Cognitive Control in the Generation of Random Sequences:
A Computational Study of Secondary Task Effects

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

Abstract

Cognitive control processes, such as those involved in response inhibition or task switching, have been the focus of much recent research. Few studies, however, have considered how such processes work together in tasks that require multiple control processes. This paper reports a computational study of random sequence generation and the cognitive control processes involved therein. The task, which is argued to involve multiple control processes, produces several dependent measures. These measures are held to be differentially dependent on the differential efficacy of the various underlying control processes. Initial simulations demonstrate that the model is capable of reproducing subject performance on the basic task. Additional simulations explore differential interference effects of different secondary tasks (held to interfere with different control processes) on the different random generation dependent measures. The work illustrates how the putative control processes may interact in the production of successive responses during the random generation task.

Keywords: Random generation; executive processes; cognitive control; response inhibition; set shifting; monitoring.

Introduction

A substantial body of evidence suggests that behaviour in complex tasks is dependent on a number of functionally (and anatomically) distinct control functions, such as response inhibition, memory updating, task shifting and monitoring. One study which well supports this position is that of Miyake et al. (2000), who had over 130 subjects complete nine relatively simple tasks (three of which were primarily held to tap the control function of response inhibition, three to tap memory updating and three to tap task shifting) and five more complex tasks (such as solving Tower of Hanoi problems, which were thought to tap multiple control functions). Miyake and colleagues used confirmatory factor analysis on performance measures from the nine simple tasks to extract three factors, corresponding conceptually to *response inhibition*, *memory updating* and *task shifting*. They followed this up with structural equation modelling, using the three derived factors, to determine the involvement of those factors in performance of the complex tasks. The analysis supported the involvement of different subsets of the three separable factors in performance of the different complex tasks. Similar results using different batteries of tasks have been obtained with developmental (Bull et al., 2004) and neuropsychological (Stuss et al., 2005; Shallice et al., 2008) samples, while a number of other studies have focused on specific control functions (for reviews see, e.g., Monsell, 2003, and Vandierendonck et al., 2010, for task switching, and Aron, et al., 2004, and Verbruggen & Logan, 2008, for response inhibition).

In response to this empirical work, a number of computational accounts of the operation of various control functions have been proposed. For example, Jones et al. (2002) modelled a process of monitoring and adjusting for response conflict within a simple interactive activation model of two-alternative forced choice by using a measure of response conflict to modulate the baseline activity of response units – when conflict was high the baseline activity was reduced, leading to slower/more deliberate responding (see also Botvinick et al., 2001). Other researchers have focussed on different control functions. Thus, Gilbert and Shallice (2001) were able to account for the behavioural effects of task shifting by modifying an existing interactive activation model of the Stroop task to allow a form of carryover between trials, while O'Reilly and Frank (2006) have provided a computational account of possible control processes related to working memory.

As indicated by the preceding discussion, the existence of control functions is widely accepted in the behavioural, neuropsychological (and neuroimaging) literatures. Moreover, specific functions have received substantial attention in the computational literature. However, while the behavioural literature would suggest that control functions are likely to be of most importance during the performance of relatively complex tasks, the computational literature has focussed on relatively simple tasks (e.g., two-alternative forced choice and Stroop). It has, to date, not considered how control functions might interact in complex task performance. Just as critically, existing cognitive architectures such as ACT-R (Anderson et al., 2008) – systems that are routinely applied to modelling the performance of complex tasks – generally fail to make any explicit appeal to control functions of the kind postulated in the other literatures. On the basis of this architectural work, one might therefore argue that such control functions are epiphenomenal (cf. Cooper, 2010).

The purpose of this paper is to explore, from a computational perspective, how different cognitive control processes might interact in a task that appears to tap multiple such processes. We begin by describing the task – random sequence generation – together with a verbal account of the control processes that have been held to be involved in performance of the task. Target data from a dual-task study of random sequence generation is then reported which suggests that secondary tasks which tap different control processes (specifically, updating and

monitoring of working memory, task shifting and response inhibition) interfere with random generation in different ways. The focus of the paper, however, is a computational account of random generation which critically involves the control processes of memory updating and monitoring, task shifting, and response inhibition. The model is shown to be capable of reproducing control performance in the task. Additional simulations, aimed at accounting for the different dual-task interference effects, explore the possible roles of cognitive control processes in the task. The work demonstrates how control processes may interact in the performance of tasks with complex control requirements while providing additional support for the fractionation of cognitive control.

