mapping stage, this simplification appears reasonable.

We did not attempt to produce a detailed quantitative fit between the simulation results and human data; nonetheless, a

A. Results of Spellman & Holyoak (1996, Experiment 3)



B. LISA Simulation Results

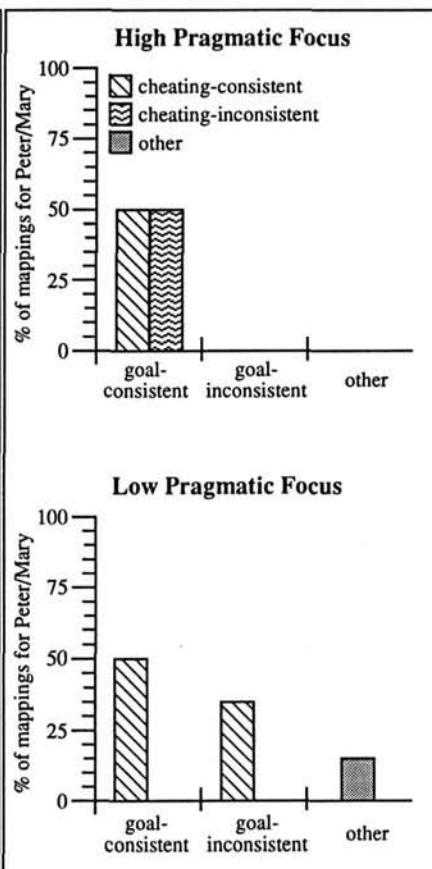


Figure 5. A: Summary of the results of Spellman and Holyoak (1996), Experiment 3. Adapted from "Pragmatics in Analogical Mapping," by B. A. Spellman and K. J. Holyoak, 1996, *Cognitive Psychology*, 31, p. 332. Copyright 1996 by Academic Press. Adapted with permission. B: LISA's simulation results under different levels of pragmatic focus.

close qualitative correspondence is apparent, with LISA generally yielding a cleaner form of the pattern obtained with people. In the high pragmatic-focus simulations (collapsing across the P-hi and R-hi conditions), LISA's Peter-Mary mappings were invariably consistent with the goal-relevant relation (either professional or romantic) but were entirely independent of the irrelevant cheating relation. LISA effectively ignored the constraint imposed by the cheating relation, instead producing, with equal probability, one of the two Peter-Mary mappings consistent with the goal-relevant relationship. In the low pragmatic-focus simulations (collapsing across the P-lo and R-lo conditions), LISA was again more likely to produce mappings that were goal-consistent rather than goal-inconsistent, but this preference was much weaker than in the high pragmatic-focus simulations. On the other hand, in the low-pragmatic-focus simulations, LISA reliably selected mappings that were consistent with the cheating relation. Thus LISA, like people, produced mappings that were dominated by the goal-relevant relation alone when pragmatic focus was high but that were also influenced by goal-irrelevant relations when pragmatic focus was low.

These results illustrate two important properties of LISA's mapping algorithm. First, as illustrated by the results of the high pragmatic-focus simulations, LISA will stochastically produce different mappings, each consistent with the goal-relevant relation but independent of goal-irrelevant relations, when the latter are only given a limited opportunity to influence mapping. Second, as illustrated by the low pragmatic-focus simulations, small differences in selection priority (i.e., order and frequency) are sufficient (when no single relation is highly dominant) to bias LISA in one direction or another in resolving ambiguous mappings. More generally, the simulations of the Spellman and Holyoak (1996) results demonstrate that LISA is sensitive to the constraint of pragmatic centrality (Phenomenon 8) and that it can successfully find alternative mappings for a single ambiguous analogy (Phenomenon 9).

Mapping performance with an "unnatural" analogy problem. We now examine LISA's ability to map variants of an "unnatural" analogy, illustrating how it deals with Phenomena 10 and 11 in Table 1. As noted earlier, LISA operates in parallel (or semi-parallel) with respect to the proposition or propositions

that fire together within a single phase set (i.e., prior to updating the mapping connections). However, in so far as working-memory limits restrict the number and complexity of propositions that can fit into a phase set, LISA must necessarily build mappings incrementally (in a similar manner to such systems as IAM; Keane et al., 1994).

These properties cause major variations in the model's performance as a function of the structure of the analogy. LISA's performance with the convergence analogs indicates that its incremental approach to mapping can successfully mimic the results of parallel constraint satisfaction under some circumstances, even when the analogs are relatively large and semantically dissimilar. Despite their size and complexity, the convergence analogs are natural in that (a) they contain semantically similar predicates that aid in finding an approximately isomorphic structure, and (b) each analog exhibits a high degree of textual coherence, with propositions interconnected by role-filler relationships. The first property plays a critical role in LISA's ability to bootstrap the mapping (that is, to find at least a few correspondences early in mapping), and the second plays a critical role in its ability to use those established mappings to constrain otherwise ambiguous future mappings.

Not all analogies have these desirable properties. We will now examine LISA's performance on the boys–dogs analogy (Table 2), a mapping problem that lacks them. As we argued earlier, this problem is intuitively unnatural. People seem to solve it with difficulty, using re-representation strategies based on counting the occurrences of objects and predicates in each analog. Such strategies are unavailable to LISA, and initial explorations revealed that LISA has great difficulty solving this problem. It failed to generate the six correct correspondences (three object mappings and three predicate mappings), not only when one proposition at a time was placed in the phase set but even when an entire analog (five propositions) was placed in the phase set. Only when all propositions in *both* analogs were placed in the phase set together did LISA succeed reliably using the representations in Table 2. The size of the phase set required for success (10 SPs) clearly exceeds the postulated capacity of human working memory. LISA thus predicts that the boys–dogs problem could not be reliably solved by humans without the use of re-representation strategies or external memory aids.

We then explored LISA's performance with variants of the boys–dogs problem that increased either the referential coherence of the individual analogs or the semantic similarity of the analogs to each other. In all of these runs, the boys analog served as driver. Keane et al. (1994) showed that human performance on this problem improves if the analogs are presented in a format that encourages participants to begin by forming a correct mapping between a pair of propositions. In their Experiment 2B, the two analogs appeared side-by-side on a sheet of paper. One version was formatted much like Table 2, with no highlighting of correct correspondences. A second version used the same propositions, but reordered so that the first proposition in each analog described a singleton, an individual with only one rather than two associated properties (i.e., "Steve is smart" and "Fido is hungry"). These two propositions were thus visually highlighted as a potential match, which is in fact correct. People reached the correct solution more rapidly in the latter, singleton-first condition.⁷

LISA can simulate this order effect by assuming that people notice the first proposition in each analog list, and re-represent it with some common predicate, such as *first* or *singleton*. Accordingly, we augmented the representation of each analog in Table 2 with one additional proposition stating explicitly which fact was first. When these marked propositions in fact corresponded (singleton-first condition), LISA correctly mapped all of the objects and predicates across the two analogs, even when only one proposition at a time entered the phase set. In this case, presentation format influences the semantic similarity of the analogs; the similarity constraint then bootstraps the incrementally emerging mappings, which are subsequently supported by the repeated predicates and objects. This result indicates that LISA can account for Phenomenon 10: An initial correct correspondence aids in finding subsequent mappings.

In addition to showing an effect of proposition order, Keane et al. (1994, Experiment 1B) also demonstrated that people map the boys–dogs analogy more easily if some or all of the corresponding predicates are semantically similar. LISA was applied to two modified versions of the analogy that matched conditions used in Keane et al.'s study. In the one-similar condition, the "Fido is hungry" proposition in Table 2 was changed to "Fido is clever," creating a similarity with the predicate in the corresponding boys proposition, "Steve is smart." In the all-similar condition, the predicate in each of the dogs propositions was replaced with one that was similar in meaning to the corresponding boys proposition.

LISA successfully solved both variants with one proposition at a time entering the phase set. An analysis of LISA's mapping performance during the first cycle revealed that, on average, it approaches the solution more rapidly in the all-similar condition than in the one-similar condition (although, in both cases, there is an effect of the order in which propositions are selected to fire). We ran LISA for one cycle on the one- and all-similar analogies and recorded its mapping progress after one, two, three, and five propositions. Mapping connections were updated after each proposition. As a mapping score, we calculated the number of correct predicate mappings from Analog 1 (boys) to Analog 2 (dogs). (The number of correct object mappings did not differ between the two similarity conditions.) As shown in Figure 6, the number of predicates correctly mapped grows more rapidly in the all-similar condition than in the one-similar condition, although the difference is small. Perhaps more interesting is the fact that the growth curve is smoothly negatively accelerated in the all-similar condition but not in the one-similar condition. This difference reflects the fact that, in the all-similar condition, each predicate in Analog 1 can activate its corresponding predicate in Analog 2 on the basis of semantics alone. In the one-similar condition, only one predicate mapping (*smart*

⁷ Keane et al. (1994) attributed their finding of a singleton-first advantage to the fact that a singleton proposition was listed first in each analog. However, their results are consistent with the more general possibility that mapping will be facilitated whenever the first propositions in each of the two analogs in fact correspond to one another. That is, Keane et al.'s design confounded "singleton is first in each analog" with "first propositions in each analog correctly map to one another." In the absence of evidence to the contrary, we assume that the latter, more general interpretation holds (Phenomenon 10 in Table 1).

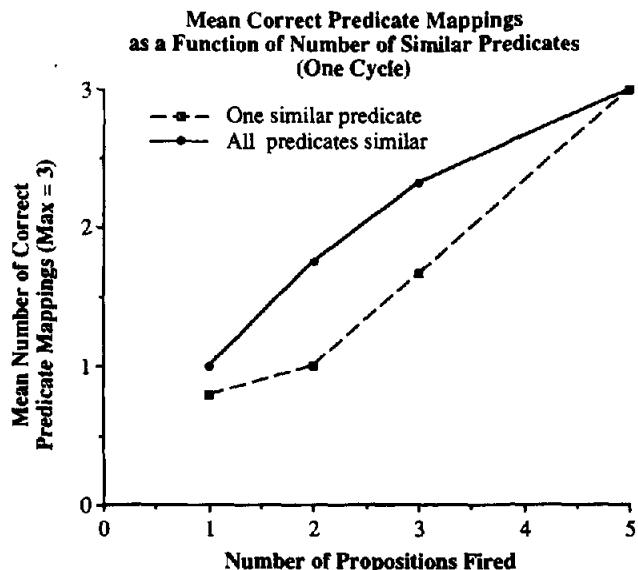


Figure 6. Mean number of correct predicate mappings (from the boys analog to the dogs analog) as a function of the number of propositions fired (1, 2, 3, or 5) and the number of semantically similar predicates (one similar vs. all similar). Means are calculated over two runs of each selection order. There were 5 orders in the one-proposition case, 4 in the two-proposition case, 3 in the three-proposition case, and 6 in the five-proposition case.

to *clever*) has this advantage; all others must be generated on the basis of the structural constraints embodied in other emerging mappings. These simulations illustrate that LISA is sensitive to semantic similarity in mapping (Phenomenon 7 in Table 1).

Table 8 summarizes the qualitative results of the simulations based on the boys-dogs problem. LISA is unable to reliably solve the unnatural basic version of this mapping problem (Phenomenon 11), unless it is granted a psychologically unrealistic working-memory capacity. However, it can solve the problem within realistic capacity limits when the representations are augmented to establish a correct initial correspondence and when similarity of predicates contributes to finding the isomorphic correspondences.

Finding mappings that violate the n-ary restriction. LISA's

failure to solve the basic boys-dogs analogy under realistic assumptions about its working-memory capacity suggests that the model (unlike ACME) is not overpowered relative to humans in its ability to map unnatural analogies. A related question concerns whether or not LISA is *underpowerful* in the way that previous models are. As discussed earlier, all previous analogy models obey an inviolable *n*-ary restriction, which prohibits mapping a predicate with *n* arguments to one with a number of arguments other than *n*. To test whether LISA can find apparently natural mappings that require freedom from the *n*-ary restriction, we gave it the two mapping problems based on size relations that we discussed earlier. For the mapping from "Abe is tall" and "Bill is short" onto "Chris is taller than Dean," the semantic representations of *tall* and the first role of *taller*, as well as of *short* and the second role of *taller*, were assumed to share one common feature (out of a total of four features each). This modest semantic overlap was sufficient to allow LISA to map *Abe* to *Chris* and *Bill* to *Dean*—mappings that require rejecting the *n*-ary restriction.

