Abstract

Surrogate reasoning is reasoning whose task is partially taken over by operations on external aids, such as sentences, diagrams, physical models, mathematical models, and computers. Drawing on the basic concepts in situation theory, we present a semi-formal model of surrogate reasoning. We claim that the relative advantages and disadvantages of different forms of surrogate reasoning can be explained with reference to the ways in which the default constraints on surrogates intervene in the processes of reasoning. We define and examine two prominent patterns of such constraint intervention (dubbed "free rides" and "overdetermined alternatives"). We also introduce the notion of "constraint projection" and try to capture the general framework in which different forms of constraint intervention take place in surrogate reasoning. Key words: surrogate reasoning, diagrammatic reasoning, constraint projection, situation theory, modeling.

Surrogate reasoning

Jon Barwise and Atsushi Shimojima
Indiana University

Introduction

Surrogate reasoning in problem solving A "problem" is a mystery, a question, a source of uncertainty. Problems range from the mundane, like finding your way to some place you have never been before, to the profound, say that posed by Fermat's Last Theorem or understanding the origin of life. "Problem solving" involves eliminating or reducing the uncertainty by finding an a answer to the problem. "Reasoning" is the process by which one starts with a problem and attempts to arrive at a solution.

Following Barwise and Etchemendy (in press), we suggest conceptualizing of a problem in terms of a space of unexplored possibilities, the mystery being what structure this space has and what resides there. In terms of this metaphor, we can think of problem solving as finding out enough about the space of possibilities to eliminate the mystery at hand. In these terms "reasoning" is the process of exploring the space of possibilities presented to us by the problem.

The most usual way to think about problem solving within cognitive science is as a variety of thought. After all, we do think when we solve problems, and sometimes it seems that that is just about all we do. Conversely, a fair amount of thinking can be seen as problem solving: solving the problem of getting around in the world. Indeed, it is not uncommon within cognitive science to find thinking and problem solving more or less identified.

This identification is a mistake. On the one hand, there are forms of thought that are not problem solving. But more to the point of this paper, there is frequently more than thinking involved in finding a solution to a problem. If a problem is at all difficult, we typically solve it with the use of external aids. Such aids include things like pencil and paper, calculators, computers, sentences, diagrams, physical models, mathematical models, and human experts. It is this sort of activity which we refer to as *surrogate reasoning*: the exploration of a space of possibilities using tools external to that space.

Once realized, the ubiquity of surrogate reasoning is almost too obvious to belabor. But given the lack of attention that has been paid to it, perhaps it deserves a bit more space. Suppose you are going to move to a new town and decide to build a new home. Various problems arise: where should you build it?

How can you afford it? Who will build it? And what will it be like? Typically, solving each of these problems will involve surrogate reasoning of various kinds. Take, for example, the last, that of finding a design for a house that will fit your needs, budget, and piece of land.

What tools do we use to solve this problem? Architects, sketches, drawings, furniture-shaped cutouts, scale models, and computer simulations of interior spaces. No one in their right mind would try to solve the problem of designing their house by means of pure thought, pondering just over the house itself.

Let's take an example which seems at first sight more amenable to pure thought: the solution to a mathematical problem. One of the most obvious but under appreciated facts about mathematical reasoning is that it is done on things like pencil and paper (or chalk and blackboards). Whether one is trying to prove a theorem or just compute 345×429, the use of surrogates is more or less indispensable.

To list a few more examples of surrogates used in problem solving:

Sentences We record information at hand in the form of sentences, and derive further information by using those sentences both as memory aids and reference sites. For example, when we construct a proof within an axiomatized theory such as ZFC set theory, we typically start with writing down the assumptions in the form of sentences. We then refer to the axioms and the sentences we have written down, and add more sentences that are "derivable" on the pre-determined inference rules. These new sentences typically let us read off pieces of information different from our initial assumptions.

Diagrams We often obtain a quicker and easier solution of a problem by representing information in a diagram. We use versions of Venn diagrams or Euler circles in solving problems about inclusion and membership relations among sets and individuals. Diagrams are more or less indispensable tools in geometry proofs. Maps, flow-charts, blue-prints, graphs, and tables are used not only as final displays of information, but also as aids to an on-going process of problem solving.

Physical models Architects use various kinds of architectural models in designing a building. They use models to facilitate their reasoning about the outlook, the sun-orientation, the ventilation, the acoustic condition, the lighting condition, and the temperature distribution of the building that they have in mind. Researchers use dummies in car crush tests to predict the possible damages that a human body receives in various situations of car crush.

Mathematical models and computer simulation Scientists, system analysts, and product designers often use mathematical models in place of physical models. Mathematical models are typically presented as mathematical theories often consisting of a system of equations whose solutions provide quantitative predictions of the target phenomena. In many cases, mathematical models are implemented as computer programs that let computers display the behaviors of the targets in one way or another.

Heterogeneous surrogates It is rather rare that only one type of surrogates is used in a problem solving. We record the behaviors of a physical model in the form of sentences. Geometry proofs are most natural when a mixture of sentences and diagrams is used. Even a mixture of different types of diagrams is used in analyzing the behaviors of complex system, such as computer hardware and human bodies.

Main questions At first sight, surrogate reasoning can seem like a strange thing to do. Rather than reason solely about the problem at hand, we have to also reason about the aid or aids used, and about their relationships to the problem. In this way we seem to replace one problem by three problems. However, as the above examples show, there are good reasons to think that there can be a large payoff: one may well be able to solve problems by means of surrogate reasoning that one could not solve without it. On the other hand, for a given problem, various forms of surrogate reasoning may be more or less appropriate than others, where appropriateness is judged in terms of some combination of efficiency and reliability. For example, calculators provide a more efficient and reliable solutions to a wide range of calculation problems than abacuses or pencil and paper do, which are in turn more efficient and reliable than resorting to mental arithmetic. Asking experts often provides a quicker solution to a problem than working it out on one's own. Venn diagrams let us solve a range of problems in a more perspicuous fashion than traditional sentence-based rules.[1]

In this paper we start to explore the relative advantages and disadvantages of different forms of surrogate reasoning in different tasks of problem solving. The main question we address in this paper is this: what are the factors that make a particular form of surrogate reasoning particularly appropriate for a given problem domain. What exactly are the mechanisms by which some forms of surrogate reasoning gain or lose these relative advantages? Are there principled explanations of these differences in efficiency and reliability of different forms of surrogate reasoning?