The Random Generation Task

In random generation tasks subjects are provided with a response set (e.g., integers from 0 to 9) and required to generate a sequence of responses from this set such that the sequence is subjectively random. The task is of interest because, despite the apparent loose specification of the task, subjects exhibit strong biases, producing sequences that deviate from true randomness in reliable ways. For example, repeat responses (i.e., the same response on two consecutive trials) are typically generated with lower than expected frequency (e.g. Rapoport & Budescu, 1997; Towse & Valentine, 1997).

There are numerous ways of measuring the degree to which a sequence is random. For example, in a truly random sequence one would expect, over the course of a sufficiently long sequence, that the frequency of each response is equal. One would also expect the frequency of response pairs (i.e., R_1 followed by R_2) to be equal, so that it is not possible to predict with greater than chance accuracy the next response given the previous response. Towse and Neil (1998) survey a range of measures of randomness and show, through factor analysis of subjects' responses, that the different measures of randomness cluster into several factors. Thus, several measures of randomness index "equality of response usage" (i.e., whether all responses are generated with roughly equal frequency, or whether there is a bias towards some responses and against others). Similarly, several other measures index "prepotent associates" (i.e., whether some pairs or "bi-grams" of responses – associates – occur more frequently than would be expected by chance).

The various measures of randomness are also affected by the format of the response. For example, responses may be generated verbally (Baddeley et al., 1998; Towse, 1998), in writing (Towse & Valentine, 1997), or using a keyboard (Baddeley, et al., 1998; Towse, 1998). If responses are generated with two hands on a keyboard, then subjects tend to alternate hands more frequently than appropriate. Biases towards prepotent associates are therefore specific to the format of the response. In a similar vein, equality of response usage tends to be poorer when the response set is internalised (as in verbal digit generation), in contrast to when the response set is externally realised (as in selection

from a keyboard) and hence when random generation involves selecting from that externally realised set.

Random generation tasks have a surprisingly long history in psychological research (see Wagenaar, 1972, for an early review) and have been widely used in examining cognitive control processes (e.g., Baddeley, et al., 1998; Miyake et al., 2000). To understand why control processes might be relevant, it is useful to consider a possible process model of random generation. Suppose one is attempting to generate the nth response in a series, having already generated n-1 responses. A possible response somehow comes to mind, perhaps because it is in some way associated with the previous response (e.g., if generating digits and the previous response was 8, the possible response 4 might come to mind, corresponding to a half of 8). Before producing the response, one must then decide if it is sufficiently random given the previous n-1 responses. Thus it is necessary to monitor ones likely responses, maintain an up-to-date record of previous responses, and possibly inhibit a potential response if it is deemed "too predictable".

This process account of random generation is basically that of Baddeley et al. (1998; see also Rapoport & Budescu, 1997), but random generation was also one of the complex tasks investigated by Miyake et al. (2000). Rather than considering a specific processing account of random generation, Miyake et al. used an analysis of individual differences together with structural equation modelling to determine the relation between their three specific control processes – response inhibition, memory monitoring and updating and task shifting – and the factors found (by factor analysis) to underlie random generation. They found that measures of randomness associated strongly with "equality of response usage" were correlated with the control process of memory monitoring and updating. That is, subjects who performed well on memory monitoring and updating tasks tended in random generation to produce all responses with roughly equal frequencies, in contrast to subjects who performed poorly on memory monitoring and updating tasks, who tended to show biases towards some responses and away from others. Similarly, measures of randomness associated strongly with "prepotent associates" were inversely correlated with the efficacy of the putative control process of response inhibition. Thus, subjects who performed poorly at response inhibition tasks tended in random generation to produce some pairs of responses more frequently than others. These findings seem plausible, but they would benefit from being embedded within a process model for a complete understanding of the operation of control processes in random generation.

