Psychological Review
2001, Vol. 108, No. 4, 847–869

0033-295X/01/$5.00 DOI: 10.1037//0033-295X.108.4.847

THEORETICAL NOTE

Serial Modules in Parallel: The Psychological Refractory Period and Perfect Time-Sharing

Michael D. Byrne
Rice University

John R. Anderson
Carnegie Mellon University

The authors describe ACT-R/perceptual-motor (ACT-R/PM), an integrated theory of cognition, perception, and action that consists of the ACT-R production system and a set of perceptual-motor modules. Each module (including cognition) is essentially serial, but modules run in parallel with one another. ACT-R/PM can model simple dual tasks such as the psychological refractory period (PRP), including subtle results previously explained with executive process interactive control (EPIC, D. E. Meyer & D. E. Kieras, 1997a). The central difference between the theories is that EPIC's productions can fire in parallel, whereas in ACT-R/PM, they are serial. Results from three PRP-like experiments with more demanding cognitive requirements indicate that cognitive processing for the 2 tasks need not overlap. ACT-R's activation-based retrieval processes are critical in accounting for the timing of these tasks and for explaining the dual-task performance decrement.

Computational theories are not new to experimental psychology. Traditionally, computational theories of cognition have been just that, theories of cognition and cognition alone. Similarly, computational theories of perception have typically focused almost exclusively on perceptual phenomena. The same is true of work on motor control. What have been far less common are theories that seriously attempt to integrate cognition, perception, and action. This is surprising, because there are numerous domains to which such a theory might be applicable, such as mental workload, manual tracking, divided attention, time and motion analysis, paced tasks and time stress, resource-conflict matrices, some kinds of errors, the eye–hand span in typewriting, and many more.

This is not to say there have been no theories concerned with such integration. The model human processor (MHP) of Card, Moran, and Newell (1983) was originally presented as a summary of the state of the field's current knowledge about cognition and performance. The MHP, although never implemented as a computational model, specified the timing for cognition, perception, and motor "processors." The MHP also stipulated that these processors ran in parallel with one another, though each processor was itself serial. That is, each processor could perform one operation at a time, but multiple processors could be working at one time. John developed a framework based on the MHP for making specific predictions about high-speed expert performance in telephone operators (Gray, John, & Atwood, 1993) and transcription typists (John, 1996). Her analyses provide clear demonstrations of the necessity for parallelism between cognition, perception, and motor stages of processing in fairly complex tasks. Particularly relevant to the current article, she described the notion of the critical path for any task (see also Schweikert, 1980). That is, computation by one of the stages may depend on the output of other stages, and there can be a critical path of dependencies among the parallel stages. We make use of this technique later to understand various dual-task effects.

However, the MHP was never fully instantiated into a computational theory. Newell (1990) described some efforts to do this with the Soar architecture and described some of the difficulties involved. The best computational instantiation of such a theory to date is executive process interactive control (EPIC; Meyer & Kieras, 1997a, 1997b). EPIC represents a substantial advance in the computational modeling of cognition, perception, and action. However, EPIC has focused on what might be called "low-level" cognition such as performing simple dual tasks and the like. It is not, nor is it intended to be, a theory of memory, problem solving, learning, and other "high-level" cognition. There has still not been a comprehensive computational theory with applications across a broad range of cognition as well as perception and action.

This article proposes just such a theory, called ACT-R/perceptual-motor (ACT-R/PM). This theory is a synthesis of the ACT-R production system (Anderson, 1993; Lebiere & Anderson, 1998) with a set of EPIC-inspired perceptual-motor modules. The theory itself is still fairly young, but we believe it represents a significant advance over ACT-R in numerous areas. We first describe the theory and then show how the theory applies to some existing data in the realm of simple and rapid dual tasks, an area previously too low level to be covered by ACT-R. We then move

Michael D. Byrne, Department of Psychology, Rice University; John R. Anderson, Department of Psychology, Carnegie Mellon University.

This research was supported by National Science Foundation Grant SBR 94-21332, Office of Naval Research Grant N00014-96-I-0491, and National Institute of Mental Health Grant 2732-MH19102. We thank Wayne Gray and the Washington ACT-R Guild for their comments on an earlier version of this article.

Correspondence concerning this article should be addressed to Michael D. Byrne, Department of Psychology, Rice University, MS-25, 6100 Main Street, Houston, Texas 77005-1892. Electronic mail may be sent to byrne@acm.org.

"up" a level to dual tasks with more complex cognitive requirements, presenting three new dual-task experiments that blend traditional dual-task conditions with more traditionally cognitive tasks. These experiments provide new insights into dual-task phenomena, which we believe we have captured in a series of ACT-R/PM models.

ACT-R/PM

ACT-R/PM is an extension of the ACT-R production system theory of cognition (Anderson, 1993; Anderson & Lebiere, 1998). ACT-R/PM is organized as depicted in Figure 1. In many ways, this system is similar to, and was certainly heavily influenced by, Kieras and Meyer's (1996) EPIC system. In ACT-R/PM, there are four perceptual-motor modules that communicate with central cognition, which is realized as a production system (in this case ACT-R). ACT-R involves a spread of activation over declarative memory followed by a selection of a production to fire. Although spreading activation is parallel, cognition is serial at the level of production selection. Similarly, each of the perceptual and motor modules is itself more or less serial (exceptions are noted below in the module descriptions). However, central cognition and the various modules all run in parallel with one another. For example, the production system could be retrieving something from long-term declarative memory while the vision module is shifting attention in the visual array and the motor module is preparing to press a key. This is in agreement with the original MHP, which consisted of a collection of serial processors acting in parallel with one another. The modules are similar in form to EPIC's "processors"; we chose the term *module* to emphasize that each module is, from the perspective of the other components of the system, a black box, and may be thought of as an encapsulated module.

In the ACT-R production system, production selection is under the control of the current goal, and a single production is selected that matches that goal. Executing this production can involve the retrieval of elements from declarative memory (called *chunks*). The latency for retrieval of chunk i is governed by the retrieval time equation (Anderson, Lebiere, & Lovett, 1998)

$$Time_i = Fe^{-A_i}, \quad (1)$$

where A_i is the activation of chunk i. The activation of chunk i is both a function of the base-level activation and the activation from elements focused in the goal. This is described by the activation equation

$$A_i = B_i + \sum W_j S_{ji}, \quad (2)$$

where B_i is the base-level activation of chunk i, W_j is the source activation (the amount of activation spread from the chunks in the goal chunk), and S_{ji} is the strength of association between goal source j and chunk i. A chunk's base-level activation, B_i, is a function of recency and frequency of access. W_j is determined by the number of source elements in the goal; a total of 1.0 units of source activation is divided evenly among all elements in the goal, and thus total activation is a dynamic property of the current state of the goal. S_{ji} is a measure of the strength of association between

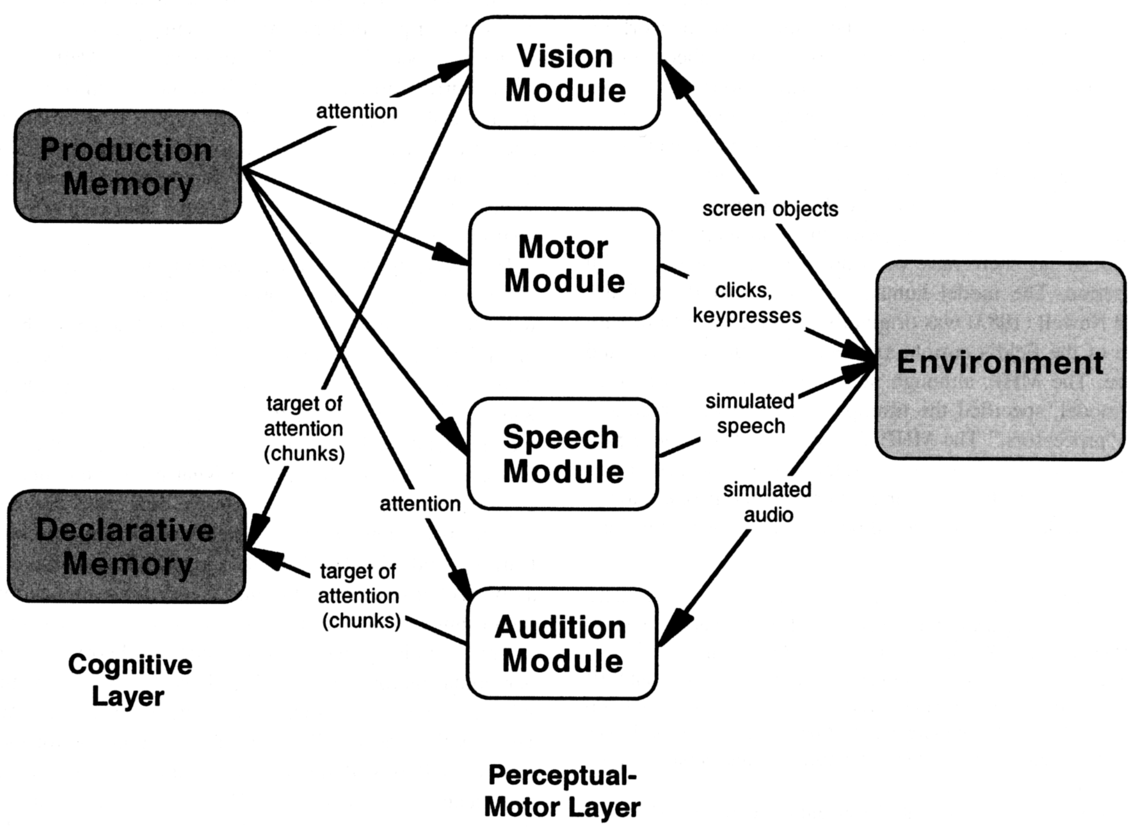


Figure 1. ACT-R/perceptual-motor system diagram. The cognitive layer and each of the perceptual-motor modules run in parallel, but each component is itself serial.

chunks j and i, based on how often chunk i was needed when element j was a goal element. (See Anderson, Lebiere, & Lovett, 1998, for further details.) These activation equations form the backbone of ACT-R's successful program of modeling results from the memory literature such as list memory (Anderson, Bothell, Lebiere, & Matessa, 1998) and the fan effect (Anderson & Reder, 1999). They also prove the basis for the difficulty effects in modeling our experiments.

These two features, serial production firing and activation-based retrieval from declarative memory, are the primary differences between ACT-R/PM's underlying production system and EPIC's cognitive processor. In other respects ACT-R/PM and EPIC are quite similar. As in EPIC, the components of ACT-R/PM work in parallel with one another. Also as in EPIC, each of the perceptual-motor modules is essentially serial in nature. The perceptual-motor modules in ACT-R/PM receive commands from the action, or THEN, side of production rules and relay information back to the cognitive system through declarative memory chunks.

The motor module in ACT-R/PM is based directly on the specifications of EPIC's manual motor processor, described in Meyer and Kieras (1997a) and in more detail in Kieras and Meyer (1996). Their formulation, in turn, is based on a synthesis of the motor movement literature (e.g., Abrams & Jonides, 1988; Rosenbaum, 1980; etc.). ACT-R/PM's motor module receives commands from the production system that specify a movement style (e.g., PUNCH, as in punch a key) and the parameters necessary to execute that movement (e.g., LEFT hand and INDEX finger). Movements are divided into two phases, preparation and execution. A movement must be prepared before it can be executed. The motor module is serial in that it may prepare only one movement at a time and may execute only one movement at a time, though it may be preparing one movement while executing another. If a production requests a movement while one is currently being prepared, the request is ignored. Motor preparation is based on movement features. The style of the movement is considered a feature, and each parameter of the movement passed to the motor module is also a feature. Thus, the movement "punch the key below the left index finger" requires the preparation of three features, PUNCH, LEFT, and INDEX. Each feature normally requires 50 ms of preparation.

When the preparation of a movement is complete, the cognition layer is informed by a change in a special declarative chunk representing the state of the motor module. After preparation, the movement may be executed if the motor module is not already executing a movement. If a movement is currently being executed, then the newly prepared movement will be queued and will not be executed until the current movement and all other movements in the queue have been executed. Movement execution takes time according to the nature of the movement itself, with larger aimed movements taking more time than simple "burst" movements such as keypunches. Movement initiation is a subphase of movement execution: After the initial 50 ms of motor execution, the cognitive layer is signaled (via a change in the state chunk) that movement initiation has completed. When movement execution has completed, the cognition layer is signaled again. The speech module is also based directly on EPIC's vocal motor processor and works on the same principles as the motor module in that speech outputs are first prepared and then executed.

ACT-R/PM's vision module is an enhanced version of the ACT-R visual interface (Anderson, Matessa, & Lebiere, 1998). The visual requirements of the models to be discussed here are rather minimal, and so the details of the visual system are not central to our predictions. The key feature of note is the MOVE-ATTENTION operator, which shifts attention in the visual field and produces a chunk representation of the object on which attention has been focused. This operator has a latency, estimated from other models (Anderson, Matessa, & Lebiere, 1998), of 135 ms. Again, this module is serial in that only one item may be attended at a time and only one shift of attention may be executed at a time.

The audition module is a hybrid of EPIC's auditory processor and ACT-R/PM's attention-based vision module. Again, because the auditory requirements are rather minimal in these experiments, the details are less central. This module is again serial in that only one sound may be attended at a time. The parameter of central interest is the "tone recoding time," which is the time it takes the audition module to produce a declarative memory representation of an attended tone. The value of this parameter is estimated from model to model.

In many ways, ACT-R/PM represents a synthesis of ACT-R and EPIC. Traditionally, ACT-R's domain has been higher level cognition, including phenomena such as list memory (Anderson, Bothell, Lebiere, & Matessa, 1998), working memory (Lovett, Reder, & Lebiere, 1999), choice (Lovett & Anderson, 1996), scientific reasoning (Schunn & Anderson, 1998), skill acquisition (Anderson, Fincham, & Douglass, 1999), fan effect (Anderson & Reder, 1999), and the like. One of the most prominent applications of EPIC has been to dual-task interference phenomena, primarily the psychological refractory period (PRP) paradigm (Meyer & Kieras, 1997a, 1997b). Of principal concern, then, is whether ACT-R/PM can successfully model low-level tasks such as the PRP. We address this paradigm and other simple dual tasks in the first half of this article. In the second half of the article we consider three new experiments that involve more complex cognition than is typical of PRP experiments. Because ACT-R has a long history in the memory and problem-solving areas, it is reasonable to expect that ACT-R/PM can cover new ground in between the traditional PRP paradigm and so-called higher level cognition.

Modeling Simple Dual Tasks

The first question raised by ACT-R/PM regards how well it can handle traditional dual-task phenomena, in particular the PRP and perfect time-sharing. The PRP literature is both larger and more well developed than the perfect time-sharing literature and is considered first.

The PRP

The PRP paradigm is a simple form of dual tasking that has been studied extensively in the laboratory for half a century (see Meyer & Kieras, 1997a, and Pashler, 1994, for comprehensive reviews of the PRP literature). PRP experiments require participants to perform two tasks, usually called Task 1 and Task 2, which generally consist of simple responses to the presentation of simple stimuli. Typically, Task 1 and Task 2 are choice reaction tasks (e.g., say the word "high" on detection of a high-pitched tone and the word "low" for a low-pitched tone), and the stimulus modality, response modality, and task difficulty are often manipulated in PRP experiments. Participants are instructed to complete Task 1 before completing Task 2 or to give Task 1 higher priority than Task 2. Finally, there is a delay between presentation of the Task 1

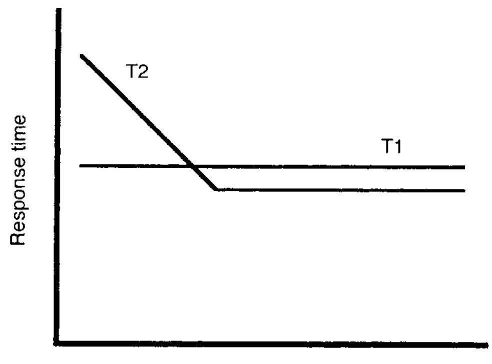


Figure 2. Idealized psychological refractory period (PRP) curve. Task 1 response time (T1) is insensitive to stimulus onset asynchrony (SOA), whereas Task 2 response time (T2) is elevated at short SOAs (the PRP effect).

stimulus and the Task 2 stimulus called the *stimulus onset asynchrony* (SOA). Response times in PRP experiments are typically plotted as a function of SOA, as in Figure 2, which represents somewhat idealized results typical of many PRP experiments.

