Psychological Review
2006, Vol. 113, No. 4, 887–916

0033-295X/06/$12.00 DOI: 10.1037/0033-295X.113.4.887

Hierarchical Schemas and Goals in the Control of Sequential Behavior

Richard P. Cooper
Birkbeck, University of London and University College London

Tim Shallice
University College London and Scuola Internazionale Superiore di Studi Avanzati

Traditional accounts of sequential behavior assume that schemas and goals play a causal role in the control of behavior. In contrast, M. Botvinick and D. C. Plaut (2004) argued that, at least in routine behavior, schemas and goals are epiphenomenal. The authors evaluate the Botvinick and Plaut account by contrasting the simple recurrent network model of Botvinick and Plaut with their own more traditional hierarchically structured interactive activation model (R. P. Cooper & T. Shallice, 2000). The authors present a range of arguments and additional simulations that demonstrate theoretical and empirical difficulties for both Botvinick and Plaut's model and their theoretical position. The authors conclude that explicit hierarchically organized and causally efficacious schema and goal representations are required to provide an adequate account of the flexibility of sequential behavior.

Keywords: control of routine behavior, localist versus distributed representations, simple recurrent networks, neuropsychological impairments of action

It has become a commonplace in many areas of psychology over the past 50 years that there exist discrete representations that correspond to qualitatively different states of the organism. The accessing, activation, or selection in some other way of one of these states rather than another is held to have qualitatively different effects on the selection of subsequent states and of subsequent behavior. Moreover, the selection of the current state is held to be the result of the effecting of discrete operations or rules, typically by analogy with a computer program.

More recently, there has been a challenge to this perspective. It has been strongly argued that this assumed discreteness both of the representations and of the structures that select them arises from the familiarity of such concepts in other domains (e.g., computer science) rather than reflecting the operation of the underlying mechanism in the human mind. Instead, the apparent discreteness reflects inputs or outputs rather than the states of the internal mechanisms, which are better represented as regions or trajectories within continuous state spaces created by connectionist networks.

The core issues in the debate, which has raged since 1985 (e.g., Broadbent, 1985; McClelland & Rumelhart, 1985; Pinker & Prince, 1988; Rumelhart & McClelland, 1985; etc.), have principally concerned areas in psycholinguistics and neurolinguistics where the existence of discrete representations (e.g., phonemes, morphemes) and of discrete operators (syntactic rules) was made plausible by developments in independent disciplines such as phonology and linguistics (e.g., Chomsky, 1957, 1980; Chomsky & Halle, 1968). In addition, the debate has concerned areas where rule-based mappings had already been postulated on other grounds, as in spelling-to-sound translation in reading (e.g., Coltheart, Curtis, Atkins, & Haller, 1993; Plaut, McClelland, Seidenberg, & Patterson, 1996; Wijk, 1966; see also Zorzi, Houghton, & Butterworth, 1998).

However, despite over 15 years of research in these areas, it would be premature to say that the structuralist view has been convincingly rejected in any one of them. Yet there are other areas of psychology where the assumption of discrete internal units is mainly driven by their apparent behavioral manifestations and not by any other well-organized discipline such as linguistics. In particular, concepts such as *schemas, scripts,* and *frames* with a complex intellectual history involving neurology, philosophy of mind, and artificial intelligence (AI) lack the independent support provided by other empirical disciplines that concepts like phonemes have. Although, in the initial development of connectionism (McClelland & Rumelhart, 1986; Rumelhart & McClelland, 1986), these less well-anchored concepts were widely viewed as being naturally explicable by the new approach, they often continue to be used as representing discrete units.

Consider the concept of schema. It has been involved in a variety of areas, such as memory (Bartlett, 1932), perception (Evans, 1967), and action (Schmidt, 1975). An early use as far as empirical science is concerned was that of Head (1920) in his analysis of disturbances of the somatosensory system. He used the

Richard P. Cooper, School of Psychology, Birkbeck, University of London, London, England, and Institute of Cognitive Neuroscience, University College London, London, England; Tim Shallice, Institute of Cognitive Neuroscience, University College London, London, England, and Scuola Internazionale Superiore di Studi Avanzati, Trieste, Italy.

We are grateful to Nicolas Ruh for many helpful discussions of the simple recurrent network (SRN) model, particularly in relation to the role of the training set in shaping the model's attractors and the role of the resulting context representations in the SRN model's implementation of choice. We are also grateful to Gordon Brown for his comments on the effect of rate of production on error profiles, to Matthew Botvinick for providing additional details of his implementation of the SRN model and for insightful comments on an earlier draft of this article, and to Padraic Monaghan for insightful comments on an earlier draft of this article.

Source code for the reimplementation of Botvinick and Plaut's (2004) model and for the five simulations reported in the Appendix is available from http://www.psyc.bbk.ac.uk/people/academic/cooper_r/models/.

Correspondence concerning this article should be addressed to Richard P. Cooper, School of Psychology, Birkbeck, University of London, Malet Street, London WC1E 7HX, United Kingdom. E-mail: R.Cooper@bbk.ac.uk

concept when referring to “organised models of ourselves” that “modify the impressions produced by incoming sensory impulses in such a way that the final sensations of position, or of locality, rise into consciousness charged with a relation to something that has happened before” (Head, 1920, pp. 607–608). Bartlett (1932) generalized the notion to apply to the units in which the memory of all past experience is held and not merely to the positions of parts of the body. In neither case was the concept more than very vague. However, Bartlett’s usage contained the idea that forgetting could lead to loss of detail in the schema while the core structure remained intact. In the mid-1970s, Bobrow and Norman (1975) made the idea somewhat more explicit by proposing that “each schema is a self-contained memory structure, capable of performing operations because it contains procedural definitions of its potential functions and operations” (p. 138). Moreover, the idea that the structure would contain argument slots and explicit default values was proposed by Rumelhart and Ortony (1977).

These last two ideas were then combined in the domain of action control by Norman and Shallice (1980, 1986) in an informal model. It was argued that the effecting of routine behavior involves producing behavior routines, controlled by schemas, with the constraints imposed by the specific environment being mediated through argument selection. It was also claimed that nonroutine behavior is controlled in a qualitatively different manner. On this approach, the key theoretical issues for the control of routine behavior are the organization and selection of schemas, whereas a key theoretical issue for the control of nonroutine behavior is the mechanism by which nonroutine behavior interfaces or interacts with routine behavior.

Norman and Shallice’s (1980, 1986) schema-based account of routine behavior was motivated largely by phenomenological and neuropsychological evidence, but the mechanism they hypothesized—contention scheduling—has considerable similarity to concepts in AI approaches to planning and engineering solutions to the control of robots (Gat, 1998; Glasspool, 2005; Maes, 1989; see also Shallice, 1988, pp. 350–352). A common approach within these domains is to delegate common behaviors to one system (the routine subsystem) that at any one time selects from a library of simple routines (the plan library). These behaviors relate to situations where the goals and context are familiar or an immediate response is required. The routine control subsystem is supplemented by a deliberative subsystem—a processing-intensive planning system that is invoked when no suitable behavior is available in the routine subsystem’s plan library.

One aspect of the concept of schema in the Norman and Shallice (1980, 1986) account is that every manifestation of a particular type of routine behavior depends upon the activation and selection of one particular internal unit, its schema node. Moreover, the hierarchical structure that is frequently manifested in routine actions is assumed to be controlled by the activation of a hierarchy of schema nodes. In this respect, the Norman and Shallice model merely echoes the ideas of many theorists (e.g., Fuster, 1989; Humphreys & Forde, 1998; Miller, Galanter, & Pribram, 1960).

A second concept used in the Norman and Shallice (1980, 1986) approach that naturally complements that of schema is that of goal or purpose, which may be defined as a state of affairs that an agent aims to achieve. Here, too, the concept derives from a simple mixture of phenomenology and functional biology; there is no separate discipline through which the subject’s goal or goals at any particular time can be specified. Goals, even more than schemas, have a long history within psychology (cf. Miller et al., 1960). They have become central to the control of behavior in production-system cognitive architectures such as Soar (Laird, Newell, & Rosenbloom, 1987) and ACT–R (Anderson et al., 2004), where their role is to effectively limit production rules that might be applied in a situation to the subset of productions relevant to the current goal. Within more mainstream cognitive psychology, goals serve a similar function. Thus, in Duncan’s (1993) account of attentional selection and behavioral control, they provide a means for selecting from all possible stimulus–response relationships just those relevant at the current point in time.

Given the preceding definition of a goal, a schema may be seen as a means of achieving a goal or subgoal. More generally, recent computational accounts of the contention scheduling system (Cooper, Schwartz, Yule, & Shallice, 2005; Cooper & Shallice, 2000; see also Cooper, Shallice, & Farringdon, 1995) take schemas to be goal-directed structures, with goals serving to mediate schema–subschema relationships. Thus, schemas achieve goals and, apart from at the lowest level of the schema hierarchy, consist of partially ordered sets of subgoals (which may themselves be achieved by other schemas). Again, there is a parallel with planning systems from the AI literature, where goals and methods (the AI equivalent of schemas) may be structured in an *and/or tree* (see, e.g., Charniak & McDermott, 1985), with multiple methods possible for any goal (the *or* component of the tree), but each method consisting of a conjunction of subgoals (the *and* component of the tree). These views see goals as playing a critical role in guiding behavior (without distinguishing between routine and nonroutine domains).

There is, however, another way of conceiving of a concept like schema. Within the context of early connectionism, Rumelhart, Smolensky, McClelland, and Hinton (1986) argued,

> There is no representational object which is a schema. Rather, schemata emerge the moment they are needed from the interaction of large numbers of much simpler elements all working in concert with one another. Schemata are not explicit entities, but rather are implicit in our knowledge and are created by the very environment that they are trying to interpret—as it is interpreting them. . . . In our case, nothing stored corresponds very closely to a schema. (Rumelhart et al., 1986, pp. 20–21)

From this tradition, Botvinick and Plaut (2004; see also Botvinick & Plaut, 2002) questioned the functional roles of both schemas and goals and the need for assuming hierarchical structures with their simple recurrent network (SRN) model of routine action. Specifically, they claimed that their model provides a good account of a range of empirical phenomena without recourse to either construct. Although Botvinick and Plaut (2004) claimed that they did not deny “the existence or psychological importance of explicit goal representations” (p. 424), they speculated that “much of cognition and behavior . . . may share a basic reliance on mechanisms of the sort illustrated in the present [SRN] model” (p. 424). What makes their claim particularly important is that the schema concept in Norman and Shallice (1980, 1986) is a hybrid with interactive activation aspects but also with symbolic rule-following ones. If Botvinick and Plaut are right, then the symbolic rule-following aspects of the models are an unnecessary postulate, and the potential power of connectionist models is clear. The current article, however, challenges their view by investigating the limits of their model. This involves reports, detailed in the Appendix, of new simulations conducted with a reimplementation of Botvinick and Plaut's SRN model to address critical issues. The investigation raises a series of principled difficulties—ones that are both theoretical and empirical in character and that challenge the eliminativist view.

Two Models of Routine Action Selection

In Botvinick and Plaut's (2004) critique of schema-based hierarchical models of action, they took as their key example the interactive activation model of routine action selection (henceforth referred to as the IAN model) proposed by us and our colleagues (Cooper & Shallice, 2000; Cooper et al., 2005; see also Cooper et al., 1995), which in turn is based on the informally specified contention scheduling part of the theory of Norman and Shallice (1980, 1986). The IAN model has both activation-based and symbolic aspects. The activation-based component consists of three interactive activation networks (see Figure 1). Nodes within the schema network represent goal-directed action schemas of varying levels of complexity (ranging from, e.g., *prepare instant coffee* at the highest level to *pick up implement* at the lowest level). Nodes within the object representation network correspond to ways of using objects present in the immediate environment (e.g., *fork as an implement* or *juice glass as a target*). Nodes within the effector network correspond to special-purpose cognitive subsystems that can be recruited to act upon the world (e.g., motor subsystems for each effector). All nodes have continuous valued activation levels that vary according to standard principles of interactive activation (McClelland, 1992) and with nodes functioning as leaky accumulators (Usher & McClelland, 2001). Nodes that correspond to schemas that are mutually exclusive (e.g., because they have overlapping requirements for special-purpose cognitive subsystems) have mutually inhibitory links, whereas nodes corresponding to object representations have excitatory links to schemas that are routinely performed with the corresponding objects and vice versa.

The IAN model also has symbolic aspects. Links between nodes in the networks explicitly reflect rulelike relationships between the elements represented by the nodes. For instance, within the schema network, links exist between superordinate schemas and lower level schemas that achieve the subgoals of the superordinate schema. Also, schemas have an argument structure, with the arguments being filled by the outputs of the object representation networks. In addition, activation flow between schema nodes is gated by preconditions and postconditions that relate to the achievement of goals.

Normal functioning of the IAN model begins with direct excitation of an intended schema. This excitation, which is assumed to typically originate from a separate deliberative subsystem, *the supervisory system,* causes the schema's activation to rise. When it exceeds the selection threshold (a parameter of the model), activation is passed from that schema to any schemas that may achieve the original schema's subgoals (subject to ordering constraints that are stated in the form of preconditions on subgoals). Normally, one subschema then becomes active and is selected. This process continues until a schema that corresponds to a simple action (e.g., *pick up*) is selected. The corresponding action is then performed, with the object to which the action is applied determined by the most active relevant item in the object representation network. On completion of an action, its corresponding schema is inhibited, allowing another schema to become activated and another action to be performed. Figure 2 illustrates the activations of schema nodes as time progresses while performing a typical routine task, that of preparing instant coffee. The figure shows how the activation of nodes accumulates over time and how the activation of lower level schema nodes occurs within the context of active higher level schema nodes.

We and our colleagues have shown how the processes of interactive activation implemented within the IAN model can result in

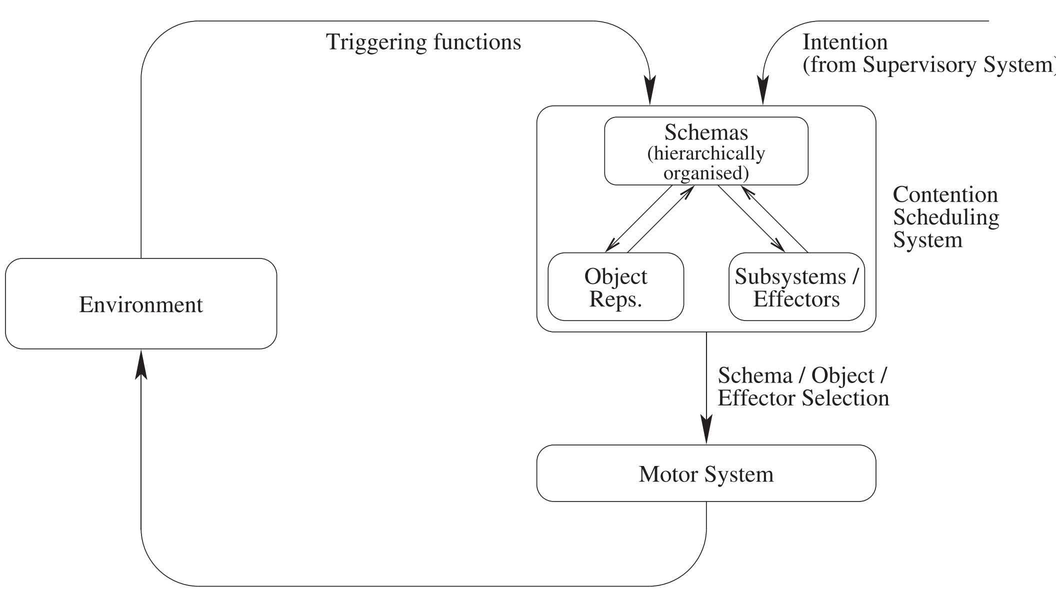

Figure 1. Functional components of the interactive activation model. Object Reps. = object representations.

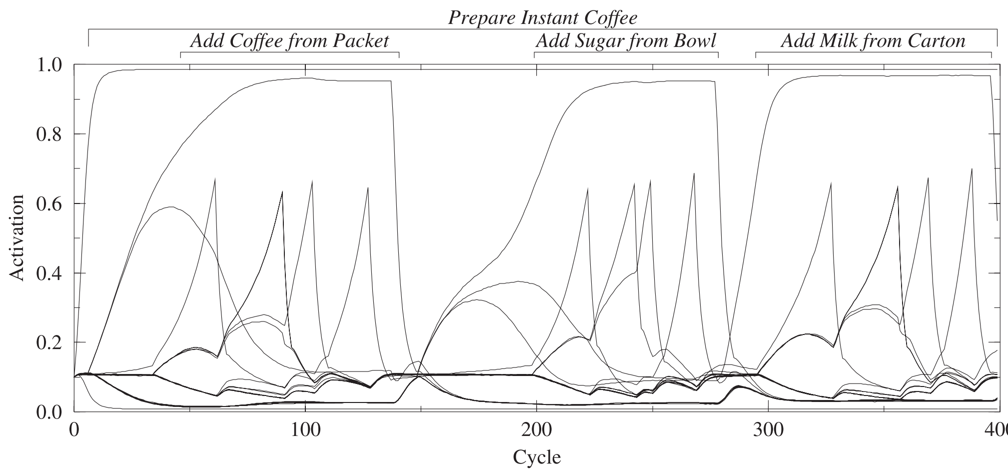

Figure 2. Activation profiles of schema nodes within the interactive activation model during the task of preparing instant coffee. From "Contention Scheduling and the Control of Routine Activities," by R. P. Cooper and T. Shallice, 2000, *Cognitive Neuropsychology, 17,* p. 319, Figure 5. Copyright 2000 by the Psychology Press. Reprinted with permission. See http://www.psypress.co.uk/journals.asp.

extended sequences of behavior, such as those involved in everyday behavioral routines, for example, preparing a mug of instant coffee (Cooper & Shallice, 2000) or preparing and packing a child's lunch box (Cooper et al., 2005). We have also shown that the IAN model can mimic the effects of neurological damage that impairs execution of routine or everyday action.

Botvinick and Plaut's (2004) approach operates on two levels. Verbally, for instance, they accepted that goals are useful for the cognitive system as a whole and assumed that learning skills is a two-stage process. However, their specific implementation, the SRN model, lacks both of these characteristics. Indeed, an attractive feature of their SRN model—which was developed largely in response to the IAN model—is that it demonstrates that a recurrent connectionist framework is capable of reproducing extended sequences of actions comparable in complexity to those achieved by the IAN model without explicit goal representations. Recurrent activation, the basic sequencing mechanism of the SRN model, uses activation propagating around a set of neuronlike units. Within the SRN model (see Figure 3), input units representing held or fixated objects are activated by features present in the representation of the environment. On each time cycle, activation is passed along weighted connections from these units to a set of hidden units, which also receive recirculated activation. Further weighted connections lead from the hidden units to a final set of output units, which encode possible actions, such as fixating on specific objects or picking up the fixated object. Conceptually, the recirculated activation of the hidden units provides a

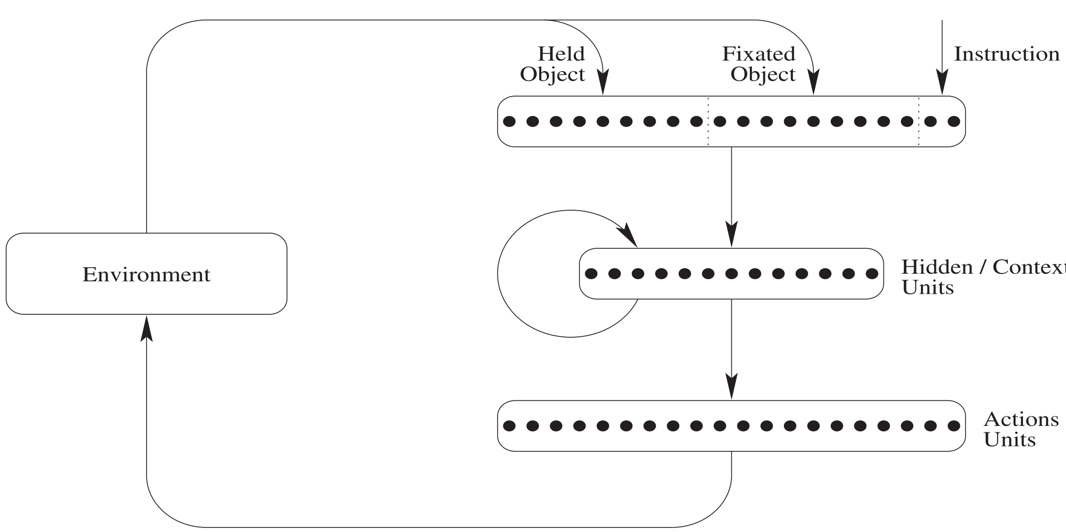

Figure 3. Functional components of the simple recurrent network model. Adapted from "Representing Task Context: Proposals Based on a Connectionist Model of Action," by M. Botvinick and D. C. Plaut, 2002, *Psychological Research, 66,* p. 300, Figure 1, with kind permission of Springer Science and Business Media. Copyright 2002 by Springer-Verlag.

running context. Input activation is incorporated into this context both to provide an output and to generate an updated context for use with the next input. Crucially, the weights of all the connections are acquired from a set of input–output exemplar sequences, and the hidden unit representations that develop from this learning process are distributed across the units and not open to transparent unit-by-unit interpretation. Thus, individual hidden units do not encode specific actions or the position of an action or subtask within a task.