We will show that there is a path to such principled explanations. The gist of our idea is the following. In surrogate reasoning, one chooses a certain collection

[1]There are additional considerations involved in the choice of a particular form of surrogate reasoning. Some forms of surrogate reasoning are easier to learn than others. Some are more widely applicable than others. Some forms of surrogate reasoning not only solve a particular problem, but also give greater general insight into the problem domain, insight that may be useful in solving future problems. We, however, confine ourselves to the issues of reliability and efficiency in this paper.

of objects as "source objects" to reason about other "target objects" by using some "semantic conventions." We operate on the source objects, get information about the resulting source objects, and then interpret that information to obtain information about the target objects.

The source objects are themselves objects in the world, and as such subject to certain constraints imposed by natural (and other) laws. Now, depending on the ways we operate on and interpret our source objects, these constraints intervene in the above process of derivation in various ways. Consequently these constraints may affect the overall processes of problem solving. Our basic thesis is that most advantages and disadvantages of various forms of surrogate reasoning can be explained with reference to the ways these constraints on source objects intervene in the processes of problem solving. Let us call this thesis the "constraint hypothesis."

Surrogate reasoning and cognitive science Science attempts to solve problems. As a branch of science, cognitive science attempts to solve the problems about cognition. Probably the most profound question addressed by cognitive science has to do with the nature of thought. What is it to think? How do people do it? What are the cognitive processes involved, and how are they related to the physical processes in the brain?

Despite this overwhelming trend, we think that the subject of surrogate reasoning is a very important one for cognitive science. One reason is that, as we have seen, it is a ubiquitous form of cognitive activity. In fact, it may be that the natural domain of cognitive science is problem solving, with its reliance on surrogates, not thinking *per se*. We think this shift in focus could give a new grip on old problems.

There is another reason to think that surrogate reasoning is an important subject of cognitive studies. Cognitive science makes great use of metaphor in trying to account for the nature of cognition. "The mind is like a Turing machine." "Thinking is computation." "Thinking is manipulating sentences in the language of thought." "Thinking is the construction and examination of mental models."

Metaphor is itself a form of surrogate reasoning. We try to understand one thing in terms of something else. But that is not the point we are trying to make here. Rather, look at the things that thinking is being likened to in these various metaphors: computation on a computer, manipulating sentence tokens in some language, the constructing and examination of models. These are all powerful tools used in surrogate reasoning. It is as though, being impressed by the power of some tool for the use of problem solving, we take the function of that particular tool to be paradigmatic of all thought. One could even take the naive view that "thinking is listening to the voice of reason" as a metaphor where thinking is likened to another form of surrogate reasoning: taking advice from an expert. This view is surely unhelpful, though, since it amounts to nothing more than a homunculus theory of mind.

This is not to say that all such metaphors are wrong-headed. It may be that the tools used in particular forms of surrogate reasoning will provide illuminating models for understanding various forms of cognitive activity. We hope it will. Indeed, this is what makes us think that an understanding of the varieties and mechanisms of surrogate reasoning can make a direct contribution to the ongoing discussions in cognitive science.[2]

We will start our main discussion by spreading out the framework in which we understand surrogate reasoning in general. After defining key concepts such as "constraint" and "reasoning system," we will illustrate two particular ways, the cases of "free rides" and "overdetermined alternatives," in which the default constraints on surrogate objects intervene in the process of reasoning. We will then generalize our discussion and define when a reasoning system "projects" a constraint onto its target domain. The notion captures the general framework in which various forms of constraint intervention can take place in surrogate reasoning. We will close our discussion by reformulating our constraint hypothesis in terms of this general notion of constraint projection.

Basic concepts

Our approach to the problem of understanding surrogate reasoning is mathematical. We are developing a mathematical model of the process and using the model to understand the process. Reasons of time, space, and appropriateness for the audience prohibit us from giving the details of this model here. We will, instead, work at a rather informal and intuitive level.[3]

Target domain Reasoning is a process of problem solving. We see this process as an intentional one: a reasoner is trying to solve a problem *about* something. This thing may be very concrete (a car that won't work or a city you need to find your way around) or abstract (the natural numbers or the economic situation in the nation), but it is there (or at least presumed to be there). We assume that the target t of a problem is a certain *situation*, which can be classified by *states of affairs*, or *infons*, that hold in it.

Given a target situation t, we assume that there is a fixed set Θ of states of affairs that can possibly classify t. Given the set Θ, there is in turn a fixed set

[2]Indeed, some theorists do not only take surrogate reasoning as a useful model of thinking, but also *identify* thinking with a manipulation of mental surrogates. For those people, thinking is a species of surrogate reasoning, and our previous contrast between surrogate reasoning and thinking will seem misguided. Although we do not commit ourselves to such literalist views, our discussions of surrogate reasoning will have even more direct implications for them.

[3]Those familiar with situation theory, say Barwise (1989) and Devlin (1991), will find this framework already familiar.

T of possible situations that the members of Θ can classify. We call the pair $\mathcal{T} = \langle T, \Theta \rangle$ the *target domain* of the instance of reasoning in question. If t and θ are members of T and Θ respectively, we write $t \models \theta$, read "t supports θ, to indicate that θ hold in t. We call members of Θ "target infons."

We conceptualize a (well-formed) problem as a pair $P = \langle \Theta_I, \Theta_G \rangle$ of subsets of Θ. Intuitively, Θ_I is the initial information or "clues" that a problem solver initially has about the target situation t. Each member of Θ_G is possibly true information about t that counts as an answer to the problem P. Thus, when Holmes wonders who murdered Smith, he is addressing the problem $P = \langle \Theta_I, \Theta_G \rangle$ where Θ_I is the total information he possesses about the target situation t in which the murder occurred, and Θ_G comprises all the target infons of the form:

> X murdered Smith.

If Homes wonders whether or not Jones murdered Smith, then Θ_G would contain only two members:

> Jones murdered Smith.
>
> Jones does not murder Smith.

Holmes may just be wondering what happened in the murder situation t. In that case, Θ_G coincides with Θ—any information that can classify t counts as an answer to his question. If Holmes wonders what actions he should take to arrest Jones safely without hurting him, then the target situation of his problem is a future situation t' in which Holmes takes actions to arrest Jones. The initial information Θ_I of his problem comprises some background information about t' such as:

> Holmes arrests Jones.
>
> Jones has already shot a man.
>
> Jones has a gun.

and the possible answers Θ_G comprises the infons of the form:

> If Holmes does X_1, and then X_2, ..., and then X_n, then Holmes is safe and Jones is not hurt.

where X_1, ..., X_n are types of actions.