There are several things to note about Figure 2. First, note that Task 1 is unaffected by SOA, a typical PRP finding. This makes sense in that participants are instructed to give Task 1 priority over Task 2, and flat curves for Task 1 are taken to mean they have done just that. Second, notice the curve for Task 2 (called a *PRP curve*). The Task 2 reaction time is elevated at short SOAs and gradually falls until it is more or less flat as well. The elevation at short SOAs, called the *PRP effect,* indicates some kind of delay in responding to Task 2 and is the source of the term *psychological refractory period.* The PRP effect has been used to argue for a variety of limitations and properties of the human cognitive–perceptual–motor system, including bottlenecks of various kinds (see Meyer & Kieras, 1997a, and Pashler, 1994, for discussions of this and numerous other debates that PRP experiments have sparked). Generally, although there are exceptions, the slope of the PRP curve at short SOAs is very close to -1, which further suggests some kind of bottleneck. That is, if Task 2 had to wait for completion of Task 1, each unit of advance time on Task 1 would mean a unit of time saving on Task 2.

Most relevant to the current discussion, the curves also suggest some kind of parallelism—as with most PRP experiments, the Task 2 reaction time at very small (or 0) SOAs is typically measurably less than the sum of Task 1 and Task 2 reaction times at long SOAs. If the human system were completely serial, the Task 2 response time at 0 SOA should be the simple sum of the Task 1 and Task 2 base reaction times, but it is usually found to be less in PRP experiments. This suggests that there is some kind of overlapping of the Task 1 and Task 2 processing. This is precisely the kind of result that would be impossible to model in ACT-R without the perceptual-motor parallelism of ACT-R/PM.

Response Selection Bottlenecks

PRP results have typically been explained with the notion of a response selection bottleneck (RSB). The idea behind the RSB hypothesis is that, although there is some parallelism in the system that allows parts of the two tasks to be processed concurrently, there is some part of the tasks where such concurrent processing cannot take place. This basic RSB model as described by Pashler (1994) is illustrated in Figure 3. Figure 3 is a schedule chart depicting the stages of processing in the PRP paradigm. Sequential dependencies, such as Task 1 motor processing waiting for Task 1 response selection, are represented with lines connecting the boxes. Finally, the critical path through the chart as a whole, which is the set of stages that determines the total Task 2 response time, is presented with boldface lines and boxes. As this figure illustrates, the assumption is that response selection is performed by central cognition and that central cognition is serial, which produces the bottleneck. That is, the perceptual and motor components of the two tasks can, in principle, be done in parallel, but the cognition for Task 2 must wait for the completion of Task 1 cognition. This relatively straightforward kind of stage model has a great deal of explanatory power and has successfully explained a wide variety of PRP results such as the -1 slope of the PRP curve at short SOAs, the fact that Task 1 is unaffected by SOA, and so on (see Pashler, 1994, for further details). Notice how this account resembles the MHP, described earlier. There are multiple processing streams (perception, cognition, motor) that run in parallel with one another, but each one can perform only one operation or computation at a time.

The EPIC–Strategic Response Deferment Account

Claims of parallelism between the cognitive, perceptual, and motor subsystems are hardly controversial and are almost certainly necessary to explain real-world human performance on complex tasks. Although the RSB model allows for such parallelism, it assumes that cognition is serial. Meyer and Kieras, with their EPIC theory of multiple task performance (Meyer & Kieras, 1997a, 1997b), have questioned whether it is necessary to assume any serial central stage that becomes a bottleneck. In EPIC, cognition for two or more tasks can run entirely in parallel, and all bottlenecks are associated with peripheral (perceptual-motor) processes or explicit scheduling decisions. If EPIC were to perform the two tasks in parallel in a PRP experiment, there would be no guarantee that Task 1 would be completed before Task 2. Therefore, Meyer

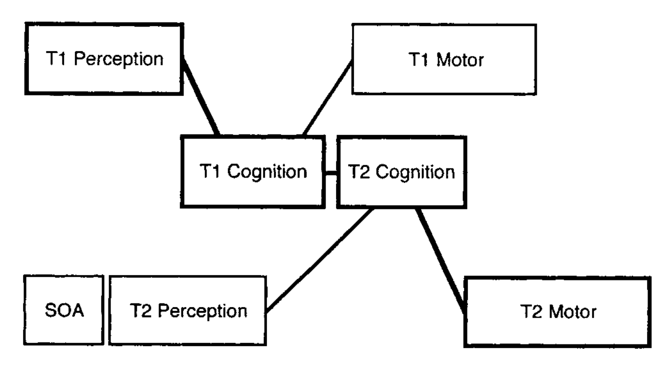


Figure 3. Response selection bottleneck model of the psychological refractory period effect. Because central cognition is serial, the response selection component of Task 2 (T2) is forced to wait for the completion of response selection for Task 1 (T1), thus delaying the T2 response. Boldface lines and boxes indicate the critical path through the chart as a whole. SOA = stimulus onset asynchrony.

and Kieras, (1997a, 1997b) have proposed that participants strategically defer responding to Task 2 before responding to Task 1. That is, some stage of processing of Task 2 is not allowed to proceed until it has been "unlocked" by an executive process that is triggered by the completion of some critical stage of Task 1. Typically, the transmission of the Task 2 response from central cognition to the appropriate output processor, which occurs after response selection, is the stage that must wait. This forms the basis of the EPIC–strategic response deferment (SRD) model of the PRP paradigm. The EPIC-SRD scheduling strategy has several desirable properties. It provides an explanation not only for the PRP effect but for the -1 slope generally found in PRP curves as does the RSB model. However, EPIC-SRD in addition predicts the absorption of difficulty effects at short SOAs.

Subadditive difficulty effects are both subtle phenomena and relatively rare occurrences. In the PRP paradigm, it is common for researchers to use more than one Task 2 in an experiment, a "hard" task and an "easy" task. The difference in response times between the two versions is referred to as the difficulty effect. The RSB hypothesis predicts that, if the source of the difficulty effect is longer response selection time in Task 2, then the difficulty effect should be uniform across all SOAs. This is because the full cost for Task 2 response selection must be paid in all cases, so extending the Task 2 response selection time always produces the same effect regardless of SOA. However, in their careful examination of the literature, Meyer and Kieras (1997a, 1997b) discovered that there were cases in which the difficulty effect was substantially reduced at shorter SOAs (e.g., Hawkins, Rodriquez, & Reicher, 1979). This is referred to as the *subadditive difficulty effect.*

Meyer and Kieras (1997a, 1997b) explained subadditive difficulty effects using their EPIC-SRD theory. In the EPIC-SRD model, at short SOAs the Task 2 processing is waiting after response selection to be unlocked (this is termed *postselection slack*). Under these conditions, lengthening the response selection time for Task 2 should not affect the Task 2 response time as long as the additional difficulty is less than the amount of postselection slack. When the SOA is long, the deferment does not affect Task 2 processing because Task 2 will be unlocked before the end of Task 2 response selection, and thus the full impact of response selection difficulty will appear in the Task 2 response time. There are, of course, conditions under which the analysis is more complicated. If the stimuli for the two tasks are presented in the same modality (e.g., both are visual and an eye movement between the two stimuli is required), then there may be a bottleneck in perceptual processing. A perceptual bottleneck may eliminate the subadditive nature of the difficulty effect by delaying the beginning of Task 2 response selection such that there is no longer postselection slack. That is, the time spent waiting for the response modality to free up causes a delay in the beginning of Task 2 response selection, which results in the full difficulty effect at all SOAs.

The Subadditive Difficulty Effects of Schumacher et al. (1999)

The RSB model does predict subadditive difficulty if the difficulty effect manifests before the bottleneck stage; in particular, when the difficulty manipulation makes the perceptual component of Task 2 more difficult. There is fairly clear empirical support for this phenomenon (e.g., Pashler & Johnston, 1989). However, the empirical support for subadditive difficulty effects that are in response selection or later stages has been less clear (for discussion of this topic, see Meyer & Kieras, 1997a; Pashler, 1994; Van Selst & Jolicoeur, 1997). To clarify this issue, Meyer and Kieras and colleagues (Schumacher et al., 1999) conducted a series of four PRP experiments in which they successfully produced subadditive difficulty effects. These results clearly establish the validity of subadditive, postperceptual difficulty effects as a legitimate empirical phenomenon. Because they are the most illustrative, we consider their Experiments 3 and 4.

In Schumacher et al.'s (1999) Experiment 3, Task 1 was tone discrimination—a 1120-Hz tone versus a 1450-Hz tone were mapped to the left middle and index finger key, respectively. Task 2 was a visual–manual task. In this task, an "O" replaced one of four horizontal dashes on the screen. In the easy condition, participants responded to the "O" appearing in the far left, middle left, middle right, and far right locations with the right index, middle, ring, or little finger key, respectively. In the hard version of Task 2, the same locations were mapped to keys in an inconsistent order: ring, index, little, and middle finger. The results are shown in Figure 4a. In this experiment the difficulty effect dropped from 150 ms at a 1,000-ms SOA to 100 ms at a 50-ms SOA. This was interpreted to be evidence for concurrent Task 1 and Task 2 response selection, but there is a potential caveat. De Jong (1993) proposed a dual-bottleneck theory in which there is a movement production bottleneck if the Task 1 and Task 2 responses share modality. In the case of a movement production bottleneck, subadditive difficulty would be observed even with serial response selection because of the second bottleneck.

Schumacher et al.'s (1999) Experiment 4 was designed to address the multiple-bottleneck hypothesis as well as the standard RSB hypothesis by shifting the response modality for Task 1. Experiment 4 was identical to Experiment 3 except that the Task 1 response was vocal rather than manual. Participants responded to the 1450-Hz tone by saying "high" and to the 1120-Hz tone by saying "low." The results are shown in Figure 4b. At the longest SOA, the difficulty effect is approximately 90 ms, and at the shortest SOA it is approximately 40 ms. This is a difficult result for a serial RSB theory to explain. Although it is possible that the results of Experiment 4 are chance results, the consistency of the Experiment 4 results with the results of Schumacher et al.'s (1999) Experiments 1 through 3 suggests this is not the case.

Schumacher et al. (1999), in fact, argued that these results show that RSBs are not structural or immutable. They implicitly claimed that cognitive parallelism is necessary to explain these results and that the explanation based on EPIC is the correct explanation. However, as we show below, what these data really provide evidence for is strategic deferment that creates slack in which Task 2 difficulty can be absorbed. This can be modeled in either a system with central parallelism such as EPIC or a system with central seriality such as ACT-R/PM.

Explaining Subadditive Difficulty Effects

At first blush, it might appear that ACT-R/PM should not be able to explain subadditive difficulty effects, because the underlying ACT-R production system should not be able to do response selection in parallel for multiple tasks, as ACT-R can fire only one production per cycle. However, as noted by Schumacher et al. (1999), the motor system in EPIC is effectively a serial bottleneck system; each motor unit is itself serial. Because most of ACT-R/

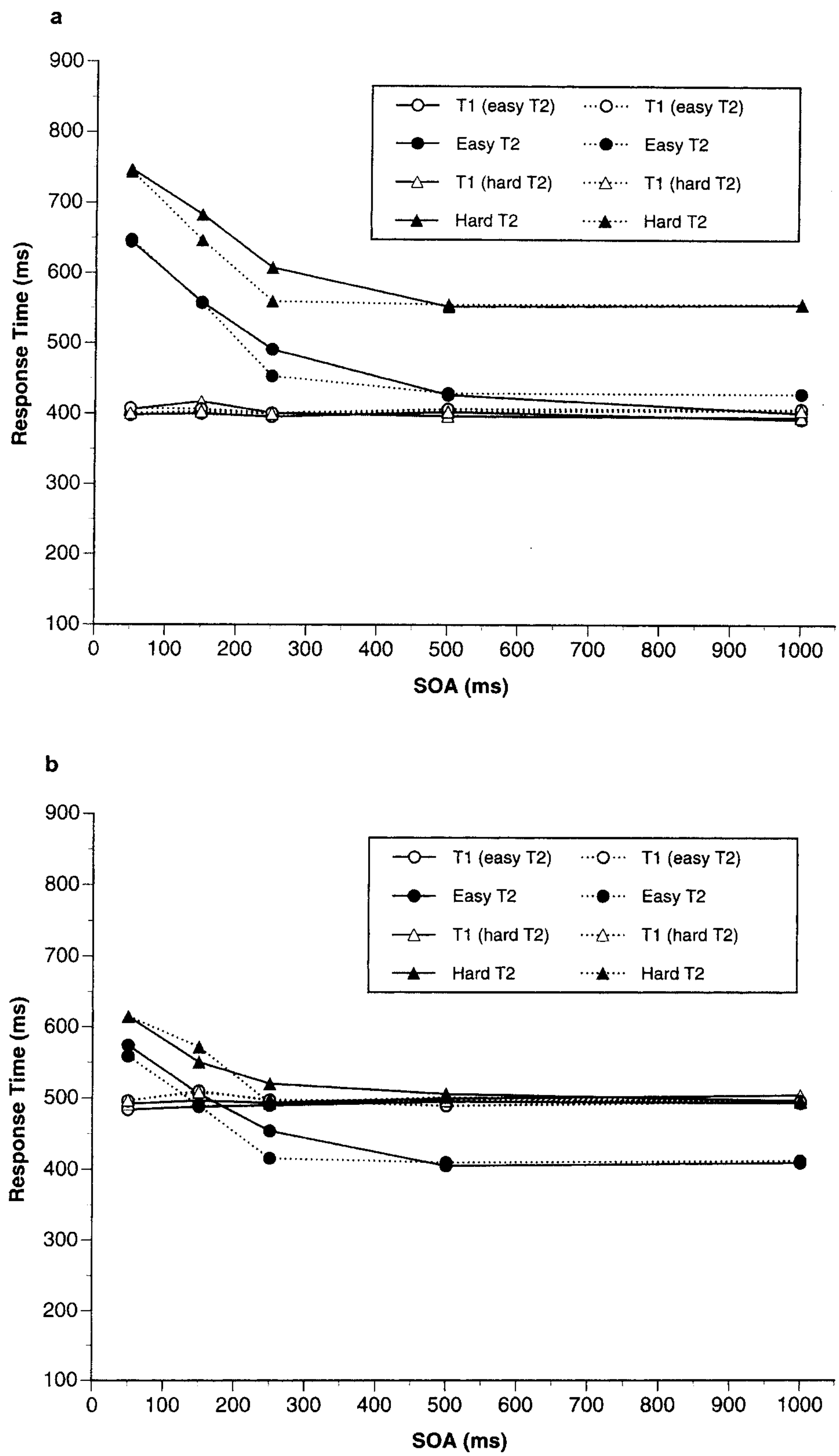


Figure 4. Results of Schumacher et al.'s (1999) Experiment 3 (a) and Experiment 4 (b) along with ACT-R/perceptual-motor simulation results. Data are solid lines; simulation results are dotted lines. T1 = Task 1; T2 = Task 2; SOA = stimulus onset asynchrony.

PM's motor system is based on EPIC's, this makes ACT-R/PM effectively a double-bottleneck system, much like the idea proposed by De Jong (1993). That is, while the cognitive system is waiting for one response to be executed by the motor system, it can be performing the cognition for the second task. This is potentially a slack period in which Task 2 difficulty effects can be absorbed at short SOAs. Thus, ACT-R/PM can predict the subadditive effects in Schumacher et al.'s (1999) Experiment 3 (their Experiments 1 and 2 have the same property) because of the need to avoid jamming the motor output with multiple requests.

