The Hierarchies and Systems that Underlie Routine Behavior: Evidence from an Experiment in Virtual Gardening

Nicolas Ruh (nruh@brookes.ac.uk),
Oxford Brookes University, Department of Psychology
Gipsy Lane, Oxford OX3 0BP

Richard P. Cooper (R.Cooper@bbk.ac.uk)
School of Psychology, Birkbeck, University of London
Malet Street, London, WC1E 7HX, UK.

Denis Mareschal (D.Mareschal@bbk.uk)
School of Psychology, Birkbeck, University of London
Malet Street, London, WC1E 7HX, UK.

Abstract
Previous behavioral research into the processing mechanisms that support action selection in complex sequential tasks has focused on errors or ‘slips of action’ (Norman, 1981; Reason, 1979, 1984). This work has been taken to suggest that sequential behavior is controlled by the dynamic interaction of two systems – a resource hungry “supervisory” system and an automatic “routine” system that employs hierarchically structured task representations (Norman & Shallice, 1986). To counter some of the shortcomings of purely error-based natural history studies we developed an alternative methodology which enables more controlled, direct and fine-grained investigation of the underlying processes, namely the use of action selection latencies within novel, computer-based, tasks. While the methodology is relatively general, the primary aim of the specific experiment reported here was to test the hypotheses that routine behavior is governed by (a) hierarchically structured task representations, and (b) the dynamic interaction of two systems. The experiment was additionally designed to investigate the impact of several factors on the processing difficulty of specific action selection steps, including experience, number of choices, availability of external disambiguation cues, and minor variations within subsequences. Our results provide some direct support for the Dual Systems framework, but also pose challenges with regard to further refinements and computational models.

Keywords: sequential routine action, control, supervisory attention, contention scheduling, action selection

Introduction
The vast majority of human everyday behavior is sequential in that it requires the production of sequences of simple actions through time. Generally it is the order and identity of these basic acts that determines a successful outcome, with success being defined as the achievement of an underlying purpose or goal. However, not all actions within a sequence are equal – evidence from a variety of sources suggests that some actions are more vulnerable to error than others.

One prominent account of action control that attempts to account for error is the Dual Systems framework (Norman & Shallice, 1986), which suggests that two intricately interacting systems are responsible for the flexible, yet efficient control of sequential behavior. The supervisory attention system (SS), commonly localized in prefrontal areas, is the origin of deliberative, higher level control signals (‘vertical threads’) that may modulate the more basic, parietal contention scheduling system (CS) in the process of actually selecting and executing an action. CS is held to be capable of governing even relatively long sequences of actions autonomously, provided they are highly overlearned and not perceived as dangerous.

A second important claim within the framework is that such action sequences or “routines” are generated from underlying hierarchically organized representations, or “schemas”. The observable consequence of this is that the links between individual actions within a sequence are not all equal in strength. That is, similar to the words in a sentence, some actions belong more tightly together than others (e.g. the words within a phrase or expression), always occurring in the same context and order, while the transition to the next action might be much more variable at other, so called, ‘branching points’ (or phrase boundaries, in syntax).

In general, the sequencing mechanism implemented within CS is assumed to enforce the continuation of the currently performed sequence or routine, but the strength of these ‘horizontal threads’ may vary. Upon encountering a branching point, for example, it might be more difficult to ensure the correct continuation due to interference from other possible successor actions or branches. It is at these branching points that errors will most frequently be observed (Norman, 1981; Reason, 1979, 1984) or, seen the other way around, it is for these potentially problematic transitions that an additional, modulatory top-down signal from the SS will be most needed and effective in ensuring error free performance.
A third aspect of the framework acknowledges the sometimes decisive bottom-up influence of environmental triggers on the selection of action, exemplified most clearly by neurological disorders such as the anarchic hand syndrome (Della Sala, Marchetti, & Spinnler, 1991) or utilization behavior (Shallice, Burgess, Schon, & Baxter, 1989). While an inability to voluntarily override the behaviors induced by the affordances of a perceived object can produce unfortunate results, it is important to realize that the sensitivity to such bottom-up influences from the perceived environment is beneficial in the majority of cases. It is, for example, quite practical when a perceived bend in the street induces the corresponding steering response while driving, because one does not have to remember and/or consciously decide to do so.