Within the SRN model, one action is selected on each and every processing cycle. Thus, activation does not accumulate as in the IAN model. Instead, the flow of activation typically results in one output unit being highly active on each cycle while all other output units are inactive. Figure 4 illustrates this aspect of the model's behavior while the model performs a variant of the coffee-making task as described by Botvinick and Plaut (2004). On all but one processing cycle, a single output unit is activated. In the one case when two output units are activated (Step 11), the action corresponding to the most active output unit is performed, but this corresponds to a case when two actions are equally possible, and different attempts at the coffee-preparation task (with context units initialized to different random patterns) can lead to selection of either action.

Both we (Cooper & Shallice, 2000) and Botvinick and Plaut (2004) have used our respective models to account for the occasional slips and lapses that arise in routine action under conditions of distraction or fatigue and also for the disorganization of action that occurs in certain classes of neurological patient. Significantly, the SRN model contrasts with previous accounts of routine sequential action by doing this in the absence of any explicit representations of action schemas or goals. It thus instantiates a novel theory of the organization and control of routine sequential action. For this reason, it makes an important contribution to the field.

More generally, the contrast between the IAN model and the SRN model takes the general debate on the utility for cognitive science of unitized internal representations and internal structures linking them into a new critical area. Moreover, as the hierarchical structures that link schemas of different levels on the IAN model are not derived from any external discipline but are internal assumptions of the model, this makes the a priori plausibility of Botvinick and Plaut's (2004) approach the greater. It therefore enables one to examine a model of a complex domain, which we refer to as *eliminativist*, in the a priori most plausible situation. This article therefore considers the two models directly and in particular considers whether the eliminativist aspects of the SRN model are justified.

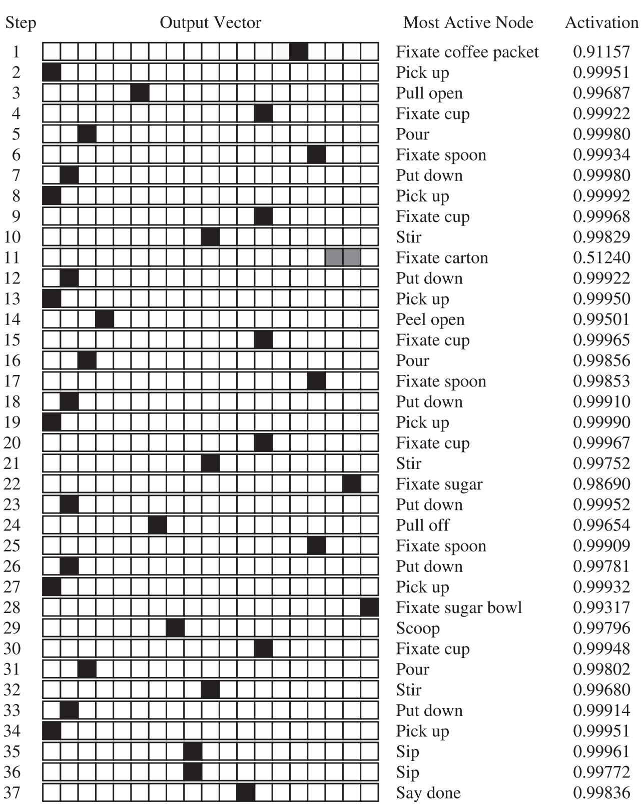

Figure 4. Activation of the simple recurrent network model's output units during the 37 steps of the coffee-making task. The shading indicates the level of activation, with white corresponding to zero and black corresponding to one.

Theoretical Differences and the Key Metatheoretical Choice

We see several theoretical differences between the models developed by Botvinick and Plaut (2004) and by us (Cooper & Shallice, 2000). Most fundamentally, the underlying computational processes—of recurrent activation with distributed representations versus interactive activation with localist representations—differ, and indeed, Botvinick and Plaut presented the representational difference as the critical one distinguishing between the two approaches. The relation between distributed and localist models of a processing domain can vary along an abstract dimension with, at the one end, the former models being more detailed implementations of the latter. In this situation, to each internal representation within the latter type of model there corresponds a clearly characterizable state of the former type. At the other end of the continuum, the internal states are not homomorphic in any simply characterizable way. Botvinick and Plaut exacerbated the conceptual difficulties their model faces by taking an extreme nonreductionist position with respect to the relation between the models. This, however, makes the contrast between models especially revealing.

We term Botvinick and Plaut's (2004) conceptual framework eliminativist because, at least within the domain of routine action, Botvinick and Plaut considered three theoretical constructs of classical action control to play no causal role. First, Botvinick and Plaut rejected the explicit representation of action schemas. As described above, we (Cooper & Shallice, 2000) have represented schemas as nodes within an interactive activation network, and those nodes play a causal role in the control and selection of routine action. Schemas are explicit in the sense that they are discrete and unitary. There is a one-to-one mapping between schemas and nodes in the schema network, and nodes in the schema network may be individually and directly activated or inhibited by other cognitive systems (specifically, by the hypothesized supervisory system during deliberate control of behavior). Botvinick and Plaut claimed instead that schema is just a descriptive term linked to the emergent regularities of the trajectories traced through the model's continuous state space. Second, our model uses a hierarchical network to structure schemas and subschemas, whereas Botvinick and Plaut claimed explicit hierarchical structure to be unnecessary for the control of routine behavior. Third, goals play an essential role in our model, whereas Botvinick and Plaut considered that much routine action is not under the control of explicit goals. These three differences place the SRN model and the IAN model in opposition with respect to the critical issues that motivate this article, as discussed in the introductory section, above.

A further difference between the models, one that is not related to Botvinick and Plaut's (2004) eliminativist position, relates to the representation of objects to which actions apply. We (Cooper & Shallice, 2000) have represented objects explicitly in a further set of interactive activation networks, with the targets of actions determined by the most active object representations, whereas Botvinick and Plaut employed a deictic scheme in which actions operate on attended objects, with attention being directed by a set of object-specific *attend* actions (such as *fixate cup* and *fixate spoon*). This approach eliminates the need for object representations to be used by the action control system.

Returning to the fundamental representational difference between the models, there has been much debate about the pros and cons of localist and distributed representations (cf. Page, 2000, and the commentaries following that target article). For example, localist representations have been argued to result in models with more perspicuous functioning (because the interpretation of the model's state is straightforward). Although the interpretation of the IAN model's state is more direct than that of the SRN model's state, Botvinick and Plaut (2004) provided extensive analyses of the hidden unit activations that largely address differences in ease of interpretation. Thus, the use of distributed representations is not, in our view, necessarily problematic. However, the many difficulties that a model in the domain of action selection needs to confront have been accentuated by Botvinick and Plaut's related metatheoretical choices, namely, their eliminativist interpretation of their model and the detailed mechanisms they proposed.

The present article is structured as follows. We first consider each of the above key theoretical issues in turn before briefly discussing three other issues—two theoretical and one empirical—that show the limitations of the Botvinick and Plaut (2004) approach clearly. For each issue that we consider, we also assess the criticisms made by Botvinick and Plaut of the IAN model. The empirical domain over which the contrasts between the two models are mainly made consists of coffee and tea making. We (Cooper & Shallice, 2000) have implemented the IAN model in terms of the subsequences involved in making coffee from the set of packets and containers typically found on the breakfast tray of a hospitalized neurological patient (this was an abstraction from the breakfast-tray task analyzed empirically by Schwartz, Reed, Montgomery, Palmer, & Mayer, 1991). This task involves adding coffee grounds, sugar, and cream to a mug of hot water. A total of 12 different types of basic action (e.g., *pick up, pour, tear*) structured in a three-level-deep hierarchy are used to realize a sequence of actions. In addition, objects are involved, both to be used and to act as distractions. Botvinick and Plaut took essentially the same basic task in their five simulations. However, in several of the simulations, they trained the model in addition on a second task, tea making. Significantly, this second task has some sequences—in particular, those concerning adding sugar—identical to subsequences of the coffee-making task.

The Role and Representation of Schemas

Within the IAN model, a schema is a complex entity. It consists of a goal, a triggering condition (i.e., a condition that specifies the degree to which states of the world excite the schema), an activation value, and a set of subgoals (with each subgoal having a precondition and a postcondition).[1] Schemas are explicit and play a causal role in determining behavior: Excitation and subsequent selection of a schema cause excitation and then selection of subschemas or actions. In contrast, the SRN model's behavior is determined by the activation of its input and hidden/context units, together with its training history (which shapes the connection weights to and from the hidden/context units). These connection weights encode a kind of sequential attractor—sequences of regions within the multidimensional space of hidden unit activations that the trained model tends to follow. Thus, it does this in the absence of explicit task instructions, as in Botvinick and Plaut's (2004) Simulation 1, where the model reproduced either coffee- or tea-preparation sequences even when no instruction unit was set, and in the presence of noise, as in their Simulation 2, in which the model tended to produce tea- or coffee-preparation sequences even in the presence of low levels of noise.

On the Importance of the Training Set

There is a strong sense in which one may equate the sequential attractors of the SRN model with schemas, although the attractors are implicit and emergent rather than explicit and prespecified. However, the method of learning employed by the SRN model—back-propagation through time with minimization of cross-entropy—means that the sequential attractors developed by the trained model are fully determined by the training set. The composition of the training set is therefore critical in determining the behavior of the model in both normal and impaired functioning.

SRNs that learn by back-propagation through time are essentially statistical devices that encode the conditional probability of an output given an input and the context established by the current task. Given the training sets employed by Botvinick and Plaut (2004) for the coffee-making task, the SRN model, for example, learns that the context and input established by adding cream leads with a probability of 1.0 to the action of adding sugar if sugar has not already been added but with a probability of 1.0 to drinking if it has. Crucially, it is easier for the network to learn temporal relations that operate over shorter times, so the effects of immediate prior context tend to be more pronounced or more robust than the effects of more distant prior context. Thus, in the above example, if the context is degraded or the task is not sufficiently well learned, the information concerning whether or not sugar has been added may be corrupted or inaccessible at the end of the routine for adding cream. This may result in sugar being added twice (once before and once after adding cream—a recurrent

[1] Subgoal preconditions encode ordering constraints and subgoal optionality, whereas subgoal postconditions enable monitoring. Basic-level schemas have no subgoals but instead interface directly with the motor system.

perseverative error) or failure to add sugar at all (an omission error).

This analysis demonstrates how the SRN model can simulate the occurrence of two important types of error that occur both in the lapses of normal subjects and in the errorful behavior of patients with frontal brain damage (cf. Schwartz et al., 1991, 1998). However, it also demonstrates that the ability of the model to generate either type of error is critically dependent on the training set. This produces two problems: one concerning the model's susceptibility to specific errors and the other concerning generalization from experience.

The first problem with the selection of the training set is that the model is especially prone to an error consisting of the omission of the subsequence B in the larger sequence A → B → C only if the training set also contains sequences including A → C. Thus, Botvinick and Plaut's (2004) training set for the coffee-making task involves four different action sequences. Critically, it includes sequences where the GROUNDS subtask is followed by SUGAR and then CREAM and others where it is followed immediately by the CREAM subtask. Without exposure to such sequences, the trained model would not be prone to omission of the SUGAR subtask. Similarly, the model is most prone to an error consisting of a recurrent perseveration of the subtask B in the larger sequence A → B → C (i.e., a delayed erroneous repetition of B as in A → B → C → B → C) if the training set also contains sequences including C → B. Again, Botvinick and Plaut's training set includes sequences where the GROUNDS subtask is followed by SUGAR and then CREAM and others where the CREAM subtask is followed immediately by the SUGAR subtask. Without such sequences in the training set, the model would not be prone to recurrent perseveration of the SUGAR subtask. On the basis of this logic, one might expect the SRN model to be prone to omission of the SUGAR and CREAM subtasks when making coffee but not prone to omission of the GROUNDS subtask, which is always the first task of coffee preparation. Similarly, the model should not be prone to recurrent perseveration of the GROUNDS subtask.

To explore these predictions, we reimplemented and trained the SRN model as described by Botvinick and Plaut (2004). Noise was then introduced, and the specific errors produced by the model were tabulated. When noise was low (at the levels used by Botvinick & Plaut, 2004, to simulate normal slips and lapses), the predictions were observed to hold: No omissions or perseverations of TEA or GROUNDS subtasks occurred in 1,000 attempts at tea making and 1,000 attempts at coffee making when the standard deviation of noise held at 0.10, the level used by Botvinick and Plaut to simulate action slips and lapses in normal subjects. In contrast, the SUGAR subtask was omitted on 464 out of 2,000 occasions and repeated 168 times, whereas the CREAM subtask was omitted on 367 out of 1,000 occasions and repeated 28 times.[2] Full details of the simulation are given in the appendix (see Simulation 1, *Analysis A*).

It is clear that the tendency of the SRN model to produce errors that consist predominantly of subsequences occurring in the model's training history is empirically unsatisfactory. For example, one anticipation error commonly produced by patients involves attempting to pour from a sealed container (e.g., De Renzi & Lucchelli, 1988; Schwartz et al., 1991). Within the domain investigated by Botvinick and Plaut (2004), this error might be manifest by the model attempting to pour from the coffee, sugar, or cream packets before opening them. The task thus provides ample opportunity for this particular error, and Botvinick and Plaut cited one such error produced by the model—pouring from the cream container before it has been opened (see Botvinick & Plaut, 2004, Table 6). However, in a sample of 47,572 errors occurring in a corpus of 22,000 trials produced by our reimplementation of the SRN model with varying levels of noise, there was not a single occurrence of this form of anticipation error. (See Simulation 1, *Analysis D*, in the Appendix for details.) Such errors are thus exceedingly rare in the behavior of the SRN model. The reason is that pouring from a sealed container is something that never happens in the training set: The probability within the training set of selecting *pour* when holding a sealed container is zero. In contrast, actions such as *put down* or *tear* (or any of the other actions related to opening) have nonzero probability of occurrence. Thus, although noise could in principle lead to *pour* being selected when holding a sealed container, *put down* or any of the various open actions are far more likely to be selected. Similar comments apply to tool omission errors (e.g., attempting to use a finger to stir the coffee), which never occur in the training set, are unlikely to arise with substantial frequency in normal behavior, but are relatively frequent in the behavior of some neurological patients (see, e.g., Rumiati, Zanini, Vorano, & Shallice, 2001). Again, common tool omissions, such as stirring or scooping without a spoon, were not observed in our error corpus.

These analyses are important because several researchers (e.g., Henson, 1998; Houghton, 1990; Houghton & Hartley, 1995) have suggested that recurrent networks essentially implement a chaining approach to sequential behavior (where the current action is determined by the previous action and the current input), and omission errors and recurrent perseverative errors are two error types that would seem unlikely within such an account (Lashley, 1951). Indeed, these and related order errors led us to express skepticism about whether recurrent networks could account for certain kinds of slips and errors (Cooper & Shallice, 2000)—skepticism that Botvinick and Plaut (2004) took as a challenge. A critical empirical question is then whether all order errors of normal subjects and patients may be traced to a training history that includes different orders of the relevant actions or subsequences. This may possibly be true, but it remains far from having been demonstrated.

The second problem with the selection of the training set is that the model needs to be trained on all legitimate sequence orders. Thus, the SRN model cannot form an abstract representation of sugaring from the four sequences of the coffee task, with two different orderings of adding sugar and cream and two different ways of adding sugar, and generalize from one version of the tea task, for example, that with SUGAR (BOWL), to the other version, in this case that with SUGAR (PACK)—see Simulation 3 in the Appen-

[2] At higher levels of noise, omission and perseveration of fragments of the TEA and GROUNDS subtasks did occur. However, in these cases, the model's behavior was in general far more disordered, with many different errors of different types occurring in combination with these omission and perseveration errors. In these cases, the action sequences produced by the model suggest that behavior either consisted of the tail end of a trained sequence or (at very high levels of noise) was effectively random.

dix for details.[3] Similarly, the SRN model cannot produce the fourth coffee-preparation sequence by abstracting from the other three. The difficulty in these cases arises from the fact that the SRN model does not represent the sugaring subtask (or any subtask for that matter) as a distinct entity separate from the context in which it occurs. Thus, if the SRN model encounters sugaring within an unfamiliar context, it may succeed in adding sugar, but the model has no way of ensuring that critical aspects of the prior context are preserved during the subtask and, therefore, no way of ensuring that, once sugar has been added, it can continue the original higher order task in an appropriate fashion.

Similar problems of the limits of generalization arise from variations in the task environment. If, for example, the SRN model is trained in a task environment with the sugar bowl initially closed when making tea but open when making coffee, it cannot succeed in either task if the bowl is not initially in the state for which the model was specifically trained. (See Simulation 4 in the Appendix for details.) The SRN model therefore makes the counterintuitive prediction that generalization of variants of a subtask (e.g., different ways of adding sugar) across tasks that share those subtasks (e.g., coffee making and tea making) is not possible.

Thus, the composition of the training set and particularly the ordering of subsequences within that set are critical in ensuring both that the model learns to produce all legitimate orders and that the model is able to produce the right kinds of order errors. Botvinick and Plaut (2004) accepted that the training history has a critical influence on the SRN model's behavior and saw this as a strength, stating, for example, that "the specifics of the sequencing mechanism are shaped by learning, with the result that they are closely adapted to the details of specific task domains" (Botvinick & Plaut, 2004, p. 420). However, they made only limited comments on the origins of the training set, and these were restricted to the way that they included single-step (i.e., nonsequential) background examples within the training set. Such examples are critical in producing many of the SRN model's errors, but to fully explore the SRN model's predictions, it is necessary to have independent justification for the selection of individual sequences and of the assignment of their frequencies in the training set. Hence, although it might appear preferable for schemas to be acquired (à la Botvinick & Plaut, 2004) rather than specified by hand (à la Cooper & Shallice, 2000), in effect, Botvinick and Plaut simply transferred the burden of schema specification from an explicit schema hierarchy to a training set. In principle, that training set might be empirically determined through observation of the sequences observed by the learner, but this has yet to be attempted. It therefore appears that Botvinick and Plaut's approach merely replaces one problematic aspect of the IAN model (hand coding of action schemas) with another (hand selection of training exemplars).

Schema Similarity

It is generally agreed that there is an element of sharing or overlap in the mental representations of similar action sequences (see, e.g., Botvinick & Plaut, 2002; Grafman, 1995; Schank & Abelson, 1977). Evidence from transfer, learning, and neurological breakdown has been cited in support of this view. Botvinick and Plaut (2004; see also Botvinick & Plaut, 2002) argued that one advantage of the SRN model in comparison to the IAN model is that the acquired schema representations automatically encode schema similarity. Evidence for this was provided, for instance, by the similarity of the multidimensional scaling plots of the SUGAR (PACK) subsequence within the different contexts of coffee preparation and tea preparation (cf. Botvinick & Plaut, 2004, Figure 4); this results in a tendency of the model when lesioned to produce capture errors (James, 1890; Norman, 1981), where behavior on one task is captured by a related overlearned action sequence (see Botvinick & Plaut, 2004, Simulation 2A). However, it is incorrect to think of schema representations within the IAN model as being disjoint and nonoverlapping. Although schema nodes may be discrete elements of the schema network, hierarchical relations between nodes mean that schemas may share subschemas (where a schema's subschemas are defined as those schemas that achieve a subgoal of the schema). Botvinick and Plaut accepted this but argued that this form of subschema sharing carries "less representational richness and flexibility than the idea of information sharing implies" (Botvinick & Plaut, 2002, p. 308).