Schumacher et al.'s (1999) Experiment 4 is different because it involves different output modalities. The key insight for explaining Experiment 4 is that in the case in which the responses are in different modalities, there is a danger of making the responses in the wrong order just because one cannot count on the seriality of the output modality to order the responses. If the Task 2 response selection is very rapid, then the system is capable of producing the Task 2 response before the Task 1 response, despite the bottleneck in central cognition. That is, if the motor system takes time to process the output command (e.g., punch the left ring finger) for Task 1, which it does in EPIC or ACT-R/PM, then the Task 2 response could possibly occur first even if the command to output the Task 2 response was given by the cognitive system at a later time than the Task 1 command. For instance, if the Task 2 response is a repeated movement (e.g., press a key again) or some other movement with a very short preparation time, the Task 2 response can be output before the Task 1 response even under the assumption of cognitive seriality, simply because the output modules run in parallel with one another. Of course, the Task 1 response will have a head start because Task 1 will get through the bottleneck first, but this does not guarantee that the response will be emitted first because the completion times for the various stages are noisy.

Thus, even with a cognitive bottleneck, experimental participants may adopt a lock-out strategy similar to the one proposed in the EPIC-SRD model. That is, they may strategically delay one stage of Task 2 processing until a particular stage of Task 1 processing is complete either to avoid jamming a single output modality (Experiment 3) or to guarantee correct response order (Experiment 4). This strategic deferment is particularly likely in experiments in which Task 2 does not involve substantial demands on central cognition because one cannot count on the cognitive components of Task 2 to provide sufficient delay. Note that the easy version of Task 2 in their Experiment 4 finishes 100 ms (20%) faster than Task 1, which suggests that this analysis is appropriate in this case.

Providing a strategic deferment introduces pre-unlocking slack in Task 2 processing, into which a difficulty effect may be absorbed. The parallel operation and sequential dependencies between the various task components in the ACT-R/PM lock-out model are difficult to analyze and discuss clearly in text, and so to aid in presentation of this argument, we refer to the schedule chart in Figure 5. This schedule chart uses the same conventions as Figure 2, but in this schedule chart, box length is proportional to the time taken by that stage. Stages that must wait for one another because the processing required makes use of a bottlenecked processor occupy a row in the chart. The rows have been labeled with the processor they represent. The narrow boxes after motor initiation are key detect times.

This chart assumes that response selection and response transmission for Task 1 can be done in a single production cycle, with a production of the following form:

IF the goal is to do the dual task
  and the current task is Task 1
  and the tone is 1120 Hz
THEN send "punch left index finger" to the motor module.

There is a similar production for the 1450-Hz stimulus in Task 1. The corresponding Task 2 productions are not allowed to fire, however, until the unlock production has fired. The unlock production checks on the progress of Task 1, and, when Task 1 has reached some critical point (in this case the completion of motor initiation for the Task 1 movement), it allows Task 2 to proceed. In the case of Task 2, response selection and response transmission

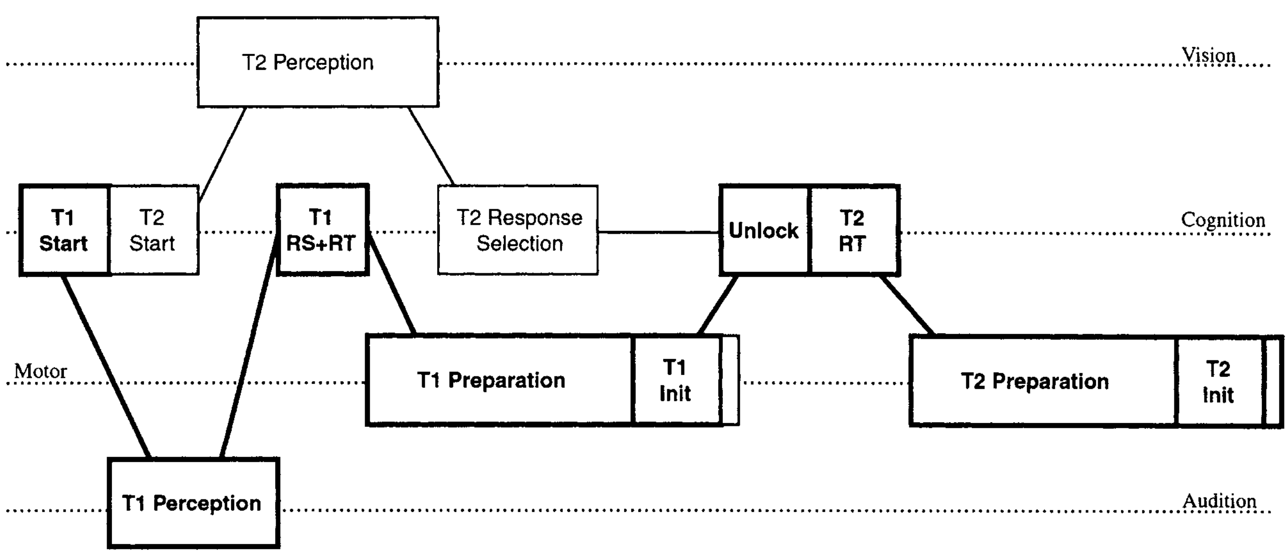


Figure 5. Schedule chart for ACT-R/perceptual-motor model of Schumacher et al. (1999), Experiment 3. T1 = Task 1; T2 = Task 2; RS = response selection; RT = response transmission; Init = motor initiation.

are separated and occur in two productions. The first production, response selection, is allowed to proceed before Task 2 is unlocked. This production retrieves the mapping between tones and fingers:

IF the goal is to do the dual task
  and there is a "O" in position X
  and there is a finger associated with position X
THEN note the finger.

A second production is responsible for actual response transmission:

IF the goal is to do the dual task
  and Task 2 is unlocked
  and a finger ⟨f⟩ has been noted
THEN send "send punch right ⟨f⟩ finger" to the motor module.

Because the response transmission production waits for unlocking of Task 2, there is potentially slack between the completion of response transmission for Task 1 and the unlocking of Task 2. The response selection production for Task 2 can fire if it completes in this slack time. When that occurs, response selection for Task 2 is overlapping with motor preparation for Task 1. Under these conditions, there will be no difference between the hard and easy version of Task 2, because the difficulty will be absorbed into this pre-unlocking slack. Note in Figure 5 that the "T2 Response Selection" box could grow considerably before affecting the overall Task 2 response time. Because the time for feature preparation for the Task 1 response is noisy, and the retrieval is itself noisy, this production will sometimes fail to fire in the pre-unlocking interval. As the SOA increases, the response selection production will have to wait longer to fire, and thus there will be less slack time in which it could fire, until the SOA becomes long enough that the response selection production always fires after unlocking. When this happens, the full difficulty effect is realized in the Task 2 response time. Thus, this model predicts a subadditive difficulty effect: full difficulty effects at long SOAs, with some absorption of the difficulty effect at shorter SOAs.

Thus, ACT-R/PM is able to predict subadditive difficulty effects in PRP in much the same way as EPIC because it has adopted the similar design of multiple serial output modules running in parallel and the EPIC-SRD principle of strategic lock-out. It adopts this principle to avoid the problem of jamming (Experiment 3) and to guarantee output order (Experiment 4). Although this shows how much we have borrowed from EPIC, it also calls into question the extent to which subadditive difficulty effects in PRP experiments with little cognitive involvement provide evidence for cognitive parallelism, because ACT-R/PM explains these effects without requiring cognitive parallelism.

So far we have given only a qualitative argument for how ACT-R/PM can produce these results. However, being able to explain such results in principle is not the same as providing quantitative fits. Thus, we constructed detailed ACT-R/PM models of the experiments in Schumacher et al. (1999). These models make the following assumptions.

Assumption 1. Uniform unlocking: All models make use of exactly the same unlocking strategy. That is, the response transmission production for Task 2 must wait for Task 2 to be explicitly *unlocked. Strategic unlocking is to avoid* response jamming in Experiment 3 and to guarantee response order in Experiment 4. The unlocking production always checks for the same state of Task 1, which is the completion of motor initiation of the Task 1 response. Unlocking takes exactly one production cycle, rather than the indeterminate amount of time taken for unlocking in the EPIC-SRD models. This potentially eliminates one free parameter.

Assumption 2. Task 1 response selection and transmission: Response selection and response transmission for Task 1 always occur in the same production. Because there is no deferment of any kind for Task 1, there is no reason to separate response selection from response transmission; doing so would only degrade Task 1 performance.

Assumption 3. Task 2 response selection and transmission: The ACT-R/PM models do response selection and response transmission similar to the EPIC-SRD model. If Task 2 has been unlocked, then response selection and response transmission happen in the same production. If Task 2 has not been unlocked, then these occur in separate productions.

Assumption 4. Dual-task goal: Dual-task performance is somewhat more complex than the simple union of the two tasks. As illustrated in the productions above, we assume that there is a single goal to perform the dual task that controls strategic deferment, rather than multiple single-task goals. Use of a special dual-task goal to manage both tasks is motivated by the need to overlap processing between the two tasks, which is not possible with alternative strategies such as a main goal that pushes subgoals for each of the two tasks.

At most, three data points are fit for each experiment: the Task 1 response time at the highest SOA, the easy Task 2 response time at the highest SOA, and the hard Task 2 response time at the highest SOA. Thus, the magnitude of the difficulty effect at the largest SOA is fit, but the size of the difficulty at shorter SOAs is not explicitly fit—it emerges from the scheduling policy and system dynamics. Thus, the size of the subadditive difficulty interaction is not fit but rather is predicted by ACT-R/PM. With this relatively simple set of assumptions in hand, we constructed and ran full ACT-R/PM models of the four experiments from Schumacher et al. (1999). Because completion times for the two tasks are noisy, 200 Monte Carlo runs of the model were performed at each difficulty condition and each SOA. The parameters used in these models are given in Table 1. The production rules and supporting LISP code used for the simulation models are available at http://chil.rice.edu/byrne/smp/. Detailed descriptions and models of Experiments 1 and 2, not discussed in this article, can also be found on this site.

Model of Experiment 3. The model for Task 1 requires one production cycle to initiate auditory attention (time t_c) followed by an audio recoding time (t_a). Then, there will be another production cycle for response selection and transmission (time t_c) followed by motor feature preparation (time $n_f * t_f$, where n_f is the number of features and t_f is the time per feature), motor initiation (time t_i), and a key closure (time t_k). Thus, $rt_1 = t_c + t_a + t_c + (n_f * t_f) + t_i + t_k = 50 + 95 + 50 + (3 * 50) + 50 + 10 = 405$ ms. Audio recoding time (t_a) was estimated to be 95 ms to fit the data. At long SOAs, when Task 2 has been unlocked before the onset of the Task 2 stimulus, the response time for Task 2 should be $rt_2 = t_c + t_v + r(a_c) + t_c + (n_f * t_f) + t_i + t_k = 50 + 135 + r(ac) + 50 + (2 * 50) + 50 + 10 = 395$ ms $+ r(a_c)$, that is, one cognition cycle to initiate perception, one shift of visual attention (time t_v), one production cycle to do response selection and transmission that requires a retrieval from long-term memory denoted $r(a_c)$, motor

Table 1
Parameters Used in the Models of Schumacher et al. (1999)

Parameter	Value	Description
	Parameters common to all four Schumacher et al. (1999) models	
t_c	50 ms	Time for one production cycle (minimum cognition time)
t_v	135 ms	Time to shift visual attention
t_f	50 ms	Time to prepare a movement feature
t_i	50 ms	Motor initiation time
t_k	10 ms	Key closure time
t_d	100 ms	Voice key closure time
F	1.0	Latency factor (in retrieval time equation)
t_a	95 ms	Tone recording time
s	0.5	Activation noise
n_{f1}	3	Number of movement features to be prepared in Task 1
n_{f2}	2	Number of movement features to be prepared in Task 2
	Parameters used in the model of Schumacher et al. (1999), Experiment 3	
a_{ce}	3.8	Activation of S-R mapping chunks in easy condition
a_{ch}	2.3	Activation of S-R mapping chunks in hard condition
	Parameters used in the model of Schumacher et al. (1999), Experiment 4	
a_{ce}	4.5	Activation of S-R mapping chunks in easy condition
a_{ch}	2.6	Activation of S-R mapping chunks in hard condition

Note. S-R = stimulus–response.

feature preparation time, motor initiation time, and finally key closure time. The difference between the easy and hard conditions in this model is the $r(a_c)$ term, which is a function of the activations of the chunks that map positions to keys.

Three parameters were estimated: the tone recoding time (relevant for Task 1), estimated at 95 ms, and the base-level activation of the chunks that map stimulus to response for Task 2 for easy and hard versions, estimated at 3.8 and 2.3 units of activation, respectively. Comparison of the simulation results and the actual data are shown in Figure 4a. For the fit from data to model $r^2 = .98$ with a root-mean-square (rms) error of 18.1 ms. No parameters were estimated to fit any data points at lower SOAs. Although the model slightly underpredicts response times for Task 2 at intermediate SOAs, the overall fit is very good and the model predicts the subadditive difficulty effect accurately.

Model of Experiment 4. The model of Experiment 4 is similar to the model for Experiment 3. Task 1 was auditory tone discrimination with vocal responses, which this time required the detection of the tone (t_d) via voice key: $rt_1 = t_c + t_a + t_c + (n_f * t_f) + t_i + t_d = 50 + 95 + 50 + (3 * 50) + 50 + 100 = 495$ ms.

Task 2 was identical to the Task 2 used in Experiment 3 and thus is based on the same equation. However, because the longest SOA times were slightly different in this experiment, slightly different activation values were used to determine r(ac). Overall, two parameters were estimated, which were the base-level activations of the chunks that map stimulus to response for Task 2. This was estimated to be 4.5 for the easy version and 2.6 for the hard version. Tone recoding time for Task 1 was not estimated for this data set; rather, the 95-ms value used for Experiment 3 was reused here because the tones used were the same tones used in Experiment 3. Comparison of the simulation results and the actual data are shown in Figure 4b. For the fit from data to model $r^2 = .91$ with an rms error of 14.4 ms. Again, no parameters were estimated to fit any data points at lower SOAs. As was the case for the model of Schumacher et al.'s (1999) Experiment 3, the model for Experiment 4 does an excellent job of predicting not merely the presence of a subadditive difficulty effect but the magnitude as well.

Overall, the ACT-R/PM models provide an excellent match to the data, particularly considering that only the points at the longest SOAs are estimated. In some sense, these models represent an implementation of the EPIC-SRD model in ACT-R/PM. However, unlike in EPIC, cognition for the two tasks never runs in parallel. For very simple tasks with small cognitive demands such as the ones typically found in PRP experiments, it may not be possible to distinguish between parallel and serial models. However, unlike EPIC, the lock-out model implemented in ACT-R/PM cannot produce subadditive difficulty effects in cases in which the cognitive demands are more sizable as in the arithmetic PRP experiment of Byrne and Anderson (1998). In such cases, there is no cognitive slack time, not even in the easy version of Task 2, and thus no time into which the difficulty effect can be absorbed. The experiments that we report below look at tasks with more sizable cognitive demands.