Secondary Task Effects on Random Generation

Cooper et al. (submitted) take an alternative approach to determining the control process requirements of random generation. In their study, subjects completed a random generation task under four conditions; first as a solitary task and then within a dual-task paradigm concurrently with each
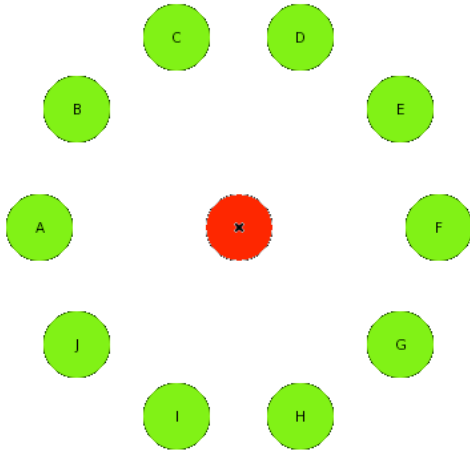
Figure 1: The clock-face layout of the random generation task of Cooper et al. (submitted). Subjects were required to click with the mouse on one letter on each trial. The mouse was then automatically repositioned in the central circle.

of three secondary tasks. The random generation task involved using a mouse to select responses from a clock-face type of display with ten options (see Figure 1). Thus the response set was externally realised. Moreover two types of prepotent associates specific to the task can be identified: opposite associates, where successive responses are 180° apart (A-F, B-G, C-H, etc.), and adjacent associates, where successive responses are adjacent in a clockwise or anti-clockwise direction (A-B, A-J, B-A, B-C, etc.).

The three secondary tasks were designed to primarily tap different control processes. Thus the *digit-switching* task was held to tap the process of task-shifting, the *2-back* task to tap memory monitoring and updating processes, and the *go-no go* task to tap response inhibition. In each of the three dual-task conditions subjects were required simultaneously to complete the random generation task (which was visual-manual in nature) and one of these secondary tasks (which were each auditory-vocal in nature). The full procedure is described in Cooper et al. (submitted).

The effects of condition on five measures of randomness are shown in Table 1. The measures are: R (which measures equality of response usage); RNG (which measures equality of bi-gram usage); RR (the proportion of responses that are repeats); AA (the proportion of responses that are adjacent associates) and OA the proportion of responses that are opposite associates). Figure 2 shows the data in a way that more clearly shows the effect of condition on each

Table 1: Mean values of measures of randomness in the control condition and each of the three experimental conditions. (CTRL = control, DS = digit-switching, 2B = 2-back, GnG = go-no go.)

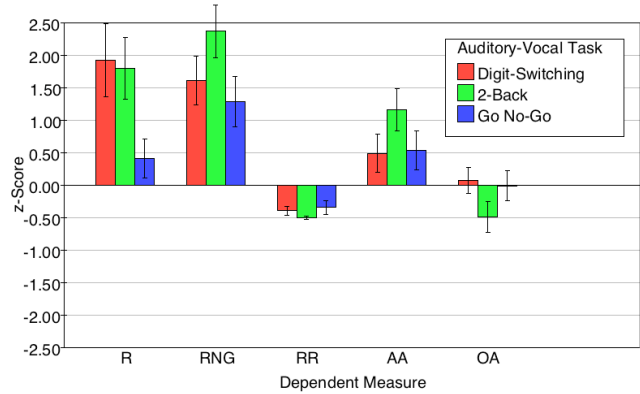|	R	RNG	RR	AA	OA
CTRL	0.962	0.300	0.014	0.259	0.131
DS	2.048	0.410	0.004	0.328	0.136
2B	1.979	0.461	0.002	0.424	0.097
GnG	1.196	0.388	0.005	0.334	0.130

Figure 2: z-scores for measures of randomness in the three dual-task conditions, based on means and standard deviations of the control condition. Error bars indicate the standard error of the mean z-score in each condition.

dependent measure. In this figure, the means and standard deviations of each dependent measure in the control condition were used to convert scores from the three experimental conditions into z-scores, clarifying the effect of each condition on the different dependent measures. As can be seen from the figure, digit-switching and 2-back have similar large effects on R, with go-no go having a lesser effect. In contrast, 2-back has the largest effect on RNG, RR, AA and OA, with digit-switching and go-no go having similar lesser effects. (All effects apparent in the figure were statistically significant except those concerning RR, for which statistical power was limited by a floor effect.)