Two considerations led Botvinick and Plaut (2002) to this negative assessment. First, they argued that higher level schemas can share subschemas only if the execution of those subschemas is "absolutely invariant with respect to context" (Botvinick & Plaut, 2002, p. 308). In fact, this is not correct. The use of preconditions and postconditions within the IAN model overcomes this difficulty: Within the IAN model, actions that are normally realized by a schema are not expressed in behavior if those actions would merely contribute toward the achievement of the current states of affairs or if the schema is terminated early because its postconditions are met. Differences in context may also arise if, for example, tea is prepared with a small teaspoonful of sugar but coffee with a heaped teaspoonful of sugar. It is true that this form of contextual variation has not been addressed within the IAN model, but it could be addressed by augmenting the IAN model with manner and quality features that are inherited by subschemas from superordinate schemas. Second, they suggested that some abstract patterns of behavior do not decompose simply into tasks and subtasks. Botvinick and Plaut (2002, p. 308) gave the example of *fixate X, reach for X, grasp X, fixate Y, move hand to Y, put down X,* where *X* and *Y* may be instantiated with different object descriptions for different tasks. Again, this form of structure sharing does not present difficulties for the IAN model. As object representations and schema nodes inhabit separate subnetworks within the model, it is possible for two different high-level schemas to activate a single *move X to Y* schema with different object representations

[3] Botvinick and Plaut (2004) suggested that their model is capable of precisely this kind of generalization, citing unpublished observations in which "systems of this sort . . . infer sequence equivalence, interchanging equivalent sequences in a way that produces overall sequences the network has not observed during training." (pp. 423–424). We were unable to replicate this effect with our implementation of the SRN model on the coffee- and tea-making tasks. Matthew Botvinick (personal communication, October 14, 2005) has confirmed that these unpublished observations relate to a scaled-down version of the model using a modified training regime. The conditions under which the SRN model itself can generalize subsequences to contexts beyond those encountered in its training history and subsequently continue with the original task without being captured by an example from the training history therefore remain to be identified.

being activated (by higher level schemas) for source and target in each case.

Capture errors are also possible in the IAN model, although their origin differs from that in the SRN model. The IAN model includes mutual interactions between schemas and object representations, and these may lead to capture errors. Thus, if an object satisfies the triggering conditions of two schemas, then it tends to excite both schemas. Furthermore, if one of the schemas is highly active, it tends to excite the representation of the object, and this in turn tends to excite the other schema. If this excitation of the other schema is not regulated (e.g., through sufficient lateral inhibition in the schema network), the schema may become inappropriately selected and capture behavior.

A related feature of the SRN model is that it automatically encodes schema frequency, with more frequent tasks creating stronger attractors than less frequent tasks. This results in frequency effects, such as the tendency for behavior on less frequent tasks to be captured by more frequent tasks (see Botvinick & Plaut, 2004, Simulation 2A). The automatic encoding of schema frequency is not present in the implementation of the IAN model, and this may appear to be a weakness of that approach. However, in the verbal description of the contention scheduling theory, Norman and Shallice (1980, 1986) suggested that different schemas may have different selection thresholds (i.e., different activation levels that result in the selection of the schema). Specifically, well-learned schemas were held to have lower selection thresholds. The evidence cited by Botvinick and Plaut (2004) in favor of their frequency-dependent encoding is not inconsistent with this basic theory. In addition, Botvinick and Plaut failed to show that the results of their Simulation 1 hold when sequence frequency is varied, as in their Simulation 2A. As discussed below in the section titled The Implementation of Choice (see also Simulation 2 in the Appendix), our own simulations suggest that the frequency of sequences in the training set must be finely balanced if the SRN model is to be able to generate all sequences on which it has been trained.

Sequential and Hierarchical Control Structures of Routine Action

On the IAN model, actions are organized through a hierarchical structure. Botvinick and Plaut (2004) rejected this approach. They argued that it has problems over how the hierarchy is learned, over how multilevel control of behavior is sequenced, and, in the interactive activation version at least, over how it accounts for error data.

Sequencing

Within the IAN model, sequential behavior results from the activation, eventual selection, and then inhibition of nodes within the schema hierarchy. An important source of schema excitation is top-down excitation from a parent schema to its component schemas. When a parent schema is selected, however, it does not excite all of its component schemas, just those whose preconditions are satisfied and whose postconditions are not satisfied. As the mechanism by which this selective excitation of component schemas is not specified further, Botvinick and Plaut (2004) claimed that the IAN model assumes "an important part of the functionality it is intended to explain" (p. 398). This claim is misguided on two counts.

First, it fails to take into account the fact that a selected schema may excite multiple component schemas if the preconditions of those component schemas are satisfied. Thus, in the IAN model as applied to the coffee-preparation task, selection of the *add sugar from bowl* schema results in excitation of schema nodes for picking up an implement, dipping an implement in an open source container, and emptying the implement into an open target container. All three subschemas receive top-down excitation. Sequential order is imposed by bottom-up excitation in the form of triggering conditions (comparable to affordances), whereby picking up an implement initially receives excitation from the representation of the environment (because that is the only component schema that may be performed given an initial state of the environment in which an implement is not held). Therefore, although gating of top-down excitation by precondition achievement is an important factor in determining the sequential order of the model's behavior, it is not the only factor.

Second, the IAN model does not deny that gating of top-down excitation is implemented in neural terms. Rather, it assumes that normal and impaired behavior may be modeled without recourse to the neural implementation of the mechanism, and the results of Cooper and Shallice (2000) and Cooper et al. (2005) support this. Thus, the issue is one of the level at which the theory is specified.

A further question raised by Botvinick and Plaut (2004) with respect to sequencing concerns the time course of reflex inhibition. Within the IAN model, units at the lowest level of the schema hierarchy are inhibited immediately after selection, allowing other low-level units to become active, but selected units higher in the hierarchy are only inhibited once all of their subgoals have been achieved. Botvinick and Plaut took issue with this, claiming of the IAN model that "the actual mechanisms responsible for goal-monitoring and schema inhibition . . . remain to be explained" (Botvinick & Plaut, 2004, p. 398). Again, this is an issue of the level at which the theory is specified, and the comments in the preceding paragraph apply.

Learning

A key advantage of the SRN model over the IAN model, according to Botvinick and Plaut (2004), is that the SRN model provides an account of the acquisition of routine action, with quasi-hierarchical structuring emerging from the model as it acquires action sequences. However, the approach to skill acquisition within the SRN model has serious failings. Thus, as discussed below in the section titled Goals and Learning, the SRN model adopts implausible assumptions concerning the role (or lack thereof) of explicit subtask structure in task acquisition.

Most clearly problematic is how the SRN model deals with the problem of catastrophic interference. First, it should be noted that the SRN model is severely subject to this potentially grave problem for many connectionist models. To ascertain this, we trained our replication of the SRN model first on preparing tea. Once the model mastered this task, the training set was changed, and the model was trained on the coffee-preparation task. Once this task was learned, the training set was then switched back to that for the tea task, and so on. Performance on each task was monitored after each training epoch. The reverse situation, learning the coffee task first, was also explored. (See Simulation 5 in the Appendix for full details.) Figure 5 shows the performance of the model on the two tasks as learning progressed. Not surprisingly, switching the training set led to immediate impairment of the previously mastered task, and the model alternated between mastery of each task as the training set was switched. Granted, less training was required on each subsequent switch for the model to regain its previous level of performance, such that after many alternations, the model did eventually acquire both tasks. However, this does not diminish the basic problem of catastrophic interference—that acquisition of a second task impairs performance on previously acquired tasks.

The requirement that the model be exposed to all training sequences on every training epoch is clearly unrealistic. To cope with this problem, Botvinick and Plaut (2004) imported a hypothetical and complex learning mechanism previously postulated to deal with the potential for catastrophic interference in learning semantic representations (McClelland, McNaughton, & O'Reilly, 1995). Botvinick and Plaut proposed that the learning of action sequences occurs in two stages. Thus, they adopted the McClelland et al. (1995) position that acquisition is initially in the hippocampus, which then trains the cortex, so as to reduce the possibility of catastrophic interference in learning multiple input–output mappings in the cortex (see Botvinick & Plaut, 2004, pp. 401, 403). This however creates a number of problems. The McClelland et al. model is controversial even for the retention of semantic (i.e., nonsequential) information (see, e.g., Nadel & Moscovitch, 1997). More critically, there is no evidence that the hippocampus can retain and order completely accurately a very long sequence of input-to-output mappings that would be required to implement hippocampal training of the action sequence.

This hypothesis also fits very poorly with other neuroscientific evidence. Learning of instrumental behaviors, an animal precursor of motor skills, involves two systems—an inflexible, automatic habit stimulus–response system and a flexible goal-directed action system (Dickinson, 1985; Knowlton, Mangels, & Squire, 1996). Thus, the process of reward devaluation affects goal-directed action, which is employed early in learning a novel action, but does not influence the operation of the habit-based system that controls action later (e.g., Balleine & Dickinson, 1998). Key structures in the implementation of automatic habit repertoires are the dorsolateral striatum and the premotor and motor cortices (Graybiel, 1998). Although Botvinick and Plaut (2004) did not draw the connection, this is clearly a system that could relate to the SRN model. Critical in the training phase is, however, the other system, the goal-directed action system, which, by contrast, requires the prefrontal cortex, the pre–supplementary motor area, and the dorsomedial striatum (Yin, Knowlton, & Balleine, 2004). Thus, it is the prefrontal cortex and the dorsomedial striatum that tend to be involved early in learning a motor skill. For instance, Jueptner et al. (1997) scanned subjects both while they were performing a new motor sequence and when they had already learned it well. The set of regions active when the task was well learned (cingulate, supplementary motor area, premotor cortex, motor cortex, left parietal cortex, basal ganglia, and cerebellum) was even more active when the task was novel. In addition, however, when the task was novel, the prefrontal cortex, particularly the right, was also strongly active. (See also Aron, Monsell, Sahakian, & Robbins, 2004, and Alexander, Stuss, Shallice, Picton, & Gillingham, 2005, for the involvement of the left prefrontal cortex in the acquisition of task switching and serial reaction time, respectively, and Hollerman, Tremblay, & Schultz, 2000, for relevant basal ganglia evidence.) Moreover, the striatal region that is indirectly connected to the hippocampus is the dorsomedial striatum, and not the dorsolateral striatum controlling habit (Devan & White, 1999; Poldrack, Prabakharan, Seger, & Gabrieli, 1999; see also Graybiel, 1998). As far as the habit system is concerned, one thus has a much more distant and tenuous anatomical relation to the hippocampus when one compares it with the links that inferior anterior temporal structures involved in semantics have with the hippocampus, namely, the ones required by the initial McClelland et al. (1995) model of overcoming catastrophic interference. Indeed, it is most plausible that any training input from the hippocampus to an habitual action system could occur only when mediated by the goal-directed action system, yet Botvinick and Plaut's appeal to hippocampal systems to overcome catastrophic interference assumes that goal directedness plays no part.

The situation with respect to learning in the SRN model is further complicated by the fact that, as Botvinick and Plaut (2004) acknowledged, learning may occur via a variety of means and that their implementation includes only one of these (learning by imitation). It is far from clear how the model might be extended to include other learning mechanisms or how such mechanisms would impact upon the model's behavior. Acquisition of routine action sequences is not in fact addressed in the IAN model, and as Botvinick and Plaut pointed out, although sequence learning has been addressed within the interactive activation framework (e.g.,

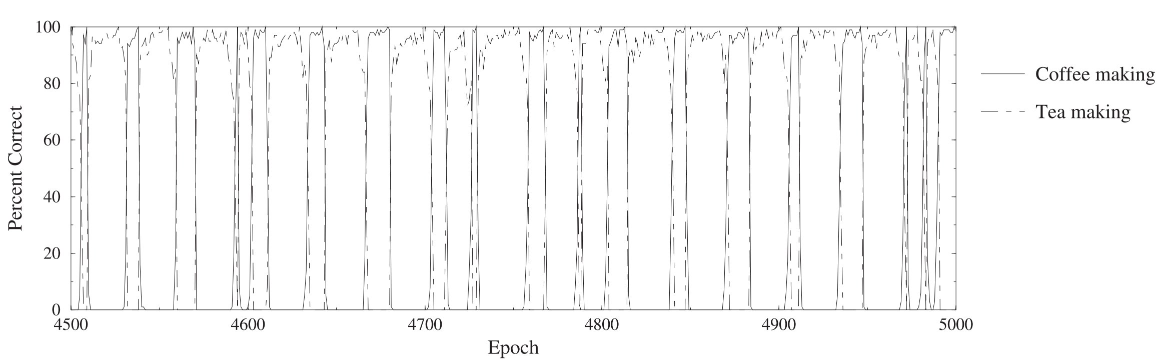

Figure 5. Mean number of cycles required to reach successive criterion states.

Burgess & Hitch, 1992; Grossberg, 1986; Hartley & Houghton, 1996; Henson, 1998; Houghton, 1990; Houghton, Glasspool, & Shallice, 1994), hierarchical interactive activation models, including those of Estes (1972), Rumelhart and Norman (1982), MacKay (1985), and Cooper and Shallice (2000), all rely upon appropriate hand-coded hierarchical structure. However, the basic approach on this set of models consists of maintaining a time-varying context representation and associating this representation at successive points in time with successive sequence nodes within an interactive activation network (i.e., a set of mutually inhibitory, semantically interpretable nodes). The approach—competitive queuing—can account for rapid (one-trial) learning of sequential structure, and Humphreys and Forde (1998) have suggested that it might be extended to the domain of routine sequential action. Botvinick and Plaut's (2004) only criticism of this work was that it assumes an ability to identify sequence boundaries—a criticism we address below. Note, however, that the supervisory processes described below as putatively responsible for the development of the schema hierarchy structure can be localized in the prefrontal cortex (Shallice, 2004; Shallice & Burgess, 1996), and this is consistent with the empirical evidence of Jueptner et al. (1997) cited above.

A further aspect of learning within the contention scheduling framework (beyond acquisition of hierarchy and sequence) relates to the acquisition of schema triggering conditions. This can be viewed as the result of Hebbian or delta-rule learning that associates representations of the environment in which a schema is performed with the schema representation within the contention scheduling system. Preliminary work has demonstrated that this can account for the acquisition of triggering conditions for individual actions (Cooper & Glasspool, 2001), and the generalization of this approach to schemas is straightforward.

Context Sensitivity and Quasi-Hierarchical Sequences

Botvinick and Plaut (2004) claimed that for the routine system itself, a nonhierarchical system is preferable. In claiming this, they cited Agre (1988) and situated themselves by analogy with the AI reactive planners of the 1980s who dispensed with intelligent planning systems (Agre & Chapman, 1987; Firby, 1987; but see Gat, 1998; Glasspool, 2005). Thus, they explicitly stated that "performance of a routine should vary with the larger behavioral context" (Botvinick & Plaut, 2004, p. 398). To illustrate this, Botvinick and Plaut asked the reader to imagine a waiter with three coffee-preparation routines (appropriate to three different customers) differing only in the amount of sugar (zero, one, or two teaspoons) added in each routine. They claimed that the IAN model, when applied to this task, could not capture the inherent similarities between the routines but would need three separate coffee-preparation schemas.

In our opinion, the example is not convincing. First, the idea that one would learn that A has one sugar and B has two sugars without using an explicit representation of one and two and a counting routine is implausible. There are many routine acts where a specific number of operations are required: in using a recipe, in making tea with a pot, in taking pills, and so on. For the Botvinick and Plaut (2004) model, numbers cannot be used explicitly, either in the skilled implementation of the task or, even more critically, when it is being learned. Furthermore, if one uses a counting routine, one needs to have a representation of what one is counting. This is debarred on the Botvinick and Plaut approach as there is no explicit representation of the sugaring subroutine—one is forced to assume that whatever processes in the brain are used in the initial training phase and whatever types of representation they produce are irrelevant for the final state of the SRN. It needs to be assumed that the system observes its own input and own output, however produced, and learns the pairings. In contrast, we would model the waiter scenario within the IAN model by assuming that, although a schema may exist for one version of the coffee task, other versions would be controlled through temporary schemas created and maintained by higher level systems throughout the task (cf. Shallice, 2004; Shallice & Burgess, 1996). This is facilitated by the explicit representation of schemas at all levels within the IAN model.

At the same time, routine behavior can indeed be highly context sensitive. A more realistic example involves the preparation of a beverage from different initial situations. On the IAN model, there do not need to exist different schemas for coffee preparation for situations in which the milk container is initially closed or initially open or different schemas for buttering toast depending on whether one is currently holding a butter knife (from a previous task) or not. Rather, schemas are held to include optional elements; their inclusion on any particular occasion is determined by the context in which the schema is performed. The association of preconditions and postconditions with subgoals within a schema within the current version of the IAN model allows for just such optional elements (Cooper et al., 2005).

A further example of context sensitivity was discussed by Botvinick and Plaut (2002). Making coffee and making cocoa both involve scooping an ingredient into the target mug. In the case of coffee, this is a moderate size scoop of sugar, whereas, in the case of cocoa, this is a large scoop of cocoa mix. Botvinick and Plaut discussed how this might be accommodated within the SRN model through the addition of an extra output unit to represent the modifier *large* (which should be activated on the same step as *scoop* when scooping cocoa, but not when scooping sugar). This form of context sensitivity can also be addressed within the IAN model through an appropriate augmentation, namely, through the addition of manner features (such as *large* or *quickly*) that act as modifiers of actions and that are specified at higher levels of the schema hierarchy and inherited by schemas at lower levels.

Goals and Subgoals: Explicit or Redundant?

An important element of the concept of a schema as employed by us (Cooper & Shallice, 2000) is that schemas are goal directed: Action schemas are invoked to achieve goals, and successful performance of a schema entails that the schema's goal is achieved. The term goal is used synonymously with purpose: Schemas are held to be purposeful, and behavior is held to consist of segments of purposeful action. This is far from a novel claim, either at the level of complex tasks (cf. Miller et al., 1960) or the level of routine activities (Schwartz et al., 1991).

Two Arguments Against Goals

Botvinick and Plaut (2004) were equivocal with respect to the importance of goals within routine behavior. At a general level, they accepted that "in some circumstances human action does involve instantiation of explicit goal representations" (Botvinick & Plaut, 2004, p. 424). At they same time, goals play no role in the functioning of the SRN model. They presented two arguments against goals as constructs that necessarily play a role in structuring routine behavior. First, they claimed that the concept of goal is too rigid to account for "the extent to which goals may be context dependent" (Botvinick & Plaut, 2004, p. 423). The only example they gave was of how one's goals in cleaning the house may vary depending on whether one is tidying up or preparing for a visit from one's mother. It is not at all clear what the sense of context is here; for instance, it does not relate to aspects of the immediate environment. Instead, the example is simply characterizable as behavior attempting to realize one of two related but distinct goals. In any case, the reservation ignored the way that, within the IAN model, contention scheduling operates in tandem with a supervisory system whose functions include the modulation of contention scheduling in nonroutine contexts. Second, Botvinick and Plaut argued that there are behaviors "for which it is not straightforward to identify discrete, explicit goals" (Botvinick & Plaut, 2004, p. 423). The example they gave was of playing a violin. Anyone who has ever tried to play the instrument is very well aware that the goal is to produce a particular type of attractive aesthetic sound, and for a novice, this is very difficult to achieve. It is true that the goal in this case is not easily made explicit. However the top-down flow of control in contention scheduling does not require that the higher levels of the structure have a full representation of all that is produced by the lower level schemas. Thus, in the case of violin playing, the higher level schemas do not need to have a representation of the individual finger, arm, and wrist movements required to produce a specific melody.