Perfect Time-Sharing

Perfect time-sharing refers to situations in which two or more tasks can be carried out in a multiple-task situation just as fast as in isolation. Several experiments have demonstrated such effects (e.g., Allport, Antonis, & Reynolds, 1972; Schumacher et al., 1997). EPIC predicts that perfect time-sharing is possible where there is a lack of input interference (e.g., stimuli for the two tasks are in different modalities) and no output interference (e.g., responses are in different modalities)—that is, when there is no peripheral (i.e., perceptual-motor) interference. It is important to note that EPIC does not predict perfect time-sharing in all such cases because participants may voluntarily adopt lock-out strategies that prevent perfect parallelism. The conditions under which such strategies will and will not be adopted are not specified in EPIC, but it is clear that perfect time-sharing is possible. The question, then, is whether or not perfect time-sharing is possible for ACT-R/PM. In fact, there are circumstances under which ACT-R/PM can produce perfect time-sharing. Consider for now the dual-task case. ACT-R/PM predicts that perfect time-sharing can occur when cognition for the two tasks involved need not occur at the same time. This can happen if the cognition for each task can occur during the perceptual or motor processes for the other task. This is unlikely to occur in tasks with even fairly moderate cognitive demands, such as arithmetic fact retrieval, but can occur when the cognitive demands are minimal. Like EPIC, ACT-R/PM also requires no peripheral interference to generate perfect time-sharing.

The Schumacher et al. (1997) experiment is such an instance of perfect time-sharing. It involved two simple choice reaction time tasks: three-choice (low–middle–high) tone discrimination with a vocal response and three-choice (left–middle–right) visual position discrimination with a manual response. Both of these tasks are simple and can be completed very rapidly by experimental participants. Schumacher et al. (1997) had experimental participants train on these two tasks separately, and they reached average response times of 445 ms for the tone discrimination task and 279 ms for the location discrimination task. Participants were then asked to do the two tasks together with simultaneous stimulus presentation and were encouraged to overlap processing of the two stimuli. In the dual-task condition, they experienced virtually no dual-task interference—283-ms average response time for the visual–manual task and 456-ms average response time for the auditory–vocal task.

We constructed an ACT-R/PM model of the two tasks and the dual task. The schedule chart for the dual-task model is presented in Figure 6. Note that there is no critical path designated in this chart because there is no dual-task interference. There are three models altogether: the single-task model for the auditory–vocal tone discrimination task, the single-task model for the visual–manual position discrimination task, and the dual-task model. The models are very simple: They perceive the stimulus, select a response, and produce that response. What is significant about the models is that they indicate that perfect time-sharing should be possible in this case. Because no stimulus properties of the visual stimulus other than location have to be identified, perceptual processing of the stimulus for the location discrimination task happens very rapidly. Response selection for this task goes on in parallel with perceptual processing of the auditory stimulus and completes before response selection for the tone discrimination task would normally begin.

Similarly, response selection for the tone discrimination task goes on in parallel with the motor preparation for the position discrimination task. The key features that enable perfect time-sharing in this task are (a) the rapid perceptual component that enables response selection for the visual–manual task to take place during auditory perception and (b) the lack of ordering constraints on the responses that enables response selection for the auditory–vocal task to proceed in parallel with motor preparation for the visual–manual task without lock out.

Simulation results are presented in Figure 7. Because completion times for the various stages are noisy, the times in the figure are based on 200 Monte Carlo runs. Only one parameter was estimated for this model, the tone recoding time for the auditory–vocal task, at 50 ms. This was again estimated from the single-task data; no additional parameters were estimated for the dual-task model. As can be seen in the figure, the model exhibits nearly perfect time-sharing. There is a very slight increase in response time associated with the dual-task condition for the tone detection task, but this effect is minimal. What is most important is that the

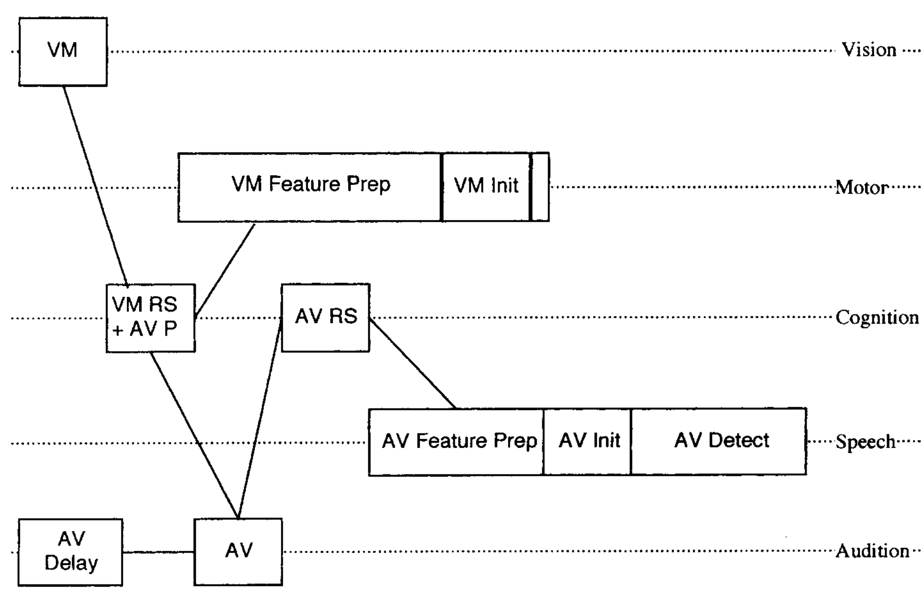


Figure 6. Schedule chart for Schumacher et al. (1997) perfect time-sharing model. VM = visual–manual task; *Prep* = *preparation; Init* = motor initiation; RS = response selection; AV P = auditory–verbal perception; AV = auditory–verbal task.

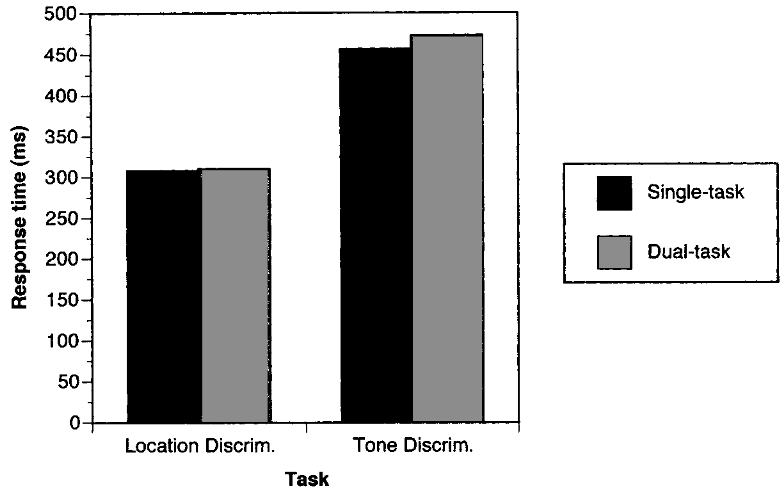


Figure 7. ACT-R/perceptual-motor simulation of Schumacher et al. (1997) perfect time-sharing results. Discrim. = discrimination.

model does not predict large effects for either task, despite serial cognition. Parallelism between cognition and the various perceptual-motor modules is sufficient to predict perfect time-sharing in cases of minimal cognitive demand. However, ACT-R predicts that such perfect time-sharing will not be found in the experiments we report below because they involve much more substantial cognitive processing.

Complex Dual Tasks

We conducted a series of experiments designed to bridge the gap between the relatively simple tasks typical of the PRP literature and more high-level memory-oriented tasks to which ACT-R has been applied. Because EPIC is arguably the premier computational theory of low-level dual-task performance and ACT-R is arguably the premier computational theory of memory and problem solving, the space where those two domains meet should be ideal for ACT-R/PM, a synthesis of the two. The first step in constructing a theory of dual-task interference in more complex situations is understanding the empirical phenomena. There are presently few data available in this area that are detailed enough to provide a basis for modeling work. Therefore, we first review an extant experiment of ours in this area and present three new experiments on this topic.

Byrne and Anderson (1998): PRP Arithmetic

Byrne and Anderson (1998) reported a PRP experiment that involved cognitively more demanding tasks than are typically seen in the PRP paradigm. In our experiment, which we refer to as the PRP arithmetic experiment, Task 1 involved retrieving the product of two auditorily presented digits and responding with a spoken answer. Task 2 involved single-digit addition tasks in which a particular sum was presented—for example, 6 + 7 = 12 or 6 + 7 = 13—which participants responded to with one key if the assertion was correct and with another key if it was in error. Difficulty of the second task was manipulated by giving participants either large digits (hard) or small digits (easy). Figure 8 shows the results of the PRP arithmetic experiment as a function of the SOA between the first and second task and the predictions of ACT-R/PM (to be described later). There are three aspects of these data that may be considered surprising from the perspective of the EPIC-SRD explanation. First, there was no systematic interaction between SOA and the difficulty of the second task. That is, there is not the expected subadditive difficulty effect at short SOAs.

Second, we can compare the time for Task 2 at 0 SOA with the sum of the time for Task 1 and for Task 2 at long SOAs where the participants have reached asymptotic speed. The sum is only slightly less than the sum of the times for each task individually. For easy addition, the 0 SOA time is 2,070 ms, and the sum of the

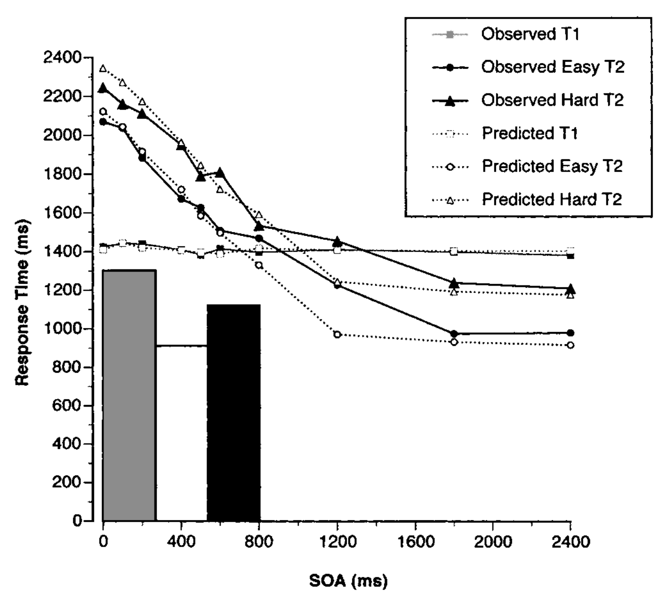


Figure 8. Psychological refractory period arithmetic experiment results from Byrne and Anderson (1998), both data and model. Single-task response times are displayed in the bars in the lower left (gray for multiplication, white for easy verification, and black for hard verification). T1 = Task 1; T2 = Task 2; SOA = stimulus onset asynchrony.

asymptotic times is 2,362 ms (this is the sum of 1,381 ms, the Task 1 time at an SOA of 2,400 ms, and 981 ms, the Task 2 time at an SOA of 2,400 ms), a difference of only 292 ms. For hard addition, the 0 SOA time is 2,247 ms, as compared with the 2,592-ms sum of the asymptotic times (1,381 ms and 1,211 ms), a 345-ms difference. Thus, despite the fact we greatly increased the cognitive component, we saw very little time-sharing.

Third, we can compare the multiplication (Task 1) times and the asymptotic addition (Task 2) times in Figure 8 with the times to perform these tasks in isolation. These single-task times are shown as bars in Figure 8 for visual comparison. Single-task multiplication took an average of 1,303 ms; single-task easy addition, 913 ms; and single-task hard addition, 1,123 ms. These times are all reliably smaller than the corresponding asymptotic dual-task times (1,381 ms, 981 ms, and 1,210 ms, respectively). This dual-task slowing is not predicted by EPIC or by traditional RSB models.

These results all suggest that increasing the complexity of the cognition involved in the tasks changes the nature of the dual-task interference. An ACT-R/PM model of this experiment was constructed and reported in Byrne and Anderson (1998), and the fit of the model to the data is also presented in Figure 8. As detailed there, the model explains all three of these unexpected effects on the basis of serial cognition and effects of diffuse activation (more details on diffuse activation are presented later). We next describe three new experiments that replicate and extend these results, and we describe ACT-R/PM models for the new experiments. As a preview, these experiments give no evidence for cognitive parallelism. We do not claim these experiments disprove EPIC because cognitive parallelism is always optional in EPIC, and so it makes no firm predictions. We do claim, however, that the results are predicted by the ACT-R/PM architecture.

Experiment 1: Parallel Arithmetic

Experiment 1, which we refer to as the parallel arithmetic experiment, was essentially a replication of the PRP arithmetic experiment reported in Byrne and Anderson (1998) with one major change: Participants were given no instructions regarding response order. That is, they were told only to complete both tasks as rapidly as possible, regardless of order (they were also warned not to group their responses). Because the two tasks had different input modalities (visual and auditory), there should be no bottleneck in input modality. Just as important, the tasks did not share output modality (one manual, one verbal). According to the ACT-R/PM model, perfect time-sharing should be impossible here because of the increased cognitive demands. However, according to EPIC, it may be possible to do these tasks completely in parallel if the primary source of the interference seen in Byrne and Anderson is strategic.

Method

Participants. Participants were 30 Carnegie Mellon undergraduates who participated for credit in a psychology course.

Stimuli and procedures. Participants were presented with two tasks: multiplication and addition. The multiplication task consisted of the auditory presentation of two 1-digit numbers, to which participants were to respond verbally with the product of the two numbers. Each number was digitized audio of a person's voice, and the audio clips were normalized for volume and length, which was fixed at 400 ms. There was a 500-ms pause between the completion of the first audio clip and the onset of the second clip. Response time was measured from the onset of the second digit. Multiplication problems were randomly generated and used the numbers from one to nine. Problems never used the same number for both multiplicands, the number one was never used as the first digit presented, and participants never received the identical problem on consecutive trials.

The addition problems were single-digit addition verification problems, presented visually. Each problem consisted of an addend, the plus sign, the second addend, the equal sign, and a result (e.g., 6 + 3 = 10). If the result was the correct answer to the addition problem, participants were to respond by pressing the 6 key on the numeric keypad section of the keyboard. If the result was incorrect, participants were to respond by pressing the 4 key on the numeric keypad section of the keyboard. Participants were instructed to use two different fingers (to prevent hand movements) of their right hand for the two keys. Addends were randomly selected digits with the constraint that digits were not repeated within a problem. For foils, the result was off from the correct answer by 2, 1, −1, or −2, and the amount of deviation was randomly selected. Half the trials were foils and half targets, randomly ordered.

Participants first completed 120 practice trials consisting of 40 multiplication-only trials, then 40 addition-only trials, and then 40 dual-task trials. After practice, participants received three blocks of trials. Each block was divided into five sets of 40 trials: one multiplication-only set, one addition-only set, and three dual-task sets. Ordering of sets within a block was random. Dual-task trials had 10 levels of SOA. SOA was defined as the amount of time elapsed between the onset of the second digit in the multiplication problem (audio) and the presentation of the addition problem. Ten different SOAs were used: −400, −200, −100, 0, 100, 200, 400, 700, 1,100, and 1,600 ms. Negative SOAs (meaning the addition problem appeared before the onset of the second digit of the multiplication problem) were included to encourage participants to overlap processing as much as possible.

An Apple Power Macintosh 8500/120 with a standard Apple PlainTalk microphone was used to present all stimuli and record all data. Audio stimuli were presented using the computer's built-in speaker. Visual stimuli were presented in 18-point sans serif (Helvetica) text on an Apple 17-in. (43.2 cm) color monitor. Latencies were measured by means of active software polling of the microphone level and keyboard state, bypassing the operating system's event-handling overhead and providing accuracy of approximately ±5 ms.

Results

An alpha level of .01 was used for all tests. Error rates were low (around 5%) for all participants and are not considered. Error trials and outliers (defined as more than 3 standard deviations from each participant's mean) were excluded from all analyses. Figure 9 presents the standard PRP curve, showing both multiplication and addition performance as a function of SOA, aggregating across blocks. It also presents as bars the times for the single tasks. Note that the effect of SOA on addition was substantial—much like the PRP effect in Figure 8—with participants speeding up by approximately a full second from an SOA of −400 to an SOA of 1,600. Responses to addition problems were consistently faster than responses to the multiplication problem at long SOAs.