Surrogate domain While we use a surrogate object to reason about a target situation, the surrogate typically goes through a sequence of change in its states. We call these states of the surrogate *surrogate situations*. Thus, a calculator is in different surrogate situations before and after we push one of its keys; a diagram is in different situations before and after we add a stroke, and so on.

In order for our reasoning to use a surrogate, we must have ways of obtaining information about the surrogate situations. Typically, this is by perception: we look at the page or abacus, we read the sentences or look at a diagram, we listen to the expert or read their report. In any case, we must somehow inspect particular situations that our surrogate enters into, and obtain information about those situations. We call the pieces of information that classify surrogate situations "surrogate infons." Thus, the surrogate situations of a calculator c can be classified by the following surrogate infons:

The numeral "0" is displayed.

The numeral "5" is displayed.

The numeral "9" is displayed.

The numeral "45" is displayed.

The numeral "0" is not displayed.

The numeral "5" is not displayed.

$\vdots$

Just as in the cases of target domains, a set of surrogate situations S and a set of surrogate infons Σ form a domain of classification $\mathcal{S} = \langle S, \Sigma \rangle$ if S exhausts all the situations that the members of Σ can classify, and Σ exhausts all the infons that can classify the members of S. We call the domain $\mathcal{S} = \langle S, \Sigma \rangle$ the *surrogate domain* of the on-going process of reasoning.

Information link In order for information σ about a surrogate situation s to give us information θ about a particular target situation t, there must be (i) some sort of semantic convention $\sigma \Rightarrow \theta$ that holds at the level of information, and (ii) some sort of signaling relation $s \rightsquigarrow t$ that holds at the level of situations. Consider the following diagram:

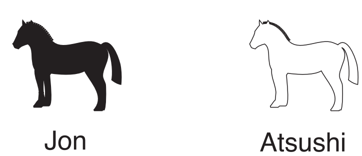

Figure 1

Let s be the particular situation on the paper that this diagram is in. Depending on what semantic convention $\Rightarrow$ and signaling relation $\rightsquigarrow$ are associated, the situation s can carry many different pieces of information about many different situations: s may be targeted at the current situation t of Jon's and Atsushi's properties, and carry the information that Jon's horse is black while

Atsushi's white; s may be targeted at the future situation t' of Jon's and Atsushi's properties as of three years later, and carry the information that Jon's horse is dead while Atsushi's alive; or s may be targeted at the current situation t'' of the horses named "Jon" and "Atsushi," and carry the information that Jon can run a hundred meters in less than ten seconds while Atsushi cannot.

Intuitively, the signaling relation $\rightsquigarrow$ determines what particular situation a surrogate situation s is targeted at, and $\Rightarrow$ determines how to translate the information about s into the information about the target situation. Thus, given a surrogate domain $\mathcal{S} = \langle S, \Sigma \rangle$ and a target domain $\mathcal{T} = \langle T, \Theta \rangle$ for reasoning, we can conceive $\rightsquigarrow$ as a binary relation from S to T and $\Rightarrow$ as a binary relation from Σ to Θ. We say that a surrogate situation s *represents* a target infon θ, and write $s \models\!\!\!\Rightarrow \theta$, if there is some σ such that $s \models \sigma$ and $\sigma \Rightarrow \theta$, that is, if some state of affairs σ that holds in s encodes the state of affairs θ relative to the semantic convention. If Θ_i is a set of infons, we write $s \models\!\!\!\Rightarrow \Theta$ to mean that $s \models\!\!\!\Rightarrow \theta$ for each $\theta \in \Theta_i$.

The pair of relations $L = \langle \rightsquigarrow, \Rightarrow \rangle$ provides what Barwise (1991) has called a "link" between the domain of source objects and the domain of target situations. The link is *reliable* if every surrogate situation is accurate. Of course surrogate reasoning is not in general reliable in this strong sense. What interests us is determining in what makes such a link more or less reliable, and what makes getting information about t indirectly, via s, more or less efficient than getting information about directly about t itself.

Domain of operations In reasoning with surrogate objects about some target situation t, reasoners do things to these surrogate objects, they operate on them in certain ways. Think of the operations in doing a computation with a calculator, computers, pencil and paper, or abacus. Or think of the operations involved in using sentences of first-order logic or Venn diagrams in solving a logic puzzle. Or think of asking an expert a question. These all involve starting with an initial surrogate situation s_I that signals a target situation t, performing some action a (or sequence of actions) on it, and getting a new surrogate situation s_O for t. Based on what information we obtain from s_O, we might perform additional actions, still in an attempt to get some desired piece of information about our target t. It is useful to think of these actions in realistic terms, as certain kinds of events in the world that start with an initial situation and result in another situation; we write this as $s_I \stackrel{a}{\longmapsto} s_O$ which is read "action a has s_I as input situation and s_O as output situation." We assume that each action has a unique pair of input and output situations.

Actions are particular events $a, a', \ldots$ in the world. These events have their own types. We suppose that these types are given by a system of infons $\omega, \omega', \ldots$ used to classify actions. Thus, we are assuming that actions used in surrogate reasoning constitute a domain of classification $\mathcal{A} = \langle A, \Omega \rangle$, just as surrogate situations and their target situations do. A domain of actions $\mathcal{A} = \langle A, \Omega \rangle$

consists of the class A of all possible actions that the members of Ω can classify, and the class Ω of all infons that can classify these actions. We write $a \models \omega$ if a is of type ω. Given a domain of actions $\mathcal{A} = \langle A, \Omega \rangle$ and a domain of surrogate situations $\mathcal{S} = \langle S, \Sigma \rangle$, we say that $\mathcal{A}$ *plays on* $\mathcal{S}$ iff every action in A has its input situation and output situation in S.

For example, multiplying 5 by 9 on a calculator c that initially displays "13" might be classified in this framework with five actions and six calculator situations as follows. (We assume that this calculator has two memory locations, the input and the register, and a display.)

$$c_0 \stackrel{a_1}{\longmapsto} c_1 \stackrel{a_2}{\longmapsto} c_3 \stackrel{a_3}{\longmapsto} c_3 \stackrel{a_4}{\longmapsto} c_4 \stackrel{a_5}{\longmapsto} c_5$$

where

$c_0 \models$ The numeral "13" is displayed.
$a_1 \models$ Atsushi taps on the "C" key.
$c_1 \models$ The input and the register contain 0; the numeral "0" is displayed.
$a_2 \models$ Atsushi taps on the "5" key.
$c_2 \models$ The input contains 5; the register contains 0; the numeral "5" is displayed.
$a_3 \models$ Atsushi taps on the "$*$" key.
$c_3 \models$ The input contains 5; the register contains 5; the numeral "5" is displayed.
$a_4 \models$ Atsushi taps on the "9" key.
$c_4 \models$ The input contains 9; the register contains 5; the numeral "9" is displayed.
$a_5 \models$ Atsushi taps on the "=" key.
$c_5 \models$ The input and the register both contain 45; the numeral "45" is displayed.