The second example involved mapping "Abe is taller than Bill" and "Bill is taller than Charles" (i.e., two binary transitive relations) onto one proposition based on the trinary predicate "top-to-bottom," with three ordered arguments (top, middle, and bottom). We assumed that the semantic features of the *top* role included three of four features representing the first role of *taller* and that the features for the *bottom* role included three of four features representing the second role of *taller*. The features of the *middle* role were assumed to include the union of the above features (reflecting the fact that the "middle" person is the one who is both taller than someone and less tall than someone). Although the exact numbers of unique and overlapping features assumed for each role are arbitrary, this general pattern of role overlap is consistent with a plausible basis for learning the trinary predicate by summing the role features of objects from two size orderings that are mapped on the basis of binary relations (a point to which we return in the General Discussion). Using the two binary relations jointly in the phase set, LISA successfully mapped the two binary propositions onto the single trinary proposition, generating mappings of *Abe* to *top*, *Bill* to *middle*, and *Charles* to *bottom*. These simulations reveal that LISA is indeed able to find sensible mappings that violate the *n*-ary restriction (Phenomenon 12). The model thus demonstrates a degree of flexibility in mapping

Table 8
Summary of LISA's Mapping Performance on Variants of the Boys-Dogs Problem

Variant	No. propositions in phase set	Performance
Basic (see Table 2)		
No similar predicates	1	Failure
No similar predicates	5 (1 analog)	Failure
No similar predicates	10 (both analogs)	Success
With one similar predicate	1	Success
With all predicates similar	1	More rapid success
Augmented		
With "first" propositions embedding singleton proposition in each analog	1	Success

that surpasses that of any previous computational model (but appears to be equaled by human reasoners).

Phylogenetic and Ontogenetic Change

We now consider how LISA may account for the increases in analogical ability that occur across species (Phenomenon 13) and over the course of human cognitive development (Phenomenon 14). LISA does not by any means provide a definitive explanation or even description of phylogenetic and developmental change; rather, as a theory of the adult human architecture for analogical comparison, it provides a set of potential hypotheses about constituent mechanisms that may have evolved and may undergo maturation. Adult human competence in analogy, as modeled by LISA, has the following requirements:

1. The capacity to perform dynamic binding (which is necessary for propositional thought),
2. Hierarchically organized structure units that store bindings in LTM,
3. Both parent and daughter modes of processing (to enable mapping of hierarchical propositions despite the one-level restriction),
4. The capacity to learn mapping connections, and
5. Attentional control of a phase buffer (working memory) that strategically drives the comparison process.

This list, although not exhaustive, at least provides some relatively specific hypotheses about capacities that may undergo phylogenetic or developmental change. Our final set of simulations explores one particularly simple variable in LISA that may account for differences in the complexity of analogies that can be mapped by different individuals. This variable is the size of the phase set: the number of distinct role bindings (SPs) that can be considered together before updating the mapping connections. Developmental and phylogenetic limitations on the size of the phase set may reflect a number of specific factors. The most intuitive is physiological: As noted in the Introduction, the temporal properties of neural firing necessarily limit the number of groups that can be simultaneously active but mutually desynchronized. However, this is by no means the only variable (or necessarily even one of the variables) that might affect phase set capacity. Others include (a) attentional and strategic variables (i.e., what an individual knows to place into the phase set together, as related to requirement 5 above), (b) differences in the way a problem is represented, and (c) differences in the attentional–cognitive resources required by other aspects of the task or by other tasks that must be performed concurrently (e.g., a person who is distracted may evidence a smaller phase set capacity than a person who is free to devote full resources to the mapping task). We are not committed to any strong claims about the origins of developmental or phylogenetic differences in phase set size. Rather, we are more interested in how such differences may manifest themselves in an individual's ability to perform mapping tasks of varying complexity.

Halford and his colleagues (Halford, 1992, 1993; Halford & Wilson, 1980) have proposed that the maximum complexity of mappings that can be processed by a reasoner is limited by working-memory capacity, as defined by the number of independent dimensions of variation that jointly constrain a decision. They have also noted that each argument of a predicate (i.e.,

each role binding) can be interpreted as a dimension of variation. Halford et al.'s (1994) STAR model of mapping provides a computational realization of this view of capacity limits by using a tensor-product representation. In their model, each argument must be assigned to a separate dimension in the tensor product. Although LISA achieves role binding in a very different way (by synchrony rather than conjunctive units), its approach is consistent with Halford's general analysis of the capacity requirements for mapping. In particular, each additional available slot in the phase set permits an additional role binding to be considered in parallel (i.e., prior to revising mapping connections): Slots in LISA's phase sets map (roughly) onto dimensions in Halford et al.'s tensor products.

It follows that increases in the size of the phase set will be accompanied by increases in the maximal complexity of the analogical mappings that can be computed, where complexity is defined according to a metric of the sort proposed by Halford and colleagues. Halford (1993) argues that human cognitive development involves maturational stages related to increases in the number of dimensions that can be considered together. He distinguishes four levels of mapping complexity based on the maximal number of dimensions that can be processed in parallel: one (attribute mapping, attained by roughly age 1 year), two (relational mapping, at about age 3 years), three (system mapping, about age 5 years), and four (multiple system mapping, about age 12 years).

Based on their performance on tasks such as match-to-sample, it can be argued that primates in general, including monkeys, are capable of mapping objects on the basis of similar attributes (i.e., one-dimensional representations, equivalent to a single SP in LISA; Holyoak & Thagard, 1995). Work by Gillan et al. (1981) indicated that with training in explicit coding of binary relations, chimpanzees can acquire the capability to map on the basis of a single binary relation (i.e., two-dimensional representations, or two SPs). This capability is evidenced in the success of "language"-trained chimpanzees on proportional analogies based on familiar relations. Similar problems are within the competence of 4-year-old human children (Goswami, 1989; Goswami & Brown, 1989). There seems to be no compelling evidence that any nonhuman species can reliably map problems at higher complexity levels than relational mapping.

The direct equivalence of number of SPs in LISA and number of independent dimensions according to Halford's taxonomy breaks down at the level of system mapping and beyond. The reason is that in Halford et al.'s (1994) taxonomy, the number of independent dimensions can be fewer than the number of overt role bindings. A good example is provided by transitive inference, as in the mappings of size orderings considered earlier. Halford et al. (1994) argued that a composition of two binary relations (which in LISA must involve four SPs) is equivalent in complexity (system level) to a single trinary relation (three SPs in LISA). This type of equivalence between representations with different syntactic structures illustrates the importance of developing a mapping model that breaks free of the *n*-ary restriction. As we demonstrated previously, LISA is indeed able to map two binary transitive relations onto a single trinary relation, thereby discovering the type of correspondence that can be derived from Halford et al.'s formal analysis. LISA's ability to map syntactically distinct structures that are equivalent

in information and formal complexity leads to the possibility that inductive mechanisms could allow the model to form new predicates that effectively compress the information in multiple propositions into a single proposition (as in the translation from two binary *taller* propositions to the single trinary *top-to-bottom* proposition).

The version of LISA presented here does not include such compression mechanisms (although we have implemented them in a different version); accordingly, our simulations of the model's performance on problems at the system level and beyond were simplified by allocating enough slots in the phase set to hold all of the overt SPs at a given complexity level (rather than the formally definable minimal number). Our goal was to ascertain whether the model, in fact, achieves reliability in mapping problems at successive levels of complexity as the phase set is suitably expanded.

We have already shown that when the phase set can include two binary relations, LISA can map them successfully onto an equivalent trinary relation. This performance is at the system level of Halford et al.'s (1994) taxonomy of complexity. We also tested LISA's ability to map two binary relations representing a size ordering (e.g., "Abe is taller than Bill" and "Bill is taller than Charles") onto two binary relations describing an ordering on a semantically dissimilar dimension (e.g., "Communism is left of socialism" and "Socialism is left of capitalism"). The representations of the objects (people and political systems) had no features in common; the relations (*taller than* and *left of*) had one shared feature (out of four). When only one binary relation at a time was allowed in the phase set (relational mapping), LISA was unable to reliably find the isomorphic mappings (i.e., Abe → communism, Bill → socialism, Charles → capitalism). However, when the phase set was expanded to include two binary relations (system mapping), LISA consistently found all of the correct object mappings. This qualitative improvement in performance with increased size of the phase set captures the shift from relational mapping (which depends on similarity of first-order relations) to more abstract system mapping (which can find correspondences based on isomorphism even when the mapped relations are highly dissimilar).

To determine whether a comparable shift in LISA's performance with size of phase set would obtain when the overall size of the analogs was increased, we also simulated mapping performance for a structure-learning task that Halford and Wilson (1980, Experiment 1) administered to 4-year-old and 5- to 6-year-old children. The children saw the spatial layout sketched in Figure 7. This layout is based on a mathematical group structure, the cyclic 4-group. A cyclic 4-group is defined by four cyclically ordered states (denoted by p, q, r, and s in Figure 7) and up to four operators that transform an input state into an output state. The possible operators are *null* (leaving the input state unchanged, as in null [p, p]); *clockwise* (move one step clockwise from the input, as in clockwise [p, q]), *counterclockwise* (move one step in the counterclockwise direction, as in counterclockwise [p, s]); and *diagonal* (move across the diagonal, as in diagonal [p, r]).

In the initial learning phase, a white house was placed at each corner of the layout, and a toy truck was placed in front of one of the houses. The children were then shown three geometric figures (e.g., triangle, circle, dumbbell) and had to learn by

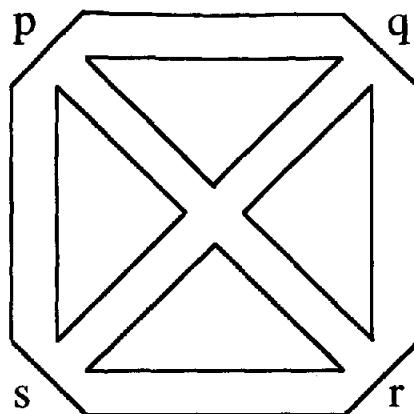


Figure 7. Spatial layout of experimental environment used by Halford and Wilson (1980, Experiment 1). The letters are shown for clarity of exposition, but the corners were unmarked on the apparatus. Adapted from "A Category Theory Approach to Cognitive Development," by G. S. Halford and W. H. Wilson, 1980, *Cognitive Psychology*, 12, p. 381. Copyright 1980 by Academic Press. Reprinted with permission.

trial and error to move the truck in accord with the operator represented by each geometric figure. For example, if the truck was located at the house on corner p, and *triangle* represented the diagonal operator, then the correct response was to move the truck to corner r. For different groups of children, the three figures represented null, clockwise, and counterclockwise, or else null, clockwise, and diagonal. As the null operator is not theoretically significant in interpreting Halford and Wilson's (1980) findings, we did not include it in our simulations; hence, we will refer to the two experimental groups as the *clockwise–counterclockwise* and the *clockwise–diagonal* conditions, respectively. After the children learned to interpret all of the geometric figures correctly, they learned the system a second time with a new set of figures representing the same operators. This initial learning phase was intended to ensure that the participants acquired a schema representing the underlying structure of the layout and operators.

In the subsequent criterial phase of the experiment, the children were presented with a series of additional problems embodying the same relational structure as they had learned in the initial phase (either clockwise–counterclockwise or clockwise–diagonal), but with entirely new surface elements. The states were represented by four toy houses of different colors (red, green, blue, yellow). One of these houses was initially placed on a corner of the layout (e.g., blue might be placed on r), and the child was given the remaining three. The child then saw a series of operators represented by circle, star, and cross, each defined by a move from one colored house to another. The operators were formally equivalent to those the child had learned in the initial phase. For example, if *star* meant diagonal, and if red belonged on state p, then the child had to learn (by trial and error with feedback) that applying *star* to green (on r) meant moving to red (on p). The child was encouraged to place the three unassigned houses onto the layout in accord with the meaning of the operators, rearranging the tokens as often as they liked. When a learning criterion was reached (12 successive

correct predictions of the output state given an input and operator), or a maximum number of trials was presented (72), the child advanced to a new problem, again with the identical underlying structure. The main dependent measure was the number of problems (out of a maximum of three) for which the child achieved criterion.