Both the EPIC-SRD account and traditional RSB accounts suggest that there should be no increase in response time due to the fact that the tasks are executed in a dual-task situation (i.e., no slowing of the first task and no slowing of the second task at long SOAs). In the PRP arithmetic experiment, a dual-task slowdown was found for both Task 1 and Task 2 at long SOAs. Because there were effects of SOA on both addition and multiplication in this experiment, it is appropriate to compare both Task 1 and Task 2 only at the largest SOA, where participants were presumably at or

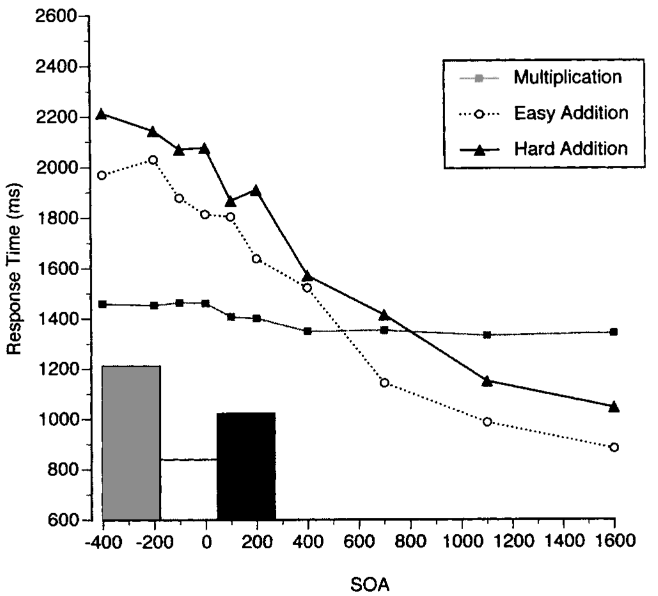


Figure 9. Psychological refractory period curve for the parallel arithmetic experiment, showing response time as a function of stimulus onset asynchrony (SOA) for multiplication and both hard and easy verification. Single-task response times are displayed in the bars in the lower left (gray for multiplication, white for easy verification, and black for hard verification).

near asymptotic speed (as the longest SOA is considerably longer than the single-task times). Despite this conservative procedure, multiplication was approximately 135 ms slower in the dual-task condition than in the single-task condition (1,341 vs. 1,207 ms), which was a reliable difference, $F(1, 29) = 15.53, p < .001$. Also like the PRP arithmetic results, the dual-task times are longer for addition, but this increase (approximately 60 ms) was not statistically reliable, $F(1, 29) = 0.80, p = .38$.

Unlike the PRP arithmetic results, SOA did have an effect on multiplication response time, $F(9, 261) = 9.23, p < .001$. This effect is likely due to the fact that participants sometimes responded to the addition task first. Figure 10 presents the proportion

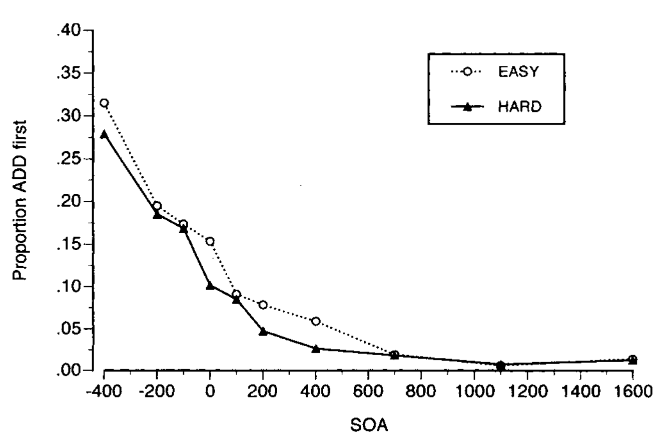


Figure 10. Proportion of trials in the parallel arithmetic experiment where response to the addition verification (ADD) preceded response to the multiplication versus stimulus onset asynchrony (SOA), by difficulty.

of the trials on which participants responded to the addition problem (Task 2) before responding to the multiplication problem (Task 1). As can be seen in Figure 10, they responded to the addition task first more often at negative SOAs and for easy addition problems, and the effect of SOA is indeed reliable, $F(9, 261) = 19.02, p < .001$, as is the effect of difficulty, $F(1, 29) = 12.33, p = .001$, though the interaction is not, $F(9, 261) = 1.77, p = .074$.

In fact, if such trials are eliminated, many of these effects disappear. Limiting the data to only trials on which multiplication was responded to first produces the graph in Figure 11. The data are generally tidier in this graph. The data are also remarkably similar to the data from the PRP arithmetic experiment presented in Figure 8. Note that there are large effects of SOA on latency, and the slope from −400 ms to 700 ms is very close to −1. Thus, it does not seem that participants behave any differently when the requirement to respond in a particular order is removed. When they respond to addition second, it appears they must wait for multiplication to complete just as when they were instructed to do so.

Another key result for the purposes of the present discussion is the addition response time in Figure 11. If cognition for the two tasks can go on in parallel, in the absence of explicit instructions to wait for the completion of Task 1, one might expect that Task 2 should not be slowed by temporal overlap with Task 1. This is clearly not the case, as the effect of SOA is quite robust, $F(9, 261) = 226.07, p < .001$. Further, although there was again an effect of difficulty, $F(1, 29) = 28.61, p < .001$, and an interaction between difficulty and SOA, $F(9, 261) = 5.55, p < .001$, there is no statistical evidence that the difficulty effect is smaller at shorter SOAs. The difficulty effect is approximately 175 ms for the three negative SOAs, approximately 200 ms for the three positive SOAs

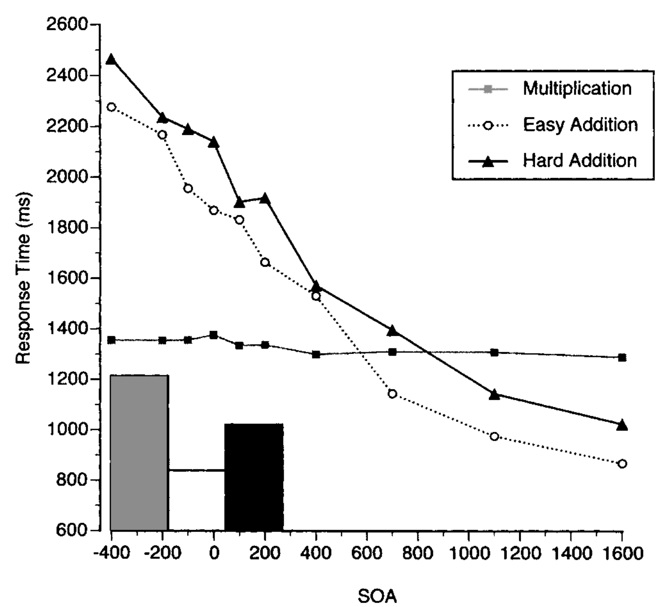


Figure 11. Psychological refractory period curve for the parallel arithmetic experiment when only multiplication-first responses are considered, showing response time as a function of stimulus onset asynchrony (SOA) for multiplication and both hard and easy verification. Single-task times are displayed in the bars in the lower left (gray for multiplication, white for easy verification, and black for hard verification).

between 0 and 200 ms, and approximately 150 ms for the four longest SOAs.

Discussion of Parallel Arithmetic

Despite the lack of scheduling instructions, the PRP effect observed here, much like the one observed in the Byrne and Anderson (1998) experiment, was quite large. In fact, there was very little dual-task savings whatsoever, even at SOAs around 0. The data in Figure 11 from the parallel arithmetic experiment are remarkably similar to the data from the standard PRP arithmetic experiment in Figure 8. This suggests that Task 2 slowing is not just a strategic decision of participants in response to standard PRP instructions. It is not inconceivable that an EPIC model of this task would produce perfect time-sharing, because there is no incentive for participants to adopt a strategic deferment and the tasks do not share input or output modalities. However, because the onset of the first auditorily presented number for the multiplication task preceded the visual presentation of the equation for the addition verification task, participants may have developed a strategic bias and simply chosen to respond to the multiplication problem first. Because the auditory stimulus was transient and the visual was not, this was also the sensible ordering if participants were going to order their encoding. The next experiment looks at a situation in which participants could not induce a task ordering by ordering the encoding of the stimuli.

Experiment 2: Fixed–Free Arithmetic

The results of the parallel arithmetic experiment suggest there is more to dual-task interference than merely strategic postponement. However, it is not the most direct test of people's ability to perfectly time-share. This is because experimental participants may have scheduled the two tasks to minimize the opportunity for cognitive overlap by not processing the Task 2 stimulus until they had completed Task 1. Experiment 2 addresses this issue by combining the Task 1 and Task 2 stimuli into one stimulus.

Further, the effects of the instructional manipulation are primarily inferred by cross-experiment comparison to the PRP arithmetic experiment in Byrne and Anderson (1998). Experiment 2 more directly assesses the impact of the task instructions by adding a between-subjects manipulation. Half the participants were instructed to respond in a specific order, as is typical in PRP experiments, and half the participants were instructed to respond in any order they wanted (though they were instructed not to group their responses).

Method

Participants. Participants were 34 Carnegie Mellon undergraduates who participated for credit in a psychology course.

Stimuli and procedures. On each experimental trial, participants were presented with three digits, the first and last of which were underlined (e.g., <u>3</u> 4 <u>7</u>). There were two single tasks: addition verification and arithmetic production. Addition verification proceeded as in the parallel arithmetic experiment: Participants were to judge whether or not the first two digits (left to right) summed to the third. If they did, participants were to respond by pressing the 6 key on the numeric keypad section of the keyboard. If the third digit was not the sum of the first two, participants were to respond by pressing the 4 key on the numeric keypad. Participants were instructed to use two different fingers of their right hand (to prevent hand movements) for the two keys. Again, addends were randomly selected digits with the constraint that digits were not repeated within a problem. For foils, the result was off from the correct answer by 2, 1, −1, or −2 and the amount of deviation was randomly selected such that the stimulus presented always consisted of exactly three digits.

The arithmetic production task involved verbally reporting either the product or the sum of the outer (first and last) digits. If the block of trials was an addition block, participants would respond "10" to the <u>3</u> 4 <u>7</u> stimulus and "21" if the block of trials was a multiplication block. On the basis of past research, it was anticipated that the multiplication problems would be more difficult, and so this was intended as a difficulty manipulation. There were also dual-task trials. In dual-task trials, participants were asked to do both the verification and the production tasks for each stimulus.

Participants first completed 200 practice trials consisting of 40 verification-only trials, then 40 production-only trials doing addition, then 40 trials of production-only with multiplication, then 40 dual-task trials doing addition as the production task, then 40 dual-task trials with multiplication as the production task. After practice, participants received three blocks of trials. Each block was divided into five sets of 40 trials: one verification-only set, one addition production-only set, one multiplication production-only set, one dual-task set with addition as the production task, and one dual-task set with multiplication as the production task. Ordering of sets within a block was random.

Materials. Materials were identical to those used in the parallel arithmetic experiment.

Design. Roughly half (16) of the participants were instructed to respond in the dual-task condition in whatever order they wanted. This was termed the free condition because participants were free to respond in any order. The other half (18) of the participants were instructed to respond to one task (the verification problem) first, as is typical in PRP experiments. This was termed the fixed condition, because participants were required to respond in a fixed order. Note that in the fixed condition, this experiment is similar to a PRP experiment with an SOA of 0, except that both Task 1 and Task 2 make use of the same stimulus.

Results

An alpha level of .01 was used for the more powerful within-subjects tests, and .05 for all between-subjects tests. Error rates were again low (again around 5%) for all participants and are not considered. Error trials and outliers (defined as more than 3 standard deviations from each participant's mean) were excluded from all analyses. Contrary to expectation, we found no difference between the mean times for addition and multiplication production in either the single task (1,070 ms vs. 1,107 ms), $F(1, 33) = 1.80$, $p = .189$, or the dual task (2,005 ms vs. 1,980 ms), $F(1, 33) = 0.51, p = .481$. Therefore, in all analyses we collapsed over this factor.

We chose to split the data from the free condition according to the order in which participants responded. Participants in the free condition responded slightly more often to the production task first (overall 59% of the time, ranging on a per-participant basis from 1% to 99%). For each task, verification and production, we can report five mean times: mean time for the single task for the fixed-order participants, mean time for the dual task for the fixed-order participants, mean time for the single task for the free-order participants, mean time for the dual task for the free-order participants when they performed verification first (when it was Task 1), and mean time for the dual task for the free-order participants when they performed verification second (when it was Task 2). These data are reported in Figure 12a for the verification task and in Figure 12b for the production task. There are a number of important effects. First, single-task response times were faster (and approximately equal for free- and fixed-order participants) than

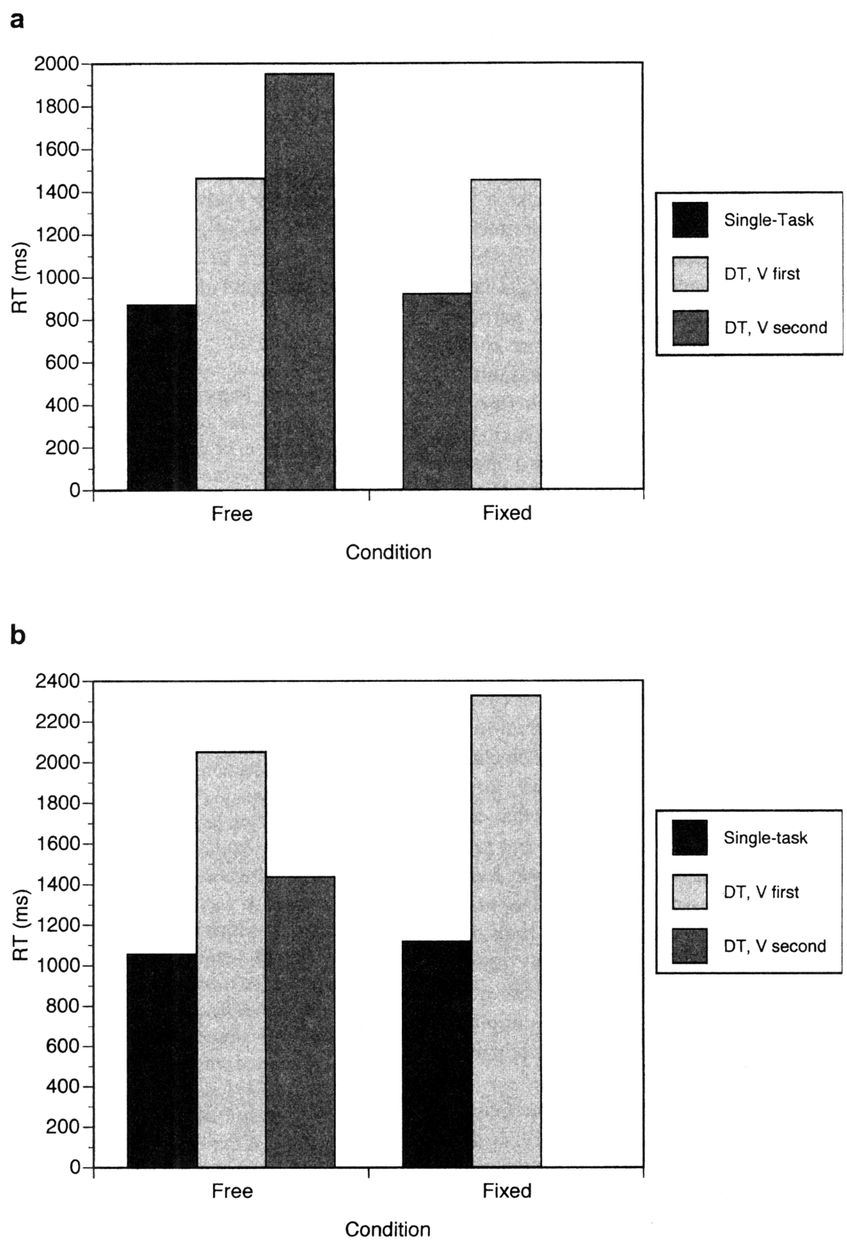


Figure 12. a: Response times (RTs) for the verification task in the fixed–free arithmetic experiment, with dual-task (DT) responses for participants in the free condition split by response order. b: RTs for the production task in the fixed–free arithmetic experiment, with DT responses for participants in the free condition split by response order. V = verification.

dual-task times, even when compared against the task participants did first; the average difference between single-task time and Task 1 time in dual task was 539 ms, $t(33) = 10.82, p < .001$. This is the dual-task effect. Second, participants were much slower to perform the Task 2 even in the free-order condition (the average difference between Task 2 time and single-task time was 1,073 ms, $t(33) = 17.03, p < .001$. In fact, the dual-task effect is so large in Task 2 that the participants' average time for Task 2 was reliably larger than the sum of their times for the single responses. That is, the mean difference between the Task 2 time and the sum of the single-task times was 57 ms, reliably greater than 0, $t(33) = 3.27$, $p = .003$. Thus, the evidence for parallelism is weak at best.