Methods Our notion of "method" is a generalization of the notion of "rule of inference" in logic. Roughly speaking, a method is a set of rules that specify what types of actions can be taken under what circumstances. For the purpose of this paper, we can model such a system of instructions by means of a class $\mathcal{M}$ of pairs $\langle \theta, \omega \rangle$ of target infons θ and action types ω. Intuitively, an individual rule $\langle \theta, \omega \rangle$ allows the following action: given that the target situation $t \models \theta$, carry out any action a such that $a \models \omega$. Thus, the method $\mathcal{M}$ *allows* a sequence of actions $a_1, \ldots, a_n$ of the type $\omega_1, \ldots, \omega_n$ on the assumptions $\{\theta_1, \ldots, \theta_n\}$ if for each ω_i, $\mathcal{M}$ contains a member $\langle \theta, \omega_i \rangle$ such that $\theta \in \{\theta_1, \ldots, \theta_n\}$.

Constraints The concept of a "constraint" is a central one in our model of surrogate reasoning. Intuitively, a constraint (in the sense used here) is a constraint on the way different states of affairs can hold in a situation. A constraint may hold on a set of situations either logically or more locally. Let us spell out the idea.

Let $\mathcal{D} = \langle D, \Delta \rangle$ be any domain of classification, either of targets situations, surrogate situations, or actions. A *constraint* is a pair of any subsets Δ_1, Δ_2 of Δ, and we use the notation $\Delta_1 \vdash \Delta_2$ to denote the constraint. A constraint may or may not hold on the domain $\mathcal{D}$. We say that the constraint $\Delta_1 \vdash \Delta_2$ *logically holds on* $\mathcal{D}$ iff for every situation d in D, if every infon in Δ_1 holds in d, then at least one infon in Δ_2 holds in d. Thus, a logical constraint is a constraint that governs every situation in the domain *without exception.*

Even if a constraint is not a logical constraint, and thus allows certain exceptions, it might be something that governs "normal" situations, something fairly regular and reliable. The notion of "local" constraint is to capture this kind of constraints. Let D_ℓ be the set of normal situations in D. (So, D_ℓ is a subset of D.) We say that a constraint $\Delta_1 \vdash \Delta_2$ *locally holds on* the domain $\mathcal{D}$ iff for every situation d in D_ℓ, if every infon in Δ_1 holds in d, then at least one infon in Δ_2 holds in d. Notice that every logical constraint is a local constraint. Thus, we can conceive the set of local constraints as a system of a stronger, extra-logical laws that govern the domain. We will use "T_ℓ," "S_ℓ," and "A_ℓ" to denote the sets of normal situations in the target domain $\mathcal{T}$, the surrogate domain $\mathcal{S}$, and the domain of actions $\mathcal{A}$ respectively.

Examples of local constraints Let $\mathcal{M}$ and $\Rightarrow$ be the method and the semantic convention adopted in an instance of surrogate reasoning. We assume that there is a range of "normal" target situations T_ℓ that $\mathcal{M}$ and $\Rightarrow$ are designed to cover, and there is a system of "target constraints" that reasoning with $\mathcal{M}$ and $\Rightarrow$ is supposed to capture. This system of target constraints often consists of special, extra-logical constraints, such as geometrical laws, physical laws, and social conventions, depending upon the application $\mathcal{M}$ and $\Rightarrow$ are designed for. These are the examples of *local constraints* holding on the target domain.[4]

Let us turn to the cases of surrogate domains and domains of actions. Given an operation method $\mathcal{M}$ and a semantic convention $\Rightarrow$, there is a range A_ℓ of "normal" operations that $\mathcal{M}$ and $\Rightarrow$ are designed for, and a system of local operational constraints that they are designed to work under. Operations outside A_ℓ are logically possible, but "abnormal" operations, which constitute exceptions to the given operation method $\mathcal{M}$. Furthermore, there is a range of normal surrogate situations S_ℓ that $\mathcal{M}$ and $\Rightarrow$ are designed for, and a system of local constraints on surrogate situations that they are designed to work under.

Imagine that you are trying to solve a logic problem with Venn diagrams. You draw three partially overlapping circles a, b, and c. You then shade the

[4]Of course, there are many pairs of methods and semantic conventions that are designed to capture only the logical constraints governing target domains—first-order calculus, Venn diagrams, and semantic tableau are all associated with such pairs.

complement of c with respect to b, and shade the intersection of b and a.

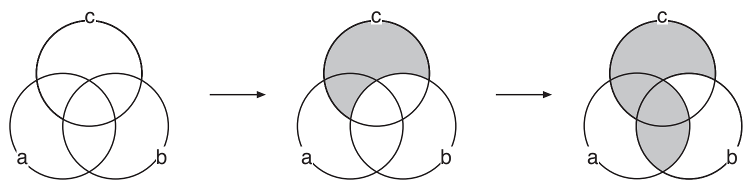

Figure 2

Having done all this, you will probably expect that the intersection of the c and a is shaded. In fact, this is an assumption made by Venn himself—the method and the semantic convention associated with Venn diagrams are designed to work under the assumption that this constraint holds.

This is not a logical constraint since there are some exceptional circumstances in which it does not hold. Think of drawing the diagram with the toy pen, whose "magic" ink fades away a few seconds after it is put on paper, or imagine drawing the diagram into a computer which automatically distorts, moves, and sometimes erase what you draw. In these circumstances, even if you execute a sequence of operations described above, there is no guarantee that the intersection of the circle a and the circle c gets shaded. Still, the operational constraint that you and Venn have assumed is fairly reliable—reliable enough for the normal practices of logic students to depend on it.

Even if we bar the above tricky circumstances, and assume that everything that we draw is preserved until the end of the derivation, the assumed constraint is not a logical necessity. The constraint can be used in our reasoning only because the following *local* constraint holds on the surrogate domain:

{Three circles, a, b, c, partially overlap,
The complement of c with respect to b is shaded,
The intersection of b and a is shaded}
$\vdash$ {The intersection of b and a is shaded}

This is certainly not a logical necessity—it is an extra-logical constraint rooted in the geometrical and physical properties of the circles being drawn.