Three Arguments for Goals

Regardless of the above objections to assuming the involvement of goals, it is appropriate to ask what purposes are served by goals within the IAN model and how or to what extent these purposes are achieved within the SRN model. Goals are critical within the IAN model for four different types of reason. First, they provide a source of activation for the units controlling behavior. Second, they allow one to distinguish between different roles of actions within a sequence and hence compute and assemble specific actions that are necessary in a given situation. Third, goals enable schemas to be treated as interchangeable. Fourth, goals facilitate the learning process by helping to realize the chunking structure of a longer sequence in terms of specific subroutines. The first of these types of reason is specific to the IAN model. The remaining three are of general significance.

Goals and enabling, crux, and tidying actions. Goals allow a distinction to be made between critical behaviors and enabling or tidying behaviors. This is realized in the concept of a crux action (Schwartz et al., 1991). Within many routines, certain actions are more important to successful completion of the routine than others. Thus, in adding sugar to a beverage, the crux action is that in which the sugar is actually added (either pouring the contents of the sugar packet into the mug or emptying a spoonful of sugar into the mug). Other actions in the sequence serve to enable successful execution of the crux (e.g., removing the lid of the sugar bowl) or serve a subsidiary tidying function to enable further action (e.g., discarding the empty sugar packet). Intuitively, the crux action within a sequence is the one that achieves the primary goal of the sequence. It is also the one that is most essential. Thus, if the sugar bowl's lid has already been removed, the action may be safely omitted. Even the act of discarding the spoon on completion of the sequence can be omitted if the spoon is required by the next sequence; indeed, discarding the spoon should be omitted if the next action would only result in picking it up again. What cannot be omitted is the crux action: the act of actually depositing sugar into the mug.

The concept of a goal and the related concepts of enabling, crux, and tidying actions facilitate the efficient assembly of action sequences into novel combinations as required even in simple situations when, for example, one is required to butter two slices of toast and one spontaneously assembles two instances of the *butter toast* schema by leaving out the inessential tidying-up actions of the first instance and the preparatory actions of the second instance and running the crux actions together. Although the original IAN model (Cooper & Shallice, 2000) is not capable of such flexibility, a revised model does show precisely this flexibility, largely because of the explicit goal-directed nature of schemas (cf. Cooper et al., 2005). Within the revised model, each subgoal of a schema has a precondition and a postcondition. When a schema is selected, activation is passed to the nodes for schemas corresponding to subgoals that have preconditions that are met and postconditions that are not. At the same time, nodes for selected schemas corresponding to subgoals whose postconditions are met receive inhibition. Transferring butter from the butter container to the butter knife and from the butter knife to the toast are two subgoals of the *butter toast* schema, but the postcondition of the second (that a butter knife, without butter on its blade, be held) matches the precondition of the first. Moreover, the postcondition of the *butter toast* schema generally is met once butter has been applied to the toast, even if the knife is still held. Running two versions of the sequence together therefore results in the first instance of *butter toast* being inhibited and deselected once butter has been transferred to toast (prior to discarding the knife), and selection of the second instance does not activate picking up of the knife (because it is already held). The net result is that the transfer actions of both instances are performed without an intervening put-down/pick-up of the knife. Note that this behavior is achieved within the IAN model through preconditions and postconditions associated with the subgoals of a schema and without explicit marking of crux, enabling, and tidying actions. The model therefore does not require that each schema have precisely one crux action.

Analogous processing occurs in the more complex situation of task interleaving, where objects that are to be used again may be spontaneously left in an appropriate state for later use and novel action sequences that maintain a joint purpose are constructed on the fly (cf. Joe, Ferraro, & Schwartz, 2002). The explicit representation of goals within the IAN model means that it is fully compatible with all of the requirements of interleaving, and in the semicomplex task of preparing and packing a child's lunch box, the model is able to either prepare all items before packing them or interleave the preparation and packing operations (Cooper et al., 2005).[4]

Goals and the interchangeability of schemas. Within the coffee-preparation task, adding sugar may be achieved either by using a packet of sugar or by using a sugar bowl. It is the shared goal (of sweetening the beverage) that allows these schemas to be interchanged. Botvinick and Plaut (2004, p. 423) suggested that their model can give the impression of being goal directed without any representation of goals because it can learn to perform the two different sugar-adding subsequences as if they were interchangeable. Although this is true, as discussed above, the SRN model as presented by Botvinick and Plaut can only do this if it is trained with each and every variant of sugar addition in the context of each and every task that employs sugar (but see footnote 3, above). It cannot spontaneously generalize or transfer this learning to use different sugar-addition methods in a related task (e.g., making tea) because, in its current instantiation, the model has no way of representing subtasks as discrete entities and no way of knowing how to preserve context information (e.g., whether it is making tea or coffee) across a subtask (e.g., adding sugar) unless it has received explicit training on that variant of the task. Simulations supporting this claim are described in Simulation 3 in the Appendix.

Goals and learning. Botvinick and Plaut (2004, p. 397) argued that learning presents a serious difficulty for hierarchical approaches to action. In particular, they suggested that a significant factor limiting the extension of existing approaches to learning within interactive activation networks (e.g., Grossberg, 1986; Houghton, 1990) to the learning of hierarchical structure concerns the determination of sequence boundaries. Indeed, if one considers, say, the perception of familiar speech units (Saffran, 2001), such a criticism is appropriate. However, this relates to perception, not production. Furthermore, if subsequences achieve subgoals and subgoals are explicit at least initially in learning, then the problem of determining sequence boundaries dissolves. Thus, the idea that the child, when learning coffee making, would have difficulty in separating out the subroutines conceptually or that the child, when adding sugar, would not understand that sugar is sweet and would not have the goal of making the drink sweet seems highly implausible. Goals can therefore facilitate the learning process by helping to realize the chunking structure that breaks down longer sequences of perception–action pairs into the products of specific subroutines.

We suggest that tasks such as coffee preparation are primarily acquired through instruction of the contention scheduling system by the supervisory system. High-level goal-directed problem solving would initially be responsible for developing solutions to simple subtasks such as adding sugar or milk to a beverage. Schemas that embody these solutions develop with practice within contention scheduling and are then available for use in more complex tasks, such as preparing coffee, which again are controlled initially through biasing of behavior by a supervisory system but which, with practice, are also transferred to contention scheduling. In this way, hierarchical structure is not abstracted by unguided imitation or observation of lengthy, apparently purposeless action sequences. Rather, it develops as a result of top-down problem solving and bears strong similarities to the mechanism of learning by chunking within the Soar cognitive architecture (Laird et al., 1987; Newell, 1990; see also Duncan, 2001). Alternatively—or in conjunction—learning may use imitation, and studies on learning by imitation have suggested that the processes involved mediate action execution through explicit or implicit goals (Wohlschläger, Gattis, & Bekkering, 2003). Moreover, in a fuller model including a supervisory system, goals would allow the system to institute monitoring and checking procedures (see Shallice, 2004).

Doing Without Goals

The case for goals, even in the performance of routine or everyday activities, appears strong. How then does the SRN model achieve its impressive performance in the absence of any goal representations? The answer is twofold. First, at the level of the complete task, Botvinick and Plaut's (2004) instruction units serve to specify an intention. Although the network learns to immediately encode this intention in its context units, it is nevertheless initialized with an intention (as, in our opinion, it should be). This is true in both their basic simulations of tea and coffee preparation (Botvinick & Plaut, 2004, Simulation 1, where two mutually exclusive instruction units are employed) and the additional simulations of coffee preparation with zero, one, or two sugars (Botvinick & Plaut, 2004, Simulation 1A, where three mutually exclusive instruction units are employed). In addition, in all cases, the model is trained to select the *say done* action when the goal is achieved.

Second, close inspection of the SRN model's performance reveals that performance is not that impressive; it lacks the kind of behavioral flexibility seen in everyday human action. Although the model is able to learn six action sequences (including four with 37 steps) and although those sequences contain some overlap in the form of subsequences that notionally achieve subgoals, the learned action subsequences cannot be combined in novel ways, and as noted above, the model breaks down if the task environment in which it is applied is not identical to that in which it was trained (e.g., if the lid has been left off the sugar bowl).

Linking Actions With Objects

Botvinick and Plaut (2004) employed a deictic scheme to link actions to objects. It is implemented through separate fixate or attend actions for each object relevant to the task. There is much evidence in support of a deictic scheme (e.g., Hayhoe, 2000; Land, Mennie, & Rustead, 1999). However, such a scheme is orthogonal to the issue of the representation of schemas or the underlying computational processes—it is fully consistent with either explicit representations of schemas or implicit schemas and with either a recurrent or an interactive computational substrate (see below). What is not independent is the detailed implementation of such a scheme. Botvinick and Plaut avoided any internal representation of objects. The key difference is therefore not the use of deictic reference but the explicit representation of objects within the IAN model.

[4] Notwithstanding this, intentional interleaving (as in the six-element task introduced by Shallice & Burgess, 1991) would appear to require some mechanism to switch away from the current task even when that task is progressing adequately. We assume this to be a supervisory system function that operates by intentional inhibition of an ongoing schema and excitation of an alternative schema.

Studies of the action selection of neurologically intact individuals and certain classes of neurological patient have provided several strands of evidence that may help to differentiate the approaches to linking actions with objects. Thus, patient studies (e.g., De Renzi & Lucchelli, 1988; Giovannetti, Libon, Buxbaum, & Schwartz, 2002; Humphreys & Forde, 1998; Rumiati et al., 2001; Schwartz et al., 1991, 1998) have revealed that object substitution errors, in which an appropriate action is performed with an incorrect object (e.g., a fork is used in place of a knife for spreading butter on toast), are relatively common in the actions of many different patient groups. Mild forms of such errors also occur in the slips and lapses of unimpaired individuals (Reason, 1979, 1984). If objects are represented implicitly only in terms of fixate or attend actions, then such errors must arise from either the selection of an incorrect fixate action or an error in some additional, downstream system responsible for performing fixate actions. Representing objects explicitly and incorporating a mechanism for associating objects with actions (as in the Cooper and Shallice, 2000, approach) constitute one way of effectively specifying this additional system, except that the interactions between object representation units and schema units within the IAN model mean that the system is not downstream from the action selection system but is reciprocally and interactively connected with it.

The Influence of Distractor Objects

There is substantial empirical evidence for the effect of the presence of unattended or distractor objects on action selection. Studies of reaching behavior both in normal subjects (Pratt & Abrams, 1994; see also Tipper, Lortie, & Baylis, 1992) and in neurological patients (Riddoch, Edwards, Humphreys, West, & Heafield, 1998) have shown that action initiation in reaching tasks is delayed when distractors are present as compared with when they are not present. In a related fashion, Meegan and Tipper (1998) found that normal subjects made nonnegligible numbers of errors in a simple reaching task when distractors were present. Furthermore, several neuropsychological group studies with action-disordered patients, including closed head injury patients (Schwartz et al., 1998), left-hemisphere stroke patients (Buxbaum, Schwartz, & Montgomery, 1998), right-hemisphere stroke patients (Schwartz et al., 1999), and dementia patients (Giovannetti et al., 2002), have demonstrated that the presence of additional distractor objects in the local environment affects error profiles when performing a range of simple activities of daily living, with all patient groups omitting more actions when distractor objects were present than when they were not.

The presence of distractor objects can also lead to outright object substitution errors. These account for one of the four main categories of reported slips of routine action by normal subjects (Reason, 1984), and such errors made up 17% of all errors observed by Schwartz et al. (1998) in the behavior of a healthy control group completing a range of everyday tasks. Both Schwartz et al. and Humphreys and Forde (1998) reported that object substitution errors accounted for approximately 10% of the errors produced by their frontal patients.

If objects are represented only in terms of relevant fixate or attend actions and, in addition, environmental input to the system is limited only to the representation of the objects that are attended and held, as in the SRN model, then only those objects that are actually attended or held can influence later action selection. The SRN model therefore provides no account of how the presence of distractor objects may slow action selection or lead to omission errors. This insensitivity of the SRN model's behavior to distractor objects is not an implementation issue that can be addressed by increasing the number of objects in the task environment (and hence increasing the number of associated fixate actions that might be produced by the model). To have any effect, such a modification would need to be accompanied by training of the modified model on tasks involving the new objects. However, an object can only affect the SRN model's behavior if the SRN first fixates the object, and that fixate action is internally generated by associations learned through the SRN. So, extending the model in this way does not result in more errors when distractor objects are present than when they are not because the distractor objects are not fixated.

The SRN model also provides no real account of object substitution errors, either in slips of normal action selection or in errors of impaired action selection. Thus, although Botvinick and Plaut (2004) provided an example of an object substitution error produced by their model (stirring with the coffee packet), our reimplementation of the SRN model revealed that such errors were exceedingly rare. Table 1 shows the number of object substitution errors produced by the reimplementation at various levels of noise. Each cell represents the cumulative results of 1,000 attempts at the tea task and 1,000 attempts at the coffee task. As the table illustrates, object substitution errors made up less than 0.5% of all errors when noise was 0.10 or less (the level used by Botvinick and Plaut, 2004, to simulate normal slips of action). This rose to a maximum of less than 2.0% at higher values of noise. (For full details of the simulations and a breakdown of the specific object substitution errors, see Simulation 1, *Analysis B*, in the Appendix.)

To understand why object substitution errors are rare in the behavior of the SRN model following damage, consider the error of pouring the coffee into the sugar bowl (assuming that the sugar bowl is open). The correct target is the coffee mug, so, once the unopened coffee packet is held, the sequence of actions should be *tear packet, fixate mug, pour.* For the object substitution error to occur, the central *fixate mug* must be replaced by *fixate sugar bowl* without affecting the subsequent *pour.* This is unlikely to happen because, after fixating on the sugar bowl, pouring is not likely to be supported by either the model's context representation or its inputs. It is unlikely to be supported by the context representation because that representation must have been corrupted to generate the error in the previous fixate action. It is also unlikely to be supported by the input because the training set does not include any cases of pouring the coffee packet into the sugar bowl, so that configuration of held and fixated objects should not facilitate pouring. In fact, within the SRN model, all object substitution errors must begin with an incorrect fixate action, but this fixate action is likely to result in subsequent actions being captured by the fixated object rather than being driven by the task or subtask in which the fixate error occurred. In contrast, the explicit representation of objects in the IAN model means that all objects physically present in the environment can influence the selection of actions within the model. Thus, Cooper et al. (2005) demonstrated that, when the IAN model is appropriately damaged, the addition of distractor objects to the representation of the environment produces error profiles similar to those reported by Schwartz et al. (1998) for action disorganization patients.

Table 1
Object Substitution Errors as a Percentage of Total Errors Produced by the Single Recurrent Network Model With the Standard Deviation of Noise Varying From 0.00 to 0.50

	Noise *SD*										
Error type/statistic	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
Total object substitution errors	0	0	9	37	57	89	94	103	104	133	134
Total errors	0	922	1,909	3,056	4,077	4,940	5,473	6,126	6,604	7,091	7,374
% object substitution errors		0.0	0.5	1.2	1.4	1.8	1.7	1.7	1.6	1.9	1.8

Deictic Reference in the IAN Model

The above effects argue for the explicit representation of objects at some level in the action selection system, yet the evidence for fixate or attend actions remains strong. In recent work (Cooper, in press), we have therefore incorporated deictic object selection into the IAN model. Specifically, the array of pick-up actions (*pick up source*, *pick up target, pick up implement,* and *pick up theme*) has been replaced with an array of fixate actions (*fixate source, fixate target, fixate implement,* and *fixate theme*) and a single pick-up action. As in the SRN model, all arguments of actions are set through prior fixate actions (so *pick up* operates on whatever is being fixated, and *pour* pours whatever is held into whatever is fixated). All task schemas have been adjusted accordingly (so all *pick up source* actions have been replaced by a schema of the form *fixate source, pick up,* and all *pour* actions have been replaced by a schema of the form *fixate target, pour,* etc.). This approach takes from Botvinick and Plaut (2004) the use of deictic reference but differs because it retains the explicit representation of objects, which we take to be essential in accounting for the distractor and object substitution effects enumerated above.

The modified model was applied to five multiple-object tasks commonly used to assess ideational apraxia, such as lighting a candle and juicing an orange. The model was able to perform all tasks without error using the same parameter settings as in other recent work (i.e., as in Cooper et al., 2005). More critically, Rumiati et al. (2001) described two ideational apraxic patients with tendencies toward different types of action error. Patient DR's dominant error type involved misusing objects (e.g., attempting to cut an orange with a knife by using a pushing, rather than a sawing, motion), whereas Patient FG's dominant error type involved using objects correctly but in the wrong location (e.g., striking a match on the inside of the drawer of the matchbox). Both patients produced a number of other errors characteristic of ideational apraxia. The modified IAN model was able to provide good quantitative fits to the error profiles of both patients by assuming, in one case, that the patient's deficit affected the link strengths from the object representation to the schema networks and, in the other, that the deficit affected the reverse links. (See Cooper, in press, for further details.) It is unclear how the pattern of impairments and particularly the differences between patients might be accounted for by the SRN model.

The Implementation of Choice

A key feature of the SRN model is its ability to simultaneously encode two different methods for adding sugar—from a packet or from a bowl—and to automatically select between the two. This is important because, within the Cooper and Shallice (2000) approach, the two different methods would correspond to two different schemas with a common goal.

A simple attempt at implementing this kind of choice within the general framework of the SRN model would be to employ two different fixate actions: *fixate sugar packet* and *fixate sugar bowl*; the two sugaring subsequences would then begin with different fixate operations. However, this approach fails, at least in the sense that when trained in this fashion, the model always adds cream before adding sugar, whereas when sequences in which sugar is added first appear in the training set, they are never spontaneously reproduced. The reason for this lies in the statistical structure of the training set. With two distinct subsequences for adding sugar, the transition probabilities in the coffee-related subset of the training set are as follows:

GROUNDS → SUGAR (PACK)	0.25
GROUNDS → SUGAR (BOWL)	0.25
GROUNDS → CREAM	0.50

With these transition probabilities, the context established upon stirring in the coffee grounds strongly favors adding cream, and the activation of the output units after stirring in the coffee grounds reflects this, with *fixate cream carton* being approximately 0.50 and *fixate sugar packet* and *fixate sugar bowl* each being approximately 0.25. Noise in the initial values of the context units means that these activations are only approximate, but that noise is never sufficient to change the superiority of *fixate cream carton*. Consequently, the winner-take-all approach at the output layer of the SRN model means that sugar is never added immediately after the coffee grounds.

Botvinick and Plaut (2004) ignored this issue. Instead, they employed a single *fixate sugar* action that initiates both sugar-adding subsequences. This action results in fixation moving to either the sugar bowl or the sugar packet with equal probability. Because the fixated object is input to the network, this then allows the network to proceed with whichever of the two sugaring subsequences fits with the fixated object. The net effect of this merging of the initial stages of the two sugar-adding subsequences is that the transition probabilities to the first action of adding sugar and adding cream after stirring in the grounds are equal. This is critical if the model is to reproduce all training sequences, as demonstrated in Table 4 of Botvinick and Plaut (2004).

On the positive side, the implementation of the sugaring subtasks within the SRN model demonstrates that the model is able to develop behaviors that are responsive to environmental feedback: It is the environmental feedback that first differentiates which of the two sugaring subsequences is being performed, and this feedback is sufficient to ensure that whichever subsequence is selected by the environment is performed without error. There are, however, three serious difficulties with the approach. First, the model imposes strong constraints on the tasks that it might acquire. In particular, if two tasks share an initial subsequence, the SRN model can acquire those tasks only if all options at the point of divergence are equally common in the training set. If all options are not equally common, then any option that is underrepresented is produced with greatly reduced frequency, if at all. Simulation 2 in the Appendix demonstrates this difficulty. Second, and more critically, the *fixate sugar* action effectively implements a goal: that of adding sugar. The choice of which method of adding sugar is selected is left to the environment. Thus, the model has no way of influencing which subsequence might be selected. Although it appears that the model adds sugar from the packet on 50% of occasions and from the bowl on the remaining 50% of occasions, there is no way for the model to bias selection toward either alternative because that selection is entirely the product of the implementation of the *fixate sugar* action, which itself is random. Third, the model cannot adapt the probability of each sugar-adding subsequence to reflect the probability of the subsequences in the training set. Thus, if, during training, sugar is normally added from the bowl when making coffee but from the pack when making tea, this bias cannot be reflected in the trained model's performance (see Simulation 2 in the Appendix).