These results appear to conflict with those of Miyake et al. (2000) described in the previous section. For example, while the 2-back task – held to tap memory monitoring and updating – had a significant effect on the R measure (as would be predicted), its effect was similar to that of the digit-switching task – held to tap set shifting (which would not be predicted). More critically, the effect of the go-no go task – held to tap response inhibition – on bi-gram measures (RNG, RR, AA and OA) was less than that of the memory monitoring and updating task. In contrast, the results seem more consistent with the verbal process model of Baddeley et al. (1998) described in the previous section. The following two sections present a computational model of the task based on this verbal process account, together with simulations that explore the possible effects of secondary tasks and hence the relevant control processes.

A Model of the Random Generation Task

The model of random generation described here was developed within COGENT (Cooper & Fox, 1998), a graphical object-oriented environment for cognitive modelling. COGENT allows information processing models to be sketched as box-and-arrow diagrams. Such a diagram may then be fleshed out into a fully functioning model by providing if/then rules and property settings for each box. Figure 3 shows the box-and-arrow structure of the random generation model. The model consists of three buffers (shown as rounded rectangular boxes) and four processes
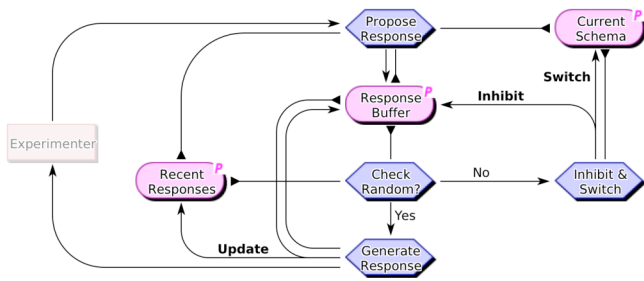
Figure 3: The box-and-arrow diagram of the random generation model.

(shown as hexagonal boxes). Buffers store information while processes transform information or copy it from one buffer to another. Arrows with pointed heads show information flow (implemented by message passing), while arrows with blunt heads indicate querying of buffer contents by processes.

The model functions as follows. When prompted by the *Experimenter* (the faded rectangular box), *Propose Response* attempts to propose a single random response from the response set. How this is done is discussed below, but the result is added to *Response Buffer*, a temporary storage device with a capacity of just one item. When an element is present in *Response Buffer*, the *Check Random?* process evaluates that response in the context of previous responses to determine whether it is subjectively random. If so, the *Generate Response* process is triggered and the proposed response is generated (and sent to the *Experimenter*). If not, an additional process, *Inhibit & Switch*, is invoked to inhibit generation of the proposed response. This process also switches the current schema that is used to produce a potential response by *Propose Response*. This process can then propose an alternative response, which will then be added to *Response Buffer* and a further round of evaluation for randomness will take place. This part of the model will loop until a proposed response is considered by *Check Random?* to be sufficiently random.

On the first trial *Propose Response* generates its proposal by selecting at random from the response set. On subsequent trials, however, it selects a response by applying a "schema" to (its recollection of) the previous response. Schemas implement associations between responses. Thus one schema might implement the association of selecting the opposite response, while other schemas might implement the associations of selecting an adjacent response. *Current Schema* stores the schema that is, at a given point in time, being used to generate responses. Like *Response Buffer*, it is limited to storing just one item (i.e., one schema) at a time. The schema itself is generated by *Inhibit & Switch*. We assume that schema generation may itself be modelled as a random process with the probability of generating any particular schema being a function of that schema's prepotency. For example, the schema for selecting a response that is diametrically opposite to the previous response is assumed to be selected more frequently than the

schema for, say, selecting a response that is 72° clockwise from the previous response.

Recent Responses maintains a record of recently generated responses. This record is used in two ways: *Check Random?* uses it to test whether a proposed response is subjectively random. *Propose Responses* uses it to provide the seed for generating the next proposed response from the model's (recollection of its) previous response and the current schema. Thus, unlike the other buffers its capacity is not limited to one. For the purposes of the simulations reported here, it is allowed an unlimited capacity but decay is imposed on its elements. Thus, there is a small probability that an element placed in the buffer on processing cycle *n* will disappear from the buffer on each subsequent processing cycle.