Throughout this paper, we use the phrase "a default constraint on surrogates" to mean a local constraints on either the domain $\mathcal{A}$ of actions or on the domain $\mathcal{S}$ of surrogates. When we discuss the possible ways in which the default constraints intervene in the process of reasoning, we will see that some of the cases are rooted in the local constraints on the surrogate domain but that some rely on constraints on actions.[5]

[5] It is interesting to observe that a local constraint on a surrogate domain $\mathcal{S}$ can be reflected in a local constraint on a domain of actions that plays on $\mathcal{S}$. For example, the geometrical constraint described above is reflected in the operational constraint on Venn diagrams described earlier.

Reasoning systems Given an instance of surrogate system, there are four elements associated with it: a target domain $\mathcal{T}$, a surrogate domain $\mathcal{S}$, a domain of actions $\mathcal{A}$, a method $\mathcal{M}$, and a semantic convention $\Rightarrow$ associated with it. Furthermore, as we have just seen, a method $\mathcal{M}$ and and a semantic convention $\Rightarrow$ are always designed on the assumption that there are a set T_ℓ of normal target situations to capture, a set S_ℓ of normal surrogate situations to work on, and a set A_ℓ of normal actions to work with. We capture this association of different elements with the notion of "reasoning system." Thus, a *reasoning system* is a quintuple $\mathcal{R} = \langle \mathcal{T}, \mathcal{S}, \Rightarrow, \mathcal{A}, \mathcal{M} \rangle$ consisting of:

- a target domain $\mathcal{T} = \langle T, T_\ell, \Theta \rangle$,
- a surrogate domain $\mathcal{S} = \langle S, S_\ell, \Sigma \rangle$,
- a semantic convention $\Rightarrow$ defined on $\Sigma \times \Theta$,
- a domain of actions $\mathcal{A} = \langle A, A_\ell, \Omega \rangle$ that plays on $\mathcal{S}$, and
- a method $\mathcal{M}$ defined on $\Theta \times \Omega$.

Note that the domains $\mathcal{T}$, $\mathcal{S}$, and $\mathcal{A}$ now have their "normal" subsets T_ℓ, S_ℓ, and A_ℓ incorporated in them. These subsets serve to characterize the sets of local constraints holding on their relevant domains. We will deal with instances of surrogate reasoning that can be modeled by a reasoning system in this sense.

Free rides and overdetermined alternatives

Our claim is that most of the advantages and disadvantages of particular forms of surrogate reasoning can be explained with reference to the ways the default constraints on the surrogate objects intervene in the process of problem solving. But how exactly do the default constraints on surrogate objects intervene in a processes of problem solving? In this section, we will illustrate two typical patterns of constraint-intervention, which we dub "free rides" and "overdetermined alternatives."[6] Let us start with "free rides."

Compare the following three stages of a problem solving scenario:

Scenario 1, Stage 1 A ninety-three year old man is asked to describe the geographical features of the town in which he grew up, as accurately as possible, by memory.[7] The problem is about the locations of various geographical features of his home town in a particular period, and about their spatial relationships. He

[6]Shimojima (in press-a) proposes formal definitions of the phenomena of free rides and overdetermined alternatives in diagrammatic reasoning. Our own treatment of those phenomena will remain informal, but will cover a wider range of surrogate reasoning.

[7]Hohauser (1982) reports that a Polish man named Harry Lieberman was actually engaged in this type of activity. The book does not explain why his home village was so important.

first tries to solve the problem by pure thinking. Starting with his fragmented memories about the town and relying on the principles of geometry, he performs mental deductions of further information. Through these deductions, he tries to obtain a largest possible, consistent body of information about his home town. (The result of a deduction may or may not be consistent with the rest of his memory. If it is not consistent, he has to revise at least a part of his memory. So, the process is not monotonic.) This process is not a case of surrogate reasoning, if we preclude the possibility of "mental surrogates" (such as mentalese and mental models).

Stage 2 The man gives up the mental deduction method, and starts writing down the fragments of his memory one by one in the form of sentences. He then writes down what he can deduce from these sentences on the basis of the principles of geometry. This way, he hopes, he will eventually have a large collection of sentences that describe the geographical features of his home town. (Again, the process may not be monotonic.) This is a case of surrogate reasoning, whose source objects are the sentences he writes down on the piece of paper.

Stage 3 He gets bored or frustrated writing down sentences. Instead of representing the fragments of his memory in the form of sentences, he now tries to draw a approximate map of his home town. On the basis of fragments of his memory, he draws lines and curves on a sheet of paper to represent the streets, pathways, rivers, and such. He then uses wood blocks to represent the buildings that he remembers to have existed, and places them on his map, to represent the approximate locations of those buildings. He keeps revising and supplementing the map, and eventually obtains a map that represents his home town to the best of his memory. The source objects in this case are the drawings and the wood blocks on the piece of paper.

What we call "free rides" occur in Stage 3. Suppose the man remembers the locations of a river, two roads, and several houses, and constructs the following tentative map:

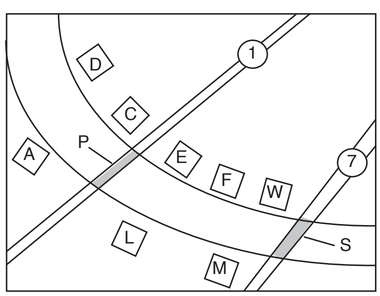

Figure 3

He thinks that the house K was between the houses L and M. On this assumption, he puts a wood block that represents the house K between the wood blocks representing L and M on the map:

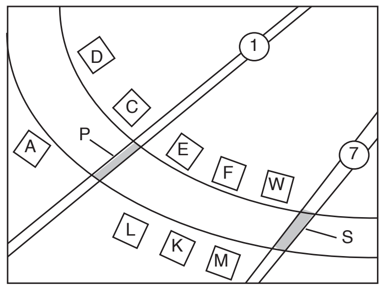

Figure 4

Notice that, in virtue of geometrical constraints that govern the map, the wood block K is now in various spatial relationships with other objects on the map: it is across the block F over the river line, it is closer to road line 1 than the block M is, while M is closer to the bridge symbol S than K is, K and A has a road line 1 in between, and so on. The map obtains these properties as consequences of the man's operation of placing the block K between the blocks L and M. And of course, these properties encode various pieces of information on the semantic convention that he adopts—that the house K was across the house F over the river, that it was closer to road 1 than the house M was, that the house M was closer to the bridge S than the house K was, the house K and the house A had the road 1 in between, and so on. Yet, he does not have to apply additional operations to obtain these pieces of information. He obtains them "for free," thanks to the geometrical constraints governing the map. To get the sense of utility of the free rides in this case, imagine how many deduction steps would be needed if he tried to obtain the same results with pure thought or sentences on the basis of the principles of geometry.