One way around the first of these difficulties may be to reinterpret the activation of the vector of output units as a frequency distribution and select action probabilistically from this distribution. On this approach, actions that are rare within a context would still have a chance of being selected. In addition, the approach would not lead to high error rates in the absence of noise because, as demonstrated in Figure 4, the activity of output units for correct actions is almost always near 1.0, whereas the activity of incorrect output action units is invariably small (typically less than 0.01). This approach would also address the third difficulty if, in addition, the *fixate sugar* action were to be replaced with separate *fixate sugar packet* and *fixate sugar bowl* actions. The second difficulty, however, remains.

Accounting for Error Data in Coffee-Making Tasks

Having considered general issues arising from conceptual differences between how the SRN and IAN models work, we turn now to the specific issue of the models' accounts of error data. In particular, we consider how each of the models accounts for the range of errors produced by patients with action disorganization syndrome (ADS; Humphreys & Forde, 1998; Schwartz et al., 1991, 1998) and ideational apraxia (De Renzi & Lucchelli, 1988; Rumiati et al., 2001), as well as recent empirical evidence relating to errors in neurologically healthy adults following interruption (Botvinick & Bylsma, 2005). This consideration raises some substantive difficulties for the SRN model, primarily because the implicit representation of schemas and the lack of any explicit representation of objects in that model mean that all errors are essentially capture errors.

The SRN Model

A major criticism of the SRN model lies in its inability to fit empirical findings. Although the model does capture several effects (e.g., the effect of relative task frequency on capture errors, the monotonic increase in independent actions with severity of damage, and the increase in omission errors with damage), close inspection of its behavior reveals several deficiencies in the empirical fits offered by the model.

1. The relative frequency of error types. We have already discussed how the SRN model has great difficulty in producing object substitution errors (see the section titled Linking Actions With Objects, above). It is also unclear how the model might account for specific anticipatory errors (e.g., attempting to pour without first opening a container; see the section titled *On the Importance of the Training Set,* above) or specific tool omission errors (e.g., using a finger instead of a utensil to stir a drink or spread butter). Equally as critically, although the SRN model predicts increasing rates of omission errors with severity, it tends to produce more omission errors than either controls or patients. Omission errors were indeed common in Schwartz et al.'s (1998) patients, making up 38% of errors, by contrast with only 3% of the errors of controls. However, Botvinick and Plaut (2004) reported that at noise of 0.2 (equivalent to a mild impairment on their criteria), 77% of their model's errors were omissions. Our reimplementation of the SRN model replicated this, although there was considerable variability resulting from the initial randomization of network weights. Table 2 shows the average proportion of omission errors produced by 10 instances of the SRN model at varying levels of noise. At low levels of noise, the model produced, on average, more than 10 times the proportion of omission errors produced by Schwartz et al.'s healthy control subjects, whereas, at high levels of noise, the omission rate is still at least double that observed in patient behavior. (See Simulation 1, *Analysis C*, in the Appendix for further details.)

One might argue that the differences in frequency of error types between the behavior of the SRN model and the results of the patient studies could be addressed through some minor modifica-

Table 2
Omission Errors as a Proportion of Total Errors Produced by the Simple Recurrent Network Model With the Standard Deviation of Noise Varying From 0.00 to 0.50

	Noise *SD*										
Error type/statistic	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
Total omission errors	0	383	925	1,581	2,175	2,730	3,285	3,835	4,354	4,810	5,126
Total errors	0	922	1,909	3,056	4,077	4,940	5,473	6,126	6,604	7,091	7,374
% omissions		41.5	48.5	51.7	53.3	55.3	60.0	62.6	65.9	67.8	69.5

tion of the SRN model. However, it is unclear what that modification might be. For example, augmenting the SRN with a thresholding mechanism such that an action is performed only when the SRN's output exceeds that threshold does not help. As is clear from Figure 4 (especially Step 11 of the coffee-making task), when the SRN model is run without noise, the activation of output units tends to be all or none. A threshold of less than 0.99 would have no effect on most selections. The only exception is when multiple outputs are possible, where the coffee-making task requires a threshold of at most 0.5 for normal functioning. In simulations that we do not report in detail here, combining a threshold with noise led to an even greater tendency toward omission errors and away from, for example, object substitution errors.

In fact, the root cause of many of these empirical difficulties is that all sequential action within the SRN model is guided by the sequential attractors that the model develops from its training set. Consequently, all errors result ultimately from the context units corresponding to a point in one sequential attractor drifting to a nearby point either at a different stage in the same sequential attractor or in another sequential attractor. They are essentially a form of capture error. Omissions or perseverations result if the drift is to some nonconsecutive point in the same sequential attractor. Explicit capture errors result if the drift is to another sequential attractor, but given the structure of Botvinick and Plaut's (2004) training corpus, such errors are hard to positively identify, and they may easily be misidentified as any type of error (including omission or perseveration).[5] Omissions and perseverations are common in the SRN simulations because there are just six attractors, and these attractors are so closely related that genuine capture errors appear more like omission or perseveration errors (e.g., coffee preparation being captured by tea preparation is equivalent to coffee preparation with the omission of the CREAM subtask). It is likely that the addition of more distinct tasks to the training set would result in fewer omission and perseveration errors but more clear-cut capture errors. Although the reduction in omission errors would be in line with empirical findings, the corresponding increase in capture errors would not. In any case, this analysis further demonstrates the sensitivity of the model's behavior to its training set.

2. The effect of the presence of distractor objects on error profiles. In a series of group studies, Schwartz and colleagues have demonstrated a reliable effect of the presence of distractor objects on the error profiles of ADS patients (Buxbaum et al., 1998; Schwartz et al., 1998, 1999). Closed head injury patients, left-hemisphere stroke patients, and right-hemisphere stroke patients all tended to produce more omission errors when distractor objects were present. Whether the SRN model could account for this effect given that it contains no separate representation of objects (see the section titled *The Influence of Distractor Objects,* above) is unclear. One possibility is that distractors could produce impairments in fixation behavior, but this, as well as its consequences for behavior, needs to be demonstrated. In contrast, noise within the schema network of the IAN model has recently been shown to yield the pattern of behavior observed in the various patient groups (Cooper et al., 2005).

3. Disorders affecting the rate of action and effects of the rate of action on error profiles. The SRN model selects one action on every processing cycle. It is therefore difficult to see how it can give an account of the action-related impairments of rate occurring in disorders such as Parkinson's disease or amphetamine psychosis. Similarly, it cannot account for any effects of the rate of action on error profiles as would be anticipated by extrapolating results for studies of speech production (e.g., Dell, Burger, & Svec, 1997) to the action domain (cf. Gupta & Dell, 1999; MacKay, 1985; Vousden & Brown, 1998). Although the IAN model has not been applied to modeling the latter effects, it has addressed empirical phenomena in both Parkinson's disease and amphetamine psychosis (Cooper & Shallice, 2000).

4. Dissociations between individual patients and between patient groups. The SRN model attempts to account for all action errors in essentially the same way: through corruption of the context representation by the addition of noise. Yet, even ignoring disorders of rate, there are distinct cognitive-level action-related disorders. Many, although not all, commentators, for example, have differentiated the generalized action disorganization found in patients with prefrontal and supplementary motor area lesions (cf. Duncan, 1986; Humphreys & Forde, 1998; Luria, 1966; Schwartz et al., 1991) on the one hand from ideational apraxia (De Renzi & Lucchelli, 1988; Rumiati et al., 2001) that may occur following damage to left temporoparietal regions on the other hand. It is, however, accepted that these deficits may have similar behavioral consequences on semicomplex everyday tasks (Buxbaum et al., 1998). Surprisingly, there is no clear quantitative study differentiating the properties of these types of patient. However, as discussed above, Rumiati et al. (2001) described two ideational apraxic patients with tendencies toward different types of error. Extrapolation from qualitative differences between the patterns of disorder exhibited by patients to qualitative differences between loci of impairment can be somewhat hazardous (see, e.g., Plaut, 1995; Shallice, 1988; but see Bullinaria, 2003). However, with only one source of error, it is unclear how the SRN model could lead to such a differentiation in hypothetical patient patterns. By contrast, as discussed in more detail earlier, Cooper (in press) showed how one pattern of symptoms fits with an impairment of the pathway from action schema representations to object representations, whereas the other would reflect an impairment to the reverse pathway.

5. Susceptibility to error following interruption. Botvinick and Plaut (2004) demonstrated that, in the SRN model, context information within the hidden units is more sensitive to noise within subtasks (e.g., within adding sugar) than between subtasks (e.g., between adding sugar and adding cream). On the basis of this and specific simulations, they predicted that interruptions that occur within subtasks are likely to result in more errors at the following subtask boundary than interruptions that occur between subtasks. This prediction has since been confirmed by empirical work (Botvinick & Bylsma, 2005) in which a coffee-making routine involving adding sugar and cream was interrupted at unpredictable points by a short subtraction task.

The IAN Model

Botvinick and Plaut (2004) made three criticisms of our account (Cooper & Shallice, 2000) of error data: that although ADS pa-

[5] See Simulation 3 in the Appendix for evidence in support of this interpretation of the SRN model's error behavior.

tients produce several types of error, it was necessary to vary different parameters in the model to simulate different error types; that the IAN model did not naturally produce one type of error—recurrent perseverations—without the addition of special mechanisms; and that the IAN model did not reproduce the finding reported by Schwartz et al. (1998) that more severely impaired patients tended to produce disproportionately more omission errors.

The use of different parameter variations in the simulation of different error types was an expository device we (Cooper & Shallice, 2000) used to clarify how the various error types might arise within the IAN model. In fact, only one parameter study was reported in detail: that involving the reduction in top-down excitation within the schema network coupled with a complementary increase in the bottom-up excitation within the network. This study was motivated by theoretical claims that ADS is a consequence of this form of dysfunction (Schwartz et al., 1991). The effects of increasing noise within the IAN model—the mechanism employed by Botvinick and Plaut (2004) to capture simultaneously the range of ADS error types—were not reported. However, more recent work has suggested that an imbalance in top-down and bottom-up excitation within the schema network is more consistent with another action selection disorder, utilization behavior (Boccardi, Della Sala, Motto, & Spinnler, 2002; Lhermitte, 1983; Shallice, Burgess, Schon, & Baxter, 1989), whereas increased noise within the schema network does indeed produce the full range of error types (Cooper et al., 2005). Furthermore, increased noise also leads the model to reproduce the relation between omission errors and severity reported by Schwartz et al. (1998), as well as an additional effect—the effect of distractor objects on patients' error profiles—that would appear problematic for the SRN model (as discussed in the preceding section).

The failure of the basic IAN model to exhibit recurrent perseverative errors is also not of major concern. Humphreys and Forde (1998) reported two patients with extensive frontal damage and behavior characteristic of ADS. The patients were comparable in terms of the severity of their action selection impairment, yet the perseverative errors of one patient were generally of the recurrent type, whereas those of the other patient were generally of the continuous type. Thus, the two types of perseverative error can dissociate. Sandson and Albert (1984) also suggested that the two types of perseveration result from different forms of neural damage. However, Shallice, Venable, and Rumiati (2005) argued that the relative immediacy of perseverative actions across patients lies on a continuum rather than forming a dichotomy. Thus, the empirical phenomena in this domain are not well established. However, it is likely that recurrent perseverative errors would be produced by the IAN model if the model's representations of achieved subgoals were to be corrupted. Therefore, these observations are consistent with the account offered of perseverative errors by the IAN model, and although they do not preclude a variety of other models, they would, as discussed above, appear to present a problem for any model (such as the SRN model) in which a single impairment necessarily leads to both forms of perseveration.

With regard to the specific difficulties relating to accounting for dissociations between patients and patient groups within the SRN model, it should be noted that within the IAN model, there are at least two sources of content errors—over schemas and over object representations. This allows us to suggest that different types of patient might have qualitatively different sorts of content errors, namely, concerning schemas and object arguments, respectively (Cooper, in press; Cooper et al., 2005). Similarly, as discussed above, the disorder of utilization behavior fits naturally with the account of the control of routine action offered by the IAN model.

The IAN model has not been applied to the Botvinick and Bylsma (2005) paradigm, and it remains to be demonstrated that the IAN model can match the success of the SRN model in this situation. However, contrary to Botvinick and Bylsma, the observed error pattern does not seem to be counterintuitive. If the interruption leads to insufficient information being available in the action system to allow the action to be completed, then what additional information is required to resume? In the within-subtask case, it is necessary to resume the current subtask and recall which subtasks from the overall task have been completed, as the subject is required to vary the order of subtasks from trial to trial. Only the second of these need be recalled in the between-subtasks case. Botvinick and Bylsma did not score errors on resumption of the current subtask, but errors are unlikely as it is generally possible to infer one's position in the subtask from the state of the task environment. Making such an inference could, however, potentially interfere with one's recollection of completed subtasks and hence result shortly thereafter in the kind of omission or perseveration errors observed by Botvinick and Bylsma.

The Place of Routine Action Systems in the Overall Architecture: The Interface With Higher Cognition

The neuroscience evidence strongly supports the existence of two systems in the acquisition of action sequences—a habit-learning system and a goal-directed system. In humans, these would correspond to the contention scheduling and the supervisory systems. Botvinick and Plaut (2004) emphasized that their model is intended as one of routine habitual action, but they accepted that nonroutine action probably requires additional mechanisms. Indeed, they appeared to accept the position of Norman and Shallice (1980, 1986) that behavior is the product of a routine system modulated or biased by one or more other systems when nonroutine behavior is required. However, Botvinick and Plaut provided no account of how other systems might interact with their proposed routine system. This is critical because, although it may be argued that much behavior consists of routines, those routines are generally assembled in nonroutine ways. Thus, processes such as error correction, inhibition of an inappropriate or undesirable behavior, and interleaving of behavioral routines all present a major issue for the model: How can the output of the proposed routine system be controlled or biased by other systems in these situations?

Within the SRN model, this excitatory or inhibitory biasing is problematic. Consider the case of error detection and correction. Clearly, mechanisms exist by which one may compare intention and effect (to use the phrasing of Luria, 1966). Without such mechanisms, Reason's (1979, 1984) self-report diary studies would have yielded no data, for his participants would not have been aware of their errors. Equally clearly, once a person detects an error in his or her actions, he or she is also normally able to correct that error. Thus, if, on preparing to shave, a person picks up deodorant instead of shaving foam, the person generally detects this before significant harm is done, puts down the deodorant, picks up the shaving foam, and resumes the shaving routine. How might one give an account of this fragment of behavior if routine action is controlled by a system such as that of Botvinick and Plaut (2004)? Accounting for the slip itself, through corruption of the context representation, is only a small part of the solution (although we have already discussed the difficulties that the SRN model has in producing object substitution errors). To detect the slip once it has occurred, some mechanism is required to compare actual behavior with expected behavior. Botvinick and Plaut provided no account of this, but the mechanism would seem to require some representation of the expected or intended behavior. One also requires a mechanism to plan and effect the repair. Planning might reasonably be relegated to some other system, but effecting the repair is problematic because it requires performing a simple but potentially novel subsequence of actions, which the SRN model cannot do. Finally, one needs a mechanism to resume the interrupted shaving sequence from the point at which the error arose. The SRN model provides no mechanism whereby a routine can be entered midway through. It would seem necessary either to resume the routine from scratch or to learn the tail ends of all routines from each and every step within the routine just in case they might be required for error correction.

These difficulties reflect a more general difference between models that use localist as opposed to distributed representations—a difference Page (2000) referred to as ease of manipulation. Botvinick and Plaut's (2004) eliminativist position limits the extent to which representations used by the routine system can be communicated to and manipulated by the nonroutine system. The simplest way in which the SRN model might be interfaced with higher level control follows from the observation that the SRN model effectively performs the actions of a schema if its hidden units are initialized to appropriate activation levels (and the fixated and held objects are also appropriate for the first step of the schema). Hence, one might envisage a system that maintains associations between higher level representations of schemas and the hidden unit patterns that result in those schemas being performed. A supervisory system could then interface with the SRN model to yield controlled behavior (when required) by deliberately instantiating the hidden units with the corresponding activations. This is clearly what Botvinick and Plaut had in mind when they suggested the addition of "a new group of units dedicated to representing desired states of the system or environment" (Botvinick & Plaut, 2004, p. 424; see also Botvinick, 2005). There is also a sense in which the instruction units already present in the SRN model do just this for the two basic tasks of preparing tea and preparing coffee. However, this approach to interfacing the routine system with a controlling system requires discrete (although not necessarily localist) representations of schemas elsewhere. Critically, it requires a one-to-one mapping between such representations and all schemas at all levels (i.e., for both complete tasks such as preparing coffee and subtasks such as adding sugar and cream) and from all possible starting states of the world (i.e., from starting states in which any object may be held or fixated). These additions would result in the SRN model mimicking the explicit hierarchical structuring of schemas and goals within the IAN model. It also does not address the issues of how the controlling system might inhibit an undesirable behavior or how the higher level system might monitor behavior and know either when a deliberately triggered schema has been completed or when an error has occurred.

Now, consider how the fragment of behavior described above might be accounted for within a system built around the IAN model. The error might arise because some corruption in the processing of the object representation network (modeled as noise) results in the representation of the deodorant container winning the competition within the source object representation network. Once the error has occurred, however, monitoring mechanisms that check the context-specific postconditions of a schema would detect the error. Note that these mechanisms make use of an explicit representation of expectations. In this case, monitoring mechanisms would detect a mismatch between the expected and actual states of the system and invoke appropriate error-correction mechanisms. The shaving schema could then be temporarily suspended while the correction is applied. Performance of the shaving schema would then be resumed from where it was suspended. The use of explicit goals, preconditions, and postconditions within the IAN model means that it is not necessary to store the entire state of the system while the repair is being carried out—all that needs to be stored is the high-level goal. Reactivating schemas for this goal on completion of the repair would not result in repetition of any completed subschemas (because their postconditions would be met) but would instead result in activation and, hence, selection of all appropriate remaining subschemas. Thus, although the interface with higher cognitive functions does not raise significant issues for the IAN model, it raises major issues for any model of routine action that avoids explicit representation of schemas or goals and more generally for Botvinick and Plaut's (2004) eliminativist position.

Conclusion

Our prime concern in this article has been to present a series of arguments for the explicit representation of schemas and goals within the system or systems responsible for the generation and control of routine sequential action. In doing this, we have contrasted our model (Cooper & Shallice, 2000) and that of Botvinick and Plaut (2004). To develop the argument, we have aimed to clarify the key differences between the approach of Botvinick and Plaut and our approach to routine sequential action, to demonstrate that Botvinick and Plaut's criticisms of the hierarchical interactive activation approach are not substantive, and, most critically, to provide a set of major problems that the recurrent network approach currently faces. We do not believe these difficulties are necessarily insurmountable hurdles to the basic SRN approach. However, in our view, they provide a set of daunting challenges for Botvinick and Plaut's eliminativist position with respect to schemas and goals.

The great attraction of the SRN model is that it replaces hand-coded specification of a model by gradual shaping of connection weights with a learning algorithm. However, the end product does not produce, when damaged, errors such as those produced by patients. In particular, it does not produce anticipation errors or object substitution errors at more than a minuscule rate, and it produces inappropriate rates of omission errors. These characteristics and others, such as its inflexibility in behavior, stem from a very basic flaw that is a natural consequence of its architecture. This is that it can only produce—even as errors—sequences of actions on which it has been substantively trained. As a consequence of this characteristic, for other critical aspects of its behavior, the training set has to be fine-tuned to produce the appropriate output. In other words, the negative property of being hand coded is merely transferred from the weights for the IAN model to the training set for the SRN model. Most critically, the attempt to do away with hierarchical control structures and goals fails. To avoid postulating hierarchical control structures for schemas while not suffering from catastrophic interference, the SRN model is forced to adopt a learning procedure—hippocampal training of the habit system—that is neuroscientifically and cognitively implausible. Moreover, the arguments presented against goals are weak, and the use of goals has been shown to have many advantages. Thus, goals allow functionally equivalent schemas to be interchanged and noncritical preparatory and tidying actions to be dispensed with when appropriate. In addition, studies of learning by imitation have suggested that goals greatly facilitate learning.