How should *Check Random?* work? Random generation is known to be a task that produces large individual differences, and one aspect of the task that may be open to individual differences is the subjective assessment of what is or is not random. One could certainly envisage this being a complex process – at least for subjects who perform well on the task. For current purposes, however, we adopt a very simplistic criterion of subjective randomness: namely that if a response is present in *Recent Responses* then it cannot be sufficiently random. While this might seem implausible, simulations demonstrate that it yields a surprisingly good account of the experimental data.

The model as described is underspecified in two critical ways. Neither the rate of decay of elements from *Recent Responses* nor the probability distribution of schemas (as required by the schema generation sub-process of *Inhibit & Switch*) have been specified. These are effectively free parameters of the model. A series of simulations was performed to explore the effects of these parameters. In each case, 36 blocks of 100 trials were simulated (corresponding to the 36 subjects tested by Cooper et al., submitted), and the resultant sequences scored according to the measures of

Table 2: Mean simulated values of measures of randomness as a function of memory decay rate for two distributions of prepotent responses. (Note: The memory decay rate is the half-life in cycles of memory elements, that is, the number of cycles an element remains in a buffer on average before the probability of it decaying is 50%.)

Half-Life	R	RNG	RR	AA	OA
10	0.937	0.405	0.006	0.379	0.016
20	0.737	0.275	0.014	0.260	0.045
30	0.790	0.262	0.018	0.241	0.054
40	0.885	0.251	0.019	0.234	0.064

a) All schemas equi-probable.

Half-Life	R	RNG	RR	AA	OA
10	0.621	0.398	0.011	0.433	0.059
20	0.753	0.278	0.014	0.327	0.108
30	0.771	0.263	0.018	0.285	0.131
40	0.924	0.256	0.024	0.281	0.141

b) Strong bias towards opposite and adjacent responses.

randomness described in the previous section. Results are shown in Table 2.

Consider first the case where all schemas are equi-probable (Table 2a). Here, R scores (which can in theory range from 0 to 100) are reasonably similar to those obtained from human subjects (which ranged from 0.962 in the control condition to 2.048 in the digit-switching condition; see Table 1). RNG scores are also comparable to those obtained with human subjects. The three measures related to specific bi-grams show that the primary difference between sequences generated by human subjects and the model lies in the model's tendency to produce too few opposite associates (around 6 in 100 when the half-life of recent responses is greater than 20, compared to 13 in 100 for human subjects).

But why, given the mechanism for checking randomness, do repeat responses occur at al? In fact, such responses can be proposed for two reasons: either the "generate repeat" schema is selected and applied to the immediately preceding response, or the immediately preceding response decays from *Recent Responses* and the schema that was applied (successfully) to the response produced on trial *n-2* to generate a proposed response on trial *n-1* is applied again on trial *n* with the response from trial *n-2*. Repeat responses proposed via the first of these are typically rejected by *Check Random?* as being insufficiently random (because unless they decay at a critical moment they will still be in *Recent Responses*). Thus, repeat responses are generally produced by the model because it essentially "forgets" that it has produced the same response on the previous trial.

The low rate of opposite associates arises from a similar interaction of processes. Here the issue is that the "generate opposite" schema is unusual in that applying it twice in succession will produce the sequence $R_1 R_2 R_1$. If R_1 has not decayed from *Recent Responses* when it is generated the second time it will be suppressed by *Check Random?*, thus causing the model to produce fewer repeat responses than would be expected by chance.

The low rate of opposite associates may be ameliorated by assuming that the "generate opposite" schema has a relatively high probability of controlling *Propose Response*. The figures in Table 2b were generated by assuming that this schema was three times more likely to be selected by the switching sub-process of *Inhibit & Switch* than the "generate adjacent" schemas, which were in turn slightly more likely than the schemas for generating responses that bear other relations to the previous response. Note in particular that both of the last two lines of Table 2b provide a reasonable fit to the subject data from the control condition, with all simulated data being within one standard deviation of the observed means.