Free rides occur with simpler, purely diagrammatic surrogates. Consider the following scenarios:

Scenario 2 A logic students uses Venn diagrams to check the validity of the following syllogisms: All Cs are Bs. No Bs are As, Therefore no Cs are As. He draws three (partially overlapping) circles on a sheet of paper, and labels them "As," "Bs," and "Cs." He represents the premises of the syllogism by shading the complement of the B-circle with respect to the C-circle (figure 5) and then shading the intersection of the B-circle and the A-circle (figure 6). He observes that the intersection of the C-circle and the A-circle is shaded as a result. He concludes that the syllogism is valid.

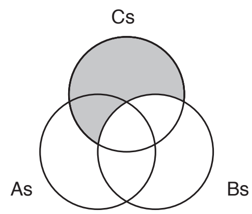

Figure 5

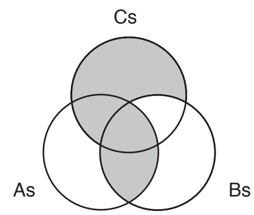

Figure 6

Scenario 3 Another logic student uses Euler circles to solve the same problem. She represents the premises by drawing a circle labeled "Cs" inside a circle labeled "Bs" (figure 7), and drawing a circle labeled "As" completely outside the B-circle (figure 8). She observes that the C-circle and the A-circle do not overlap as a result. She concludes that the syllogism is valid.

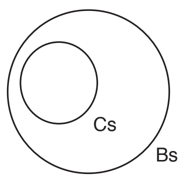

Figure 7

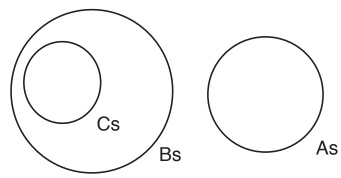

Figure 8

In each of these scenarios, the reasoner gets a free ride from the premises to the conclusion of the syllogism—the student applies certain types of operation to represent the premises in a diagram, and in virtue of the geometrical constraints, the diagram comes to bear the property that encodes the conclusion. The relevant constraints are different in the two scenarios—in scenario 2, the relevant constraint is one that governs the shadings of different areas of overlapping circles, while in scenario 3, the relevant constraint is one that governs the enclosure-disclosure relations among circles of different sizes. However, these different constraints spare the logic students the same deduction steps—they would have to go through two applications of modus ponens and a universal generalization if they used standard first-order calculus.

We can describe these processes of free ride in general terms as follows:

1. To represent a set Θ_1 of assumptions, a reasoner applies a sequence of operations $a_1, \ldots, a_n$ that realizes a set Σ_1 of infons in a surrogate situation s_O.

2. In virtue of a local constraint $\Sigma_1 \vdash \{\sigma\}$ holding on the surrogate domain, the output situation s_O is constrained to support an extra infon σ.

3. On the semantic convention $\Rightarrow$ that the reasoner adopts, the surrogate infon σ encodes a target infon θ. (The reasoner thus gets a free ride from the assumptions Θ_1 to a piece of information θ.)

Note that the constraint $\Sigma \vdash \{\sigma\}$ governing the surrogate domain plays a crucial role in this process of free ride. To wit, suppose there were no *non-trivial* geometrical constraints that govern the configurations of the wood blocks and the lines on the map in scenario 1. Then putting the K block between the L block and the M block would only result in the K block between the L and the M block, and nothing else. No additional facts would hold on the map, and hence no additional information could be read off from the map. Also, if the blocks and the lines on the map should obey no *regular* constraints so that they display different configurations every time the man applies the same operation, then he could not use the map as a reliable aid for problem solving.

This observation leads us to consider the possibility of *incorrect* free rides. The information the reasoner obtains "for free" at stage 3 may or may not be a consequence of the assumptions that he started with. We call a free ride *correct* if there is a (trivial or non-trivial) constraint on the target domain relative to which the obtained "free" information is a consequence of the assumptions. A free ride is *incorrect* if there is no such constraint on the target domain. Happily, all the instances of free ride we have seen so far are all correct. The large amount of information that the man reads off after putting the K block between the L and the M block is a consequence of the assumptions that he has made in constructing the map thus far. There are geometrical constraints that govern the real houses, rivers, and roads that make the former a consequence of the latter. The information the logic student reads off from the final Venn diagram is a consequence of the assumptions that he has made in drawing the diagram. There is a constraint governing the sets As, Bs, and Cs that makes the former a consequence of the latter. Similarly for the case of Euler diagrams.

What is an example of incorrect free rides? Consider the following scenario:

Scenario 4 A dummy used in a car accident test has poor "biofidelty"—its head, neck, throat, and knees are unfaithful to human counterparts in relevant respects such as flex, extension, and fragility . Not knowing this defect, testers assume that a certain type of car accident happens, and let a model accident happen to this dummy in a test car. In virtue of the constraints that govern the dummy and the test car, the dummies respond to the impact in a particular way, which the testers interpret to obtain information about the consequence of the assumed accident to a human body.

There is a sense in which the testers obtain this information by a free ride: they apply certain operations to the dummy on the assumption that a certain type of car accident happens. In virtue of a default constraint governing the dummy, it enters into a surrogate situation that represents a piece of information about the real accident. However, due to the poor biofidelty of the dummy, there might not be any constraint governing real human bodies that justifies this consequence. Thus, the free ride they get may well be incorrect. A similar case of incorrect free rides is possible in any process of simulation—what is generally called "validity" of a simulation model is a special case of "correctness" of free rides in surrogate reasoning.

Let us now turn to the notion of overdetermined alternatives. Consider the following continuation of stage 3 of scenario 1.

Scenario 1, Stage 4 After putting the K block between the L block and the M block, the man remembers that the house B was somewhere between the house A and the house K. Before forgetting it again, he wants to represent this information in his map by placing the B block between the A block and the K block. However, this requires him to put the B block either between the A block and the road line 1, or between the road line 1 and the L block, or between the L block and the K block. He cannot decide which alternative to take, since each alternative has its own semantic content that he does not want to presume. Thus, he has to give up recording a precious fragment of his memory in his map, until he obtains further information that allows him to take a particular alternative.

In general terms, we can describe the phenomenon of overdetermined alternatives in the following way:

1. To represent a set Θ_1 of assumptions, a reasoner applies a sequence operations $a_1, \ldots, a_n$ that realizes a set Σ_1 of infons in a surrogate situation s_O.