In summary, the Dual Systems framework (Norman & Shallice, 1986) makes three important assumptions about the control of complex sequential behaviors. Namely it postulates that (a) sequential behavior is hierarchically organized, (b) behavior arises from the interplay of a deliberate control system (SS) and a more basic routine system (CS), and (c) environmental triggers may have an influence on the selection of an action within CS.

Notwithstanding the popularity of the Dual Systems framework it has to be acknowledged that it stands on a somewhat limited empirical basis. Data with regard to the normal population comes almost exclusively from diary studies (Reason, 1979, 1984) or anecdotal collections (Norman, 1981), and, by virtue of the sole dependent measure of erroneous actions, is necessarily indirect with regard to the usually correct performance in highly overlearned, if complex, tasks. Further complications arise with respect to the interpretability and comparability of erroneous actions in the context of non-standardized everyday behavior, which may involve routinized behavior interspersed with sporadic control. For example, do slips of action reflect breakdown at the level of CS, SS, or the interaction of the two?

In the following we present an experiment within a new paradigm that allows us to test the three assumptions of the Dual Systems framework in a more direct way. The paradigm involves normal participants performing (and learning to perform) complex sequential tasks with graphical objects on a computer screen. This has the double advantage of increasing the objectivity and interpretability of the error measure, as well as giving access to the online measure of action selection latencies. Specifically the action latencies, it is argued, provide a more fine grained measure of the processing difficulties at individual action selection steps. This detailed online information on the relative difficulty of choosing the next action in a sequence furthermore allows us to trace the impact and interaction of various factors throughout the course of increasingly routinized performance of the tasks. We interpret our results in the light of the Dual Systems framework and recent computational modelling work (Botvinick & Plaut, 2004; Cooper & Shallice, 2000; Ruh, 2007) that attempts to capture the interaction of such factors in the control and routinization of complex sequential activities.

### Method

A set of artificial hierarchically structured tasks was designed for the experiment. Each task required the production of one of three different kinds of fertilizer for an artificial plant which, when treated with the correctly assembled concoction, would respond by either growing larger, getting bushier or producing flowers. The task was computer controlled, with fertilizers being assembled by manipulating pictorial objects (ingredients and tools) on screen via drag and drop with a standard computer mouse. Figure 1 shows a typical screen set-up.

Table 1: Task variants. Capitals denote which of seven possible ingredients has to be added (assignment of letters to objects was pseudo-randomized). Numbers in parenthesis indicate how many portions of the ingredient are required.

<table>
<thead>
<tr>
<th>Task</th>
<th>Set-up</th>
<th>Action</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Make plant bushy:</td>
<td>Set-up – A – B1 – X(1 portion) – C1 – fertilize</td>
<td>30%</td>
<td></td>
</tr>
<tr>
<td>2) Make plant grow:</td>
<td>Set-up – A – B2 – X(3 portions) – C2 – fertilize</td>
<td>30%</td>
<td></td>
</tr>
<tr>
<td>3) Make plant produce 1-4 flowers:</td>
<td>Set-up – A – B3 – X(1-4 portions) – C1/C2 – fertilize</td>
<td>4 x 10%</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1: Typical display during the Virtual Gardener experiment. The seven objects aligned on top are the ingredients; the four objects aligned vertically below represent the tools. Other objects from left to right represent: the plant, the pot on its stove and lighting the stove) then adding four
different ingredients in the correct order and finally feeding
the plant with the mixture using a syringe-like tool. Three
task variants (with different goals) were used (see Table 1).

The subsequence of actions required to add any ingredient
followed a common structure which entailed several basic
actions, i.e., putting the ingredient on the worktop, chopping
the ingredient with the cutting tool, moving the chopped
ingredient into the pot and stirring the pot. Participants were
introduced to the “Virtual Gardener” task by means of
pictorial recipes that were accessible between trials. Visual
feedback concerning the correctness of performance was
provided at the end of each trial.