The most direct comparisons between the fixed and free conditions are when participants choose to respond in the free condition in the same order as the fixed condition—that is, verification first, production second. The verification latencies (see Figure 12a) are remarkably similar across conditions. On the other hand, the production latencies show an advantage of over 250 ms for participants in the free condition, though this difference is not reliable, $t(30) = 1.76, p = .089$.[1] This may indicate some additional ability to parallelize in the free condition, but the evidence for this is weak.

Although the participants' Task 2 completion time looks a lot like the behavior observed at 0 SOA in the previous experiment, their first-task completion times were longer than the first-task times from Experiment 1 and much longer than the single-task completion times in this experiment. This suggests that participants were completing some components for Task 2 before completing Task 1. In particular, because there was just one stimulus, we suspect that they encoded all aspects of the stimulus first—including those relevant to the other task. In the production single task, participants just had to encode the first and third digit, whereas all three digits would have to be encoded for the dual task. In the verification single task, participants could just encode the first two digits, retrieve the sum or product, and match the third digit without fully encoding it.

Discussion of Fixed–Free Arithmetic

The dual-task effects in this experiment were sizable in all cases. Again, Task 1 slowdown as a result of being in a dual-task situation is not necessarily predicted by simple response-selection bottleneck models or EPIC. However, the effect is a fairly large one, around 500 ms for both conditions. Considering that the single-task times for these tasks are approximately 1 s, that represents a fairly sizable penalty. Dual-task effects in Task 2 are equally interesting. Despite the absence of any clear perceptual-motor bottlenecks, Task 2 was slowed substantially in both the fixed and free conditions. The nearly equal size of this effect across the two conditions again suggests this is not the result of a strategic deferment as a result of priority instructions. This implies that even if cognitive parallelism were possible here, participants choose not to use it.

There is a caveat to this conclusion, however. These two tasks potentially generate interference in verbal working memory, or the articulatory loop (Baddeley, 1986). The total number of digits that participants could have had to manage at once was as high as six (three for the stimulus, one for the retrieved answer to the verification problem, and up to two for the answer to the production problem). Thus, fixed–free arithmetic may have generated a bottleneck not strictly in response selection, but in the verbal working memory of the participants. We believe this possibility to be remote, because participants most likely did not attempt to do response selection for both tasks at once. However, verbal working memory interference does provide a possible explanation for the sizable dual-task effects found here. It would thus be useful to discover what people's behavior is when this is considerably less likely.

Experiment 3: Pattern Math

Experiment 3, which we refer to as the pattern math experiment, was designed to replicate the results of the fixed–free arithmetic experiment but with some critical changes. First, we wanted to be sure that modality-specific working memory issues could not be the locus of dual-task interference, so one of the two tasks in pattern math did not make use of verbally coded materials. Second, an account based on parallel response selection with strategic deferment would predict that when participants are instructed to give one task higher priority, then difficulty effects in Task 2 should be absorbed into postselection slack (assuming Task 1 response selection completes before Task 2 response selection). The fixed–free arithmetic experiment failed to get a difficulty effect manipulating multiplication versus addition. Therefore, we decided to return to the manipulation that had worked in parallel arithmetic (addend size) as we wanted to further test this effect.

Method

Participants. Participants were 44 Carnegie Mellon undergraduates who participated for either $10 payment, $5 payment and credit in a psychology course, or double experimental credit in a psychology course.

Stimuli and procedures. As in Experiments 1 and 2, there were three kinds of trials in Experiment 3, two single tasks and a dual task combining the two single tasks. The first task was termed *pattern classification.* Participants were trained on 10 stimuli consisting of three # characters placed in a 3 × 3 array (see Figure 13a for an example). All patterns making up straight lines were removed, and thus there were 22 possible patterns. For each participant, 10 patterns from the possible 22 were randomly selected, half of which were designated as A patterns and half of which were designated as B patterns. When a pattern appeared, participants were to indicate whether the pattern was an A pattern or a B pattern with a keypress response. A patterns were indicated with the 4 key on the numeric keypad section of the keyboard, and B patterns with the 6 key on the numeric keypad. Participants were instructed to use two different fingers of their right hand (to prevent hand movements) for the two keys. Participants were explicitly discouraged from creating verbal codes for the patterns, and self-reported compliance with this instruction was excellent.

The second task was addition verification. Participants saw three numbers, the third of which was presented in boldface type (e.g., 6 9 **15**). If the first two numbers summed to the third, that is, the numbers represented a correct addition fact, participants were to respond by speaking aloud the word "true." Similarly, if the first two numbers did not sum to the third, then participants were to respond with the word "false." Stimuli were randomly generated with the constraint that the first and second digits were never the same. For foils, the third digit differed from the correct sum by 1, 2, −1, or −2. Easy trials consisted of addends from 1 through 4 and hard trials of addends 6 through 9. The dual-task consisted of a pattern with the # characters replaced by numbers appropriate for the addition verification task. An example is shown in Figure 13b.

Participants in the experiment were first trained on the 10 patterns. In the training phase, participants first viewed all 10 patterns one at a time for as long as they wished, but for a minimum of 3 s. Next, they were presented with the patterns one at a time in random order and asked to classify them. They received blocks of training trials until they correctly completed three training blocks. Each training block consisted of all 10 patterns in random order; all 10 responses in a training block had to be correct in order for the block to be considered correct.

After pattern classification training, participants completed 120 practice trials consisting of 40 pattern classification trials, then 40 addition verification trials, and then 40 dual-task trials. After practice, participants received three blocks of trials. Each block was divided into five sets of 40 trials: one pattern classification set, two addition verification sets, and two dual-task sets. Ordering of sets within a block was random. Easy and hard verification trials were intermixed randomly within sets; each set consisted of 20 easy problems and 20 hard problems.

[1] This test has fewer degrees of freedom because some participants in the free condition responded to the production task first most of the time. Thus, they did not contribute sufficient data to be included in this comparison.

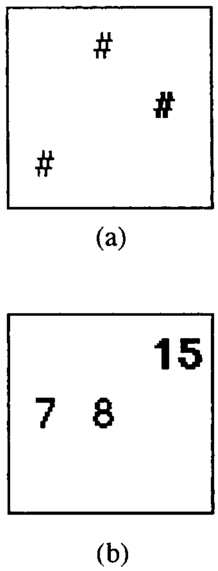


Figure 13. a: Example pattern stimulus used in the pattern math experiment. b: Example dual-task stimulus used in the pattern math experiment.

As in fixed–free arithmetic, half of the participants were assigned to the free condition, meaning that in dual-task trials they were free to respond to the two tasks in any order. Half of the participants were assigned to the fixed condition, in which they were instructed that "highest priority is the pattern classification task—always try to finish that task as quickly as possible." Participants in the fixed condition were given a warning by the experimental software if they responded to the verification task first. Again, in the fixed condition, this experiment is similar to a PRP experiment with an SOA of 0, except that both Task 1 and Task 2 make use of the same stimulus.

Results

Participants in the free condition responded to the addition verification first (opposite of the order for the fixed-condition participants) just over half of the time (59%, ranging on an individual basis from 8% to 100%). Results based on response order, with the fixed condition included for reference, for classification and verification appear in Figure 14a and 14b. The most direct comparison between the free- and fixed-order conditions is when participants respond in the same order in the free condition as they do in the fixed condition. These times are remarkably similar in the two conditions: For pattern completion, there was no evidence for a main effect of condition, $F(1, 41) = 0.49, p = .49$, and for addition verification, there was no evidence for a main effect of condition, $F(1, 41) < 0.01, p = .96$, suggesting that in the free condition participants were behaving the same as in the fixed condition. Unlike in Experiment 2, there is no evidence for a Task 2 advantage in the free condition.

Participants' response times for the second task in the dual-task condition were remarkably close to the sum of their times in the single-task condition, again suggesting that there is little parallelism in the dual task. Participants were much slower to respond to the first task in the dual-task condition than in the single-task condition; the average difference between single-task time and Task 1 time in the dual task was 508 ms, $t(43) = 10.87, p < .001$. Again, this suggests that they were performing some encoding of the stimuli for both tasks before proceeding on to perform either task. In the case of pattern classification, they had to encode the identity of the digits in the dual-task situation, which they did not have to do in the single task. In the case of verification, they had to encode the location of the digits in the dual task, which they did not have to do in the single task.

Finally, there is the issue of subadditive difficulty effects in Task 2. When that task is verification, the argument size manipulation produced a reliable difficulty effect in both the single-task and dual-task conditions. If participants are doing response selection for the two tasks in parallel but with response deferment for Task 2, the strategic postponement in the fixed condition should have caused this difficulty effect to be systematically smaller in the dual-task case. However, the difficulty effect should have been unchanged in the free condition—essentially, an interaction between condition and dual-task effect is predicted. In fact, there was no statistical evidence that the difficulty effect was smaller in that case; the difficulty effect was smaller in the free condition, but this difference was not reliable: for the interaction, $F(1, 41) = 2.47$, $p = .124$.

Discussion of Experiment 3

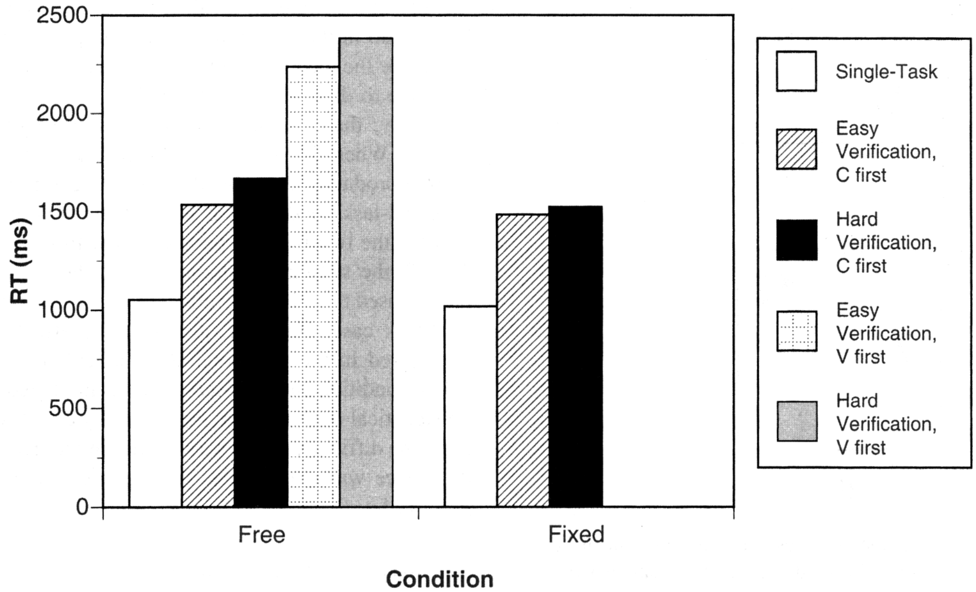
Overall, these results bear striking similarity to the results found in Experiment 2. Dual-task effects are present for both Task 1 and Task 2, with no evidence of a difference between the fixed and free conditions. Not only is the dual-task effect on Task 2 nonzero for the participants in the free condition but it is not reliably different from the Task 2 dual-task effect for the fixed condition. That is, the dual-task penalties were the same regardless of task instructions. This again suggests that there is no strategic difference between the fixed and free conditions and that the dual-task interference observed here is structural. The large and equal dual-task cost in both conditions certainly seems to suggest some kind of unavoidable bottleneck. Because the tasks shared stimuli and did not share output modality, the bottleneck appears to be cognitive rather than perceptual or motor. The fact that the results of Experiment 3 in many ways mirror the results of Experiments 1 and 2 suggests that these are not chance findings.

ACT-R/PM Models of the Current Experiments

What makes ACT-R/PM novel is the combination of EPIC-like perceptual-motor modules and the ACT-R theory of cognition. The key demonstration of this synthesis is in modeling the three new experiments, which are dual-task situations but also have more demanding cognitive components than traditional dual-task paradigms like the PRP. Parameters used in these models are listed in Table 2.

An ACT-R/PM Model of the Parallel Arithmetic Experiment

The ACT-R/PM model for the parallel arithmetic experiment involved combining two single-task models, one for each task. We describe these models individually and their fit to single-task performance. Then we describe their combination in the dual-task model. For both multiplication and addition, the model goes through four steps: (a) perceive, (b) encode, (c) retrieve, and (d) respond. For multiplication, the basic perception process consists a



b

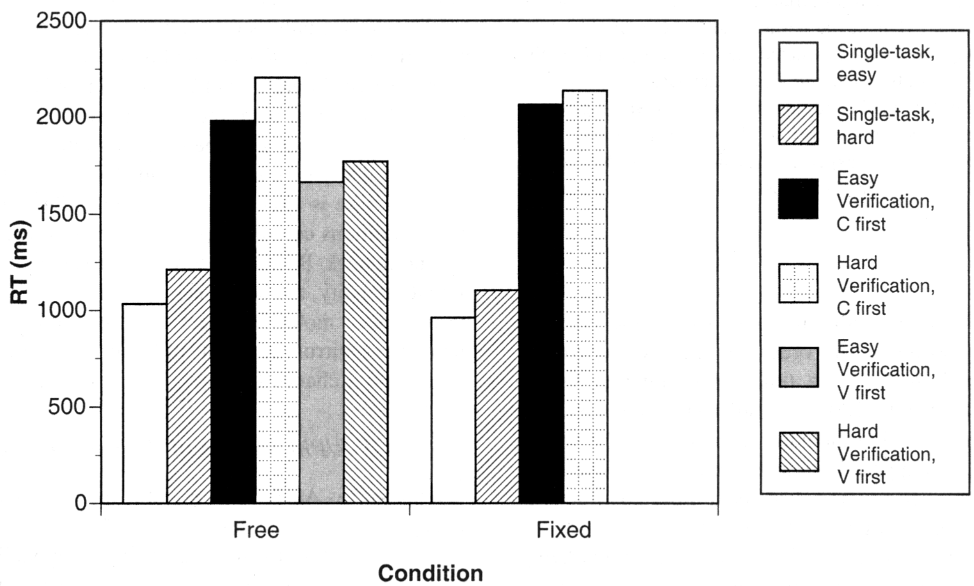


Figure 14. a: Dual-task pattern classification results from the pattern math experiment, with responses for participants in the free condition split by response order. b: Dual-task addition verification results from the pattern math experiment, with responses for participants in the free condition split by response order. RT = response time; C = classification; V = verification.

of creating a chunk representing the sound. Encoding consists of retrieving a chunk from memory that maps the raw chunk produced by the listen operator (e.g., the sound of the word *three*) into a semantic chunk representing the number (e.g., 3). Once both numbers have been encoded and stored in the goal, retrieval of the product begins. The two operands serve as retrieval cues from which activation spreads. A fact is retrieved that corresponds to the multiplication fact involving those two numbers. Responding in the multiplication task involves retrieving a chunk that maps the semantic representation of the result (e.g., 27) to the verbal code for the number (e.g., the audio string *twenty-seven*), and then initiation of the speech signal. For addition, the process is similar. The model begins by moving attention to the equation on the screen, which it takes in as a single phrase. The two operands are extracted from the representation of the phrase and then encoded into semantic units as was done for multiplication. The model then

Table 2
Parameters Used in Models of Experiments 1 Through 3

Parameter	Value
Experiment 1	
Standard deviation of the activation noise (s)	0.27
Latency factor (F)	0.65
Auditory digit recode time	350 ms
Auditory digit detection delay	200 ms
Base-level activation of encoding chunks	3.00
Experiment 2	
Standard deviation of the activation noise (s)	0.30
Latency factor (F)	0.55
Base-level activation of encoding chunks	4.00
Experiment 3	
Standard deviation of the activation noise (s)	0.30
Latency factor (F)	0.65
Base-level activation of encoding chunks	3.10
Base-level activation of pattern–finger chunks	−2.95

retrieves the answer and compares it to the answer in the probe. If they match, a production fires to punch the ring finger, and if there is a mismatch, another production fires to punch the index finger. There is one difference between this model and the one in Byrne and Anderson (1998). Participants in this experiment had faster single-task times, and so the latency factor (F in the retrieval time equation, Equation 1) was reduced from 0.85 to 0.65 to better fit the single-task times recorded by participants in this experiment. Otherwise, all processes and parameters are the same as those used in the Byrne and Anderson (1998) model.