2. In virtue of a constraint $\Sigma_1 \vdash \Sigma_2$ holding on the surrogate domain, the output situation s_O is constrained to support at least one of the alternative infons Σ_2.

3. On the semantic convention $\Rightarrow$ that the reasoner adopts, each of these alternative infons encodes a target infon θ that does not follow from the initial assumptions Θ_1 (that is, the constraint $\Theta_1 \vdash \{\theta\}$ does not hold on the target domain).

Note that the constraint $\Sigma_1 \vdash \Sigma_2$ governing the surrogate domain plays a crucial role in this phenomenon of overdetermined alternatives.

The trouble with an overdetermined alternative is that it prevents the reasoner from applying a certain type of operation without thereby choosing at least one additional state of affairs to realize in the surrogate object. Since this additional state of affairs encodes a piece of information that does not follow from the reasoner's initial assumptions, this amounts to a strengthening of his original assumptions. This leads to the inflexibility of the surrogate reasoning in question—by means of the default constraints that govern the surrogate domain, the reasoner is prevented from representing a desired combination of assumptions into his source object. He is forced to represent additional assumptions in it, or give up representing the desired information in the first place. In a worst scenario, the reasoner mistakes a consequence of the strengthened assumptions for a consequence of his original assumptions. This is a case that is generally called "an appeal to an accidental feature."

Constraint projection

Our thesis in this paper is that most advantages and disadvantages of surrogate reasoning can be explained with reference to the ways in which the default constraints on surrogates intervene in the process of problem solving. To illustrate this claim, we have characterized two special cases of constraint-intervention, free rides and overdetermined alternatives. Although these represent important cases of constraint-intervention, they by no means exhaust all the possible cases. It is time to indicate, in more general terms, how default constraints on surrogate objects can possibly intervene in the process of problem solving. This will also let us state our thesis more clearly.

Let $\mathcal{R} = \langle \mathcal{T}, \mathcal{S}, \Rightarrow, \mathcal{A}, \mathcal{M} \rangle$ be a representation system, and let $\Theta_1 \vdash \Theta_2$ be a constraint that may or may not hold on the target domain $\mathcal{T}$. We can distinguish at least three distinct ways in which $\Theta_1 \vdash \Theta_2$ might be "projected" onto our target domain by the system.

Projection by the medium We say that the *medium of* $\mathcal{R}$ *projects* a constraint $\Theta_1 \vdash \Theta_2$ if for every surrogate situation $s \in S_\ell$, if $s \models\!\Rightarrow \Theta_1$ then $s \models\!\Rightarrow \theta$ for some $\theta \in \Theta_2$. In other words, if we work in the reasoning system $\mathcal{R}$, we cannot represent Θ_1 in our surrogate situation without thereby representing some $\theta \in \Theta_2$; the local constraints on the surrogate domain $\mathcal{S}$ prevent us doing so.

Projection by the actions We say that *the actions of* $\mathcal{R}$ *project* $\Theta_1 \vdash \Theta_2$ if for every sequence of actions $a_1, \ldots, a_n$, each of which is in A_ℓ, if s_O is the output situation of this sequence of actions and $s_O \models\!\Rightarrow \Theta_1$, then $s_O \models\!\Rightarrow \theta$ for some $\theta \in \Theta_2$. Thus, we cannot represent Θ_1 in our surrogate situation without thereby representing some $\theta \in \Theta_2$, because of the restrictions upon the actions that we can take. It is possible for the actions of $\mathcal{R}$ to project a constraint without the medium of $\mathcal{R}$ doing so. Imagine that, as a matter of physical restrictions on human bodies, all the actions we can take to represent Θ_1 also represent a member of Θ_2. The constraints on the surrogate domain is clearly not responsible.

Projection by the method We say that the *method* $\mathcal{M}$ *projects* $\Theta_1 \vdash \Theta_2$ if for every sequence of actions allowed by $\mathcal{M}$, given the assumptions Θ_1, if s_O is the result of this sequence of actions, then $s_O \models\!\Rightarrow \theta$ for some $\theta \in \Theta_2$. Notice that it is not assumed that $s_O \models\!\Rightarrow \Theta_1$, namely, the assumptions Θ_1 may not be represented in the surrogate situation as a result of the actions in question. All that is required is that we apply certain operations in accord with the method $\mathcal{M}$, and then our surrogate enters into a situation that represents an infon in Θ_2. This notion of constraint projection covers a wide range of cases such as the use of a calculator, a dummy driver in a car test, a computer, and an expert.

When we employ a reasoning system $\mathcal{R}$ that projects a constraint $\Theta_1 \vdash \Theta_2$ in any of these ways, our inference from the assumptions Θ_1 to a member of Θ_2 is partially taken over by a local constraint holding on our surrogate domain or domain of actions. In the case of constraint projection by the medium, we can simply "read off" a member of Θ_2 (via the semantic convention $\Rightarrow$) as soon as we represent the information Θ_1 in our surrogate situation—we do not have to calculate out the implications of Θ with respect to some relevant system of knowledge. We enjoy (or suffer from) the same transfer of inferential burden in the case of constraint projection by the actions, although it is rooted in the local constraints on the domain of actions. In the case of constraint projection by the method, we must figure out what types of operations $\mathcal{M}$ allows us to apply on the assumptions Θ_1. Still, once we have applied one of the permitted sequences of operations, the output situation lets us read off one of the infons Θ_2.

We believe that every practice of surrogate reasoning involves a constraint projection in one of the above varieties, and a part of its inferential burden is taken over by a constraint that governs the surrogates or the actions used in the reasoning. Based upon this observation, our constraint hypothesis asserts that the advantages and disadvantages of a practice of surrogate reasoning largely depend on the fact that the reasoning systems allow the projection of constraints in these various ways. In particular, using the notion of constraint projection, we can explicate our earlier examples of free ride and overdetermined alternatives.

A *free ride* occurs when the reasoning system $\mathcal{R}$ projects a constraint of the form $\Theta_1 \vdash \{\theta\}$, namely, a constraint whose consequent is a singleton. One could consider various possibilities here, depending on how the constraint is projected, by the medium, by the actions, or by the method of the system. The free ride $\Theta_1 \vdash \{\theta\}$ is correct if $\Theta_1 \vdash \{\theta\}$ actually holds of the target domain. The use of a map of the home town (scenario 1, stage 3), of Venn diagrams (scenario 2), and of Euler diagrams (scenario 3) are all cases in which the medium of the adopted reasoning systems project various constraints upon their target domains. The projected constraints all actually hold on the target domains in the reasoning systems, and the free rides are correct. The use of a dummy in the car test (scenario 4) is a case in which a constraint is projected by the method of the underlying reasoning system. The testers operate on the dummy according to a certain method, and these operations make the dummy enter a situation that represents something about the real situation of an accident. Unfortunately, the projected constraint does not hold on the target domain of real accidents due to the poor biofidelty of the dummy. The free ride that the testers get is incorrect.