Nineteen neurologically unimpaired participants took part
in three experimental sessions each, with one session lasting
between 45 and 80 minutes. During this time they were
required to first learn and then repeatedly perform all
versions of the artificial routine task. Participants carried out
one block (40 trials) in session one and two blocks (80
trials) in each of sessions 2 and 3. In addition to the task
variants (see Table 1), each block contained trials in which
the cooking pot was transparent (50%) and trials with a
concurrent secondary task (50%) in a counterbalanced 2 × 2
design. The transparent pot betrayed the color of its content,
thus giving an implicit disambiguation cue for the ingredient
to add in the last subsequence – though this was not
explicitly brought to the attention of participants. The
secondary task (an auditory monitoring task) was intended
to interfere with the supervisory system by limiting the
availability of central processing resources. The varying
number of portions of ingredient X influenced the length of
this subsequence and thus the span of time over which the
task context information (the current goal) had to be
internally maintained. Furthermore, this variation made the
subsequence-final stirring action less regular, such that it
followed adding an ingredient in most but not all cases.

Action selection latencies (i.e., the time between
consecutive actions) and the nature and position of all
erroneous actions were recorded during the experiment.

Results and discussion

Latencies were transformed into pure processing time by
subtracting the amount of time required for a pure reaching
action over the given distance. Pure reaching times were
calculated by applying Fitts’ Law (Fitts, 1954), with
parameters obtained from a renormalization task performed
midway through the first session. Transformed latency data
were analyzed by means of within subjects ANOVAs to test
for the hypothesized presence of main effects and
interactions between the two factors (usually Action ×
Subsequence), followed by planned comparisons
(Bonferroni corrected) between the points or conditions of
specific interest. In the interest of brevity, only the most
relevant statistics are reported here.

Hierarchical organization

Figure 2 shows the mean transformed latencies for different
but comparable actions (get ingredient versus get tool for
cutting versus get tool for stirring) in each of the four kinds
of subsequence. Overall, latencies at the beginning of a
subsequence (get the ingredient) were significantly longer
than latencies of within subsequence actions (e.g., get the
tool to cut the ingredient; cut vs. get; p < 0.001). The get
stirrer action (stir), which is required after adding each
individual ingredient except when multiple instances of X
must be added, is slower to initiate than the cut action,
which is required after each and every ingredient is placed
on the worktop (cut vs. stir; p < 0.001), but faster than the
get ingredient action (stir vs. get; p = 0.024). These latter
effects are driven by the differences in subsequences B and
C, which is where task variants deviate in terms of which
ingredient has to be added next (i.e. ‘branching points’).

![Figure 2: Action selection latencies for three comparable
actions (reaching for an ingredient, a cutting tool, or a
stirring tool) through the four subsequences.](Image)

This pattern of results is compatible with the notion of an
underlying task representation that is hierarchically
organized. In this view, locally congruent alternative
continuations interfere with action selection and thus
increase selection time (i.e. latency). Note that interference
does not seem to be restricted to making an actual choice:
all stir actions were slower than the cut actions although an
actual choice was only present in subsequence X where, in
some conditions, stirring had to be withheld until several
portions of the ingredient had been added.

Supervisory influence

The difference in selection latencies between performance
of the task in isolation and in conjunction with the
secondary task is shown in Figure 3. In the dual-task
condition, performance was slowed at all points, but this
effect was generally very small (~20 ms). The two points
that are significantly more affected are get in B and get in C
(in B: get vs. cut; p = 0.02, get vs. stir: p = 0.024; in C: get
vs. cut; p = 0.031, get vs. stir: p = 0.022; all other
comparisons are not significant). These results support
the second primary hypothesis in that the points in the task that
were most susceptible to interference (the ‘branching
points’) proved to be most slowed when the availability of
central processing resources (i.e., the supervisory system)
was reduced.
External disambiguation cue