Because the underlying structure of the criterial problems was isomorphic to that of the learning problems, each criterial problem could potentially be solved by mapping propositions describing the binary input-output relations in a criterial problem (target) onto an isomorphic representation of the schema acquired during the initial phase (source). Table 9 presents the propositional representations that were provided to LISA, for both the clockwise-counterclockwise and clockwise-diagonal conditions. Propositions P1-P4 (representing the clockwise operator) in the source and target are identical for the two conditions; propositions P5-P8 represent either the counterclockwise or the diagonal operator, depending on the condition.

These two conditions, despite being closely matched (both involve analogs describable by eight binary relations based on two operators and four states from a cyclic 4-group), differ in their relational complexity, according to Halford and Wilson's (1980) taxonomy. The clockwise-counterclockwise condition only requires mapping at the relational level, as an isomorphic mapping can be computed by considering just one binary relation at a time. The reason is that the target operators, *cross* and *star*, can be mapped either to clockwise and counterclockwise, respectively, or the reverse: Either assignment would generate an isomorphic mapping. By contrast, clockwise and diagonal are asymmetrical operators and, hence, are not interchangeable: *Cross* must map to clockwise and *star* to diagonal. As such, the clockwise-diagonal condition requires finding a mapping at the system level by considering two binary relations (both clockwise and diagonal) together. As the difference in complexity

level predicts, Halford and Wilson found that children at both ages almost always reached criterion in the clockwise-counterclockwise condition but that 5- to 6-year olds performed significantly better than 4-year-olds in the clock-diagonal condition.

To simulate this Age \times Complexity interaction, we had LISA map the two pairs of structures in Table 9 either with a phase set that included a single binary proposition (relational level) or two binary propositions (system level). To simulate placing one house correctly for the child, in all runs we coded the blue house (target) with the same features as state p (source). Because the critical interaction should emerge even when the information about the target analog (which served as driver) is ordered in an optimal fashion, LISA was tested with two orderings for each condition that satisfied the following constraints: (a) all propositions involving the fixed object mapping (i.e., blue \rightarrow p) were selected before any others, (b) the first proposition selected had the fixed object as one argument; and (c) propositions alternated systematically from one operator to the other. In addition, the clockwise-diagonal orderings obeyed the additional constraint that the second proposition selected (like the first) also had the fixed object as one argument. For the clockwise-counterclockwise simulations, the orders used were (a) P3 P4 P7 P8 P1 P2 P6 P5 and (b) P7 P8 P3 P4 P6 P5 P1 P2; for the clockwise-diagonal simulations, the orders were (a) P1 P5 P2 P3 P7 P4 P6 P8 and (b) P4 P6 P3 P2 P7 P1 P5 P8. Propositions were placed in the phase set individually, except for the first two propositions in the two clockwise-diagonal orders (which were buffered together). Each order was run eight times, for a total of 16 simulations per condition.

We regarded a simulation run as a success if all mapping weights corresponding to correct mappings were substantially larger than any weight corresponding to an incorrect mapping. Specifically, a mapping was regarded as correct if, for every object, the weight from that object to the corresponding object in the other analog was at least 10% greater than the weight to any noncorresponding object. The simulation results yielded an "age" (in LISA, phase-set size) by complexity level interaction of the sort observed by Halford and Wilson (1980), under the assumption that the size of the phase set increases at about age 5 years. The clockwise-counterclockwise analogs were mapped successfully on all 16 runs, even though the phase set included only one proposition (relational mapping). In contrast, LISA correctly solved the clockwise-diagonal problem on only 2 of 16 runs in which the phase set was restricted to a single proposition but on 12 of 16 runs in which it was allowed to place the first two propositions into the phase set together.

The above simulations focus on the shift from the relational to the system levels in Halford's (1993) taxonomy of complexity. We also applied LISA to a mapping problem at the multiple-system level, which according to Halford's theory should only be solved reliably if two trinary propositions (more generally, four independent dimensions) can be considered together. The analogy we gave LISA involved a chain of social introductions: "Abe introduced Bill to Charles" and "Bill introduced Charles to Dave" (driver) mapped onto "Alice introduced Barbara to Cynthia" and "Barbara introduced Cynthia to Doris" (recipient). The people in each analog can only be uniquely mapped if the information in the two driver propositions is integrated.

Table 9
LISA Representations Used to Simulate Structure-Learning Results of Halford and Wilson (1980)

Initial structure (source)	Criterial structure (target)
Clockwise-counterclockwise condition	
P1 (clockwise p q)	P1 (cross red green)
P2 (clockwise q r)	P2 (cross green blue)
P3 (clockwise r s)	P3 (cross blue yellow)
P4 (clockwise s p)	P4 (cross yellow red)
P5 (counterclockwise p s)	P5 (star red yellow)
P6 (counterclockwise s r)	P6 (star yellow blue)
P7 (counterclockwise r q)	P7 (star blue green)
P8 (counterclockwise q p)	P8 (star green red)
Clockwise-diagonal condition	
P1 (clockwise p q)	P1 (cross red green)
P2 (clockwise q r)	P2 (cross green blue)
P3 (clockwise r s)	P3 (cross blue yellow)
P4 (clockwise s p)	P4 (cross yellow red)
P5 (diagonal p r)	P5 (star red blue)
P6 (diagonal r p)	P6 (star blue red)
P7 (diagonal q s)	P7 (star green yellow)
P8 (diagonal s q)	P8 (star yellow green)

As would therefore be predicted, LISA found the isomorphic mappings when both trinary propositions entered the phase set together, but not when they were entered one at a time.

According to Halford (1993), the level of multiple-system mapping exemplified by the above problem is the highest complexity level reliably attained by humans. However, as we already demonstrated for the boys-dogs mapping problem, LISA's mapping capability can be made overpowered relative to humans by granting the model a superhuman working-memory capacity. We expanded the "introductions" example to the next complexity level (five independent dimensions) by adding a third trinary proposition to each analog: "Charles introduced Dave to Ed" (driver) and "Cynthia introduced Doris to Edith" (recipient). According to Halford's theory of complexity, finding the isomorphic mapping between the two sets of five individuals (in the absence of differentiating semantic overlap or, of course, attending to the mnemonic first letters of the names) requires processing all three trinary propositions in parallel. In fact, LISA was able to solve this "level 5" problem only when all three driver propositions (i.e., 9 SPs) were entered into the phase set together.

In summary, LISA is able to model phylogenetic (Phenomenon 13) and ontogenetic (Phenomenon 14) increases in mapping capability as the consequences of increases in the size of the working memory (phase set) available to the driver analog. As we noted earlier, LISA performs a close approximation to parallel constraint satisfaction in mapping the proposition or propositions that enter the phase set together but maps in a more serial manner across phase sets. Thus, increases in the size of the phase set translate into greater capacity for parallel constraint satisfaction, which in turn increases the maximum formal complexity of the problems that can be successfully mapped.

General Discussion

Analogical Processing With Distributed Representations

Analogy plays an important role in many aspects of human cognition, and the processes of analogical access and mapping are fundamental to analogical thinking. Previous models of analogical access and mapping have made important contributions to our understanding of the constraints on human analogical reasoning. However, these models are limited in that they fail to capture both some of the strengths of human analogical thought (such as our ability to discover correspondences between predicates that take different numbers of arguments) and some of its limitations (such as the difficulty we have solving "unnatural" analogies). More important, it is unclear how the representations and processes these models use for analogical mapping could be adapted for other cognitive operations, such as schema induction, in which analogy plays an important role. In large part, this limitation reflects a failure to represent knowledge in a way that captures both the flexibility and structure sensitivity of human knowledge representation.

We have presented a theory of analogical access and mapping that exhibits both sensitivity to structural relations (the hallmark of symbolic cognition) and flexibility in the face of imperfect matches (a strength of distributed representations). LISA's per-

formance hinges on five assumptions that form the core of the theory:

1. Propositions are represented in active memory as distributed patterns that specify the semantic content of predicates and objects, with case roles dynamically bound to their fillers.

2. These structures are encoded in LTM in a way that statically binds semantic primitives into predicates and objects, and role-filler conjunctions into propositions.

3. Analog retrieval is a process of guided pattern classification, in which units representing stored propositions compete to respond to the distributed patterns generated by an active driver analog.

4. Analogical mapping is performed by augmenting analog retrieval with a capacity to *learn* the correspondences generated during the process of pattern classification.

5. Dynamically binding roles and objects into propositions and learning analogical correspondences consume (finite) working-memory resources.

A number of properties follow from these assumptions. The most apparent are a variety of properties related to the limitations of working memory. The upper bound on the complexity of the mappings that can be computed is determined by limits on the capacity of working memory (as given by the size of the phase set). In particular, performance on analogies that require attention to interlocking structural constraints is limited by the size of the phase set and the associated buffers that control changes in mapping connection weights. LISA is also sensitive to the semantic content of an analogy and strategic variables, such as the order in which propositions are selected to fire, and the grouping of propositions into phase sets. As a result, LISA's mapping capacity is not tied strictly to the formal complexity of the mapping problem. For example, LISA can solve complex, semantically rich analogies with greater ease than it can solve formally simpler but semantically impoverished analogies. It is important to note that the meaning of *semantically rich* is intimately tied to what LISA "knows": Depending on how an analog is represented, LISA can be made to perform like either an expert or a novice.

Capacity limits, sensitivity to semantics, and sensitivity to strategic variables constitute formal disadvantages of LISA's approach to analogical mapping (in the sense that LISA is weaker than mapping engines based on massively parallel constraint satisfaction and related algorithms), but they also constitute a source of behavioral predictions. Human reasoners are subject to some of these limitations (e.g., order effects; Keane et al., 1994), and it is very plausible that they are sensitive to others. In addition to formal disadvantages, LISA's approach to mapping also affords formal and theoretical advantages, including freedom from the *n*-ary restriction, the unification of access and mapping, and a form of knowledge representation that is useful for other processes, such as inference and schema induction (Hummel & Holyoak, 1996, in press). Together, these properties make LISA both weaker and more powerful than other models of analogical access and mapping.

LISA, like any model that integrates distributed representations of concepts with dynamic binding, is subject to the one-level restriction: At the level of semantic primitives, activating more than one level of a hierarchical structure in a single group (i.e., time slice) creates ambiguity about the role-filler bindings.

Accordingly, LISA activates only one level of hierarchy at a time at the semantic level. LISA can nonetheless exploit hierarchical structure to map propositions based on higher-order relations. Central to this capacity is the ability of a P unit to act both as an object filling a role in one proposition (daughter mode) and as a pointer to its constituent role-filler bindings (parent mode). Augmented with this duality of processing modes, the single degree of freedom that synchrony (or alternative dynamic binding codes) provides for binding suffices to capture structural relations within multilevel propositions.

More generally, LISA exploits both local and distributed representations and both serial and parallel processing. The units that encode structural relations are strictly localist, but the meanings of individual concepts are distributed over multiple semantic units. During mapping, driver propositions are activated serially; at a finer time scale, the firing of elements associated with distinct roles are desynchronized and, hence, serial. This serial processing is crucial in representing the bindings of objects to roles. At the same time, recipient propositions respond in parallel to the semantic patterns generated by the role-filler bindings of an active driver proposition. The integrated system provides distributed representations of propositional meaning while maintaining systematicity of knowledge, thus solving the core problem for distributed representations posed by Fodor and Pylyshyn (1988).

Capacity-Limited Versus Massively Parallel Constraint Satisfaction

LISA clearly demonstrates sensitivity to all of the mapping constraints postulated by the multiconstraint theory of analogy: isomorphism, semantic similarity, and pragmatic centrality. Yet it does so using radically different representational and processing assumptions than those embodied in ACME (Holyoak & Thagard, 1989), the first computational instantiation of the multiconstraint theory. ACME represents propositions as symbolic list structures and does not provide any representation of concept meaning. It uses symbolic processing to transform these structures into a network of units that explicitly represent all syntactically legal potential correspondences. The optimal set of correspondences is then computed by massively parallel constraint satisfaction, with no theoretically motivated capacity limits. In contrast, LISA represents propositions and their constituent concepts as patterns distributed over semantic units. It does not build any new units to represent correspondences; rather, it learns correspondences by modifying weights on connections between structure units. LISA, unlike ACME, operates under inherent capacity limits. Across phase sets, LISA maps in an incremental fashion that is highly sensitive to the order and frequency with which propositions within the driver become active.

One of the more surprising results of the simulations reported here is that for many analogy problems, LISA's more incremental approach to mapping yields results that closely approximate those obtained by full parallel constraint satisfaction. A case in point is the simulations of mapping convergence analogs, which were performed using representations highly similar in size and semantic overlap to those provided to ACME in previous work (Holyoak & Thagard, 1989). Whereas ACME, in effect, consid-

ered the two analogs in their entirety at once, LISA was only allowed to consider one driver proposition at a time. Nonetheless, both models yield the same sets of correspondences. The reason for the equivalent performance is that semantically rich analogs of this sort, even if drawn from disparate domains, will generally include some corresponding concepts (possibly higher-order relations) that tend to map on the basis of overlapping semantic features. To the extent that the mapping is semantically driven, it is not necessary to consider large structural units at once to find the optimal correspondences.

For more formal analogies, however, LISA's capacity-limited and incremental constraint satisfaction results in performance that differs sharply from that of ACME with its massively parallel approach. To take the most extreme case, LISA is utterly incapable of mapping two isomorphic analogs in the absence of *any* overlap of semantic features between them. The initial transmission of activation from the driver to the recipient is necessarily routed through semantic units; in the absence of any semantic overlap, the activity of the driver, however feverish, will simply leave the recipient cold. For problems in which the analogs do share semantic features (even minimally), the mapping process will at least get started. However, if semantic similarity does not suffice to determine the optimal mapping, so that consideration of structural relations is critical, then LISA's performance on structurally complex analogies varies with the size of the phase set and the order in which driver propositions are activated. We have argued that LISA's limitations in solving such problems are, in fact, more psychologically realistic than ACME's unbridled success.

The flip side of the comparison between LISA and ACME (and all other previous mapping models) is that LISA is able to map analogs that violate the *n*-ary restriction. LISA, unlike any other model of which we are aware, can find sensible mappings between the elements of propositions with different numbers of arguments. This ability follows directly from LISA's fundamental representational innovation relative to previous models: Arguments are represented not as static lists of localist symbols but as distributed patterns of activation over semantic units. Moreover, mapping is performed not by subjecting all potential legal correspondences to parallel constraint satisfaction but by using the semantic representation of elements in one analog to directly activate elements of other analogs. Thus, even though constraint satisfaction based on local representations is more powerful as an approach to analogical mapping, pattern matching based on distributed representations conveys a greater degree of humanlike flexibility. This gain in flexibility sets the stage for the computational integration of analogy with schema induction.

Future Directions

Scaling up. The simulations reported here generally used very small analogs. The representations were kept small for both a theoretical reason (to highlight specific representational properties to which LISA is sensitive) and a practical reason (the computer resources at our disposal were severely limited). It is clear that further work is required to establish the degree to which LISA scales up to larger representations and (especially with respect to analogical access) larger knowledge bases.

Nonetheless, there are reasons to be optimistic. Because LISA does not need to generate new "mapping units," the space it requires to represent correspondences is proportional to the square root of that required by ACME. Each proposition in LISA is encoded by a small number of SP units (one per case role). The number of SP units required to represent an analog therefore grows linearly with the size of the analogs, and the number of connections between SPs across analogs grows linearly with the product of the number of propositions in the source and target.

The greatest challenge that must be faced for LISA to handle large knowledge bases centers on the fact that the model depends on a detailed semantic decomposition of concepts, including predicate-argument structures. In our simulations we used very informal semantic representations. More realistic applications of the model will require careful attention to difficult issues in meaning representation. We would argue, however, that these issues must be faced in any case—the nature of meaning lies at the very heart of human cognition. Moreover, theorists have recognized that human analogical thinking involves the capacity to re-represent knowledge in ways that go beyond the limits of current models (Gentner, 1989). LISA's theoretical task leads toward confronting the problem of understanding conceptual semantics, whereas previous models of analogy have generally postponed it.

In particular, LISA's ability to overcome the *n*-ary restriction sets the stage for the incorporation of representational schemes more sophisticated than flat feature vectors. It has often been argued (e.g., Jackendoff, 1983) that lexical concepts often have a close conceptual relationship with more complex relational forms. For example, causative verbs such as *lift* (e.g., "John lifted the hammer") have very similar meanings to structures based on an explicit higher-order relation, *cause* (e.g., "John caused the hammer to rise"). In such cases, the causative verb serves as a "chunked" representation of a more elaborate predicate-argument structure.

The LISA architecture suggests at least two ways in which such chunked information may be established and exploited. The first relates to the parent–daughter distinction currently implemented with P units. When a P unit is active in daughter mode (i.e., as an argument of another proposition), it functions as a chunked representation of its propositional content. This chunked representation is useful for mapping only insofar as mapping connections have already been established between the P unit and others: Because it cannot express its semantics directly, a P unit in daughter mode can only affect the recipient analog through learned mapping connections. The similarity between this mode of operation and chunking as it is often discussed in the cognitive literature (e.g., Halford et al., 1994) is obvious: Efficiency (here, the need to occupy only a single slot in the phase set rather than one slot per case role) is purchased at the price of flexibility and generality (the proposition can only affect other propositions with which it has established mapping connections). In our current implementation, predicate and object units behave as if they are permanently in a sort of modified daughter mode: Each case role and object occupies only one slot in the phase set, and in this slot, it is represented as a flat feature vector without any articulated structure. Placing a predicate or object into its own version of a parent mode would

mean treating each role (or object) not as a flat feature vector but as a whole structure with its own SPs. For example, "lift" could be expanded into "cause to rise," or "Mary" into a structure expressing the relations among her attributes. This kind of expansion would permit structured comparisons between predicates and objects (Markman & Gentner, 1993a; Markman & Wisniewski, 1997), analogous to the structured comparisons LISA currently performs on whole propositions.

A second way in which the LISA architecture addresses the notion of chunked predicates is in its capacity to solve mappings that violate the *n*-ary restriction. A mapping system that is free of the *n*-ary restriction can potentially recognize correspondences between chunked structures (e.g., as the "taller than" propositions can be chunked into a linear ordering) despite their differences in form. That is, freedom from the *n*-ary restriction can permit mapping between chunked and fully elaborated structures without requiring an explicit prior process of lexical decomposition to equate their forms (a process that is psychologically questionable; Fodor, Fodor, & Garrett, 1975).

Symbol grounding and relations between analogical mapping and perception. The potential for chunking in a LISA-style architecture raises broader issues in meaning representation, including symbol grounding and relations between perceptual and cognitive representations (as discussed by Barsalou, 1993; Jackendoff, 1983; and Lakoff & Johnson, 1980). With the notion of chunked predicates and objects, LISA hints at a kind of recursive representation for meaning that may ultimately ground itself in basic perceptual primitives. In its current implementation, LISA can represent and map hierarchical propositions of arbitrary depth (Hummel, Melz, et al., 1994). Analogously, it is possible to imagine structures for roles and objects that are, themselves, deeply embedded recursive structures. The depth to which a role or object would need to be decomposed for the purposes of mapping would depend on the task at hand. For example, mapping "John lifted the hammer" onto "Bill raised the book" may require little or no decomposition of the predicates "lift" and "raise," which will have substantial overlap in their semantic features. On the other hand, mapping "John lifted the hammer" onto "Bill pushed the cart," where the predicates have less feature overlap, may be more likely to depend on decomposition of "lift" into "cause to rise" and "push" into "cause to move laterally," thereby making explicit the parallelism of their internal structures. Recursively, "rise" and "move laterally" might be decomposed into structures relating simpler predicates, with basic perceptual primitives representing motion and locations in space residing at the very bottom. Such conjectures about the recursive decomposition of concepts into relations among simpler concepts are by no means novel. Nevertheless, LISA may be the first working model of human cognitive architecture to suggest specific algorithms and representations that can make such an approach computationally realizable.

This optimism is encouraged by the close theoretical links between LISA and models of perception, such as Hummel and Biederman's (1992; Hummel & Stankiewicz, 1996) JIM model of object recognition. LISA's most basic operations—which perform analog access by guided pattern classification—rest on representations and processes that permit the classification of distributed representations of structure. The upper layers of the

JIM model perform exactly the same function (except that here, the representations are structural descriptions of object shape). As such, it is no accident that LISA resembles (and operates like) the upper layers of JIM: Take away LISA's mapping connections and attach a front end to generate structural descriptions from visual images, and the result would be JIM. Integrating LISA with a perceptual front end for symbol grounding would therefore be straightforward. The architectures and operations necessary for structural description (of visually perceived shapes) and LISA-style analogical mapping are virtually indistinguishable.

There is one more parallel between LISA and JIM that warrants mention because it speaks directly to the consequences of their shared approach to the representation of structure. As we have discussed extensively throughout this article, LISA can solve analogies that violate the n -ary restriction, mapping (for example) a proposition with two arguments onto a subset of a different proposition with three. The missing case role poses no special difficulty for LISA. Similarly, JIM can map a two-part structural description of an image onto a three-part structural description stored in memory (a capacity that permits JIM to recognize objects based on a subset of their parts, as when one part is invisible because it is occluded by another surface). LISA's freedom from the n -ary restriction and JIM's capacity to recognize objects from a subset of their parts both result from the fact that the basic elements (case roles or object parts) are not tied to specific positions in a list or vector. Most models of analogy represent case roles as elements in an ordered list; in a similar way, most models of object recognition (e.g., Poggio & Edelman, 1990; Ullman & Basri, 1991) represent object features as elements in a vector of coordinates. Coding elements as a list (or vector) yields models that are exquisitely sensitive to the number of elements (case roles or features) in the to-be-compared structures: The match between a four-element feature vector and a five-element vector in memory is mathematically undefined. LISA's departure from this representational convention (like JIM's) frees it from this kind of sensitivity to list (or vector) position.

Detailed analyses of analogical processing. Earlier models (arguably with good reason) have generally been directed more at understanding the basic constraints that govern human analogical thinking than at understanding its psychological mechanisms (cf. Keane et al., 1994). LISA, however, provides avenues for theoretical analysis of analogical processing at a finer level of detail than previously possible. In many cases, the model leads to predictions that have yet to be empirically tested. For example, LISA theoretically unifies analogical access and mapping by a single assumption: New structural correspondences can be learned during mapping but not during access. The theory therefore predicts that structural consistency will influence access only when the relevant correspondences can be computed from preexisting connections. This principle underlies LISA's simulations of the dissociations between access and mapping observed by Ross (1989) but remains to be tested using analogs that instantiate additional types of structural relations.

With respect to mapping, LISA accounts for the influence of processing goals on mapping ambiguous analogies (Spellman & Holyoak, 1996) in terms of variations in the order and frequency with which driver propositions enter the phase set. It follows

that other factors that influence order and frequency of firing should also alter people's preferred mappings for ambiguous problems. Many other predictions can also be derived concerning the influence of firing order on mapping. A basic principle is that mapping will be facilitated if propositions that, when considered alone, can be uniquely and correctly mapped are selected before propositions that generate ambiguous mappings, rather than the reverse order. When the flow of control obeys this ordering principle, the clear initial mappings will be able to help disambiguate later mappings that would otherwise be ambiguous. This principle can manifest itself in many different structures. In some cases, for example, a clear mapping of a higher-order relation can disambiguate lower-level correspondences; however, for other analogs, the constraints will be reversed such that clear lower-level correspondences can disambiguate the mappings for higher-order relations (Hummel, Melz, et al., 1994).

LISA predicts that mapping performance will be influenced not only by the order and frequency with which propositions are fired but also by the manner in which propositions are grouped together into phase sets. The simulations reported here demonstrate that some analogy problems (e.g., mappings between two sets of transitively ordered objects) can only be solved when multiple propositions are fired within a single phase set. What we did not emphasize, but which is also predicted by the model, is that for structurally complex analogies it also matters which propositions are grouped together in a shared phase set. In mapping tasks based on mathematical group structures (Halford & Wilson, 1980), for example, LISA's performance proved to be highly sensitive to both the order and grouping of propositions in the phase set. Further theoretical analysis of the basis for such grouping effects should yield additional predictions about the optimal flow of control in mapping.