Overdetermined alternatives are imposed when a reasoning system projects a constraint $\Theta_1 \vdash \Theta_2$ while there is no $\theta \in \Theta_2$ such that $\Theta_1 \vdash \theta$ actually holds. Notice that this can happen even if the projected constraint actually holds. The difficulty is that the output situation s_O of the reasoner's action inevitably represents an infon θ that does not follow from the assumptions Θ_1. Again, one can consider various possibilities here, depending on how the constraint is projected, by the medium, by the actions, or by the method of the system. The example we have seen (scenario 1, stage 4) is a case in which the medium of the reasoning system projects a constraint $\Theta_1 \vdash \Theta_2$. Although the projected constraint holds on the target domain $\mathcal{T}$ of geographical situations of the actual village, there is no $\theta \in \Theta_2$ such that $\Theta_1 \vdash \theta$ actually holds on $\mathcal{T}$. So, as long as the man tries to represent the assumptions Θ_1 in the map, the map will represent an extra piece of information $\theta \in \Theta_2$ which does not follow from Θ_1.

Thus, the cases of free rides and overdetermined alternatives are instances of constraint projection allowed by a reasoning system. And indeed, these two varieties of constraint projection explain the reliability and efficiency of many systems of surrogate reasoning. The degree of reliability of a system depends on the extent to which the free rides provided by the system are correct, that is, the extent to which the constraints $\Theta_1 \vdash \{\theta\}$ projected by the system in fact hold of the target domain. The more that hold, the more reliable it is. The degree of efficiency of a system depends on the set of free rides provided: if system $\mathcal{R}_2$ projects all the constraints $\Theta_1 \vdash \{\theta\}$ projected by $\mathcal{R}_1$, but not vice versa, then, other things being equal, $\mathcal{R}_2$ will be more efficient than $\mathcal{R}_1$. There is, of course, a tension between reliability and efficiency. The more efficient, the more projected constraints, and so the greater possibility for the projected constraints that do not actually hold of the target domain.

To add to this tension, the degree of reliability of a system also depends on the cases of overdetermined alternatives imposed by the system, that is, the set of projected constraints $\Theta_1 \vdash \Theta_2$ such that no member of Θ_2 follows from Θ_1. If system $\mathcal{R}_2$ projects all the "overdetermined" constraints $\Theta_1 \vdash \Theta_2$ projected by $\mathcal{R}_1$, but not vice versa, then, other things being equal, $\mathcal{R}_2$ will be less reliable than $\mathcal{R}_1$.

Although the cases of free rides and overdetermined alternatives are important varieties of constraint projection, they by no means exhaust all. Depending on what we stipulate about the projected constraint and its relations to the semantic convention, the method, the target domain, the surrogate domain, and the domain of actions, we might be able to define other varieties of constraint projection that provide illuminating criteria for the advantage and disadvantage of particular reasoning systems.[8] Thus, we are far from claiming that we have provided a comprehensive set of such criteria. Instead, our constraint hypothesis asserts more broadly that we can explain most advantages and disadvantages of

[8] For example, Shimojima (in press-b) contrasts the cases of free rides to what he calls the cases of "half rides." Although his paper does not make it explicit, the cases of half ride do constitute a variety of constraint projection.

particular systems of surrogate reasoning *in this direction*, by detailed studies of different patterns of the constraint projection they offer.

Eventually, we wish to define the advantages and disadvantages of a reasoning system with respect to a specific task of problem solving. In other words, we want the notion of *suitedness* of a reasoning system to particular tasks of problem solving. In our view, this requires us to first define what constitutes a *valid solution* of a problem, and what it is for a solution of a problem to *rely on* a constraint $\Theta_1 \vdash \Theta_2$ on the target domain. Unfortunately, it is not possible for us to fully address these issues in this paper, and we must save our ambition for a later opportunity.

Conclusions

In this paper we have suggested that the sort of analysis that has gone into the study of formal logic can be extended from proofs using sentences in a formal language to an incomparably richer and more important domain: that of people using a variety of tools to reason about a variety of problem domains. As a first step in this project, we have proposed the following thesis: the strength and weakness of particular forms of surrogate reasoning are largely rooted in the different ways in which default constraints governing surrogate objects intervene in the process of problem solving. To substantiate this hypothesis, we have illustrated two special cases of constraint intervention, free rides and overdetermined alternatives, and then introduced the notion of "constraint projection," which captures the general framework in which various forms of constraint intervention take place in surrogate reasoning.

Our purpose here will be served if readers begin to explore some of the topics that emerge when one takes seriously the empirically obvious fact that most human reasoning of any real difficulty relies crucially on the use of tools external to the target domain. We believe that the study initiated here will eventually illuminate various cognitive models that seem to liken thinking processes to the uses of external tools such as sentences, models, diagrams, pictures, and machines. As long as the developers of those models draw any of their insights from the actual uses of these tools, an explicit study of surrogate reasoning has the potential to clarify, disentangle, and enrich the contents of the models being developed.

References

Barwise, J. (1989). *The Situation in Logic.* Stanford, CA: CSLI Publications.

Barwise, J. (1991). Information Links in Domain Theory. In S. Brooks, M. Main, A. Melton, M. Mislove, & D. Schmidt (Eds.), *Proceedings of the Mathe-*

matical Foundations of Programming Semantics Conference (1991), (pp. 168–192). Berlin, Germany: Springer Verlag.

Barwise, J., & Etchemendy, J. (in press). Logic, Proof, and Reasoning. In A. R. Malachowski (Ed.), *A Companion to Logic*. London: Blackwell Publishers.

Devlin, K. (1991). *Logic and Information*. Cambridge, England: Cambridge University Press.

Hohauser, S. (1982). *Architectural and Interior Models* (Second ed.). New York: Van Nostrand Reinhold Company.

Shimojima, A. (in press-a). Operational constraints in diagrammatic reasoning. In J. Barwise & G. Allwein (Eds.), *Diagrammatic Reasoning*. Oxford, England: Oxford University Press.

Shimojima, A. (in press-b). Reasoning with Diagrams and Geometrical Constraints. In D. Westeståhl & J. Seligman (Eds.), *Language, Logic and Computation: The 1994 Moraga Proceedings*. Stanford, CA: CSLI Publications.