As in the models for the simple PRP experiments, a single dual-task goal was used in each model. This requires the use of an additional slot to keep track of task state, which in this case was the operator currently relevant (i.e., multiplication or addition). The presence of this additional operator in the goal means that the operands receive less source activation because source activation must be divided evenly among more elements. This single-goal approach reduces basic task-switching overhead but predicts that retrieval times may be slower because of the divided source activation W_j in the activation equation (Equation 2). In this model, the dividing of the source activation among the two operands and the operator does indeed slow down retrieval of the arithmetic facts. The net effect of this single-goal-with-operator strategy is to minimize task switching time, but in such a way as to cause a resulting slowdown in arithmetic fact retrieval. This produces a dual-task effect, but one based on increased retrieval time, not on goal management cost.

There is also the question of overlapping cognitive and perceptual-motor operations. Although it is theoretically possible to overlap in a wide variety of places (such as Task 2 perception with Task 1 perception–encoding–retrieval), the overall savings in time seen in the data, although clearly present, are not great enough to indicate more than a small amount of overlap. In fact, the amount of saving (around 200 ms at an SOA of 0 ms) suggests that the savings might come from simply overlapping the speech module's processing of the response in Task 1 with the visual module's perception of the Task 2 stimulus. This is the approach that was taken in constructing the dual-task model. All the parameters other than the latency factor (F) that were used in this model are identical to those in Byrne and Anderson (1998).

The latency for the model to complete a given trial is dependent on several factors. First, there are various parameter values that affect the time to completion, such as digit delay and recoding times, ACT-R's latency scale (F), the amount of activation noise, and the base-level activations of the chunks used for digit encoding. Parameter values used to fit this model are given in Table 2. Because activation noise is used in this model, results tend to vary from trial to trial. Also, the type of problem and the numbers that appear in each problem both affect response latency. These have an impact because of the different base levels and associative strengths used for the arithmetic facts. The values were taken from Lebiere and Anderson's (1998) work on mental arithmetic. In general, facts involving smaller numbers are retrieved faster than facts involving larger numbers, and addition facts are generally faster than multiplication facts. Because operands are randomly generated for each problem, this adds variability to the model. For these reasons, all model predictions are based on the mean of 200 Monte Carlo runs of the model. Overall, though, the model fits the data quite well. Figure 15 presents the data and the results of the simulation. The fit of the model produces $r^2 = .98$. The quality of the fit of the ACT-R/PM model is impressive given that all of the productions and chunks and all but one of the parameters were taken from the model of a previous experiment with a fixed-order constraint.

An ACT-R/PM Model of the Fixed–Free Arithmetic Experiment

Although it may appear on the surface that the fixed–free arithmetic experiment is similar to the parallel arithmetic experiment at an SOA of 0, this was not the case. In particular, participants showed considerably more dual-task interference to the first stimulus to which they responded in both the fixed-order and free-order conditions. We believe this additional dual-task interference is based on the fact that the same stimuli were encoded for both tasks simultaneously (because they were the same stimuli), thus making it possible to retrieve the answer to either arithmetic operation at any time. Participants thus had to maintain more task state information, that is, which task they were currently working on (verification or production). In addition, the stimuli themselves were ambiguous with respect to which production task participants had to perform (either addition or multiplication), and this required the participants to retain additional information during the course of the trial. The maintenance of this additional state information, as well as the possibility of retrieval of the "wrong" fact for the currently relevant task, created considerable dual-task interference even in the first task to which participants responded. There are two sources of this dual-task interference. First, the maintenance of this additional state information meant that goal activation was more diffuse, which slows down retrievals. Second, the model would occasionally (approximately 25% of the time) retrieve the wrong arithmetic fact (e.g., multiplication instead of addition) for the task at hand, because the wrong production was selected in conflict resolution, and would then have to reset the goal and retrieve again to get the appropriate fact. Several participants reported this Stroop-like interference on at least some trials.

The general strategy used in the construction of the model for this experiment was the same as the one used in Experiment 1

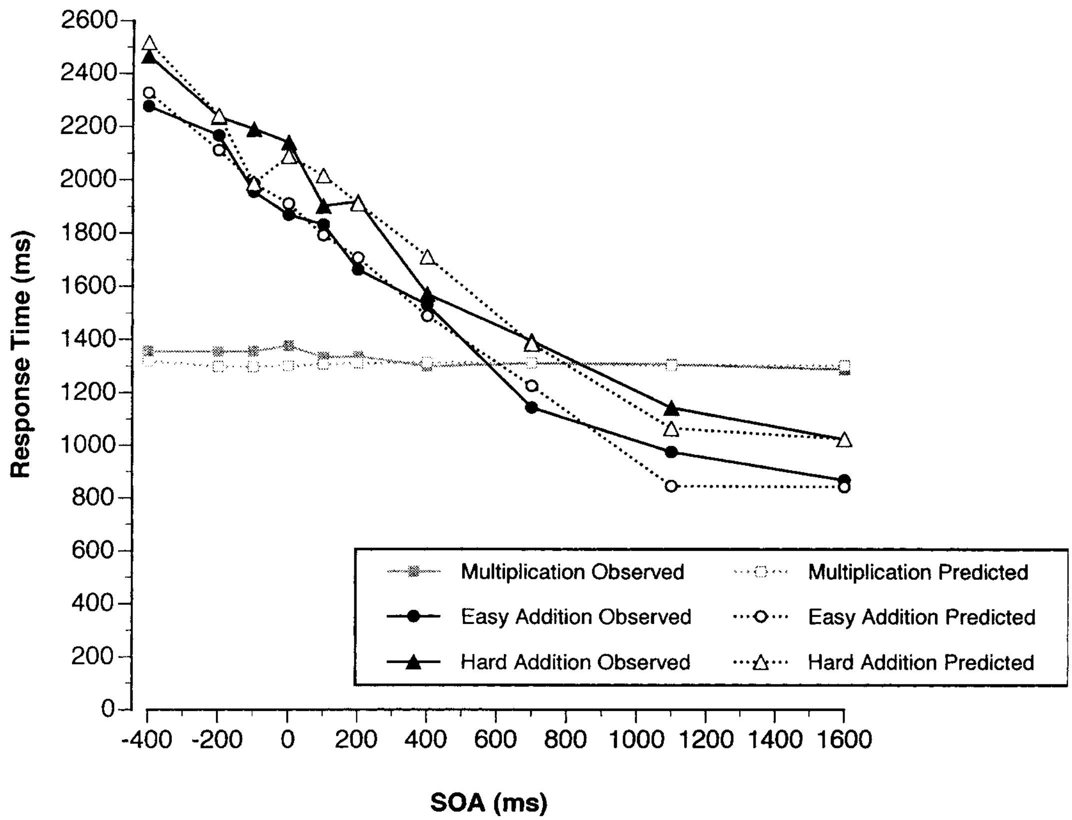


Figure 15. Model and data for the parallel arithmetic experiment. Solid lines depict the results when participants responded to the multiplication task first, and dotted lines depict simulation results. SOA = stimulus onset asynchrony.

and Byrne and Anderson (1998). All numerical parameter fitting was done based on models of the single-task situations, and then those parameters were used to predict performance in the dual-task situation. Again, these were essentially four-step models that went through the stages of (a) perceive, (b) encode, (c) retrieve, and (d) respond. The single-task addition verification model was identical to the addition verification model used for Experiment 1 and Byrne and Anderson, with the exception of the latency factor and base-level activations used. These participants were slightly faster than the participants in those two experiments and thus required slightly different parameter values. In the case of both dual-task and single-task arithmetic production, the goal chunk carried with it an additional piece of information, the operation (addition or multiplication) to be performed. The single-task models were used to estimate model parameters, presented in Table 2. Base-level activations and associative strengths for arithmetic facts were again based on the Lebiere and Anderson (1998) work, as in the Experiment 1 and Byrne and Anderson models. These parameters were not reestimated for the model of the dual-task situation. The fit of the model to the data is presented in Figure 16 (because of noise, model means are based on the average of 400 Monte Carlo runs). Overall the model does a good job of predicting the amount of dual-task interference based on single-task performance. The fit of the model produces r^2 = .96.

An ACT-R/PM Model of the Pattern Math Experiment

In the pattern math experiment, the stimuli for the two tasks were again the same (characters on the screen), but different aspects of those stimuli were relevant for the two tasks. For pattern classification, the location of each character was relevant, but the identity of the stimulus was not (e.g., # vs. *3*). For addition verification, the opposite was true: Identity was relevant, but location was not. Unlike in the fixed–free arithmetic experiment, the encoding stage of the two tasks produced different results. Thus, although it would have been possible to do the encodings for both tasks simultaneously in the dual-task situation, in practice this is not a useful strategy for ACT-R/PM, because goal activation in such a situation becomes so diffuse that retrievals are impossible. That is, in this experiment six things (three numbers, three locations) have to be encoded for the two tasks, rather than three things in the fixed–free arithmetic experiment. In the fixed condition and approximately half the time in the free condition, pattern classification was done first in the dual task. In that case, when the model encodes the locations of the characters (which is necessary for pattern classification), it also notes the physical identities of the numbers (e.g., the character *3*) but does not translate or encode those into semantic form (e.g., the chunk THREE that appears in arithmetic facts). The model stores a reference pointer to these identities in the goal for later retrieval. Storing this Task 2-relevant reference while encoding the stimuli for Task 1 does diffuse goal

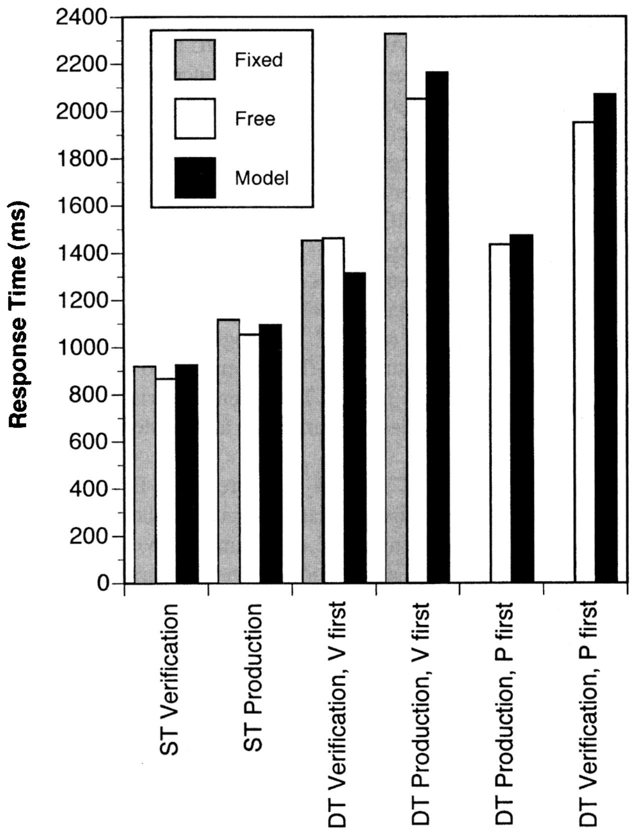


Figure 16. Model and data for the fixed–free arithmetic experiment. White and gray bars represent human data from the free and fixed conditions, and black bars represent the model's performance averaged over 400 runs of the model. ST = single task; DT = dual task; V = verification; P = production.

activation somewhat, and this diffusion is the source of the Task 1 dual-task slowing observed. It is also partly the cause of Task 2 slowing as well because it requires later retrieval. However, storing this single reference, as opposed to storing three additional stimuli encodings, prevents goal activation from becoming too diffuse to successfully retrieve the associations between patterns and response categories.

Again, the same general strategy was used: Models were constructed of the single-task situations, and all numerical parameters fit to those data; then, those parameters were used to predict dual-task performance. The addition verification model was nearly identical to previous verification models except that the response modality was different for this experiment (vocal instead of manual). Model parameters (shown in Table 2) were estimated from the single-task data. Again, base-level activations and associative strengths for arithmetic facts were based on the Lebiere and Anderson (1998) work. No additional parameters were estimated for the dual-task model; instead, the parameters fit from the single-task models were reused. Figure 17 presents the fit of the model to the data (again, based on the mean of 400 Monte Carlo runs because of noise). Again, considering that the dual-task model is not a parameter fit but rather a prediction of the model with dual-task effects simply emerging from ACT-R's activation dynamics, the fit is excellent with r^2 = .98. This again highlights the importance of a strong underlying theory of cognition, as the dual-task slowing due to more diffuse goal activation is again quite substantial.

This illustrates the power of a cumulating synthesis of two computational theories originally designed for somewhat different purposes but sharing some common history. We believe this represents an important theoretical advance, because it is difficult to see precisely how these results would be obtained in an EPIC model. We do not deny that it may indeed be possible to construct EPIC models that would produce response times that fit the data just as well as the ACT-R/PM models, because the EPIC models could incorporate executive strategies that produce behavior that effectively serializes cognition. However, it is not clear what would motivate such strategies, particularly in the cases of our fixed–free arithmetic experiment and our pattern math experiment. Adopting a deferment-based policy would not improve compliance with the instructions given and would merely serve to lengthen the experimental session. Given the number of trials involved in these experiments, we are strongly disinclined to believe that participants voluntarily adopted a strategy that would have prolonged these sessions, particularly because it has been shown in other contexts that participants can and do select strategies that allow them the most rapid performance (Gray & Boehm-Davis, 2000). If it had been possible for participants to do the tasks in parallel without interference in Experiments 2 and 3, then they should have done so.