Another area in which LISA may lead to novel predictions concerns potential asymmetries in analogical access and mapping. There is empirical evidence that for some analog pairs, transfer is asymmetrical. Asymmetries have been observed in studies of analogical problem solving (Bassok & Holyoak, 1989; Burns, 1996; Ghoshal et al., 1988; Reed, Ernst, & Banerji, 1974), inference (Gentner & Bowdle, 1994), and metaphor interpretation (Glucksberg & Keysar, 1990; Ortony, 1979). Most such asymmetries have been interpreted in terms of post-mapping processes (but see Bassok & Olseth, 1995). Although previous analogy models can account for asymmetries that arise in the aftermath of mapping, they uniformly posit that the mapping process itself is inherently symmetrical: The correspondences between two analogs will be identical regardless of the direction of mapping (e.g., Falkenhainer et al., 1989; Holyoak & Thagard, 1989).

In contrast, LISA predicts that the mapping process itself (in addition to post-mapping processes) may sometimes yield asymmetries in the correspondences between two analogs. The basis for potential mapping asymmetries lies in the differences between the operation of the driver and the recipient, a distinction that has no parallel in previous analogy models. For example, the operation of the driver, but not the recipient, will be influenced by the grouping of propositions in the phase set. It follows that any variable (e.g., textual coherence) that affects phase-set groupings could potentially generate asymmetries in

mapping. For example, if one analog is more coherent than the other, correspondences may be computed more readily when the more coherent analog serves as the driver rather than the recipient.

Analogical inference and schema induction. One of our major arguments in favor of the use of distributed representations for access and mapping is that such an architecture is advantageous for generating inferences and inducing abstract schemas in the aftermath of analogical mapping. Although detailed discussion of inference and schema induction is beyond the scope of this paper, we have, in fact, implemented these processes in LISA (Hummel & Holyoak, 1996, *in press*) and can sketch the essence of the extensions.

The basic idea, adapted from a model of learning structural descriptions of objects (Hummel & Saiki, 1993), is very simple. During access, only existing propositions in a recipient are available to respond to the driver. However, during mapping, new "unrecruited" structure units (*i.e.*, structure units with initially random connections to semantic units and to one another) are added to the recipient analog. Driver-to-recipient mapping is performed as usual. As a result, one of two things will happen whenever a proposition is selected to become active in the driver: If that proposition corresponds to an existing proposition in the recipient, then it will simply activate that proposition. However, if the active driver proposition does not activate any existing proposition or propositions in the recipient, then unrecruited units in the recipient will have the opportunity to respond. The unrecruited units will learn to respond to the semantic patterns generated by the active driver proposition (*and* to the role-filler bindings embodied in those patterns): The recipient analog will have created (*inferred*) a proposition where previously there had been none. This guided encoding of new propositions into LTM is LISA's implementation of inference.

Schema induction is similar except that the unrecruited units reside in a completely different analog. These units are thresholded so that they can only respond to (*and* learn about) semantic primitives that are common to both the driver and recipient analogs. The result is a kind of intersection discovery in which a new schema is generated to represent what is common to the two instances. LISA's distributed representations of meaning are critical to this process because they allow the intersection to flexibly emerge from the individual analogs. In both inference and schema induction, learning is completely unsupervised (*i.e.*, it operates without external feedback), reflecting the fact that analogical thinking is driven by sensitivity to its own internal constraints rather than by an external "teacher." It is unclear how a system without detailed representations of concepts (such as ACME or SME) would perform this type of intersection discovery.

In the extended LISA architecture that handles inference and induction, the stages of analogical transfer can be interwoven rather than being strictly serial. For example, the overall flow of control might involve (a) using an incomplete target to access one or more schemas in LTM, (b) using these schemas to make inferences that elaborate the target, (c) using the elaborated target to access related analogs in LTM, (d) using the retrieved analogs to generate additional inferences about the target, and (e) using multiple analogs to induce a new schema that captures their commonalities. LISA leads to a conception of analogical

thinking in which performance depends on the integration of relatively automatic comparison processes with strategic control of driver operation and triggering conditions for inference and learning.

Neural basis of analogical thinking. Although little is yet known about the neural mechanisms that support analogical thinking, the LISA architecture provides some hypotheses regarding the basic system requirements for analogy—aspects that are likely to be biologically determined in a relatively direct way. These include the capacity to perform dynamic binding, the availability of hierarchically organized structure units that store bindings in LTM (including new units that can be recruited to make inferences and induce schemas), the capacity to learn mapping connections, and attentional control of the phase buffering and selection processes that drive comparison. It has been suggested that the prefrontal cortex supports the working memory required to manipulate complex structural relations (Robin & Holyoak, 1994). Damage to that area would therefore be expected to degrade analogical performance in roughly the manner in which LISA's performance degrades when the size of the phase set is reduced. It is important to note that the model predicts that frontal damage will not lead to across-the-board deterioration in analogical ability but rather to selective difficulty in finding mappings that require integration of structural constraints across multiple propositions. The ability to find mappings based primarily on preexisting semantic connections between the analogs would be expected to be relatively spared.

Conclusion

Although a great deal of additional work remains to be done, LISA's success in simulating core phenomena associated with human analogical access and mapping is encouraging. By introducing mechanisms to perform dynamic binding, it is possible to achieve the flexibility afforded by distributed representations while maintaining the structure sensitivity so critical to propositional reasoning. The cognitive architecture embodied in LISA may help us to understand how human thinking can, at its best, appear to transcend the constraints imposed by a limited-capacity working memory.

References

- Andersen, S. M., Glassman, N. S., Chen, S., & Cole, S. W. (1995). Transference in social perception: The role of chronic accessibility in significant-other representations. *Journal of Personality and Social Psychology*, 69, 41–57.
- Anderson, J. R. (1983). *The architecture of cognition*. Cambridge, MA: Harvard University Press.
- Barnden, J. A. (1994). On the connectionist implementation of analogy and working memory matching. In J. A. Barnden & K. J. Holyoak (Eds.), *Advances in connectionist and neural computation theory: Vol. 3. Analogy, metaphor, and reminding* (pp. 327–374). Norwood, NJ: Ablex.
- Barsalou, L. W. (1993). Flexibility, structure and linguistic vagary in concepts: Manifestations of a compositional system of perceptual symbols. In A. F. Collins, S. E. Gathercole, M. A. Conway, & P. E. Morris (Eds.), *Theories of memory* (pp. 29–101). Hillsdale, NJ: Erlbaum.
- Bassok, M., & Holyoak, K. J. (1989). Interdomain transfer between

- isomorphic topics in algebra and physics. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 15, 153–166.
- Bassok, M., & Olseth, K. L. (1995). Object-based representations: Transfer between cases of continuous and discrete models of change. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 21, 1522–1538.
- Bassok, M., Wu, L. L., & Olseth, K. L. (1995). Judging a book by its cover: Interpretative effects of content on problem-solving transfer. *Memory & Cognition*, 23, 354–367.
- Brown, A. L., Kane, M. J., & Echols, C. H. (1986). Young children's mental models determine analogical transfer across problems with a common goal structure. *Cognitive Development*, 1, 103–121.
- Burns, B. D. (1996). Meta-analogical transfer: Transfer between episodes of analogical reasoning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 22, 1032–1048.
- Catrambone, R., & Holyoak, K. J. (1989). Overcoming contextual limitations on problem-solving transfer. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 15, 1147–1156.
- Desmedt, J., & Tomberg, C. (1994). Transient phase-locking of 40 Hz electrical oscillations in prefrontal and parietal human cortex reflects the process of conscious somatic perception. *Neuroscience Letters*, 168, 126–129.
- DeSoto, L. B., London, M., & Handel, L. S. (1965). Social reasoning and spatial paralogic. *Journal of Personality and Social Psychology*, 2, 513–521.
- Duncker, K. (1945). On problem solving. *Psychological Monographs*, 58(Whole No. 270).
- Eckhorn, R., Bauer, R., Jordan, W., Brish, M., Kruse, W., Munk, M., & Reitboeck, H. J. (1988). Coherent oscillations: A mechanism of feature linking in the visual cortex? Multiple electrode and correlation analysis in the cat. *Biological Cybernetics*, 60, 121–130.
- Eckhorn, R., Reitboeck, H., Arndt, M., & Dicke, P. (1990). Feature linking via synchronization among distributed assemblies: Simulations of results from cat visual cortex. *Neural Computation*, 2, 293–307.
- Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14, 179–212.
- Falkenhainer, B. (1990). A unified approach to explanation and theory formation. In J. Shrager & P. Langley (Eds.), *Computational models of scientific discovery and theory formation* (pp. 157–196). San Mateo, CA: Morgan Kaufmann.
- Falkenhainer, B., Forbus, K. D., & Gentner, D. (1989). The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, 41, 1–63.
- Fletcher, C. R. (1986). Strategies for the allocation of short-term memory during comprehension. *Journal of Memory and Language*, 25, 43–58.
- Fletcher, C. R., & Bloom, C. P. (1988). Causal reasoning in the comprehension of simple narrative texts. *Journal of Memory and Language*, 27, 235–244.
- Fletcher, C. R., Hummel, J. E., & Marsolek, C. (1990). Causality and the allocation of attention during comprehension. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 16, 233–240.
- Fodor, J. D., Fodor, J. A., & Garrett, M. (1975). The psychological unreality of semantic representations. *Linguistic Inquiry*, 6, 515–532.
- Fodor, J. A., & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. In S. Pinker & J. Mehler (Eds.), *Connections and symbols* (pp. 3–71). Cambridge, MA: MIT Press.
- Forbus, K. D., Ferguson, R. W., & Gentner, D. (1994). Incremental structure mapping. In A. Ram & K. Eiselt (Eds.), *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society* (pp. 313–318). Hillsdale, NJ: Erlbaum.
- Forbus, K. D., Gentner, D., & Law, K. (1995). MAC/FAC: A model of similarity-based retrieval. *Cognitive Science*, 19, 141–205.
- 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). New York: Cambridge University Press.
- Gentner, D., & Bowdle, B. F. (1994). The coherence imbalance hypothesis: A functional approach to asymmetry in comparison. In A. Ram & K. Eiselt (Eds.), *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society* (pp. 351–356). Hillsdale, NJ: Erlbaum.
- Gentner, D., & Gentner, D. R. (1983). Flowing waters or teeming crowds: Mental models of electricity. In D. Gentner & A. L. Stevens (Eds.), *Mental models* (pp. 99–129). Hillsdale, NJ: Erlbaum.
- Gentner, D., Rattermann, M., & Forbus, K. (1993). The roles of similarity in transfer: Separating retrievability 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.
- Gholson, B., Eymard, L. A., Long, D., Morgan, D., & Leming, F. C. (1988). Problem solving, recall, isomorphic transfer, and non-isomorphic transfer among third-grade and fourth-grade children. *Cognitive Development*, 3, 37–53.
- Gick, M. L., & Holyoak, K. J. (1980). Analogical problem solving. *Cognitive Psychology*, 12, 306–355.
- Gick, M. L., & Holyoak, K. J. (1983). Schema induction and analogical transfer. *Cognitive Psychology*, 15, 1–38.
- Gillan, D. J., Premack, D., & Woodruff, G. (1981). Reasoning in the chimpanzee: I. Analogical reasoning. *Journal of Experimental Psychology: Animal Behavior Processes*, 7, 1–17.
- Glucksberg, S., & Keysar, B. (1990). Understanding metaphorical comparisons: Beyond similarity. *Psychological Review*, 97, 3–18.
- Goldstone, R. L. (1994). Similarity, interactive activation, and mapping. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 20, 227–247.
- Goldstone, R. L., & Medin, D. L. (1994). Similarity, interactive activation, and mapping. In K. J. Holyoak & J. A. Barnden (Eds.), *Advances in connectionist and neural computation theory: Vol. 2. Analogical connections* (pp. 321–362). Norwood, NJ: Erlbaum.
- Goldstone, R. L., Medin, D. L., & Gentner, D. (1991). Relational similarity and the nonindependence of features. *Cognitive Psychology*, 23, 222–262.
- Goswami, U. (1989). Relational complexity and the development of analogical reasoning. *Cognitive Development*, 4, 251–268.
- Goswami, U. (1992). *Analogical reasoning in children*. Hillsdale, NJ: Erlbaum.
- Goswami, U., & Brown, A. (1989). Melting chocolate and melting snowmen: Analogical reasoning and causal relations. *Cognition*, 35, 69–95.
- Gray, C. M. (1994). Synchronous oscillations in neuronal systems: Mechanisms and functions. *Journal of Computational Neuroscience*, 1, 11–38.
- Gray, C. M., Konig, P., Engel, A. E., & Singer, W. (1989). Oscillatory responses in cat visual cortex exhibit inter-column synchronization which reflects global stimulus properties. *Nature*, 338, 334–337.
- Gray, C. M., & Singer, W. (1989). Stimulus specific neuronal oscillations in orientation columns of cat visual cortex. *Proceedings of the National Academy of Sciences, USA*, 86, 1698–1702.
- Halford, G. S. (1992). Analogical reasoning and conceptual complexity in cognitive development. *Human Development*, 35, 193–217.
- Halford, G. S. (1993). *Children's understanding: The development of mental models*. Hillsdale, NJ: Erlbaum.
- Halford, G. S., & Wilson, W. H. (1980). A category theory approach to cognitive development. *Cognitive Psychology*, 12, 356–411.