We have concentrated on differences between the IAN and the SRN models; there are also similarities between them, and progress seems most likely to come through the development of a hybrid system that builds upon both. To this end, it is relevant to note that, in both models, schemas, whether explicit and hand coded or implicit and emergent, play a key role in determining behavior and that, in both models, action results from the interaction of bottom-up inputs, schemas available to the system that are triggered by those inputs, and current activity in the system. Finally, both models are consistent with a dual-systems approach to the control of action—with one system for routine action and a second for nonroutine action and with the second operating by modulating or biasing the first. It can be argued that the conceptual interface, the bridge law, between the symbolic domain and the parallel distributed one is that between the attractor basin and the symbol in Newell's (1990) sense. In addition, Cooper (2003) has suggested that recurrent networks and interactive activation networks may be reconciled through the mapping of nodes within the interactive activation network to discrete point attractors (as opposed to sequential attractors) within the recurrent network. We therefore see the prime error of the Botvinick and Plaut (2004) framework to be the eliminativist position they have taken on implementation. If one rejects that perspective, then, we believe, one can be optimistic about the development of a model that functions at one level according to the principles of Botvinick and Plaut and at another according to our (Cooper & Shallice, 2000) principles.

There are clearly numerous ways in which the SRN model might be modified to address specific issues that we have raised. For example, in the section titled The Implementation of Choice, above, we have suggested that output unit activations might be interpreted as representing the frequencies of each action at each step. As discussed above, this has some advantages, but it does not address the issue of how intentional control might bias the system toward one or another method of, for example, adding sugar. Alternatively or in conjunction, one might attempt to train a recurrent network to settle to a point attractor state before selection of an action, with each point attractor corresponding to a different action. This approach would have the advantage of being able to make contact with data on the rate or timing of action. One might also explore how intentional control could be used to bias such a network toward or away from specific attractors. It is, however, not obvious how to produce a detailed implementation of such a scheme. A third possibility would be to augment the basic SRN with a bank of goal units that feed (together with the units representing the fixated and held objects) into the hidden layer. In such a model, the network would need to be trained with, in addition to its environmental inputs and motor outputs, a representation of its current hierarchy of goals and subgoals. When taken in conjunction with the point attractor approach, such nodes may be seen as providing a kind of activation gradient across such attractors, implementing Lashley's (1951) insight that multiple responses may be simultaneously activated, with competitive processes ensuring that only one is selected at any time. A fourth possibility is that one might attempt to develop an SRN model that uses a learning algorithm that does not presuppose a training set.

Working along the last of these lines, Ruh, Cooper, and Mareschal (2005) have demonstrated how an SRN embedded within an actor–critic architecture using reinforcement leaning (Sutton & Barto, 1998) can learn goal-directed multistep sequences. Reinforcement learning has two substantial advantages over standard back-propagation through time as described by Williams and Zipser (1995) and used by Botvinick and Plaut (2004) to train the SRN model.[6] First, reinforcement learning does not assume a training set. The network generates initially random sequences of actions, and learning occurs through positive or negative feedback reinforcing desirable behaviors and extinguishing undesirable ones. Of course, it is necessary to specify which behaviors are desirable and undesirable. However, a second advantage of reinforcement learning is that by giving positive feedback when a goal is achieved and using temporal difference learning (Sutton & Barto, 1998), action becomes goal directed. The network does not learn explicit sequences; rather, it learns to select actions that move it from whatever state it happens to be in toward a goal state. With sufficient exposure to possible input states, this gives the network an inbuilt sensitivity to the initial state of the world and an automatic error-recovery mechanism: Whatever the initial state of the world, the network selects actions that move it toward its goal, and if an error occurs (momentarily causing the network to move away from its goal), the network automatically resumes moving toward its goal on the next processing step. This approach is a significant departure from Botvinick and Plaut's SRN model, and it currently has significant limitations. The implementation of Ruh et al. can learn to achieve only one goal at a time, for example. At the same time, that implementation has no representation of its goal. It is likely that the extension to multiple goals requires such a representation and the integration of this representation with the reinforcement-learning mechanism.

None of these suggestions address the issue of the relation between object representations and the action selection system(s)—another aspect of the SRN model with which we have taken issue. Nevertheless, they demonstrate that the objections raised in this article are not objections to SRNs per se. As argued above, they are objections to Botvinick and Plaut's (2004) eliminativist metatheoretical position. Most critically, this article has attempted to demonstrate that even in a domain as loose as the organization of everyday routine action, one cannot simply dispense with units or discrete states representing action subroutines and goals.

[6] Both reinforcement learning as discussed here and standard back-propagation through time involve propagation of an error signal back through the network. The difference lies in the origin of the error signal and the information contained within it.

References

Agre, P. E. (1988). *The dynamic structure of everyday life* (Tech. Rep. No. 1085). Cambridge, MA: Massachusetts Institute of Technology, Artificial Intelligence Laboratory.

Agre, P. E., & Chapman, (1987). Pengi: An implementation of a theory of activity. In *Proceedings of the Sixth National Conference on Artificial Intelligence* (pp. 268–272). Menlo Park, CA: AAAI Press.

Alexander, M. P., Stuss, D. T., Shallice, T., Picton, T. W., & Gillingham, S. M. E. (2005). Impaired concentration due to frontal damage from two distinct lesion sites. *Neurology, 65,* 572–579.

Anderson, J. R., Bothwell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of mind. *Psychological Review, 111,* 1036–1060.

Aron, A. R., Monsell, S., Sahakian, B. J., & Robbins, T. W. (2004). A componential analysis of task-switching deficits associated with lesions of left and right frontal cortex. *Brain, 127,* 1561–1573.

Balleine, B. W., & Dickinson, A. (1998). Goal-directed instrumental action: Contingency and incentive learning and their cortical substrates. *Neuropharmacology, 37,* 407–419.

Bartlett, F. C. (1932). *Remembering.* Cambridge, England: Cambridge University Press.

Bobrow, D. G., & Norman, D. A. (1975). Some principles of memory schemata. In D. G. Bobrow & A. M. Collins (Eds.), *Representation and understanding: Studies in cognitive science* (pp. 131–149). New York: Academic Press.

Boccardi, E., Della Sala, S., Motto, C., & Spinnler, H. (2002). Utilisation behaviour consequent to bilateral SMA softening. *Cortex, 38,* 289–308.

Botvinick, M. (2005). Modeling routine sequential action with recurrent neural nets. In J. J. Bryson, T. J. Prescott, & A. K. Seth (Eds.), *Modeling natural action selection: Proceedings of an international workshop* (pp. 180–187). Edinburgh, Scotland: AISB Press.

Botvinick, M., & Bylsma, L. M. (2005). Distraction and action slips in an everyday task: Evidence for a dynamic representation of task context. *Psychonomic Bulletin & Review, 12,* 1011–1017.

Botvinick, M., & Plaut, D. C. (2002). Representing task context: Proposals based on a connectionist model of action. *Psychological Research, 66,* 298–311.

Botvinick, M., & Plaut, D. C. (2004). Doing without schema hierarchies: A recurrent connectionist approach to normal and impaired routine sequential action. *Psychological Review, 111,* 395–429.

Broadbent, D. (1985). A question of levels: Comment on McClelland and Rumelhart. *Journal of Experimental Psychology: General, 114,* 189–192.

Bullinaria, J. A. (2003). Dissociation in connectionist systems. *Cortex, 39,* 142–144.

Burgess, N., & Hitch, G. J. (1992). Toward a network model of the articulatory loop. *Journal of Memory and Language, 31,* 429–460.

Buxbaum, L. J., Schwartz, M. F., & Montgomery, M. W. (1998). Ideational apraxia and naturalistic action. *Cognitive Neuropsychology, 15,* 617–643.

Charniak, E., & McDermott, D. (1985). *Introduction to artificial intelligence.* Reading, MA: Addison-Wesley.

Chomsky, N. (1957). *Syntactic structures.* The Hague, the Netherlands: Mouton.

Chomsky, N. (1980). Rules and representations. *Behavioral and Brain Sciences, 3,* 1–61.

Chomsky, N., & Halle, M. (1968). *The sound pattern of English.* New York: Harper & Row.

Coltheart, M., Curtis, B., Atkins, P., & Haller, M. (1993). Models of reading aloud: Dual-route and parallel distributed processing approaches. *Psychological Review, 100,* 589–608.

Cooper, R. P. (2003). Mechanisms for the generation and regulation of sequential behaviour. *Philosophical Psychology, 16,* 389–416.

Cooper, R. P. (in press). Tool use and related errors in ideational apraxia: The quantitative simulation of patient error profiles. *Cortex.*

Cooper, R. P., & Glasspool, D. W. (2001). Learning action affordances and action schemas. In R. M. French & J. Sougné (Eds.), *Connectionist models of learning, development and evolution* (pp. 133–142). London: Springer-Verlag.

Cooper, R. P., Schwartz, M. F., Yule, P. G., & Shallice, T. (2005). The simulation of action disorganisation in complex activities of daily living. *Cognitive Neuropsychology, 22,* 959–1004.

Cooper, R. P., & Shallice, T. (2000). Contention scheduling and the control of routine activities. *Cognitive Neuropsychology, 17,* 297–338.

Cooper, R. P., Shallice, T., & Farringdon, J. (1995). Symbolic and continuous processes in the automatic selection of actions. In J. Hallam (Ed.), *Hybrid problems, hybrid solutions* (pp. 27–37). Amsterdam: IOS Press.

Dell, S. G., Burger, L. K., & Svec, W. R. (1997). Language production and serial order: A functional analysis and a model. *Psychological Review, 104,* 123–147.

De Renzi, E., & Lucchelli, F. (1988). Ideational apraxia. *Brain, 111,* 1173–1185.

Devan, B. D., & White, N. M. (1999). Parallel information processing in the dorsal striatum: Relation to hippocampal function. *Journal of Neuroscience, 19,* 2789–2798.

Dickinson, A. (1985). Actions and habits: The development of behavioural autonomy. *Proceedings of the Royal Society, 308*(B), 67–78.

Duncan, J. (1986). Disorganisation of behaviour after frontal lobe damage. *Cognitive Neuropsychology, 3,* 271–290.

Duncan, J. (1993). Selection of input and goal in the control of behaviour. In D. A. Baddeley & L. Weiskrantz (Eds.), *Attention: Selection, awareness and control—A tribute to Donald Broadbent* (pp. 53–71). Oxford, England: Oxford University Press.

Duncan, J. (2001). An adaptive coding model of neural function in prefrontal cortex. *Nature Reviews Neuroscience, 2,* 820–829.

Estes, W. K. (1972). An associative basis for coding and organization in memory. In A. W. Melton & E. Martin (Eds.), *Coding processes in human memory* (pp. 161–190). Washington, DC: Winston.

Evans, S. H. (1967). A brief statement of schema theory. *Psychonomic Science, 8,* 87–88.

Firby, R. J. (1987). An investigation into reactive planning in complex domains. In *Proceedings of the Sixth National Conference on Artificial Intelligence* (pp. 202–206). Menlo Park, CA: AAAI Press.

Fuster, J. M. (1989). *The prefrontal cortex.* New York: Raven Press.

Gat, E. (1998). On three layer architectures. In D. Kortenkamp, R. P. Bonnasso, & R. Murphy (Eds.), *Artificial intelligence and mobile robots: Case studies of successful robot systems* (pp. 195–210). Menlo Park, CA: AAAI Press.

Giovannetti, T., Libon, D. J., Buxbaum, L. J., & Schwartz, M. F. (2002). Naturalistic action impairments in dementia. *Neuropsychologia, 40,* 1220–1232.

Glasspool, D. W. (2005). The integration and control of behaviour: Insights from neuroscience and AI. In D. N. Davis (Ed.), *Visions of mind: Architectures for cognition and affect* (pp. 208–234). Hershey, PA: IDEA Group.

Grafman, J. (1995). Similarities and differences among current theories of prefrontal cortical functions. In J. Grafman, K. J. Holyoak, & F. Boller (Eds.), *Structure and functions of the human prefrontal cortex* (pp. 337–368). New York: New York Academy of Sciences.

Graybiel, A. (1998). The basal ganglia and chunking of action repertoires. *Neurobiology of Learning and Memory, 70,* 119–136.

Grossberg, S. (1986). The adaptive self-organization of serial order in behavior: Speech, language, and motor control. In E. C. Schwab & H. C. Nusbaum (Eds.), *Pattern recognition by humans and machines: Vol. 1. Speech perception* (pp. 187–294). New York: Academic Press.

Gupta, P., & Dell, G. S. (1999). The emergence of language from serial order and procedural memory. In B. MacWhiney (Ed.), *The emergence of language* (pp. 447–481). Mahwah, NJ: Erlbaum.

Hartley, T., & Houghton, G. (1996). A linguistically constrained model of short-term memory for nonwords. *Journal of Memory and Language, 35,* 1–31.

Hayhoe, M. (2000). Vision using routines: A functional account of vision. *Visual Cognition, 7,* 43–64.

Head, H. (1920). *Studies in neurology.* London: Hodder & Stoughton.

Henson, R. N. A. (1998). Short-term memory for serial order: The start-end model. *Cognitive Psychology, 36,* 73–137.

Hollerman, J. R., Tremblay, L., & Schultz, W. (2000). Involvement of basal ganglia and orbitofrontal cortex in goal-directed behavior. *Progress in Brain Research, 126,* 193–215.

Houghton, G. (1990). The problem of serial order: A neural network model of sequence learning and recall. In R. Dale, C. Mellish, & M. Zock (Eds.), *Current research in natural language generation* (pp. 287–319). London: Academic Press.

Houghton, G., Glasspool, D. W., & Shallice, T. (1994). Spelling and serial recall: Insights from a competitive queuing model. In G. D. A. Brown & N. C. Ellis (Eds.), *Handbook of spelling: Theory, process and intervention* (pp 365–404). Chichester, England: Wiley.

Houghton, G., & Hartley, T. (1995, February). Parallel models of serial behaviour: Lashley revisited. *PSYCHE: An Interdisciplinary Journal of Research on Consciousness, 2,* Article 25. Retrieved June 5, 2006, from http://psyche.cs.monash.edu.au/v2/psyche-2–25-houghton.html

Humphreys, G. W., & Forde, E. M. E. (1998). Disordered action schema and action disorganisation syndrome. *Cognitive Neuropsychology, 15,* 771–811.

James, W. (1890). *The principles of psychology.* New York: Holt.

Joe, W., Ferraro, M., & Schwartz, M. F. (2002). Sequencing and interleaving in routine action production. *Neurocase, 8,* 135–150.

Jueptner, M., Stephan, K. M., Frith, C. D., Brooks, D. J., Frackowiak, R. S. J., & Passingham, R. E. (1997). Anatomy of motor learning: 1. Frontal cortex and attention to action. *Journal of Neurophysiology, 77,* 1313–1324.

Knowlton, B. J., Mangels, J. A., & Squire, L. R. (1996, September 6). A neostriatal habit learning system in humans. *Science, 273,* 1399–1402.

Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). Soar: An architecture for general intelligence. *Artificial Intelligence, 33,* 1–64.

Land, M., Mennie, N., & Rustead, J. (1999). The roles of vision and eye movements in the control of activities of daily living. *Perception, 28,* 1311–1328.

Lashley, K. S., (1951). The problem of serial order in behaviour. In L. A. Jeffress (Ed.), *Cerebral mechanisms in behaviour* (pp. 112–136). New York: Wiley.

Lhermitte, F. (1983). Utilisation behaviour and its relation to lesions of the frontal lobes. *Brain, 106,* 237–255.

Luria, A. R. (1966). *Higher cortical functions in man.* London: Tavistock.

MacKay, D. G. (1985). A theory of the representation, organization and timing of action with implications for sequencing disorders. In E. A. Roy (Ed.), *Neuropsychological studies of apraxia and related disorders* (pp. 267–308). Amsterdam: North Holland.

Maes, P. (1989). How to do the right thing. *Connection Science, 1,* 291–323.

McClelland, J. L. (1992). Toward a theory of information processing in graded, random, interactive networks. In D. E. Meyer & S. Kornblum (Eds.), *Attention and performance XIV: Synergies in experimental psychology, artificial intelligence, and cognitive neuroscience* (pp 655–688). Cambridge, MA: MIT Press.

McClelland, J. L., McNaughton, B. L., & O'Reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and the neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review, 102,* 419–457.

McClelland, J. L., & Rumelhart, D. E. (1985). Distributed memory and the representation of general and specific information. *Journal of Experimental Psychology: General, 114,* 159–188.

McClelland, J. L., & Rumelhart, D. E. (Eds.). (1986). *Parallel distributed processing: Explorations in the microstructure of cognition* (Vol. 2). Cambridge, MA: MIT Press.

Meegan, D. V., & Tipper, S. P. (1998). Reaching into cluttered visual environments: Spatial and temporal influences of distracting objects. *Quarterly Journal of Experimental Psychology: Human Experimental Psychology, 51*(A), 225–249.

Miller, G. A., Galanter, E., & Pribram, K. H. (1960). *Plans and the structure of behavior.* New York: Holt, Rinehart & Winston.

Nadel, L., & Moscovitch, M. (1997). Memory consolidation, retrograde amnesia and the hippocampal complex. *Current Opinion in Neurobiology, 7,* 217–227.

Newell, A. (1990). *Unified theories of cognition.* Cambridge, MA: Harvard University Press.

Norman, D. A. (1981). Categorization of action slips. *Psychological Review, 88,* 1–14.

Norman, D. A., & Shallice, T. (1980). *Attention to action: Willed and automatic control of behavior* (Chip Rep. No. 99). San Diego: University of California.

Norman, D. A., & Shallice, T. (1986). Attention to action: Willed and automatic control of behavior. In R. Davidson, G. Schwartz, & D. Shapiro (Eds.), *Consciousness and self regulation: Advances in research and theory* (Vol. 4, pp. 1–18). New York: Plenum Press.

Page, M. P. A. (2000). Connectionist modelling in psychology: A localist manifesto. *Behavioral and Brain Sciences, 23,* 443–467.

Pinker, S., & Prince, A. (1988). On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. *Cognition, 28,* 73–193.

Plaut, D. C. (1995). Double dissociation without modularity: Evidence from connectionist neuropsychology. *Journal of Clinical and Experimental Neuropsychology, 17,* 291–321.

Plaut, D. C., McClelland, J. L., Seidenberg, M. S., & Patterson, K. E. (1996). Understanding normal and impaired word reading: Computational principles in quasi-regular domains. *Psychological Review, 103,* 56–115.

Poldrack, R. A., Prabakharan, V., Seger, C., & Gabrieli, J. D. E. (1999). Striatal activation during cognitive skill learning. *Neuropsychology, 13,* 564–574.

Pratt, J., & Abrams, R. A. (1994). Action-centered inhibition: Effects of distractors on movement planning and execution. *Human Movement Science, 13,* 245–254.

Reason, J. T. (1979). Actions not as planned: The price of automatization. In G. Underwood & R. Stevens (Eds.), *Aspects of consciousness* (pp. 67–89). London: Academic Press.

Reason, J. T. (1984). Lapses of attention in everyday life. In W. Parasuraman & R. Davies (Eds.), *Varieties of attention* (pp. 515–549). Orlando, FL: Academic Press.

Riddoch, M. J., Edwards, M. G., Humphreys, G. W., West, R., & Heafield, T. (1998). An experimental study of anarchic hand syndrome: Evidence that visual affordances direct action. *Cognitive Neuropsychology, 15,* 645–683.

Ruh, N., Cooper, R. P., & Mareschal, D. (2005). Routine action: Combining familiarity and goal orientedness. In J. J. Bryson, T. J. Prescott, & A. K. Seth (Eds.), *Modeling natural action selection: Proceedings of an international workshop* (pp. 174–179). Edinburgh, Scotland: AISB Press.

Rumelhart, D. E., & McClelland, J. L. (1985). Levels indeed! A response to Broadbent. *Journal of Experimental Psychology: General, 114,* 193–197.

Rumelhart, D. E., & McClelland, J. L. (Eds.). (1986). *Parallel distributed processing: Explorations in the microstructure of cognition* (Vol. 1). Cambridge, MA: MIT Press.

Rumelhart, D. E., & Norman, D. A. (1982). Simulating a skilled typist: A study of cognitive-motor performance. *Cognitive Science, 6,* 1–36.

Rumelhart, D. E., & Ortony, A. (1977). The representation of knowledge in memory. In R. C. Anderson, R. J. Spiro, & W. E. Montage (Eds.), *Schooling and the acquisition of knowledge* (pp. 99–135). Hillsdale, NJ: Erlbaum.

Rumelhart, D. E., Smolensky, P., McClelland, J. L., & Hinton, G. E. (1986). Schemata and sequential thought processes in PDP models. In J. L. McClelland & D. E. Rumelhart (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition* (Vol. 2, pp. 7–57). Cambridge, MA: MIT Press.

Rumiati, R. I., Zanini, S., Vorano, L., & Shallice, T. (2001). A form of ideational apraxia as a selective deficit of contention scheduling. *Cognitive Neuropsychology, 18,* 617–642.

Saffran, J. R. (2001). Words in a sea of sounds: The output of infant statistical learning. *Cognition, 81,* 149–169.

Sandson, J., & Albert, M. L. (1984). Varieties of perseveration. *Neuropsychologia, 6,* 715–732.

Schank, R. C., & Abelson, R. P. (1977). *Scripts, plans, goals and understanding.* Hillsdale, NJ: Erlbaum.

Schmidt, R. A. (1975). A schema theory of discrete motor skill learning. *Psychological Review, 82,* 225–260.

Schwartz, M. F., Buxbaum, L. J., Montgomery, M. W., Fitzpatrick-De Salme, E. J., Hart, T., Ferraro, M., et al. (1999). Naturalistic action production following right hemisphere stroke. *Neuropsychologia, 37,* 51–66.

Schwartz, M. F., Montgomery, M. W., Buxbaum, L. J., Lee, S. S., Carew, T. G., Coslett, H. B., et al. (1998). Naturalistic action impairment in closed head injury. *Neuropsychology, 12,* 13–28.

Schwartz, M. F., Reed, E. S., Montgomery, M. W., Palmer, C., & Mayer, N. H. (1991). The quantitative description of action disorganisation after brain damage: A case study. *Cognitive Neuropsychology, 8,* 381–414.

Shallice, T. (1988). *From neuropsychology to mental structure.* Cambridge, England: Cambridge University Press.

Shallice, T. (2004). The fractionation of supervisory control. In M. S. Gazzaniga (Ed.), *The cognitive neurosciences* (3rd ed., pp. 943–956). Cambridge, MA: MIT Press.

Shallice, T., & Burgess, P. W. (1991). Deficits in strategy application following frontal lobe damage in man. *Brain, 114,* 727–741.

Shallice, T., & Burgess, P. W. (1996). The domain of supervisory processes and temporal organization of behaviour. *Philosophical Transactions of the Royal Society of London, 351*(B), 1405–1412.

Shallice, T., Burgess, P. W., Schon, F., & Baxter, D. M. (1989). The origins of utilisation behaviour. *Brain, 112,* 1587–1598.

Shallice, T., Venable, N., & Rumiati, R. I. (2005). Dissociable distal and proximal motor components: Evidence from perseverative errors in three apraxic patients. *Cognitive Neuropsychology, 22,* 625–639.

Sutton, R., & Barto, A. G. (1998). *Reinforcement learning: An introduction.* Cambridge MA: MIT Press.

Tipper, S. P., Lortie, C., & Baylis, G. C. (1992). Selective reaching: Evidence for action-centered attention. *Journal of Experimental Psychology: Human Perception and Performance, 18,* 891–905.

Usher, M., & McClelland, J. L. (2001). On the time course of perceptual choice: The leaky competing accumulator model. *Psychological Review, 108,* 550–592.

Vousden, J., & Brown, G. D. A. (1998). To repeat or not to repeat: The time course of response suppression in sequential behaviour. In J. A. Bullinaria, D. W. Glasspool, & G. Houghton (Eds.), *Proceedings of the Fourth Neural Computation in Psychology Workshop* (pp. 301–315). London: Springer-Verlag.

Wijk, A. (1966). *Rules of pronunciation of the English language.* Oxford, England: Oxford University Press.

Williams, R. J., & Zipser, D. (1995). Gradient-based learning algorithms for recurrent networks and their computational complexity. In Y. Chauvin & D. E. Rumelhart (Eds.), *Back propagation: Theory, architectures and applications* (pp. 433–486). Hillsdale, NJ: Erlbaum.

Wohlschläger, A., Gattis, M., & Bekkering, H. (2003). Action generation and action perception in imitation: An instance of the ideomotor principle. *Philosophical Transactions of the Royal Society, 358,* 501–515.

Yin, H. H., Knowlton, B. J., & Balleine, B. W. (2004). Lesions of dorsolateral striatum preserve outcome expectancy but disrupt habit formation in instrumental learning. *European Journal of Neuroscience, 19,* 181–189.

Zorzi, M., Houghton, G., & Butterworth, B. (1998). The development of spelling–sound relationships in a model of phonological reading. *Language and Cognitive Processes, 13,* 337–371.

(*Appendix follows*)

Appendix

Simulation Studies

Simulation 1: Error Rates as a Function of Noise

Rationale and Method

Botvinick and Plaut (2004) argued that their simple recurrent network (SRN) model provides an account of action errors in both normal and impaired populations. In support of this, they cited several findings concerning error rates and types (e.g., the omission-rate effect: that assuming varying severity, the proportion of omission errors rises as the overall error rate rises) and demonstrated that their model reproduces these findings. However, we have argued that specific errors (e.g., specific forms of perseverative errors) occur rarely, if at all, because of the structure of the training set. In addition, as noted in the main body of the article, the quantitative results reported by Botvinick and Plaut raise some concerns. In Simulation 1, therefore, we sought to conduct a more thorough analysis of the SRN model's specific errors following the addition of noise to context units.

The model as described by Botvinick and Plaut (2004) was reimplemented from scratch in the C programming language. The reimplementation was kept as close to the published description as possible, using the same network architecture, the same featural representations for input and output, the same learning algorithm and parameters, and the same target sequences. To the best of our knowledge, the only difference between the original and our reimplementation concerns the one-step background examples included in the training set: Botvinick and Plaut included 267 examples, whereas we found and included 339. Our own simulation studies found, however, that inclusion or exclusion of the background sequences had little observable effect on the trained simulation's behavior. To verify the correctness of the reimplementation, we reproduced several of Botvinick and Plaut's key results, including their Tables 4 and 6 and their Figures 4, 7, 8, and 15.

Botvinick and Plaut (2004) simulated errors by training their model on the full set of sequences for 20,000 epochs and then introducing noise into the context units while testing. The reimplementation was therefore trained on all sequences and background examples for 20,000 epochs. To guard against chance effects associated with specific trained networks, we repeated this procedure with different random weight initializations to give 10 trained models. Each trained model was then run 100 times with the tea instruction set and 100 times with the coffee instruction set, with the standard deviation of noise [A1] ranging from 0.00 to 0.50 by increments of 0.05 (resulting in $10 \times 200 \times 11$ output sequences). The sequences generated were logged, yielding a corpus of 22,000 sequences. Four analyses of specific errors were performed on the corpus as described in the following subsections. All analyses used an automated analysis and scoring program that applied each action in sequence to a model of the environment, transforming the model with each action and recording errors of commission along the way. Errors of omission were then determined by comparing the final state of the environment model with the expected state given correct performance.

Analysis A: Specific Types of Perseverative and Omission Errors

A central argument of this article is that the specific errors produced by the SRN model are conditioned by the model's training set. Thus, recurrent perseveration of sugar adding is likely because the training set includes sequences in which adding sugar is followed by adding cream and other sequences in which adding cream is followed by adding sugar. Similarly, omission of sugaring within coffee making (but not tea making) is predicted because coffee making can, on different occasions, legitimately involve the subsequences of adding coffee grounds and then adding cream and of adding cream and then drinking. To demonstrate the model's propensity for specific perseverative and omission errors, we therefore tabulated the perseverative and omission errors in the action corpus described above. The results are shown in Table A1, where each cell indicates the frequency of the error in 1,000 trials with the tea instruction set and 1,000 trials with the coffee instruction set.

As can be seen from Table A1, when noise was low (standard deviation of 0.05), the error pattern was as predicted: Omissions consisted almost entirely of omission of milk or sugar, whereas the most common types of recurrent perseveration involved repeated attempts at sugaring and adding cream. Perseverative adding of cream was noticeably less frequent than perseverative adding of sugar. This may be attributed to the two idiosyncrasies of the task set-up: Sugar is used in both tasks, whereas cream is used only in coffee making, and there is only one source of cream but two sources of sugar. Once cream has been added, the cream carton is open and, hence, not in the state from which adding cream has been learned, whereas sugar can be added successfully once from the packet and once from the bowl.

As noise increased, the predicted error pattern broke down. Perseverative errors became more rare, whereas omissions became more frequent, and omission of the first subtask—adding coffee grounds or steeping the tea—occurred. This is because, as Botvinick and Plaut (2004) noted, at such levels of noise, the model's behavior includes within-subtask errors. If such an error occurs within the initial subtask and prevents the crux action of that subtask from being correctly performed, then the scoring program would count an omission error. (This is consistent with the scoring procedures used in patient studies.) Nevertheless, omission of the initial subtask was still far less common than omission of the sugaring or creaming subtasks.

Analysis B: The Relative Scarcity of Object Substitution Errors

Object substitution errors, where an incorrect object is used in a task-appropriate way, make up a significant proportion of errors in the behavior of both control subjects and action disorganization syndrome (ADS) patients. Thus, object substitution errors made up 17% of all errors produced by Schwartz et al.'s (1998) control subjects and 9% of all errors produced by their closed head injury (CHI) patients. Analysis B sought to determine if the SRN model would produce comparable proportions of object substitution errors with either low levels of noise (to simulate control subjects) or higher levels of noise (to simulate ADS patients). The object substitution errors that occurred in the corpus of action sequences produced by the model were therefore tabulated. The results are presented in Table A2, which shows the absolute number of all object substitution errors produced at different levels of noise. (Recall that each cell in the table corresponds to 1,000 attempts at each of the two tasks.)

As can be seen from Table A2, the SRN model produced few object substitution errors. In fact, when the standard deviation of noise was 0.10 or less (corresponding to Botvinick and Plaut's, 2004, simulation of slips by control subjects), only 9 of the model's 2,831 errors involved object substitution. This is far fewer than would be expected on the basis of Schwartz et al.'s (1998) control data (i.e., 17% of 2,831 = 481 object

[A1] Botvinick and Plaut (2004) claimed to consider the variance of noise ranging from 0.00 to 0.50. Our studies suggest that in fact, it is the standard deviation that ranges from 0.00 to 0.50 to yield the reported error patterns, not the variance. This is a technical point, and beyond accuracy of reporting, it has no bearing on the results.

Table A1
Specific (Recurrent) Perseverative and Omission Errors Produced by the Simple Recurrent Network Model With the Standard Deviation of Noise Varying From 0.00 to 0.50

	Noise *SD*										
Type of error	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
Perseverative adding of cream	0	13	28	31	40	37	31	29	21	13	13
Perseverative adding of sugar	0	160	168	157	115	111	90	77	46	49	32
Perseverative scooping	0	0	0	1	0	3	1	1	2	4	1
Perseverative sipping	0	0	16	56	94	99	74	71	55	69	41
Perseverative stirring	0	4	7	2	15	25	15	23	13	6	14
Perseverative tea steeping	0	0	1	2	0	2	7	6	6	4	6
Tea omitted when making tea	0	0	0	3	16	31	95	134	211	301	385
Coffee omitted when making coffee	0	0	0	10	34	77	162	267	344	428	511
Sugar not added	0	191	464	760	1,014	1,188	1,339	1,421	1,537	1,591	1,655
Cream not added when making coffee	0	183	367	503	582	655	720	767	814	853	868
Drink not drunk	0	5	11	44	75	110	224	255	502	646	800
X added but not stirred in	0	4	82	232	380	545	571	635	646	650	561
Only one sip	0	0	1	29	74	124	174	156	200	341	436

substitution errors). The SRN simulation did not produce a substantially better account of the data at higher levels of noise. Object substitution errors never constituted more than 2% of the model's errors, in contrast to the 9% observed by Schwartz et al. in their CHI patient group.

Analysis C: The Overpreponderance of Omission Errors

Omission errors are a key feature of the behavior of ADS patients. Schwartz et al. (1998) found they accounted for only 3% of the errors of their control subjects but 38% of the errors of their CHI subjects. Furthermore, omission errors were found to correlate with severity, being more frequent in the behavior of more severely impaired subjects. In contrast, Botvinick and Plaut (2004) reported that at noise of 0.2 (equivalent to a mild impairment on their criteria), 77% of their model's errors were omissions (p. 417). To investigate further the SRN model's tendency toward omission errors, we tabulated all omission errors in the corpus of action sequences. The results are shown in Table A3.

Note first that the model produced all logically possible omission errors and that although omitting sugar appears to have been the most frequent omission error, there were twice as many opportunities for that error as for omission of other ingredients (because, unlike the other ingredients, sugar could be omitted from both tea and coffee). Second, although Botvinick and Plaut's (2004) result of 77% omissions at noise of 0.20 does not appear to have been replicated, closer inspection reveals that the rate of omission errors varied greatly across the 10 trained networks, with some networks yielding much higher rates of omissions than others. Botvinick and Plaut's result is therefore supported, although the variance in behavior supports our use of 10 trained networks to gather results rather than a single trained network as used by Botvinick and Plaut. Regardless of this point, the simulations support Botvinick and Plaut's claim that increasing noise in the SRN model leads to an increased proportion of omission errors (which is consistent with patient data showing increased omission errors with increased severity). However, and third, the results indicate that the SRN model is overly prone to omission errors. Just 3% of control subjects' errors were omissions. This compares poorly with the figure of 42%–48% produced by the SRN model under low noise conditions (as used by Botvinick and Plaut, 2004, to simulate normal slips and lapses). At higher levels of noise, the problem persisted, with the SRN model producing over 60% omission errors but patients typically producing less than 40% of such errors.

Analysis D: The Scarcity of Specific Anticipation Errors

Anticipation errors form a significant subset of both normal action lapses and patient action errors. These errors consist of performing one action that is dependent on the outcome of a second action before actually performing the second action. The dependency between the actions may be used to distinguish this type of error from omission of the second action (M. F. Schwartz, personal communication, May 3, 2006). As noted in the main body of this article, a common anticipation error within tasks such as beverage preparation is attempting to pour from a container without first opening the container (e.g., De Renzi & Lucchelli, 1988; Schwartz et al.,

Table A2
Object Substitution Errors Produced by the Simple Recurrent Network Model With the Standard Deviation of Noise Varying From 0.00 to 0.50

	Noise *SD*										
Error type/statistic	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
Pouring *X* into the sugar bowl	0	0	7	11	19	22	17	18	26	24	21
Pouring *X* into the cream carton	0	0	0	0	0	2	1	0	0	5	2
Scooping from *X* (*X* not sugar bowl)	0	0	1	17	26	41	34	49	52	67	62
Steeping the teabag in the sugar bowl	0	0	0	0	0	0	0	1	0	0	2
Stirring *X* with the spoon (*X* not mug)	0	0	1	9	12	24	42	35	26	37	47
Total object substitution errors	0	0	9	37	57	89	94	103	104	133	134
Total errors	0	922	1,909	3,056	4,077	4,940	5,473	6,126	6,604	7,091	7,374
% object substitution errors		0.0	0.5	1.2	1.4	1.8	1.7	1.7	1.6	1.9	1.8

(*Appendix continues*)

Table A3
Omission Errors Produced by the Simple Recurrent Network Model With the Standard Deviation of Noise Varying From 0.00 to 0.50

Error type/statistic	Noise *SD* 0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
Coffee omitted while making coffee	0	0	0	10	34	77	162	267	344	428	511
Tea omitted when making tea	0	0	0	3	16	31	95	134	211	301	385
Cream omitted while making coffee	0	183	367	503	582	655	720	767	814	853	868
Sugar omitted	0	191	464	760	1,014	1,188	1,339	1,421	1,537	1,591	1,655
Drink omitted	0	5	11	44	75	110	224	355	502	646	800
Stir omitted after adding an ingredient	0	4	82	232	380	545	571	635	646	650	561
Only one sip (i.e., one sip omitted)	0	0	1	29	74	124	174	256	300	341	346
Total omission errors	0	383	925	1,581	2,175	2,730	3,285	3,835	4,354	4,810	5,126
Total errors	0	922	1,909	3,056	4,077	4,940	5,473	6,126	6,604	7,091	7,374
% omissions		41.5	48.5	51.7	53.3	55.3	60.0	62.6	65.9	67.8	69.5

1991), and this specific error is produced by the interactive activation (IAN) model. Theoretical considerations suggest, however, that production of this error by the SRN model would be unlikely. The corpus of action sequences was therefore analyzed for all possible anticipation errors. The results, summed over levels of noise, are shown in Table A4.

As can be seen from Table A4, the SRN model is indeed prone to producing certain types of anticipation error. Although published studies have reported figures only for sequence errors, of which anticipations are one type, the overall rate of anticipations is in line with that produced by patients. (Schwartz et al., 1998, reported that sequence errors made up 20% of all CHI patient errors.) However, in 22,000 trials of the SRN model, which produced 47,572 errors, none of those errors involved pouring without first opening. This is in spite of the fact that opportunities for such an error were provided by the coffee packet, the cream container, and both of the sugar containers. Instead, all anticipation errors were of one of three specific types: sipping while partway through preparing the beverage, stirring before adding an ingredient, and pouring from an empty spoon. There are no published data on the exact breakdown of anticipation errors for patient groups on specific tasks, but the fact that the SRN model failed to produce a commonly reported anticipation error raises further concerns about the model's ability to reproduce patient error patterns.

Simulation 2: Effects of Sequence Frequency in the Training Set

Rationale and Method

An impressive feature of the SRN model is its ability to reproduce, during testing, all sequences presented to the network during testing (see Botvinick & Plaut, 2004, Table 4). We have argued, however, that this feature is critically dependent upon the frequency of sequences in the training set and that the frequency of production of sequences is greatly reduced if they are even slightly underrepresented in the training set. Simulation 2 was therefore designed to explore the SRN model's ability to reproduce the sequences on which it was trained when the frequencies of sequence in the training set were unequal.

Two training sets with the smallest practicable degrees of bias between frequencies of sequences were constructed. Training Set 1 consisted of four copies of each item from Botvinick and Plaut's (2004) original training set, less two coffee-making sequences, both of the form GROUNDS → CREAM → SUGAR (PACK) → DRINK, one with and one without the coffee instruction unit set, and two tea-making sequences, both of the form TEABAG → SUGAR (BOWL) → DRINK, one with and one without the tea instruction unit set. Training Set 2 consisted of four copies of each item from Botvinick and Plaut's original training set, less two coffee-making sequences, both of the form GROUNDS → SUGAR (BOWL) → CREAM → DRINK, one with and one without the coffee instruction unit set, and two tea-making sequences, both of the form TEABAG → SUGAR (PACK) → DRINK, one with and one without the tea instruction unit set. Training Set 1 had a slight bias to adding sugar first when preparing coffee and, when adding sugar second, to do so from the bowl. It also had a slight bias toward using the sugar pack when making tea. Training Set 2 had the opposite biases.

A total of 50 instances of the model were then trained for 5,000 epochs, 25 with Training Set 1 and 25 with Training Set 2. Training for 5,000 epochs with the modified training sets was equivalent to training the model for 20,000 epochs with the original training set. The 50 trained models were then tested 100 times with the coffee instruction unit set and 100 times with the tea instruction unit. The sequence of actions produced by each model under each condition was recorded.

Results and Discussion

Table A5 shows the percentages of each sequence produced by the 50 replications of the trained model under the two experimental conditions. The table also shows, for each condition, the relative percentage of each sequence in the models' training history. The model generally performed flawlessly, producing one of the six sequences from training on each attempt at the task (although one instance of the model produced omission errors when attempting to make coffee following training on Training Set 1). However, the frequency of production of each sequence did not reflect the sequence's frequency within the training set. When making coffee after being trained with Training Set 1, the model showed a strong bias toward adding sugar before adding cream (in the ratio of at least 6:1, whereas the ratio of such sequences in the training set was 8:7), whereas an opposite (but equally strong) bias was shown following training with Training Set 2. At the same time, the model was unbiased in its selection of sugaring

Table A4
Anticipation Errors Produced by the Simple Recurrent Network Model When Instructed to Make Tea and Coffee (Summed Over All Levels of Noise With 10 Trained Networks and 100 Trials at Each Level)

Error type/statistic	Tea making	Coffee making
Anticipation errors		
Sipping while partway through beverage preparation	859	1,139
Sipping before adding an ingredient	1,779	507
Pouring from spoon before scooping	33	44
Other anticipations	0	0
Total anticipation errors	2,671	1,690
Total errors	19,618	27,954
% anticipation errors	13.6	6.0

Table A5
Frequency of Sequences Produced by the Simple Recurrent Network Model With Unbalanced Training Sets

Type of instruction and sequence	Training Set 1		Training Set 2	
	Relative % in training	% produced in testing	Relative % in training	% produced in testing
With coffee instruction				
GROUNDS → SUGAR (PACK) → CREAM → DRINK	26.7	45.0	20.7	2.1
GROUNDS → SUGAR (BOWL) → CREAM → DRINK	26.7	41.0	26.7	1.9
GROUNDS → CREAM → SUGAR (PACK) → DRINK	20.0	6.0	26.7	48.2
GROUNDS → CREAM → SUGAR (BOWL) → DRINK	26.7	6.4	26.7	47.8
Error	0.0	1.6	0.0	0.0
With tea instruction				
TEABAG → SUGAR (PACK) → DRINK	53.3	48.3	46.7	49.3
TEABAG → SUGAR (BOWL) → DRINK	46.7	51.7	53.3	50.7
Error	0.0	0.0	0.0	0.0

Note. Twenty-five models were trained (using different randomly initialized hidden units) using each of two training sets. In Training Set 1, there was a slight bias when making coffee away from adding sugar from the pack after adding cream and when making tea away from adding sugar from the bowl, whereas in Training Set 2, the biases were reversed. Each instruction condition was tested 100 times with each trained model.

subsequence across tasks, selecting equally from the packet or the bowl, even though in training with Training Set 1, the bowl was preferred when making coffee (in the ratio of 8:7), and the packet was preferred when making tea (again in the ratio of 8:7), whereas the opposite bias was present in Training Set 2.

The strength of bias in coffee making toward adding sugar first following Training Set 1 and adding it second following Training Set 2 arose from the fact that the action selected at any step within the SRN model is simply that whose output unit is most active. The critical step in coffee making occurs after the grounds have been stirred into the water and when the system must fixate either on the cream container or on one of the sugar containers. When the training set is balanced, both actions are activated approximately equally, with any difference in activation being attributable to differences in initial excitation within the hidden units. However, when the training set is unbalanced, the most frequent action normally dominates. It is therefore selected on most occasions. These simulations demonstrate that the domination of the most frequent action is not directly proportional to the frequency of occurrence in the training set. Rather, imbalances in the training set are magnified in the testing phase. More critically, larger imbalances can result in less frequently trained sequences becoming inaccessible. Thus, when training was repeated with greater imbalances in the training set (altering the ratios of the above sequences from 8:7 to 8:6), the retrained model failed on testing to generate any instances of the lower frequency coffee-making sequence.

The lack of sensitivity in either task to the frequency of different sugaring methods is also problematic. As noted in the main body of the article, in the section titled The Implementation of Choice, above, the method of sugar addition is selected not by the SRN model but by the nondeterministic action *fixate sugar*. In Botvinick and Plaut's (2004) implementation, this action selected either the sugar packet or the sugar bowl (with equal probability). Thus, this aspect of the model was not sensitive to the frequency of different types of sugar adding in the training set. Conceivably, the SRN model could be modified so that the outcome of nondeterministic actions would be sensitive to the frequency of outcomes in the training set; however, this would not solve the problem in this case as, across tasks, the absolute frequency of each sugaring method is equal: Sugar from the bowl is preferred when making coffee, but sugar from the pack is preferred when making tea. Thus, although the behavior of the SRN model is generally sensitive to the frequency of sequences within the training set, this is not true when it comes to the implementation of choice. Botvinick and Plaut's implementation of choice appears not to have been able to capture the kind of frequency effects explored here.

Simulation 3: Failure to Generalize Interchangeable Methods

Rationale and Method

We have argued that one difficulty with the selection of the training set for the SRN model is that the model needs to be trained on all legitimate sequence orders: It cannot spontaneously transfer interchangeable methods from one situation to another. To support this argument, Simulations 3A to 3F involved training the replication of the SRN model for 20,000 epochs with a training set consisting of the full set of one-step background examples, eight coffee-preparation sequences (four with and four without the coffee instruction unit activated) and eight tea-preparation sequences (four with and four without the tea instruction unit activated). The difference between this and the Botvinick and Plaut (2004) simulations is that in each of these simulations, one version of a task was omitted and replaced with a duplicate involving the alternative form of adding sugar. Thus, in Simulation 3A, each copy of COFFEE → SUGAR (BOWL) → CREAM → DRINK was replaced with a copy of COFFEE → SUGAR (PACK) → CREAM → DRINK (so this sequence was represented in the training set twice with and twice without the instruction unit), whereas in Simulation 3B, each copy of COFFEE → SUGAR (PACK) → CREAM → DRINK was replaced with another copy of COFFEE → SUGAR (BOWL) → CREAM → DRINK, and so on. The question of interest was whether the sequence omitted in training would occur in testing. Occurrence of the omitted sequence would support successful transfer of the sugaring method to a situation in which the model had not been trained. To ensure representative results, each simulation was once again performed 10 times with different randomly initialized networks. Each trained network was tested 100 times with the coffee instruction unit set, 100 times with the tea instruction set, and 100 times with no instruction unit set.

Results

Table A6 shows the percentage of runs yielding each sequence for each of the six simulations. Because of space limitations, results when no instruction unit was set are not shown. In all cases, however, they were parallel to the presented results.

In Simulation 3A, 34.1% of trials with the coffee instruction unit set resulted in the GROUNDS → SUGAR (PACK) → CREAM → DRINK sequence, whereas none resulted in the GROUNDS → SUGAR (BOWL) → CREAM → DRINK sequence. However, 33.5% of trials yielded errors, and in all cases, those errors involved omitting the CREAM subsequence from a trial involv-

(*Appendix continues*)

Table A6
Percentage of Each Sequence Produced by the Model as a Function of the Training Set

Type of instruction and sequence	Simulation 3A	3B	3C	3D	3E	3F
With coffee instruction						
GROUNDS → SUGAR (PACK) → CREAM → DRINK	34.1	0.0	23.9	29.4	20.2	29.8
GROUNDS → SUGAR (BOWL) → CREAM → DRINK	0.0	33.3	27.9	26.8	18.9	31.1
GROUNDS → CREAM → SUGAR (PACK) → DRINK	15.6	17.0	22.5	21.2	31.3	19.4
GROUNDS → CREAM → SUGAR (BOWL) → DRINK	16.8	17.2	25.7	22.6	29.6	19.7
Error: GROUNDS → SUGAR (BOWL) → DRINK	33.5	0.0	0.0	0.0	0.0	0.0
Error: GROUNDS → SUGAR (PACK) → DRINK	0.0	32.5	0.0	0.0	0.0	0.0
Error: Other	0.0	0.0	0.0	0.0	0.0	0.0
With tea instruction						
TEABAG → SUGAR (PACK) → DRINK	50.6	51.1	48.8	48.7	51.5	0.0
TEABAG → SUGAR (BOWL) → DRINK	49.4	48.9	51.2	51.3	0.0	48.1
Error: TEABAG → SUGAR (BOWL) → CREAM → DRINK	0.0	0.0	0.0	0.0	48.5	0.0
Error: TEABAG → SUGAR (PACK) → CREAM → DRINK	0.0	0.0	0.0	0.0	0.0	42.3
Error: TEABAG → SUGAR (PACK) → CREAM → SUGAR (X) → DRINK	0.0	0.0	0.0	0.0	0.0	9.6
Error: Other	0.0	0.0	0.0	0.0	0.0	0.0

Note. In Simulation 3A, each copy of COFFEE → SUGAR (BOWL) → CREAM → DRINK in the training set was replaced with a copy of COFFEE → SUGAR (PACK) → CREAM → DRINK. In Simulation 3B, each copy of COFFEE → SUGAR (PACK) → CREAM → DRINK was replaced with a copy of COFFEE → SUGAR (BOWL) → CREAM → DRINK. In Simulation 3C, each copy of COFFEE → CREAM → SUGAR (BOWL) → DRINK was replaced with a copy of COFFEE → CREAM → SUGAR (PACK) → DRINK. In Simulation 3D, each copy of COFFEE → CREAM → SUGAR (PACK) → DRINK was replaced with a copy of COFFEE → CREAM → SUGAR (BOWL) → DRINK. In Simulation 3E, each copy of TEA → SUGAR (BOWL) → DRINK was replaced with a copy of TEA → SUGAR (PACK) → DRINK. In Simulation 3F, each copy of TEA → SUGAR (PACK) → DRINK was replaced with a copy of TEA → SUGAR (BOWL) → DRINK.

ing the sugar bowl. The remaining coffee trials involved adding cream before sugar, either from the packet or the bowl. When the tea instruction unit was set, all trials were completed successfully, with approximately half involving the sugar packet and the other half involving the sugar bowl. Thus, Simulation 3A demonstrates that when the SRN model was trained on using either sugar source when making tea but only the sugar packet when making coffee, it was not able to transfer its knowledge of use of the sugar bowl to coffee making: When the first step of the sugaring subsequence selected the bowl instead of the packet, the model's behavior appeared to be captured by tea making, and it omitted the CREAM subsequence regardless of the instruction. The opposite situation, of transfer of use of the sugar packet from tea making to coffee making, also failed, as is shown by the results from Simulation 3B. Once again, the error consisted of omitting the CREAM subsequence, and once again, this suggests that behavior had been captured by tea making.

In Simulations 3C and 3D, those coffee-preparation sequences consisting of SUGAR after CREAM appear to provide positive evidence of transfer. These simulations yielded no errors, with the omitted sequence being generated by the model at frequencies similar to the other sequences actually in the training set. However, this special case is precisely what would be expected of capture of the coffee sequence by tea making during the SUGAR subsequence. As SUGAR is always followed by DRINK in the tea-making sequence and as, at the time of the capture error, the CREAM subsequence of coffee making has already been successfully completed, capture of coffee making by tea making results in an apparently valid coffee-making sequence.

This interpretation of apparently successful transfer as capture is supported by Simulations 3E and 3F. These simulations again yielded no instances of successful transfer, but the situations in which transfer might have occurred yielded sequences containing intrusion errors: addition of the CREAM subsequence to tea making. The CREAM subsequence occurred only in training in coffee making. Furthermore, in at least some simulations (Simulation 3F), the intrusion of CREAM was followed on some occasions by addition of another SUGAR (from either source) before DRINK. In these cases, the tea-making sequence therefore appears to have been captured by one of the coffee-making sequences that occurred in training.

Discussion

This set of simulations demonstrates comprehensively that the SRN model cannot spontaneously transfer equivalent subsequences. In one sense, the task should not have been difficult as, during testing, it was the environment that dictated whether the crucial first step of sugaring resulted in fixating on the sugar bowl, which should result in the SUGAR (BOWL) subsequence, or on the sugar packet, which should lead to the SUGAR (PACK) subsequence. The model had had exposure to both subsequences and so should have had no difficulty in responding to the cue supplied by the environment (i.e., the result of fixation) and performing the appropriate subsequence. Indeed, this is precisely what happened. However, the situation was complicated by the model's need to maintain task context information (i.e., whether it was making tea or coffee and, in both cases, which if any ingredients had been added) during the subsequence so that it could return to the appropriate point in the superordinate task. As the model had no way of discriminating task context information from subtask context information, it could not spontaneously preserve task context information. In summary, these simulations show that the SRN model had not learned to generalize its subtask knowledge and transfer it to another task. Rather, it had simply drifted from the intended sequence to an unintended sequence, and in the model as it stands, there is no way to prevent this short of training the model on all variant sequences. In contrast, the explicit representation of goals within the IAN model provides a level of abstraction that automatically embodies the appropriate generalization.

Simulation 4: Variation of Initial Conditions

Rationale and Method

This set of simulations aimed to explore the ability of the SRN model to generalize its learning to different initial states of the environment. In Simulation 4A, the network was trained in the usual way with all coffee-making and tea-making sequences and all one-step background examples, but in testing, the initial state of the sugar bowl was set as being open instead of closed. In Simulation 4B, the network's training set was altered such that the sugar bowl was initially open for the tea-making sequences and closed for the coffee-making sequences. Training then proceeded in the usual way for 20,000 epochs. Thus, the model was exposed to equal numbers of sugaring subsequences involving an open sugar bowl and sugaring subsequences involving a closed sugar bowl, but each was seen only in a specific context. The question was whether the model could apply the correct sugaring subsequence out of context. The model was therefore tested by setting an instruction unit (tea or coffee) but initializing the state of the sugar bowl to the opposite of that on which the network had been trained (i.e., closed for tea making and open for coffee making). Performance on tasks involving the sugar bowl was then examined. Finally, Simulation 4C was aimed at demonstrating that the SRN model could in principle learn the desired behavior (i.e., dealing appropriately with the sugar bowl regardless of its state or of the task) if it were trained on the full set of possible subsequences. Thus, the model was trained with a training set in which each sequence involving the sugar bowl occurred in two forms: with the sugar bowl closed and with the sugar bowl open. To accommodate this in a balanced way, the training set was doubled so as to also include two occurrences of each sequence involving the sugar packet. Training still consisted of 20,000 epochs. In all three cases, 10 instances of the model were trained, and each trained model was then tested 100 times in each of the three instruction conditions (with the coffee instruction unit set, the tea instruction unit set, and no instruction unit set) with the sugar bowl either initially closed or initially open.

Results and Discussion

The average number of sequences of each type involving the sugar bowl produced by the trained models with an instruction unit set is shown in Table A7. Sequences involving the sugar pack are not shown as they are not relevant to the issue of transfer as considered here. For brevity, sequences generated when no instruction unit was set are also excluded.

The results of Simulation 4A (see Table A7, left data columns) demonstrate that the model could not spontaneously generalize to the task with an open sugar bowl. This is despite the fact that during training on both the coffee- and tea-making tasks, the model was exposed to sequences in which the open sugar bowl was used. In training, however, these sequences always involved first opening the sugar bowl. The model therefore could not recognize that when holding an empty spoon and fixating on the open sugar bowl, the appropriate action was to scoop sugar. In fact, on some occasions (6.8% for coffee and 2.2% for tea), the trained model completely ignored the state of the sugar bowl and proceeded to perform the actions of opening it even though it was already open.

Lack of transfer is also demonstrated in Simulation 4B (see Table A7, center data columns). Here, the model failed to transfer use of the open sugar bowl to the coffee-preparation task, with all attempts ending in error. The reverse case—transfer of the use of the closed sugar bowl from the coffee task to the tea task—appears to have been successful on at least some trials (with 19.6% of tea attempts proceeding correctly when the sugar bowl was closed), but this is misleading. The tea and coffee tasks may both end with the same sugaring/drinking subsequences. The apparent transfer once again reflects capture of the model's behavior by a valid coffee-making sequence, rather than transfer of the sugaring subtask.

That the model was capable in principle of using the state of the sugar bowl to employ appropriate actions is demonstrated by Simulation 4C (see Table A7, right data columns). When trained with the full set of possible sequences, the model made no errors in detecting which sugaring subsequence should be applied when.

Simulation 5: Catastrophic Interference

Rationale and Method

The goal of Simulation 5 was to determine whether the SRN model would suffer from catastrophic interference if it were trained on related tasks in succession, rather than in parallel. Botvinick and Plaut (2004) trained their implementation of the SRN model on all versions of the coffee and tea tasks on each epoch. This is ecologically implausible. Learning of everyday routine tasks is more likely to involve practice of different tasks at different times, with new tasks often being learned after old ones are mastered and without the learning of such new tasks causing significant impairment in performance of previously mastered tasks.

To test whether the SRN model was susceptible to catastrophic interference, we constructed two training sets. The first training set—Training Set 1—comprised all one-step background sequences and two instances of each version of the coffee-preparation task (one with and one without the coffee instruction unit initially set). The second training set—Training Set 2—comprised all one-step background sequences and four instances of each version of the tea-preparation task (two with and two without the tea instruction unit initially set). In Simulation 5A, the model was trained to criterion on Training Set 1, where criterion was defined as correctly

Table A7
Mean Percentage of Sequences Produced by the Model as a Function of the Sugar Bowl's State and the Training Set

	Simulation					
	4A		4B		4C	
Type of instruction and sequence	Bowl closed	Bowl open	Bowl closed	Bowl open	Bowl closed	Bowl open
With coffee instruction						
GROUNDS → SUGAR BOWL (CLOSED) → CREAM → DRINK	23.6	0.0	26.8	0.0	43.6	0.0
GROUNDS → SUGAR BOWL (OPEN) → CREAM → DRINK	0.0	0.0	0.0	0.0	0.0	44.2
GROUNDS → CREAM → SUGAR BOWL (CLOSED) → DRINK	24.0	6.8	25.4	0.0	6.0	0.0
GROUNDS → CREAM → SUGAR BOWL (OPEN) → DRINK	0.0	0.0	0.0	0.0	0.0	8.8
Other	0.0	43.6	0.0	51.0	0.0	0.0
With tea instruction						
TEABAG → SUGAR BOWL (CLOSED) → DRINK	50.6	2.2	19.6	0.0	48.0	0.0
TEABAG → SUGAR BOWL (OPEN) → DRINK	0.0	0.0	9.6	49.6	0.0	49.8
Other	0.0	49.0	21.8	0.0	0.0	0.0

Note. In Simulation 4A, the training regime of Botvinick and Plaut (2004) was used. Simulation 4B used a training regime in which the sugar bowl was initially open for tea making but was closed for coffee making. Simulation 4C used a training regime including open and closed sugar bowls in both instruction conditions.

(*Appendix continues*)

performing the coffee task on 100 out of 100 attempts. The training set was then switched, and the trained model was trained with Training Set 2 until criterion, which now involved correct performance of the tea task on 100 out of 100 attempts. The model was then tested on the coffee task. If it failed, it was trained to criterion on that task (Training Set 1) and then tested on the tea task. If it then failed the tea task, it was trained to criterion on the tea task (Training Set 2) and then tested on the coffee task. The cycle of switching criteria and training sets was repeated for either 20,000 training cycles or until the model had learned to perform both tasks. Simulation 5B was a replication of Simulation 5A with the tea and coffee tasks reversed. That is, the model was trained first with Training Set 2, then with Training Set 1, and so on. In both simulations, each training epoch was followed by a testing cycle in which the network was run on 100 occasions with the tea instruction initially set and 100 occasions with the coffee instruction initially set. The numbers of correct tea and coffee sequences generated in each case were recorded.

Results and Discussion

The dependent variable of interest was the number of correct trials (out of 100) for each task during training. This is shown for Simulation 5A in Figure 5 (in the main body of the article, above) for 500 epochs at a time well beyond initial task acquisition. From the graph, it is clear that training with a second task impaired previous learning on the first task. The model's expertise switched between the two tasks, with performance on one task falling to zero when training on the other task began. Similar behavior was observed when tea preparation was the first task trained (i.e., Simulation 5B), and when replications were attempted, it was found that although the point at which the first task was successfully acquired varied from network to network, the basic effect of catastrophic interference between tasks was consistently replicated. However, relearning a task after switching required less training on each successive attempt (presumably because there was substantial overlap between the tasks). Thus, with prolonged training and switching between tasks, networks were able to acquire both tasks.

Received July 19, 2005
Revision received January 5, 2006
Accepted January 6, 2006 ■