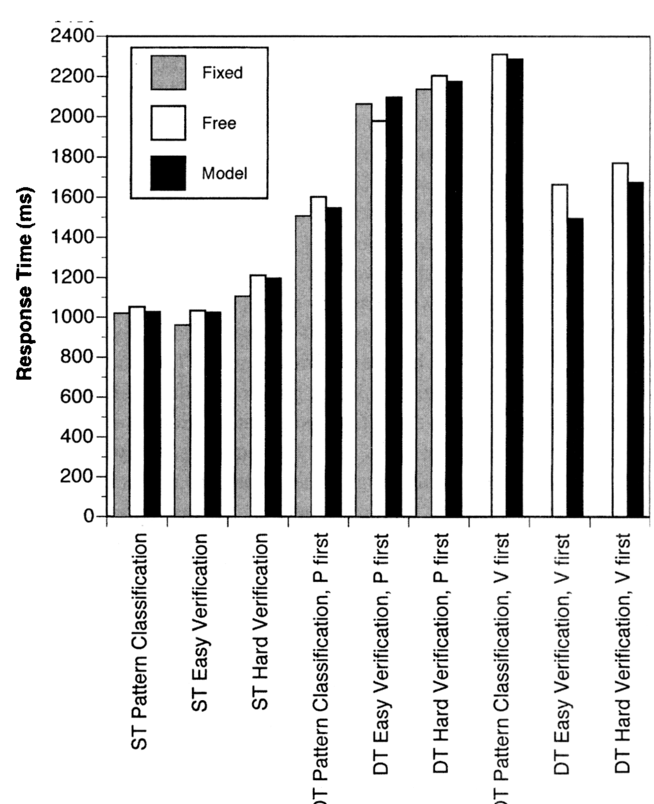


Figure 17. Model and data for the pattern math experiment. White and gray bars represent human data from the free and fixed conditions, and black bars represent the model's performance averaged over 400 runs of the model. ST = single task; DT = dual task; P = pattern classification; V = verification.

General Discussion

ACT-R/PM is a synthesis of computational theories with ACT-R as central cognition and a set of EPIC-like modules for perception and motor output. We agree with not only Meyer and Kieras (1997a, 1997b) but also Allport (1993) and Broadbent (1993) that computational modeling is a key component of understanding the complex issues involved in human multiple-task performance. Production systems also strike us as the most natural formalism, and we further agree on the need for an integrated, stimulus-to-response, information-processing architecture complete with detailed perceptual-motor specifications. Critically, the overlap in our views goes beyond mere agreements on general methodology. We consider seminal the synthesis of the motor control literature that ultimately produced EPIC's manual motor processor (see Kieras & Meyer, 1996; Meyer & Kieras, 1997a). The value of that contribution is difficult to overestimate, as a great many of the constraints that make it possible for ACT-R/PM to model PRP effects come from a detailed consideration of motor processes. This article has demonstrated the value of incorporating this work with a well-established cognitive theory, ACT-R. ACT-R/PM can successfully model PRP experiments, even those including subadditive difficulty effects, perfect time-sharing, and dual-task interference in paradigms with more cognitively complex tasks. Our work strongly suggests that examination of dual-task paradigms with more complex cognitive demands is both interesting and important to understanding multiple-task performance in general.

One of the interesting empirical and theoretical discoveries of our work was the dual-task decrement, which appears even in the first task completed in the dual task. This kind of dual-task decrement has been found in other PRP experiments (e.g., Pashler & Johnston, 1989), but such effects tend to be small or nonexistent. Dual-task decrements were as large as 500 ms in our experiments. One of the reasons why the effects were so large in our experiments was that our tasks involved significant cognitive components. We realized as we started to model these experiments that ACT-R predicted such a decrement because of the need to hold additional elements of both tasks in the goal. The need to hold information for the other task in the goal meant that there was less source activation (W_j in the activation equation, Equation 2) for the elements of a particular task. As a consequence, whenever there was retrieval required in the task, there was less activation spread to declarative memory and consequently slower retrievals. The general prediction of the ACT-R theory is that there will be a dual-task decrement whenever it is necessary to maintain elements of both tasks in working memory and it is necessary to perform retrieval. This prediction was confirmed in the experiments of Anderson, Reder, and Lebiere (1996) in which they showed that a working memory load impacted performance of an algebra task to the extent the algebra task required difficult retrievals.

ACT-R/PM and EPIC are obviously close relatives and have a number of powerful similarities. They both have multiple perceptual-motor modules and use a production system as the basis for cognition. They can both model the basic PRP results such as the -1 slope in the PRP region of the Task 2 curve, the dual-task savings, and the flat Task 1 curve. More advanced PRP effects, such as the subadditive difficulty effect, can be quantitatively modeled by both systems. And both systems are able to account for perfect time-sharing. This naturally raises the question, what is the key difference between ACT-R/PM and EPIC?

The simple and obvious (though probably incomplete) answer is serial versus parallel cognition. Although ACT-R/PM's cognitive system is not entirely serial (production selection and spreading activation processes proceed in parallel in ACT-R), it is serial in a critical way: Only one production may fire per production cycle. This is in contrast to EPIC, which can fire an unbounded number of productions in a single cycle. The seriality in ACT-R/PM strongly constrains its predictions. For instance, ACT-R/PM predicts that perfect time-sharing should be much less common than one might predict based on EPIC. Although we cannot claim our data disprove EPIC, because it is flexible enough to accommodate the results, we do claim that ACT-R's central seriality led us to predict the results of these experiments and made it natural to model them.

Although we think ACT-R/PM is favored over EPIC for these tasks, our conclusions about seriality versus parallelism are specific to these production systems' embodiments. We do not claim to have produced tests about seriality versus parallelism more generally. Reaching such general conclusions is notoriously difficult (e.g., Schweikert & Wang, 1993; Townsend & Ashby, 1983), and this was not the goal of our research. If our results are to be cast in terms more general than EPIC versus ACT-R/PM, what we have demonstrated is that doing two tasks at once results in a slowing in their joint execution roughly equal to the sum of the times for the two separately. Given the heavy cognitive involvement of these tasks, this argues for a limited-capacity central cognition and leaves open whether that limited capacity is realized in a parallel or serial system.

If one accepts the general conception of the cognitive architecture that EPIC and ACT-R/PM share, then our data better support seriality. Part of the evidence for such a cognitive architecture is its success in accounting for complex cognition. Applied to dual-task situations, ACT-R/PM essentially becomes the central bottleneck theory implemented in more detail. Thus, we think we have succeeded in the goal of establishing a bridge between the tradition of studies of high-level cognition and the tradition of detailed studies of elementary cognition.

References

Abrams, R. A., & Jonides, J. (1988). Programming saccadic eye movements. *Journal of Experimental Psychology: Human Perception and Performance, 14,* 428–443.

Allport, D. A. (1993). Attention and control: Have we been asking the wrong questions? A critical review of 25 years. In D. E. Meyer & S. Kornblum (Eds.), *Attention and performance: Vol. 14. Synergies in experimental psychology, artificial intelligence, and cognitive neuroscience* (pp. 183–218). Cambridge, MA: MIT Press.

Allport, D. A., Antonis, B., & Reynolds, P. (1972). On the division of attention: A disproof of the single-channel hypothesis. *Quarterly Journal of Experimental Psychology, 24,* 225–235.

Anderson, J. R. (1993). *Rules of the mind.* Hillsdale, NJ: Erlbaum.

Anderson, J. R., Bothell, D., Lebiere, C., & Matessa, M. (1998). An integrated theory of list memory. *Journal of Memory and Language, 38,* 341–380.

Anderson, J. R., Fincham, J. M., & Douglass, S. (1999). Practice and retention: A unifying analysis. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 25,* 1120–1136.

Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought.* Mahwah, NJ: Erlbaum.

Anderson, J. R., Lebiere, C., & Lovett, M. (1998). Performance. In J. R. Anderson & C. Lebiere (Eds.), *The atomic components of thought* (pp. 57–100). Mahwah, NJ: Erlbaum.
Anderson, J. R., Matessa, M., & Lebiere, C. (1998). The visual interface. In J. R. Anderson & C. Lebiere (Eds.), *The atomic components of thought* (pp. 143–166). Mahwah, NJ: Erlbaum.
Anderson, J. R., & Reder, L. M. (1999). The fan effect: New results and new theories. *Journal of Experimental Psychology: General, 128,* 186–197.
Anderson, J. R., Reder, L. M., & Lebiere, C. (1996). Working memory: Activation limitations on retrieval. *Cognitive Psychology, 30,* 221–256.
Baddeley, A. D. (1986). *Working memory.* Oxford, England: Oxford University Press.
Broadbent, D. E. (1993). A word before leaving. In D. E. Meyer & S. Kornblum (Eds.), *Attention and performance: Vol. 14. Synergies in experimental psychology, artificial intelligence, and cognitive neuroscience* (pp. 863–879). Cambridge, MA: MIT Press.
Byrne, M. D., & Anderson, J. R. (1998). Perception and action. In J. R. Anderson & C. Lebiere (Eds.), *The atomic components of thought* (pp. 167–200). Mahwah, NJ: Erlbaum.
Card, S., Moran, T., & Newell, A. (1983). *The psychology of human–computer interaction.* Hillsdale, NJ: Erlbaum.
De Jong, R. (1993). Multiple bottlenecks in overlapping task performance. *Journal of Experimental Psychology: Human Perception and Performance, 19,* 965–980.
Gray, W. D., & Boehm-Davis, D. A. (2000). Milliseconds matter: An introduction to microstrategies and to their use in describing and predicting interactive behavior. *Journal of Experimental Psychology: Applied, 6,* 322–335.
Gray, W. D., John, B. E., & Atwood, M. E. (1993). Project Ernestine: A validation of GOMS for prediction and explanation of real-world task performance. *Human–Computer Interaction, 8,* 237–309.
Hawkins, H. L., Rodriquez, E., & Reicher, G. M. (1979). Is time-sharing a general ability? (ONR Tech. Rep. No. 3). Eugene: University of Oregon.
John, B. E. (1996). TYPIST: A theory of performance in skilled typing. *Human–Computer Interaction, 11,* 321–355.
Kieras, D. E., & Meyer, D. E. (1996). *The EPIC architecture: Principles of operation.* Unpublished manuscript. Available ftp://ftp.eecs.umich.edu/people/kieras/EPICArch.pdf.
Lebiere, C., & Anderson, J. R. (1998). Cognitive arithmetic. In J. R. Anderson & C. Lebiere (Eds.), *The atomic components of thought* (pp. 297–342). Mahwah, NJ: Erlbaum.
Lovett, M., & Anderson, J. R. (1996). History of success and current context in problem solving: Combined influences on operator selection. *Cognitive Psychology, 31,* 168–217.
Lovett, M., Reder, L. M., & Lebiere, C. (1999). Modeling working memory in a unified architecture: An ACT-R perspective. In A. Miyake & P. Shah (Eds.), *Models of working memory* (pp. 135–182). New York: Cambridge University Press.
Meyer, D. E., & Kieras, D. E. (1997a). A computational theory of executive cognitive processes and multiple-task performance: I. Basic mechanisms. *Psychological Review, 104,* 3–65.
Meyer, D. E., & Kieras, D. E. (1997b). A computational theory of executive cognitive processes and multiple-task performance: II. Accounts of psychological refractory-period phenomena. *Psychological Review, 104,* 749–791.
Newell, A. (1990). *Unified theories of cognition.* Cambridge, MA: Harvard University Press.
Pashler, H. (1994). Dual-task interference in simple tasks: Data and theory. *Psychological Bulletin, 116,* 220–244.
Pashler, H., & Johnston, J. C. (1989). Chronometric evidence for central postponement in temporally overlapping tasks. *Quarterly Journal of Experimental Psychology, 41A,* 19–45.
Rosenbaum, D. A. (1980). Human movement initiation: Specification of arm, direction, and extent. *Journal of Experimental Psychology: General, 190,* 475–495.
Schumacher, E. H., Lauber, E. J., Glass, J. M., Zurbriggen, E. L., Gmeindl, L., Kieras, D. E., & Meyer, D. E. (1999). Concurrent response-selection processes in dual-task performance: Evidence for adaptive executive control of task scheduling. *Journal of Experimental Psychology: Human Perception and Performance, 25,* 791–814.
Schumacher, E. H., Seymour, T. L., Glass, J. M., Lauber, E. J., Kieras, D. E., & Meyer, D. E. (1997). *Virtually perfect time sharing in dual-task performance.* Paper presented at the 38th annual meeting of the Psychonomic Society, Philadelphia.
Schunn, C. D., & Anderson, J. R. (1998). Scientific discovery. In J. R. Anderson & C. Lebiere (Eds.), *The atomic components of thought* (pp. 385–428). Mahwah, NJ: Erlbaum.
Schweickert, R. (1980, August 8). Critical-path scheduling of mental processes in a dual task. *Science, 209,* 704–706.
Schweickert, R., & Wang, Z. (1993). Effects on response time of factors selectively influencing processes in acyclic task networks with OR gates. *British Journal of Mathematical and Statistical Psychology, 46,* 1–30.
Townsend, J. T., & Ashby, F. G. (1983). *The stochastic modeling of elementary psychological processes.* Cambridge, England: Cambridge University Press.
Van Selst, M., & Jolicoeur, P. (1997). Decision and response in dual-task interference. *Cognitive Psychology, 33,* 266–307.

Received September 10, 1999
Revision received August 7, 2000
Accepted August 21, 2000 ■

Instructions to Authors
Psychological Review

Authors should prepare manuscripts according to the *Publication Manual of the American Psychological Association* (4th ed.). All manuscripts must include an abstract containing a maximum of 960 characters and spaces (which is approximately 120 words) typed on a separate sheet of paper. Typing instructions (all copy must be double-spaced) and instructions on preparing tables, figures, references, metrics, and abstracts appear in the *Manual*. Also, all manuscripts are copyedited for bias-free language (see chap. 2 of the *Publication Manual*). Original color figures can be printed in color provided the author agrees to pay half of the associated production costs. For information on the types of articles and theoretical notes that are appropriate for *Psychological Review*, see the statement on the inside of the front cover. For information on the procedures that govern the handling of critiques and replies, authors should refer to the policy statement in the January 1996 issue of *Psychoplogical Review* (Vol. 103, No. 1, pp. 3–4).

Authors are required to obtain and provide to APA all necessary permissions to reproduce any copyrighted work, including, for example, test instruments and other test materials or portions thereof.

APA policy prohibits an author from submitting the same manuscript for concurrent consideration by two or more publications. In addition, it is a violation of APA Ethical Principles to publish "as original data, data that have been previously published" (Standard 6.24). As this journal is a primary journal that publishes original material only, APA policy prohibits as well publication of any manuscript that has already been published in whole or substantial part elsewhere. Authors have an obligation to consult journal editors concerning prior publication of any data upon which their article depends. In addition, APA Ethical Principles specify that "after research results are published, psychologists do not withhold the data on which their conclusions are based from other competent professionals who seek to verify the substantive claims through reanalysis and who intend to use such data only for that purpose, provided that the confidentiality of the participants can be protected and unless legal rights concerning proprietary data preclude their release" (Standard 6.25). APA expects authors submitting to this journal to adhere to these standards. Specifically, authors of manuscripts submitted to APA journals are expected to have available their data throughout the editorial review process and for at least 5 years after the date of publication.

Authors will be required to state in writing that they have complied with APA ethical standards in the treatment of their sample, human or animal, or to describe the details of treatment. A copy of the APA Ethical Principles may be obtained by writing the APA Ethics Office, 750 First Street, NE, Washington, DC 20002-4242. APA requires authors to reveal any possible conflict of interest in the conduct and reporting of research (e.g., financial interests in a test or procedure, funding by pharmaceutical companies for drug research).

Masked reviews are optional, and authors who wish masked reviews must specifically request them when submitting their manuscripts. Each copy of a manuscript to be mask reviewed should include a separate title page with authors' names and affiliations, and these should not appear anywhere else on the manuscript. Footnotes that identify the authors should be typed on a separate page. Authors should make every effort to see that the manuscript itself contains no clues to their identities.

Manuscripts should be submitted in quadruplicate. All copies should be clear, readable, and on paper of good quality. In addition, electronic versions of manuscripts, although optional, facilitate the review process. Send in Microsoft Word format to psychrev@paradox.psych.columbia.edu. A dot matrix or unusual typeface is acceptable only if it is clear and legible. Dittoed and mimeographed copies will not be considered. In addition to addresses and phone numbers, authors should supply electronic mail addresses and fax numbers, if available, for potential use by the editorial office and later by the production office. Authors should keep a copy of the manuscript to guard against loss. Mail manuscripts to the Editor, Walter Mischel, *Psychological Review,* Department of Psychology, Columbia University, 406 Schermerhorn Hall, New York, New York 10027, according to the instructions provided above.