The online measure of action selection latencies, moreover, allowed for more detailed analyses to reveal reliable effects for other factors within the experimental design. The transparent pot, for example, was informative only at one point: because the color of the liquid encoded which ingredient had been added in subsequence B, it could be used as a reliable disambiguation cue for the choice between ingredient C1 and C2 in the last subsequence. As can be seen in Figure 4, this choice was facilitated when the liquid color was visible (comparison of get actions over subsequences: C vs. A: $p = 0.049$, C vs. B: $p = 0.025$; C vs. X: $p = 0.027$, all other comparisons not significant). Note that the pot’s transparency was not mentioned anywhere in the instructions and most participants reported not having noticed that the liquid color was only sometimes visible. Even so, the results suggest that participant behavior was facilitated by the cue—a cue that, though not always available, reliably indicated the ingredient to use in the final step when it was available.

Within subsequence variations

Similarly, the varying length of the third subsequence (adding portions of X) produced reliable effects (see Figure 5). Firstly, entry into the next and final subsequence (get in C) appears to be prolonged by a longer intervening subsequence X (although testing for a linear trend yields a non-significant result: $F(1,18) = 2.80$, $p = 0.112$). Secondly, initialization of the stirring action (stir in X; linear trend: $F(1,18) = 27.39$; $p < 0.001$) took longer when more portions of X had to be added. Thirdly, and this came as a surprise, initialization of the stir action in the next subsequence (stir in C; linear trend: $F(1,18) = 15.05$; $p = 0.001$), i.e. after another ingredient had been added, showed a similar pattern—if to a lesser extent. Such a spill-over effect seems reminiscent of the ‘task set inertia’ postulated in the task switching literature (Allport & Wylie, 1999). Neither of the recent modeling approaches to routine action (i.e., Cooper & Shallice, 2000; Botvinick & Plaut, 2004) would seem to predict such an effect.

Routinization

Finally and somewhat predictably, the factor experience had a considerable impact on action selection latencies. Latencies for all actions became shorter throughout the course of the experiment (see Figure 6). Statistical analyses yielded reliable main effects for Actions and Blocks, their interaction (Actions x Blocks) and linear trends for individual Actions through Blocks, all at a significance level of $p < 0.001$. Interestingly, however, the difficult actions (subsequence initial get; this is the only Action for which a quadratic trend was also given: $F(1,18) = 16.05$; $p = 0.001$) improve more than within subsequence actions (stir), until, towards the end of the experiment, there is little difference between their respective latencies.
This tendency becomes even more apparent when looking at the effect of the secondary task (Figure 7) which, by the second half of session 3, had entirely vanished even for the most demanding actions. This is reflected by a significant interaction (Actions X Blocks: F(1.52, 25.82) = 5.46; p = 0.016). Pair-wise comparisons within individual blocks indicate that the increased vulnerability of the get action is not present any more in last two blocks (session 3), as compared to both the stir and the cut action. Taken together, and within the context of Dual Systems theory, these findings imply not only that the control signal from the supervisory system becomes increasingly dispensable, they furthermore point to a progressive flattening of the hierarchical structure in the underlying representations.

Error analyses

Analysis of the 625 errors that were committed in sessions 2 and 3 mirrors the findings of the latency data: subsequence boundaries were especially prone to error and, in addition, specifically susceptible to the aggravating influence of the secondary task. In parallel to the latency data the initial actions of subsequences B and C were most vulnerable to erroneous action selection. Consistent with previous research, erroneously selected actions were rarely unrelated to the correct ones but rather consisted of intruding actions or sequences of actions from earlier or later in the same task variant, or from a similar point in a different task variant. These results lend some support to the validity of the online measure of action selection latencies.

In a slightly different take on the error data we also found that the rate of recovery from erroneous selections was significantly reduced in trials with a secondary task. This supports the view that the SS plays an important role in monitoring and error correction.