- Halford, G. S., Wilson, W. H., Guo, J., Gayler, R. W., Wiles, J., & Stewart, J. E. M. (1994). Connectionist implications for processing capacity limitations in analogies. In K. J. Holyoak & J. A. Barnden (Eds.), *Advances in connectionist and neural computation theory: Vol. 2. Analogical connections* (pp. 363–415). Norwood, NJ: Ablex.
- Hofstadter, D. R., & Mitchell, M. (1994). An overview of the Copycat project. In K. J. Holyoak & J. A. Barnden (Eds.), *Advances in connectionist and neural computation theory: Vol. 2. Analogical connections* (pp. 31–112). Norwood, NJ: Erlbaum.
- Holland, J. H., Holyoak, K. J., Nisbett, R. E., & Thagard, P. (1986). *Induction: Processes of inference, learning, and discovery*. Cambridge, MA: MIT Press.
- Holyoak, K. J. (1991). Symbolic connectionism: Toward third-generation theories of expertise. In K. A. Ericsson & J. Smith (Eds.), *Toward a general theory of expertise: Prospects and limits* (pp. 301–335). Cambridge, England: Cambridge University Press.
- Holyoak, K. J., & Koh, K. (1987). Surface and structural similarity in analogical transfer. *Memory & Cognition*, 15, 332–340.
- Holyoak, K. J., Novick, L. R., & Melz, E. R. (1994). Component processes in analogical transfer: Mapping, pattern completion, and adaptation. In K. J. Holyoak & J. A. Barnden (Eds.), *Advances in connectionist and neural computation theory: Vol. 2. Analogical connections* (pp. 130–180). Norwood, NJ: Ablex.
- Holyoak, K. J., & Thagard, P. (1989). Analogical mapping by constraint satisfaction. *Cognitive Science*, 13, 295–355.
- Holyoak, K. J., & Thagard, P. (1995). *Mental leaps: Analogy in creative thought*. Cambridge, MA: MIT Press.
- Hummel, J. E., & Biederman, I. (1990). Dynamic binding: A basis for the representation of shape by neural networks. In *Program of the Twelfth Annual Conference of the Cognitive Science Society* (Vol. 32, pp. 614–621). Hillsdale, NJ: Erlbaum.
- Hummel, J. E., & Biederman, I. (1992). Dynamic binding in a neural network for shape recognition. *Psychological Review*, 99, 480–517.
- Hummel, J. E., Burns, B., & Holyoak, K. J. (1994). Analogical mapping by dynamic binding: Preliminary investigations. In K. J. Holyoak & J. A. Barnden (Eds.), *Advances in connectionist and neural computation theory: Vol. 2. Analogical connections* (pp. 416–445). Norwood, NJ: Ablex.
- Hummel, J. E., & Holyoak, K. J. (1992). Indirect analogical mapping. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society* (pp. 516–521). Hillsdale, NJ: Erlbaum.
- Hummel, J. E., & Holyoak, K. J. (1993). Distributing structure over time. *Behavioral and Brain Sciences*, 16, 464.
- Hummel, J. E., & Holyoak, K. J. (1996). LISA: A computational model of analogical inference and schema induction. In *Proceedings of the Eighteenth Annual Conference of the Cognitive Science Society* (pp. 352–357). Hillsdale, NJ: Erlbaum.
- Hummel, J. E., & Holyoak, K. J. (in press). From analogy to schema induction in a structure-sensitive connectionist model. In T. Dartnall & D. Peterson (Eds.), *Creativity and computation*. Cambridge, MA: MIT Press.
- Hummel, J. E., Melz, E. R., Thompson, J., & Holyoak, K. J. (1994). Mapping hierarchical structures with synchrony for binding: Preliminary investigations. In A. Ram & K. Eiselt (Eds.), *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society* (pp. 433–438). Hillsdale, NJ: Erlbaum.
- Hummel, J. E., & Saiki, J. (1993). Rapid unsupervised learning of object structural descriptions. *Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society* (pp. 569–574). Hillsdale, NJ: Erlbaum.
- Hummel, J. E., & Stankiewicz, B. J. (1996). An architecture for rapid, hierarchical structural description. In T. Inui & J. McClelland (Eds.), *Attention and performance XVI: Information integration in perception and communication* (pp. 93–121). Cambridge, MA: MIT Press.
- Huttenlocher, J. (1968). Constructing spatial images: A strategy in reasoning. *Psychological Review*, 75, 286–298.
- Inagaki, K., & Hatano, G. (1987). Young children's spontaneous personification as analogy. *Child Development*, 58, 1013–1020.
- Jackendoff, R. (1983). *Semantics and cognition*. Cambridge, MA: MIT Press.
- Johnson, J., & Pascuale-Leone, J. (1989). Developmental levels of processing metaphor interpretation. *Journal of Experimental Child Psychology*, 48, 1–31.
- Keane, M. T. (1986). On retrieving analogues when solving problems. *Quarterly Journal of Experimental Psychology*, 39A, 29–41.
- Keane, M. T. (1988). *Analogical problem solving*. Chichester, UK: Ellis Horwood.
- Keane, M. T. (1995). On order effects in analogical mapping: Predicting human error using IAM. In J. D. Moore & J. F. Lehman (Eds.), *Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society* (pp. 449–454). Hillsdale, NJ: Erlbaum.
- Keane, M. T., Ledgerway, T., & Duff, S. (1994). Constraints on analogical mapping: A comparison of three models. *Cognitive Science*, 18, 387–438.
- Keenan, J. M., Baillet, S. D., & Brown, P. (1984). The effects of causal cohesion on comprehension and memory. *Journal of Verbal Learning and Verbal Behavior*, 23, 115–126.
- Kintsch, W., & van Dijk, T. A. (1978). Toward a model of text comprehension and production. *Psychological Review*, 85, 363–394.
- Kokinov, B. N. (1994). A hybrid model of reasoning by analogy. In K. J. Holyoak & J. A. Barnden (Eds.), *Advances in connectionist and neural computation theory: Vol. 2. Analogical connections* (pp. 247–318). Norwood, NJ: Ablex.
- König, P., & Engel, A. K. (1995). Correlated firing in sensory-motor systems. *Current Opinion in Neurobiology*, 5, 511–519.
- Lakoff, G., & Johnson, M. (1980). *Metaphors we live by*. Chicago: University of Chicago Press.
- Luce, R. D. (1959). *Individual choice behavior: A theoretical analysis*. New York: Wiley.
- Markman, A. B., & Gentner, D. (1993a). Structural alignment during similarity comparisons. *Cognitive Psychology*, 23, 431–467.
- Markman, A. B., & Gentner, D. (1993b). Splitting the differences: A structural alignment view of similarity. *Journal of Memory and Language*, 32, 517–535.
- Markman, A. B., Gentner, D., & Wisniewski, E. J. (1995). Comparison and connectionism: Implications of structure-sensitive processing for connectionist models. Manuscript in preparation, Columbia University.
- Markman, A. B., & Wisniewski, E. J. (1997). Similar and different: The differentiation of basic-level categories. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 23, 54–70.
- Marshall, J. A. (1995). Adaptive pattern recognition by self-organizing neural networks: Context, uncertainty, multiplicity, and scale. *Neural Networks*, 8, 335–362.
- Medin, D. L., Goldstone, R. L., & Gentner, D. (1993). Respects for similarity. *Psychological Review*, 100, 254–278.
- Milner, P. M. (1974). A model for visual shape recognition. *Psychological Review*, 81, 521–535.
- Mozer, M. C., Zemel, R. S., Behrmann, M., & Williams, C. K. (1992). Learning to segment images using dynamic feature binding. *Neural Computation*, 4, 650–665.
- Norman, D. A. (1986). Reflections on cognition and parallel distributed processing. In J. L. McClelland, D. E. Rumelhart, & the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition* (Vol. 2, pp. 531–546). Cambridge, MA: MIT Press.
- Novick, L. R. (1992). The role of expertise in solving arithmetic and algebra word problems by analogy. In J. I. D. Campbell (Ed.), *The*

- nature and origins of mathematical skills* (pp. 155–188). Amsterdam: Elsevier.
- Novick, L. R., & Holyoak, K. J. (1991). Mathematical problem solving by analogy. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 17*, 398–415.
- Ortony, A. (1979). Beyond literal similarity. *Psychological Review, 86*, 161–180.
- Plate, T. (1991). Holographic reduced representations: Convolution algebra for compositional distributed representations. In J. Mylopoulos & R. Reiter (Eds.), *Proceedings of the 12th International Joint Conference on Artificial Intelligence* (pp. 30–35). San Mateo, CA: Morgan Kaufmann.
- Poggio, T., & Edelman, S. (1990). A neural network that learns to recognize three-dimensional objects. *Nature, 343*, 263–266.
- Pollack, J. B. (1990). Recursive distributed representations. *Artificial Intelligence, 46*, 77–106.
- Premack, D. (1983). The codes of man and beasts. *Behavioral and Brain Sciences, 6*, 125–167.
- Premack, D. (1988). Minds with and without language. In L. Weiskrantz (Eds.), *Thought without language* (pp. 46–65). Oxford, England: Oxford University Press.
- Reed, S. K., Ernst, G. W., & Banerji, R. (1974). The role of analogy in transfer between similar problem states. *Cognitive Psychology, 6*, 436–440.
- Reeves, L. M., & Weisberg, R. W. (1994). The role of content and abstract information in analogical transfer. *Psychological Bulletin, 115*, 381–400.
- Robin, N., & Holyoak, K. J. (1994). Relational complexity and the functions of prefrontal cortex. In M. S. Gazzaniga (Ed.), *The cognitive neurosciences* (pp. 987–997). Cambridge, MA: MIT Press.
- Ross, B. (1987). This is like that: The use of earlier problems and the separation of similarity effects. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 13*, 629–639.
- Ross, B. (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*, 456–468.
- Ross, B. H., & Kennedy, P. T. (1990). Generalizing from the use of earlier examples in problem solving. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 16*, 42–55.
- Rumelhart, D. (1980). Schemata: The building blocks of cognition. In R. Spiro, B. Bruce, & W. Brewer (Eds.), *Theoretical issues in reading comprehension* (pp. 33–58). Hillsdale, NJ: Erlbaum.
- Seifert, C. M., McKoon, G., Abelson, R. P., & Ratcliff, R. (1986). Memory connections between thematically similar episodes. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 12*, 220–231.
- Shastri, L., & Ajjanagadde, V. (1993). From simple associations to systematic reasoning: A connectionist representation of rules, variables and dynamic bindings using temporal synchrony. *Behavioral and Brain Sciences, 16*, 417–494.
- Smith, L. B. (1989). From global similarities to kinds of similarities: The construction of dimensions in development. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning* (pp. 146–178). New York: Cambridge University Press.
- Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence, 46*, 159–216.
- 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*, 913–933.
- Spellman, B. A., & Holyoak, K. J. (1996). Pragmatics in analogical mapping. *Cognitive Psychology, 31*, 307–346.
- Thagard, P., Holyoak, K. J., Nelson, G., & Gochfeld, D. (1990). Analog retrieval by constraint satisfaction. *Artificial Intelligence, 46*, 259–310.
- Touretzky, D., & Hinton, G. (1988). A distributed production system. *Cognitive Science, 12*, 423–466.
- Tovee, M., & Rolls, E. (1992). Oscillatory activity is not evident in the primate temporal visual cortex with static stimuli. *Neuroreport, 3*, 369–372.
- Trabasso, T., & van den Broek, P. (1985). Causal thinking and the representation of narrative events. *Journal of Memory and Language, 24*, 612–630.
- Ullman, S., & Basri, R. (1991). Recognition by linear combinations of models. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 13*, 992–1006.
- Vaadia, E., Haalman, I., Abeles, M., Bergman, H., Prut, Y., Slovin, H., & Aertsen, A. (1995). Dynamics of neuronal interactions in monkey cortex in relation to behavioural events. *Nature, 373*, 515–518.
- van den Broek, P. (1988). The effects of causal relations and hierarchical position on the importance of story statements. *Journal of Memory and Language, 27*, 1–22.
- von der Malsburg, C. (1981). The correlation theory of brain function. (Internal Rep. 81-2). Göttinger, Germany: Max-Planck-Institute for Biophysical Chemistry, Department of Neurobiology.
- von der Malsburg, C. (1985). Nervous structures with dynamical links. *Ber. Bunsenges. Phys. Chem., 89*, 703–710.
- von der Malsburg, C., & Buhmann, J. (1992). Sensory segmentation with coupled neural oscillators. *Biological Cybernetics, 67*, 233–242.
- Wharton, C. M., Holyoak, K. J., Downing, P. E., Lange, T. E., & Wickens, T. D. (1991). Retrieval competition in memory for analogies. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society* (pp. 528–533). Hillsdale, NJ: Erlbaum.
- Wharton, C. M., Holyoak, K. J., Downing, P. E., Lange, T. E., Wickens, T. D., & Melz, E. R. (1994). Below the surface: Analogical similarity and retrieval competition in reminding. *Cognitive Psychology, 26*, 64–101.
- Wharton, C. M., Holyoak, K. J., & Lange, T. E. (1996). Remote analogical reminding. *Memory & Cognition, 24*, 629–643.

Appendix A

The LISA Algorithm

The general sequence of events in LISA's operation is summarized below. The details of each of these steps are described in the subsections that follow.

General Sequence of Events

1. Construct and initialize the network. Set all inputs, activations, and mapping connection weights and buffers to zero.
2. Run the network for C cycles, where C is defined by the user. On each cycle, repeat the following until no propositions are designated (by the user) to be selected:
 - 2.1. As designated by the user, select one analog, D , to be the driver, a set of analogs, R , to serve as recipients for mapping, and a set of analogs, L , to remain dormant for retrieval from LTM. R or L , or both, may be empty sets.
 - 2.2. Initialize the activation state of the network: Set all inputs, activations, and outputs to zero.
 - 2.3. As designated by the user, select one P unit, P_s , in D to be active. Set its activation to 1.0. If P_s is NIL then proceed to step 3.
 - 2.4. Repeatedly update the state of the network in discrete time steps, $t = 1$ to $MaxT$, where $MaxT$ is 300 times the number of case roles on P_s . On each time step, t , do the following:
 - 2.4.1. Update the modes of all P units.
 - 2.4.2. Update the inputs to all units in D (except P_s).
 - 2.4.3. Update the global inhibitor.
 - 2.4.4. Update the inputs to all units in all recipient and dormant analogs.
 - 2.4.5. Update the inputs to all semantic units.
 - 2.4.6. Update all units' activations (except P_s).
 - 2.4.7. Update the buffers on all cross-analog connections.
 - 2.5. If directed to do so by the user, update the mapping weights and initialize the buffers.
3. Save the final values of the mapping connection weights.

2.4.1. Updating Proposition Modes

Because of the one-level restriction, P units operate in three distinct modes: *parent*, *child*, and *neutral*. In parent mode, P units act as the parent of a larger structure, exchanging input only with SP units below itself, that is, SPs representing its case roles. A P unit in child mode exchanges input only with SPs above itself, that is, SPs relative to which it serves as an argument. A P unit in neutral mode exchanges input with SPs both above and below itself. When the state of the network is initialized (step 2.2 above), all P units enter neutral mode except P_s , which enters parent mode. P units update their modes on the basis of their inputs from SP units above and below themselves and, in the case of P units in a recipient analog, from P units in the driver:

$$m_i = \begin{cases} \text{Parent if } SP_{\text{Below}} + Prop_{\text{Parent}} - SP_{\text{Above}} - Prop_{\text{Child}} > 0.01 \\ \text{Child if } SP_{\text{Below}} + Prop_{\text{Parent}} - SP_{\text{Above}} - Prop_{\text{Child}} < -0.01 \\ \text{Neutral otherwise} \end{cases} \quad (\text{A1})$$

where m_i is the mode of P unit i . SP_{Below} , SP_{Above} , $Prop_{\text{Parent}}$, and $Prop_{\text{Child}}$ are weighted activation sums:

$$\sum a_j w_{ij}, \quad (\text{A2})$$

where a_j is the activation of unit j , and w_{ij} is the connection weight from

j to P unit i . For SP_{Below} , j are SPs below i ; for SP_{Above} , j are SPs above i ; for $Prop_{\text{Parent}}$, j are P units in D that are in parent mode; and for $Prop_{\text{Child}}$, j are proposition units in D that are in child mode. Within an analog (i.e., for SP_{Below} and SP_{Above}), w_{ij} are 1.0 for all i and j under the same P unit and 0 for all other i and j . For $Prop_{\text{Parent}}$ and $Prop_{\text{Child}}$, w_{ij} are mapping weights (-1.0 – $+1.0$). Only P units in the recipient analog or analogs update their modes based on the mapping connections. For all other P units, $Prop_{\text{Parent}}$ and $Prop_{\text{Child}}$ are set to zero.

2.4.2. Driver Inputs

Because the driver controls the operation of the network—and, in particular, the generation of synchronized patterns on the semantic units—inputs are updated differently in the driver than in recipient and dormant analogs. Updating is nonetheless synchronous in the sense that, throughout the network, all inputs are updated before any activations are updated.

SP units. SP units in the driver drive the activity of the predicate, object, and P units below themselves, and therefore of the semantic units, and all units in the recipient or dormant analogs: Synchrony starts in the driver SPs and is carried throughout the rest of the network. Each SP consists of a pair of units, an *excitor* and an *inhibitor*, whose inputs and activations are updated separately. Each SP inhibitor is yoked to the corresponding excitor and causes the excitor's activation to oscillate. In combination with strong SP-to-SP inhibition, the inhibitors cause separate SPs to fire out of synchrony with one another.

In the driver, SP excitors receive input from three sources: (a) excitatory input from P units above themselves and from P, object, and predicate units below themselves, (b) inhibitory input from other SPs in the driver and from all driver P units except their parents and children, and (c) inhibitory input from the corresponding SP inhibitor. On each iteration, t , the net input, NE_i , to SP excitor i is:

$$NE_i = 2P_i^+ - 1.5P^- + 0.5A_i + 0.5R_i - 6 \sum_{j \neq i} E_j - 3I_i + \rho, \quad (\text{A3})$$

where P_i^+ is the activation of the P above SP excitor i , P^- is the sum of activations of all proposition units relative to which i does not serve as parent or child, A_i is the activation of i 's argument unit (i.e., object unit or child P unit), R_i is activation of i 's predicate unit, E_j is the activation of any SP excitor j ($j \neq i$) in the driver, I_i is the activation of the inhibitor on SP_i , and ρ is a random number in the range -0.1 – $+0.1$.

An SP inhibitor receives input only from the corresponding excitor, I_i , the activation of SP inhibitor i at time t , changes according to:

$$\Delta I'_i = \begin{cases} \left\{ \begin{array}{l} 0.001, I'_i \leq 0.1 \\ 0.1, I'_i > 0.1 \end{array} \right\}, E'_i > 0.5 \text{ and } r'_i \\ \left\{ \begin{array}{l} -0.00105, I'_i \geq 0.9 \\ -0.03, I'_i < 0.9 \end{array} \right\} \text{ otherwise,} \end{cases} \quad (\text{A4})$$

where r'_i is a Boolean variable that is set to *true* whenever $I'_i \leq 0$ and to *false* whenever $I'_i \geq 1$; r'_i remains at its last set value on any t when $0 < I'_i < 1$.

These operations cause the activation of an inhibitor to grow and decay in four phases. (For a qualitatively similar algorithm, see von der Malsburg & Buhmann, 1992.) When a proposition is first selected, all I_i are set to zero and all r'_i are initialized to *true*, putting the inhibitors in *slow growth* phase (top row, Equation A4). During this phase, I_i

grows by 0.001 whenever $E_i > 0.5$. The inhibitor will remain in this phase until its activation reaches 0.1, at which point I_i grows rapidly toward 1.0 (the *fast growth* phase; second row, Equation A4). Highly active inhibitors drive their excitors to inactivity (Equation A3). The slow growth parameter (0.001) was chosen to allow SP excitors to remain active for approximately 100 iterations (it takes 100 iterations for the inhibitor to enter rapid growth). When $I'_i > 1$, it is set to 1.0 and r_i is set to *false*, putting the inhibitor into *slow decay* phase (third row, Equation A4). The slow decay parameter (-0.00105) was chosen to cause an inhibitor's activity to take 95 iterations to decay to 0.9, at which point the inhibitor enters the *fast decay* phase. During fast decay, I_i drops more rapidly toward zero, giving E_i an opportunity to grow in response to an excitatory input from the P unit above itself (provided the excitor is not being inhibited by another SP). When I'_i drops below zero, it is set to zero and r_i is set to *true*, putting the inhibitor back into slow growth phase. Collectively, the growth and decay parameters in Equation A4 make it possible for three SPs (the maximum number on any linguistically natural proposition) to fire out of synchrony in a "time-sharing" fashion: All three SP excitors will have the opportunity to become active once before any excitor becomes active twice.

P, object, and predicate units. P_S^t (i.e., whichever P unit in the driver is selected at time t) does not update its input. All other driver P units (which will be in either neutral or child mode; see 2.4.2 above) receive input only from SPs above themselves. Likewise, object and predicate units receive input only from the SPs above themselves. Effectively, activation in the driver is propagated downward only. The net input, N_i , to any P, object, or predicate unit in the driver is:

$$N_i = \sum_j (\epsilon E_j - 0.25 I_j), \quad (A5)$$

where j is any SP unit above unit i , ϵ is 1.5 for P units and 2.0 for predicate and object units, E_j is the activation of the SP excitor, and I_j is the activation of the SP inhibitor.

2.4.3. Global Inhibitor

All SP units have both excitors and inhibitors, but SP inhibitors are updated only in the driver. This convention corresponds to the assumption that attention is directed to the driver and serves to control dynamic binding (see also Hummel & Biederman, 1992; Hummel & Stankiewicz, 1996). The activity of a *global inhibitor* (von der Malsburg & Buhmann, 1992) helps to coordinate the activity of structure units in the driver and recipient or dormant analogs. The global inhibitor, Γ , inhibits structure units (except P units in parent mode) in all nondriver analogs, and is itself strongly inhibited by SPs in the driver. Specifically, the global inhibitor is inhibited to inactivity ($\Gamma = 0$) by any SP excitor in the driver whose activation is greater than or equal to 0.7; it becomes active ($\Gamma = 1$) whenever no SP excitor in the driver has an activation greater than or equal to 0.7. During a transition between two driver SPs' firing (i.e., when one SP's inhibitor grows rapidly, allowing the other SP excitor to become active), there is a brief period when no SP excitors in the driver has activations greater than 0.7. During these periods, $\Gamma = 1$ and inhibits all nondriver structure units (except P units in parent mode) to inactivity. Effectively, Γ serves as a "refresh" signal, permitting changes in the patterns of activity of nondriver analogs to keep pace with changes in the driver.

2.4.4. Recipient and Dormant Analog Inputs

Structure units in recipient and dormant analogs are updated in exactly the same way except that the former receive input from the driver directly via the mapping connections, whereas the latter do not. Input to structure units can be divided into four sources: *within-proposition* excitatory input, P ; *within-class* (e.g., SP-to-SP, object-to-object, etc.) inhibitory

input, C ; *out-of-proposition* inhibitory input, O (i.e., P units in one proposition inhibit SP units in others, and SPs in one inhibit predicate and object units in others); and both excitatory and inhibitory input, M , received via the cross-analog *mapping* weights. The net input, N_i^α , to any structure unit i in analog α is the sum:

$$N_i^\alpha = P_i^\alpha - C_i^\alpha - O_i^\alpha + \rho M_i^\alpha, \quad (A6)$$

where $\rho = 1$ for α in R and $\alpha = 0$ for α in L . For P units in parent mode, O is always zero.

P units. The input terms for P units in R and L are:

$$P_i^\alpha = \delta_i^\alpha 0.5 \sum_j E_j^\alpha + \pi_i^\alpha 0.75 \sum_k E_k^\alpha - 10\pi_i^\alpha \Gamma, \quad (A7)$$

where $\delta_i^\alpha = 0$ for P units, i , in child mode and 1 for all other i ; $\pi_i^\alpha = 0$ for i in parent mode and 1 for all other i ; j are SP units below i ; and k are SP units above i . Recall that Γ is the activation of the global inhibitor (0 or 1). P units in parent or neutral mode integrate their inputs over time to a greater extent than P units in child mode. In aid of this, for P units in parent or neutral mode, E_j^α is the maximum activation achieved by SP unit j over all t since P_S (in D) was initially selected; for P units in child mode, E_j^α is simply the activation of SP unit k at time t .

$$C_i^\alpha = \frac{\kappa \sum_{j \neq i} 3a_j^\alpha}{1 + \sum_j A_j^\alpha}, \quad (A8)$$

where $A_i^\alpha = 1$ if $a_i^\alpha > 0.01$ and 0 otherwise. P units in parent or neutral mode inhibit one another only; for such units, j are other P units in parent or neutral mode, and $\kappa = 1.5$. P units in child mode exchange inhibition with one another and with object units; for such units, j are P units in child mode and object units, and $\kappa = 1.0$. P units in child mode receive out-of-proposition inhibition from SP units relative to which they do not serve as arguments:

$$O_i^\alpha = \bar{\omega} \sum_j a_j, \quad (A9)$$

where j are SP units that are neither above nor below i , and $\bar{\omega} = 0.25$.

$$M_i^\alpha = \sum_{j \in D} [a_j w_{ij} m(i, j)] \quad (A10)$$

where a_j is the action of P unit j in D , w_{ij} ($-1, \dots, 1$) is the weight on the mapping connection from j to i , and $m(i, j)$ evaluates to 1 if i and j are in the same mode or i is in neutral mode, and to 0 otherwise.

SP units. The input terms for SP excitors in R and L are:

$$P_i^\alpha = \delta_j a_j + \pi_k 0.5 a_k + 0.5 a_l - 10\Gamma, \quad (A11)$$

where a_j is the activation of the P unit, j , above SP i ; $\delta_j = 0$ if j in child mode and 1 otherwise; a_k is the activation of any object unit or P unit, k , below i ; π_k is 0 if k is a P unit in parent mode and 1 otherwise; and a_l is the activation of predicate unit, l , below i .

C_i^α (within-class inhibition) is given by Equation A8, where j ($j \neq i$) are SP units in α , and $\kappa = 1.5$. O_i^α (out-of-proposition inhibition) is given by Equation A9, where j are P units in α that are in parent mode and are not above SP i . (i.e., P units representing propositions of which SP i is not a part), and $\bar{\omega} = 1.5$.

$$M_i^\alpha = \sum_{j \in D} (a_j w_{ij}), \quad (A12)$$

where a_j is the action of SP unit j in D , and w_{ij} ($-1, \dots, 1$) is the weight on the mapping connection from j to i .

Predicate units. The input terms for predicate units in R and L are:

$$P_i^\alpha = 0.5 \frac{\sum_{j=1}^{N_i} a_j}{1 + N_i} + \sum_k a_k - 10\Gamma, \quad (\text{A13})$$

where j are semantic units to which i is connected, N_i is the number of such units, and k are SP units above i .

C_i^α (inhibition from other predicate units in α to predicate i in α) is given by Equation A8, where j ($j \neq i$) are predicate units in α , and $\kappa = 1.0$. O_i^α (out-of-proposition inhibition from SPs to predicate unit i in α) is given by Equation A9, where j are SP units not above predicate unit i (i.e., SPs with which i does not share an excitatory connection), and $\omega = 1.5$. M_i^α (cross-analog input to predicate unit i) is given by Equation A12, where j are predicate units in D .

Object units. The input terms for object units in R and L are:

P_i^α (within-proposition excitatory input to object unit i in α) is given by Equation A13, where j are semantic units to which i is connected and k are SP units above i . C_i^α (inhibition from other object units in α to object i in α) is given by Equation A8, where j ($j \neq i$) are predicate units in α , and $\kappa = 1.0$. O_i^α (out-of-proposition inhibition from SPs to object unit i in α) is given by Equation A9, where j are SP units not above object unit i , and $\omega = 0.25$. M_i^α (cross-analog input to predicate unit i) is given by Equation A12, where j are object units in D .

2.4.5. Semantic Unit Inputs

The net input to semantic unit i is:

$$N_i = \sum_{\alpha \in D, R} \sum_j a_j^\alpha w_{ij}^\alpha, \quad (\text{A14})$$

where $w_{ij}^\alpha = 1$ for any predicate or object unit, j , in analog α ($\alpha \in [R, D]$) that is connected to i ; $w_{ij}^\alpha = 0$ for all other j . The activation of a semantic unit is equal to its input.

2.4.6. Activation

P units, SP excitors, predicate units and object units (excitatory units) update their activations by:

$$\Delta a_i^\alpha = 0.3N_i^\alpha(1 - a_i^\alpha) - \phi a_i^\alpha, \quad (\text{A15})$$

where ϕ is 0.1 for units in driver and recipient analogs and 0.3 for units in dormant analogs.

2.4.7. Mapping (Cross-Analog) Connection Buffers

Mapping connections are modified (learned) between driver and recipient analogs only. The mapping connection from unit j to unit i has a weight, w_{ij} , and a buffer. The buffer has a growth term, g_{ij} , and a

decay term, d_{ij} , which are updated independently. The growth term grows whenever units i and j are active simultaneously:

$$\Delta g_{ij}^\alpha = a_i^\alpha a_j^\alpha. \quad (\text{A16})$$

Δg_{ij} is set to zero for proposition units in different modes. The decay term grows to the extent that unit i (on the receiving end of the connection) is more active than unit j (on the sending end):

$$\Delta d_{ij}^\alpha = \eta^- \lfloor a_i^\alpha - a_j^\alpha \rfloor, \quad (\text{A17})$$

where η^- , the decay rate parameter, is set to 0.00001.

2.5. Mapping (Cross-Analog) Connection Weights

At the end of each phase set (i.e., as designated by the user), the mapping buffer growth and decay terms are converted to weight values and then flushed, that is, initialized to zero. By Equation A16, buffer growth and decay terms are unbounded, so before weight updating, they are first normalized:

$$g_{ij}^c = \frac{g_{ij}^c}{g\text{-max}^c}, \quad (\text{A18})$$

and

$$d_{ij}^c = \frac{d_{ij}^c}{g\text{-max}^c}, \quad (\text{A19})$$

where c designates the class of the mapping weight (object, predicate, SP, or P unit) and $g\text{-max}^c$ is the maximum growth term on any mapping connection in class c . Normalization is performed within classes of structure unit connections (e.g., SP-to-SP connection buffers are normalized on the basis of the largest SP-to-SP connection buffer growth term, P-to-P buffers are normalized on the basis of P-to-P growth terms, etc.). Due to the normalization in Equations A18 and A19, all buffer growth and decay terms are bounded between zero and one.

The buffer growth terms are then subjected to a one-to-one (subtractive) normalization in which the magnitude of each term, g_{ij} , is converted into a growth penalty and divided among all other growth terms leading to the same recipient unit (i):

$$g_{ij} = g_{ij} - \frac{\sum_{k=1}^{n_i} g_{ik}}{(n_i - 1)}, \quad k \neq j, \quad (\text{A20})$$

where n_i is the number of mapping connections leading into unit i . As a result of this normalization, the sum, $\sum g_{ij}$, of all growth terms leading into unit i is always zero. This normalization implements the one-to-one mapping constraint by penalizing the correspondence (buffer growth) between units i and j on the basis of the correspondence between i and all other $k \neq j$.

Finally, the buffer growth and decay terms are used to update the mapping weights, w_{ij} , and initialized:

$$\Delta w_{ij} = \eta^+ g_{ij}(1 - w_{ij}) - d_{ij}(1 + w_{ij}), \quad (\text{A21})$$

where η^+ the learning rate, is set to 0.5. When $g_{ij} \leq 0$ and $d_{ij} > 0$, w_{ij} decays toward -1. The weights are truncated below -1.

(Appendices continue)

Appendix B

LISA Representations of Convergence Analogs

Propositional Representations of Analogs

Target analog ("Radiation Problem"; Duncker, 1945)

- P1 (inside tumor stomach)
- P2 (surround tissue tumor)
- P3 (between tissue raysource tumor)
- P4 (canmake raysource hirays)
- P5 (candestroy hirays tumor)
- P6 (candestroy hirays tissue)
- P7 (canmake raysource lorays)
- P8 (cannotdest lorays tumor)
- P9 (cannotdest lorays tissue)
- P10 (use doctor hirays)
- P11 (destroyed tumor)
- P12 (want doctor P11)
- P13 (destroyed tissue)
- P14 (notwant doctor P13)
- P15 (ifthen P10 P13)

Source analogs

- Close analog ("The Surgeon"; Keane, 1986)
 - P1 (inside tumor brain)
 - P2 (surround tissue tumor)
 - P3 (between tissue raysource tumor)
 - P4 (canmake raysource hirays)
 - P5 (candestroy hirays tumor)
 - P6 (candestroy hirays tissue)
 - P7 (canmake raysource lorays)
 - P8 (cannotdest lorays tumor)
 - P9 (cannotdest lorays tissue)
 - P10 (use surgeon hirays)
 - P11 (destroyed tumor)
 - P12 (want surgeon P11)
 - P13 (destroyed tissue)
 - P14 (notwant surgeon P13)
 - P15 (ifthen P10 P13)

Far analog ("The General"; Gick & Holyoak, 1980)

- P1 (inside fortress country)
- P2 (between villages army fortress)
- P3 (conform army largeunit)
- P4 (capture largeunit fortress)
- P5 (conform army smallunit)
- P6 (capture smallunit fortress)
- P7 (use general largeunit)
- P8 (captured fortress)
- P9 (blowup mines largeunit)
- P10 (blowup mines villages)
- P11 (want general P8)
- P12 (notwant general P9)
- P13 (notwant general P10)
- P14 (ifthen P7 P9)
- P15 (ifthen P7 P10)

Schema ("Convergence Schema"; Gick & Holyoak, 1983)

- P1 (inside target location)
- P2 (between asset forcesource target)
- P3 (canmake forcesource hiforce)
- P4 (canalter hiforce target)
- P5 (canmake forcesource loforce)
- P6 (cnalter loforce target)
- P7 (use goodguy hiforce)

- P8 (altered target)
- P9 (altered asset)
- P10 (want goodguy P8)
- P11 (notwant goodguy P9)
- P12 (ifthen P7 P9)

Semantic Features

Objects

- tumor: object biological negative small tumor1
- stomach: object biological positive organ stomach1
- tissue: object biological positive tissue tissue1
- raysource: object artifact forcesource raysource1
- hirays: energy radiation strong force hirays1
- lorays: energy radiation weak force lorays1
- doctor: object animate person profession medical doctor1
- brain: object biological positive organ brain1
- surgeon: object animate person profession medical surgeon1
- fortress: object building military large fortress1
- country: location place political large country1
- army: humangrp military large strong forcesource army1
- mines: object artifact explosive danger mines1
- general: object animate person profession military general1
- largeunit: humangrp military large strong force lunit1
- smallunit: humangrp military small weak force sunit1
- villages: location place humangrp small villages1
- target: object
- loforce: weak force
- forcesource: forcesource
- location: location
- hiforce: strong force
- goodguy: object animate person profession
- asset: object positive

Predicates

- inside: state location in1
- surround: state location around surround1
- between: state location intervene between1
- canmake: trans potential change generate canmake1
- candestroy: trans potential change reduce candestroy1
- cannotdest: trans potential neutral powerless cannotdest1
- ifthen: conditional
- use: trans utilize use1
- want: mentalstate goal want1
- notwant: mentalstate goal negative notwant1
- destroyed: state change reduce destroyed1
- conform: trans potential change generate conform1
- capture: trans potential change reduce capture1
- cnocapture: trans potential neutral powerless cnocapture1
- blowup: trans change reduce explosive blowup1
- captured: state change reduce capture1
- cnalter: trans potential neutral powerless cnalter1
- canalter: trans potential change generate canmake1
- altered: state change
- harmed: state change reduce

Received March 11, 1996

Revision received July 1, 1996

Accepted July 1, 1996 ■