Modelling Secondary Task Effects

How might concurrent performance of a secondary task affect random generation? Given the simulations reported above, one can rule out one simple possibility. Suppose the effect of secondary task performance (whatever the task)

Table 3: Mean simulated values of measures of randomness as a function of memory decay rate when switching efficiency is reduced to 10%. (Cf. Table 2.)

Half-Life	R	RNG	RR	AA	OA
10	3.792	0.667	0.049	0.393	0.112
20	1.993	0.451	0.074	0.355	0.100
30	1.451	0.337	0.082	0.307	0.098
40	1.454	0.328	0.081	0.349	0.103

was merely to impair working memory maintenance (modelled by increasing the speed with which elements decay from *Recent Responses*). The simulations reported in Table 2b show that while this provides a good account of the effects of secondary task performance on bi-gram measures (RNG, RR, AA and OA), it fails to account for the effect of any of the secondary tasks on the R score. Recall that this score increases in all dual-task conditions. Impairing working memory by decreasing the half-life of elements in *Recent Responses* has the reverse effect.

Two further possibilities may also be rejected. First, suppose that performance of a secondary task were to decrease the accuracy of the *Check Random?* process. Space limitations prevent presentation of full results, but simulations show that decreasing the accuracy of the relevant rule results in a large increase in the OA score – again contrary to what is observed in any of the conditions for the human data. Second, suppose that performance of a secondary task were to impair the encoding of responses as they are generated. Simulations show that decreasing the success of this process results in a large increase in the RR score – yet again contrary to what is observed in any of the conditions for the human data.

The simulations thus far argue against an account of the data of Cooper et al. (submitted) in terms of reduced efficiency or effectiveness of a single process or function. Consider then one further manipulation, namely reducing the effectiveness of the switching sub-process following proposal of an apparently non-random response, and consider this in conjunction with reduced maintenance of memory elements. Table 3 shows the effect of simultaneously reducing switching efficiency to 10% and decreasing the half-life of elements in *Repeat Responses*. Here the results are more positive. In particular, this manipulation results in increased R scores and RNG scores, coupled with decreased OA scores. AA scores are also generally higher than in the equivalent simulations when switching is 100% efficient, replicating the effect seen in all dual-task conditions of secondary task on AA.

Discussion

The above results do not provide a perfect fit to any of the experimental conditions, but they are suggestive. One possibility is that all secondary tasks impose some common load on random generation, the effect of which is to limit memory for previous responses. From Table 2 this may explain the increase in RNG and AA scores in all secondary

task conditions, coupled with the corresponding decrease in RR and OA scores in those conditions. On this account, the digit-switching and go-no go tasks would appear to impose similar memory burdens, while the 2-back task imposes a greater burden (cf. Figure 2). This is consistent with the 2-back task being a demanding memory task.

In addition to these task-general effects, however, task-specific effects seem to be required to explain the increase in R score in the digit-switching and 2-back dual-task conditions. Decreasing the efficiency of the switching sub-process can certainly account for an increase in R, as shown in Table 3, and the fit between the data in line 2 of Table 3 and subject data when random generation is coupled with the 2-back task (line 2 of Table 1) is of particular note. Yet this leaves a puzzle. The 2-back task was not intended to be a switching task. Note however that any dual-task situation is likely to result in switching between the two tasks, and this would be expected to impair the efficiency of switching between schemas within the primary random generation task. This still leaves a question over the production of repeat responses – if switching failure is behind the performance in the 2-back condition this does not explain the very low RR score in that condition. However, a limitation of the current model is that all representations are discrete. Thus, elements are either in or not in a buffer. Elsewhere, low RR scores have been attributed to inhibition of a response following its production. Elaborating the model within an activation-based system may be necessary in order to account for the effects of condition on RR.

We began by considering the role of cognitive control in complex tasks, and in particular in the generation of random sequences. The simulation results reported here provide a simple yet empirically adequate account of the basic task. Capturing the dual-task data of Cooper et al. (submitted) has proved to be more difficult, but in attempting to do so the model suggests that (a) all tasks impose an increased load which may be simulated by an increase in decay of the store of recent responses, and (b) that an additional load on the switching function may account for the increase in R score observed in two of Cooper et al.'s dual-task conditions.

References

Anderson, J.R., Fincham, J.M., Qin, Y. & Stocco, A. (2008). A central circuit of the mind. *Trends in Cognitive Sciences*, *12*, 136–142.

Aron, A.R., Robbins, T.W. & Poldrack, R.A. (2004). Inhibition and the right inferior frontal cortex. *Trends in Cognitive Sciences*, *8*, 170–177.

Baddeley, A., Emslie, H., Kolodny, J., & Duncan, J. (1998). Random generation and the executive control of working memory. *The Quarterly Journal of Experimental Psychology Section A: Human Experimental Psychology*, *51*(4), 819-852.

Botvinick, M.M. Braver, T.S., Barch, D.M., Carter, C.S. & Cohen, J.D. (2001). Conflict monitoring and cognitive control. *Psychological Review*, *108*, 624–652.

Bull, R., Espy, K.A., & Senn, T.E. (2004). A comparison of performance on the Towers of London and Hanoi in young children. *Journal of Child Psychology and Psychiatry, 45:4*, 743–754.

Cooper, R. P. (2010). Cognitive control: Componential or emergent? *Topics in Cognitive Science, 2*(4), 598-613.

Cooper, R.P. & Fox, J. (1998): COGENT: A visual design environment for cognitive modelling. *Behavior Research Methods, Instruments, & Computers, 30,* 553–564.

Cooper, R.P., Wutke, K., & Davelaar, E.J. (submitted). Differential contributions of set-shifting and monitoring to dual task interference.

Gilbert, S. & Shallice, T. (2002). Task switching: A PDP model. *Cognitive Psychology*, *44*, 297–337.

Jones, A.D., Cho, R.Y., Nystrom, L.E., Cohen, J.D. & Braver, T.S. (2002). A computational model of anterior cingulate function in speeded response tasks: Effects of frequency, sequence, and conflict. *Cognitive, Affective & Behavioral Neuroscience*, *2*, 300–317.

Miyake, A., Friedman, N.P., Emerson, M.J., Witzki, A.H., Howerter, A. & Wager, T.D. (2000). The unity and diversity of executive functions and their contributions to complex "frontal lobe" tasks: A latent variable analysis. *Cognitive Psychology*, *41*, 49–100.

Monsell, S. (2003). Task switching. *Trends in Cognitive Sciences*, *7*, 134–140.

O'Reilly, R.C. & Frank, M.J. (2006). Making working memory work: A computational model of learning in the prefrontal cortex and basal ganglia. *Neural Computation*, *18*, 283-328.

Rapoport, A., & Budescu, D.V. (1997). Randomization in individual choice behavior. *Psychological Review*, *104*(3), 603-617.

Shallice, T., Stuss, D.T., Picton, T.W., Alexander, M.P. & Gillingham, S. (2008). Mapping task switching in frontal cortex through neuropsychological group studies. *Frontiers in Neuroscience*, *2*, 79–85.

Stuss, D.T., Alexander, M.P., Shallice, T., Picton, T.W., Binns, M.A., Macdonald, R., Borowiec, A. & Katz, D.I. (2005). Multiple frontal systems controlling response speed. *Neuropsychologia*, *43*, 396–417.

Towse, J. N. (1998). On random generation and the central executive of working memory. *British Journal of Psychology*, *89*(1), 77-101.

Towse, J.N. & Neil, D. (1998). Analyzing human random generation behavior: A review of methods used and a computer program for describing performance. *Behavior Research, Instruments and Computers*, *30*, 583–591.

Towse, J.N., & Valentine, J.D. (1997). Random generation of numbers: A search for underlying processes. *European Journal of Cognitive Psychology*, *9*(4), 381-400.

Vandierendonck, A., Liefooghe, B., & Verbruggen, F. (2010). Task Switching: Interplay of Reconfiguration and Interference Control. *Psychological Bulletin, 136(4),* 601-626.

Verbruggen, F., & Logan, G. (2008). Response inhibition in the stop-signal paradigm. *Trends in Cognitive Sciences*, *12*(11), 418-424.

Wagenaar, W.A. (1972). Generation of random sequences by human subjects: A critical survey of literature. *Psychological Bulletin*, *77*, 65–72.