General discussion

We have shown, using a drag-and-drop task environment, that inter-action latencies in hierarchically structured tasks are affected by several factors including practice, the relative frequency of inter-action transitions, the presence of an attentionally demanding secondary task, the presence of environmental cues for action, and the repetition of action subsequences within a sequence. These effects are largely consistent with and predicted by the Dual Systems account of action control described in the introduction. Thus routinization of action is held to correspond to the transfer of (declarative) action knowledge from SS to (procedural) action knowledge in CS. This should lead to faster action selection and less interference from a secondary task as routinization proceeds. Similarly, CS is held to be triggered in part by environmental cues, so the presence of disambiguating environmental cues should also speed action selection, even (or especially) once a sequence has been routinized. Finally, hierarchical task structure is assumed to be present in both SS and CS, and this is reflected in greater latency at subtask initial actions (i.e., get ingredient) than at other points in an action sequence involving similar kinds of action (get stirring tool or get cutting tool).

At the same time, the Dual Systems account of action control is not well specified at the lowest levels, and so it is silent on some of the observed effects. In our view the most significant findings of this variety concern the “flattening” of hierarchical structure with practice (which we refer to as progressive routinization), and the effects of a repeated subtask sequence within a task (i.e., the slowing of selection when multiple instances of X were added, and the carry-over of this effect into the next subtask). The former result provides the first direct data with respect to the putative transfer of control from SS to CS, while the latter, as suggested above, may be related to task set inertia as observed in task switching (e.g., Allport & Wylie, 1999).

Our empirical findings also have strong implications for current computational accounts of routine action selection. Two models specifically address the domain: the interactive activation network (IAN) model of Cooper & Shallice (2000) and the simple recurrent network (SRN) model of Botvinick & Plaut (2004). These models are both really only models of CS (see Ruh, 2007, for an attempt at addressing this limitation), but our data point to complementary limitations of the models. Thus, the IAN model is, in principle, instructable at the level of SS (a pre-requisite for progressive routinization) and, by virtue of its settling characteristics, able to make action latency predictions –
Indeed it predicts greater latencies at subtask boundaries (see, e.g., Cooper & Shallice, 2000, figure 5). Crucially though, the IAN model lacks any mechanism for learning. In contrast the SRN model incorporates a well-understood (though, as argued by Cooper & Shallice, 2006, and Ruh, 2007, implausible) learning mechanism, but it cannot be instructed at the level of SS and therefore cannot address progressive routinization. In addition, the SRN model produces an action on each processing cycle (i.e., it does not settle into an attractor state for each action), so it cannot be easily related to reaction time data.

Our results seem to suggest that the system or systems underlying routine action selection have characteristics of both interactive activation and recurrent networks. What appears to be required is a model that learns to progressively routinize action sequences (in a fashion similar to that of Botvinick & Plaut (2004), though through instruction from SS and possibly based on reinforcement learning: see Ruh, Cooper & Mareschal, 2005), but that is open to supervisory intervention at any point if necessary. At the same time, for the model to make contact with the data it must make predictions of reaction times, which suggests that the action selection system is best conceived as a settling network in which environmental and supervisory influences are combined, over time, in the generation of a stable state. These constraints point towards a recurrent-type model that a) is instructable at the levels of individual actions, subtasks or complete tasks, b) gradually becomes less dependent upon instruction as tasks become more routine, and c) settles, over a number of cycles, into an attractor state corresponding to the action to be selected at each point in a task. Our behavioral data on progressive routinization suggest that, within such a model, settling time should a) be shorter for high-frequency action transitions than for low-frequency transitions, b) should decrease with increasing experience of a task, c) should decrease when multiple information sources support selection of the same action, and d) should show inertia when moving from a repeated (sub-)task to a new (sub-)task.

In closing, it is worth emphasizing that the fact that the observed effects are plausible and statistically reliable is also an important finding. There is very little empirical work on routine action selection and even less with neurologically unimpaired populations. What work there is (we know of only two relevant studies: Botvinick & Bylsma, 2005; Giovanetti, Schwartz, & Buxbaum, 2007) focuses on error patterns but is limited by the scarcity of errors under laboratory conditions. Our findings demonstrate that the “virtual gardener” experiment, and more generally the drag-and-drop environment in which latencies may be recorded, provides a methodologically valid approach to the investigation of the acquisition and performance of routine actions.

Acknowledgements

The work presented here was supported by a maintenance grant from Birkbeck, University of London to Nicolas Ruh.

References:


