SCISPACE
formerly Typeset

Open access • Journal Article • DOI:10.1037//0278-7393.15.5.898

A componential model for mental addition. — Source link

Keith F. Widaman, David C. Geary, Pierre Cormier, Todd D. Little

Institutions: University of California, Riverside

Published on: 01 Sep 1989 - Journal of Experimental Psychology: Learning, Memory and Cognition (American Psychological Association)

Topics: Poison control

Related papers:

- Cognitive arithmetic: Evidence for retrieval and decision processes in mental addition.
- A chronometric analysis of simple addition.
- Cognitive arithmetic: comparison of operations.
- The development of mental arithmetic: A chronometric approach☆
- Strategy choice procedures and the development of multiplication skill.

Share this paper:

View more about this paper here: https://typeset.io/papers/a-componential-model-for-mental-addition-324ckpo366

Journal of Experimental Psychology:
Learning, Memory, and Cognition
1989, Vol. 15, No. 5, 898-919

Copyright 1989 by the American Psychological Association, Inc.
0278-7393/89/$00.75

A Componential Model for Mental Addition

Keith F. Widaman, David C. Geary, Pierre Cormier, and Todd D. Little
University of California at Riverside

A componential model capable of representing simple and complex forms of mental addition was proposed and then tested by using chronometric techniques. A sample of 23 undergraduate students responded to 800 addition problems in a true-false reaction time paradigm. The 800 problems comprised 200 problems of each of four types: two single-digit addends, one single- and one double-digit addend, two double-digit addends, and three single-digit addends. The results revealed that the columnwise product of addends, a structural variable consistent with a memory network retrieval process, was the best predictor of mental addition for each of the four types of problem. Importantly, the componential model allowed estimation of effects of several other structural variables, e.g., carrying to the next column and speed of encoding of digits. High levels of explained variance verified the power of the model to represent the reaction time data, and the stability of estimates across types of problem implied consistent component use by subjects. Implications for research on mental addition are discussed.

Over the past 20 years, several types of models for mental addition have been proposed—for example, models hypothesizing that analog (Restle, 1970), counting (Groen & Parkman, 1972), or memory network retrieval (Ashcraft & Battaglia, 1978) processes are invoked to arrive at the solution for a given problem. Although a great deal has been learned about the manner in which persons respond to addition problems, a comprehensive model identifying the several elementary processes underlying problem solution has not been developed. The primary aim of the present study is to propose and, by use of chronometric techniques, to evaluate a general processing model specifying the processes required to solve mental addition problems of any magnitude.

Sternberg (1977) outlined the *componential analysis* approach for isolating the elementary information processes involved in solving ability problems. Chronometric, or reaction time (RT), tasks are typically used to validate proposed componential models. In the componential analysis framework, internal validation refers to the determination that RT to problems of a given domain is affected by the hypothesized elementary information processes. Internal validation may take two forms: intensive and extensive. Intensive validation refers to internal validation of processes on a single type of problem, and extensive validation denotes the finding that identical or similar processing models hold across two or more types of problem that theoretically involve the same component processes. Componential models have been proposed for several types of reasoning tasks (e.g., Sternberg, 1977; Sternberg & Gardner, 1983), for verbal ability (e.g., Hunt, 1978; Hunt, Lunneborg, & Lewis, 1975), and for spatial ability (e.g., Pellegrino & Kail, 1982), among others. In the present study, the application of componential analysis is extended to numerical facility tasks through intensive and extensive validation of elementary information processes underlying performance on four types of mental addition problem.

The most consistent finding in studies of mental addition is a problem size effect: The larger the addends in an addition problem, the longer the RT to respond to the problem. The general goal of most previous studies of mental addition has been the specification and testing of several conceptual models that could account for the problem size effect. Associated with each conceptual model is a statistical, or mathematical, model that identifies a structural variable of which RT should be a linear function. The alternative processing models are then pitted against one another by determining which statistical model best represents average RT across a sample of subjects to a set of addition problems. The fit of the statistical model best representing RT data provides evidential support for its associated conceptual model of processing. One shortcoming of previous research, discussed in detail below, has been the lack of unique predictions by competing conceptual models of mental addition and the lack of correspondence between the conceptual and statistical models for certain models of addition.

A second limitation of previous research is rather more important: the failure to specify and test a cognitive component model for mental addition. In most studies (e.g., Groen & Parkman, 1972), investigators posited several alternative counting or memory retrieval processes that might be used to

The present study was supported in part by several grants to the first author: Grants HD-14688, HD-21056, and HD-22953 from the National Institute of Child Health and Human Development, an intramural grant from the Academic Senate, and a computing grant from the Academic Computing Center, both of the University of California at Riverside. The especially helpful comments by Arthur Baroody, by three other reviewers of a previous version of this article—Michael Humphreys, Jamie Campbell, and Mark Ashcraft—and by Marilyn Smith are gratefully acknowledged. We would also like to thank Tim Lennan and Mark Mullins for assistance in computer programming. David Geary is now at the University of Missouri, Rolla, Missouri, and Pierre Cormier is now at Acadia University, Wolfville, Nova Scotia, Canada.

Correspondence concerning this article should be addressed to Keith Widaman, Department of Psychology, University of California, Riverside, California 92521.

solve addition problems and then compared the fit of the corresponding statistical models, each model having a single predictor, or structural variable. Other researchers (e.g., Ashcraft & Stazyk, 1981) used stepwise regression analyses that were essentially exploratory and data-driven in nature. Unfortunately, the results obtained by using the latter procedure were difficult to relate to any simple conceptual model of mental addition. In contrast to the preceding approaches, research on a cognitive component model for mental addition involves the specification of all cognitive components (e.g., encoding of digits) that, according to theory, should be used when responding to a given problem, rather than specifying only counting or retrieval components. The cognitive component model would therefore provide theoretical support, rather than merely empirical support, for the form of alternative statistical models. The parameters estimated in such theoretically driven models should be less biased than estimates in statistical models composed in a solely empirical fashion (Cohen & Cohen, 1983).

A third limitation associated with extant research is the rather restricted array of addition problems that has been included for study. That is, in previous research, attention has been focused largely on simple addition, which, admittedly, is (a) the simplest form of addition, (b) the most common type of problem encountered during early instruction on addition, and (c) likely an important component of the addition of multidigit addends. However, addition problems encountered in everyday life (e.g., adding the amount of a deposit to one's checking account balance) are often rather more complex than simple addition, involving two or more numbers each having several digits. Several recent studies included addition problems more complex than simple addition (Ashcraft & Stazyk, 1981; Hamann & Ashcraft, 1985). However, in these studies, differing types of complex addition problem were not systematically investigated, and results reported were inconsistent with a straightforward conceptual model for complex addition. In the present article, a general model capable of representing all types of simple and complex addition is first proposed and then tested across four types of addition problem.

Processes Proposed for the Search/Compute Stage

Five general types of process have been proposed for the search/compute stage, the stage during which the true sum of a simple addition problem is obtained. The five types of process are analog, digital (or counting), direct memory access, memory network retrieval, and rule-based, procedural processes. Because the first three types of process have been discussed in detail in several places (e.g., Ashcraft & Battaglia, 1978), they will be covered only very briefly here; more attention will be paid to the latter two types of processes.

Early Proposals for Processes Underlying Mental Addition

Analog. On the basis of research on numerical comparison (Moyer & Landauer, 1967), Restle (1970) proposed a model in which each addend is transformed into an analog representation. The analog process utilizes an internal equivalent of a number line and represents each addend as a line segment proportional in length to the magnitude of the addend. Concatenating the two line segments results in a line representing the magnitude of the sum of the addends, which could then be compared with the stated sum for the problem. Given the preceding conceptual model, it is possible to develop several alternative statistical models, each specifying one or more structural variables of which RT should be a linear function, depending on whether RT is a function of the distance a line segment is transported, the length of the transported line segment, or both. These conclusions are summarized in Table 1.

Digital processes. Digital, counting processes compose the second type of process hypothesized for the search/compute stage. Groen and Parkman (1972; Parkman & Groen, 1971) posited a mental counter that could be set at most once and then incremented an unlimited number of times in a unit-by-unit fashion. Thus, the counter could be set to zero, the first addend, the second addend, the smaller addend, or the larger addend, and then incremented a number of times equal to the remaining addend(s). After the incrementing process is complete, the recomputed sum would be compared for correctness with the stated sum.

Given the metaphor of the mental counter and its use, Groen and Parkman (1972) discussed five strategies for using the counter and translated these strategies directly into corresponding statistical models (see Table 1). For example, if a subject consistently set his or her mental counter to the larger addend and then incremented the smaller, RT should be a linear function of the smaller addend, or MIN; also, the estimated regression slope for the MIN variable would reflect the time required to increment the mental counter. Groen and Parkman (1972; Groen & Resnick, 1977; Parkman & Groen, 1971) found that RT for simple addition problems was predicted better by the MIN structural variable than by any of the remaining four structural variables for both adults and children. Adults, however, had an estimated regression slope that was $1/20$ the size of the slope parameter estimated on data from first graders; this indicates much greater proficiency by adults. It is important to note that support for the MIN model is only differential support with regard to alternative digital models; the superior fit of the MIN model is equally strong support for one version of Restle's 1970 analog model.

Direct memory access. Groen and Parkman (1972) reported that RT to "tie" problems, which have identical addends, failed to show a MIN effect, even though RT to nontie problems was strongly related to the MIN variable. Instead, RT to tie problems was equally rapid regardless of the magnitude of the addends. This suggested that the correct sums for tie problems are stored in a similar, easily accessible fashion in long-term memory and require a constant amount of time for retrieval. On the basis of this, Groen and Parkman (1972) stated the direct memory access model, which proposed that RT is unrelated to the magnitude of the addends comprising a simple addition problem if the correct sum is retrieved from a memory store. If this were so, no structural variable representing a problem size effect should explain RT variance.

Table 1
Structural Variables Consistent With Search/Compute Stage in Hypothesized Processing Models

	Conceptual processing model			
			Network retrieval	
Structural variable	Analog	Digital	Table	Nontable
First addend	X	X		
Second addend	X	X		
Larger addend (MAX)	X	X		
Smaller addend (MIN)	X	X		
Sum		X	X	
Sum squared			X	
Sum of squared addends			X	
Product			X	
Distribution of associations				X
Difficulty				X
Frequency of presentation				X
Order of presentation				X

However, Groen and Parkman (1972) reasoned that the direct memory access process would require back-up by some slower, more reliable counting process on trials on which there was retrieval failure (cf. Browne, 1906). If first graders always use counting processes and adults revert to counting on the 5% of trials on which they suffer retrieval failure, this alone could account for the twentyfold decrease in the size of the regression slope estimate for the MIN structural variable when comparing performance of children in first grade (b = 400 ms) with that of adults (b = 20 ms), assuming that counting speed were identical at the two age levels.

Memory Network Retrieval Processes

More recent research has focused on *memory network retrieval* processes, which were first proposed by Ashcraft and Battaglia (1978) as alternatives to the processes discussed by Groen and Parkman (1972). Memory network retrieval reflects the search through a stored network of number facts for the correct sum for a problem. In contrast to the direct access model, Ashcraft and Battaglia (1978) hypothesized that the time required for retrieval from the memory network of the correct sum for a problem was related to problem size. Thus, the retrieval process proposed by Ashcraft and Battaglia is assumed to take longer amounts of time with problems that have larger addends because a longer search through the memory network would be required for such problems.

Table-related processes. When developing their memory network retrieval model, Ashcraft and Battaglia (1978) used the metaphor of a square, symmetric, printed addition table as a reasonable first approximation of an adult's memory network for simple addition facts. The row and column dimensions of the table are termed entry or parent nodes and take on values from 0 to 9, which represent the addends in a simple addition problem. The correct answer for a problem is stored at the intersection of the row and column corresponding to the two addends. Thus, upon viewing the problem 7 + 6 = 13, nodal values of 7 (e.g., the row node) and 6 (e.g., the column node) would be activated, and the spreading activation would result in the activation of the correct sum of 13 stored at their intersection.

Given the preceding conceptual model, several submodels may be specified that correspond to particular ways of accessing the information stored in the tabular memory network. The first of these submodels was termed a simple *table look-up* model by Stazyk, Ashcraft, and Hamann (1982). Assuming equal-sized steps between values along each parent node, Stazyk et al. reasoned that RT should be related to the true sum for each problem. This prediction holds only under the implicit assumption that memory search occurs according to a city-block metric. Under a city-block metric, given the problem "$a + b = c$," the subject would start from the origin, move a steps along one parent node and then b steps parallel to the other parent node to arrive at the intersection, and hence the correct sum c, of the two nodal values. As noted by Stazyk et al. and shown in Table 1, the structural variable for the table look-up model, the sum of the addends, is identical to that for one of the digital models, resulting in nonuniqueness of a central prediction of the table look-up model. More importantly, several studies (e.g., Ashcraft & Battaglia, 1978) found stronger support for other memory search models.

Ashcraft and Battaglia (1978) found that RT was more strongly related to the square of the sum of the addends than to other simple functions of problem size, for example, digital model structural variables. This led Ashcraft and Battaglia to specify their *memory network retrieval* model, hereinafter referred to as the *sum squared* model. Ashcraft and Battaglia hypothesized that, rather than equally spaced distances between nodal values, an adult's memory network is square and symmetric but stretched in the direction of larger sums, with values along each of the parent nodes spaced in a systematic but nonlinear fashion. As shown in Figure 1A, in which the nonlinear spacing of nodal values is assumed to conform to the square of the addend, the distance along each parent node between 0 and 1 is 1^2, or 1 unit; the distance between 0 and 2 is 2^2, or 4 units, and so forth. Ashcraft and Battaglia claimed that a model such as that shown in Figure 1A would result in the square of the sum of the addends being the important

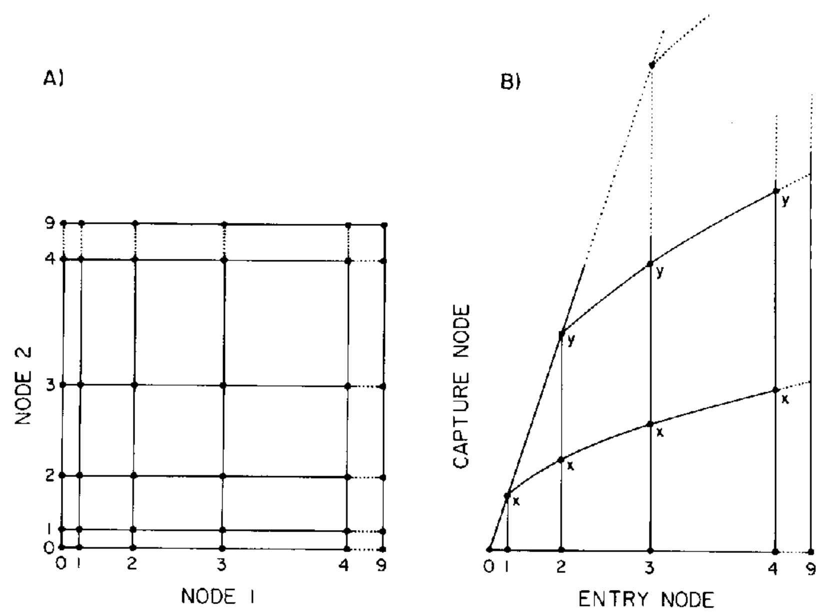

Figure 1. Conceptual models for network retrieval process (Ashcraft, 1982): (A) Symmetric model proposed by Ashcraft and his associates; (B) Asymmetric model consistent with "true-sum-squared" as structural parameter.

structural variable relating RT to problem size. On the basis of the initial study by Ashcraft and Battaglia, Ashcraft and his associates (Ashcraft & Fierman, 1982; Ashcraft et al., 1984; Ashcraft & Stazyk, 1981; Hamann & Ashcraft, 1985) launched a program of studies that consistently reported superiority of the sum squared over structural variables based on counting models (Groen & Parkman, 1972) in samples of proficient subjects.

However, a basic incompatibility between the conceptual and statistical models proposed by Ashcraft and Battaglia (1978) has heretofore gone unnoticed. Referring to Figure 1A, it is clear that the distance from the origin to a nodal value along each parent node is a simple nonlinear function of the nodal value. Figure 1A was drawn under the assumption that the stretching of the square table in the direction of larger sums was a function of the square of each of the addends. Thus, given addends a and b, the city-block metric distance from the origin to the intersection of a and b is $(a^2 + b^2)$. Ashcraft and Battaglia implicitly assumed that a city-block metric would describe search through such a memory network by stating that longer search times would result from "longer vector distances from entry nodes to intersection" (1978, p. 536). But the sum squared, or $(a + b)^2$, does *not* in general equal the sum of squared addends $(a^2 + b^2)$. Therefore, assuming the nonlinear spacing of nodal values shown in Figure 1A, the conceptual model discussed by Ashcraft and Battaglia (1978) is consistent with a statistical model in which the sum of the square (or other power) of each addend is the variable structurally relating problem size to RT. A statistical model employing the *sum of squared addends* as structural variable would therefore preserve the square, symmetric property, as well as the stretching of the nodal values, of the conceptual model developed by Ashcraft and Battaglia (1978).

The simplest revision of the Ashcraft and Battaglia model that restores compatibility between conceptual and statistical models is shown in Figure 1B. The conceptual model presented in Figure 1B is an asymmetric table with a horizontal dimension termed the *entry* node and a vertical dimension termed the *capture* node. Because of the asymmetric nature of the layout of the table, the table is entered along the entry node by moving b steps along the entry node, where b is the magnitude of the larger addend. Assuming a city-block metric, movement of a steps vertically along the capture node at b, where a is the magnitude of the smaller addend, would lead to an intersection of entry and capture node values at which the sum of the addends would be stored. According to the sum squared structural variable, problems with identical sums should lie equal city-block vector distances from the origin. For example, the distance from 2 to 3 along the entry node is equal to the distance from 2 to the first intersection vertically above 2; this leads to a prediction that the problem "3 + 0" and the problem "2 + 1" have equal search times, consistent with the identical value of the square of the sum for each problem. The tabular array in Figure 1B fans out away from the origin. The intersections labeled x in Figure 1B represent intersections containing correct sums for problems of the form "$b + 1$," where b is the larger addend, and intersections labeled y contain correct sums for problems of the form "$b + 2$." It is interesting that, in the tabular models in both Figures 1A and 1B, tie problems fall on adjacent intersection points on *straight-line* diagonals, the only nonparent node intersections following such patterns. The latter property may account for the exceptional RT results for tie problems reported in most studies of simple addition.

A fourth structural variable that is consistent with retrieval from a tabular memory network of addition facts is the *product* of addends. Although Ashcraft and his associates apparently never used the product of addends in studies of addition, Stazyk et al. (1982) found the product of the multiplicand and the multiplier to be the best predictor of RT to simple multiplication problems. In a recent study, Miller, Perlmutter, and Keating (1984) found the product of the two single-digit numbers to be the best predictor of RT for both simple addition and multiplication. Thus, it appears that the product may be more appropriate than the sum squared as the proper structural variable in statistical models for both addition and multiplication, reflecting an important functional component common to these two types of performance. But Miller et al. did not discuss any specific conceptual models that were compatible with the product as structural variable. As a result, the finding that the product of addends is the best predictor of addition RT is a rather enigmatic, unexplained phenomenon.

There is, however, a conceptual model of the network retrieval sort that is consistent with the product of addends as structural variable. This model is a geometric one and utilizes the square, symmetric, printed addition table as a metaphor for an adult's memory network for addition facts. Under this geometric model, we assume equal spacing of nodal values 0 through 9 along each of the two nodes. Next, we assume that activation of a nodal value takes a constant amount of time regardless of the magnitude of the addend. The activation of a nodal value may represent the memory network counterpart of the encoding of an addend, which may require a duration of time on the order of that required for access to name codes for letters (approximately 50–80 ms; Posner, Boies, Eichelman, & Taylor, 1969). Finally, we assume that the spread of activation through the memory network begins at the origin (i.e., the "0,0" intersection) and proceeds at a constant rate and as a linear function of the *area* of the network that must be traversed. Given these assumptions, the product of addends is equal to the area of the rectangle formed by the origin, the nodal values involved, and the intersection of the nodal values and is therefore linearly related to the area of the memory network that must be traversed to arrive at the correct sum for a problem. The geometric model has the features of a square, symmetric addition table with equally spaced nodal values, properties that Ashcraft and his associates found desirable but that had to be discarded under the sum squared model. The only additional assumptions required by the geometric model concern activation of nodal values and spread of activation through the memory network, and these assumptions appear to be reasonable and compatible with notions of spreading activation in semantic memory networks (e.g., Collins & Loftus, 1975).

Nontabular processes. Recently, several conceptual models that are not directly related to a tabular memory network have been proposed. One of the nontabular models is the *distribution of associations* model formulated by Siegler and Shrager (1984), as indicated in Table 1. In samples of subjects of rather low proficiency, Siegler and Shrager found an interesting relation between problem size and the distribution of associative strength of alternative potential answers for each problem: For problems with small addends, the distribution of associations for potential answers for a given problem tended to be rather peaked, with only one potential answer, the correct sum, clearly above a criterion for responding. For problems with larger addends, the distribution of associations tended to be flatter, with a less clear distinction of the correct sum from other potential answers. Problems with more peaked distributions of association require less time to verify because less cognitive effort is required to arrive at the correct sum; conversely, problems with flatter distributions of association require a greater amount of time to verify in order to determine which of several relatively likely answers is the correct answer.

The predictions made by the "distribution of associations" model may be highly related to those for certain table-related models. For example, if the peakedness of the distributions of associations for nontie problems is inversely and linearly related to the magnitude of each addend, the distribution of associations to a particular addition problem would be inversely and linearly related to the product of the addends. But the distributions of associations for tie problems may be very peaked, leading to a prediction of rather fast reaction times, regardless of the size of the addends. In this way, the "distribution of associations" model could account for the relatively constant RT for tie problems regardless of size of addends, while predicting a linear increase in RT as a function of product of the addends for nontie problems.

A second nontabular model, tested by Ashcraft, Fierman, and Bartolotta (1984), is based on the assumption that the speed of answering a problem should be a function of the *difficulty* associated with the problem. Ashcraft et al. used norms presented by Wheeler (1939), who reported the proportion of second-grade pupils who mastered each of the 100 simple addition problems and the ranked difficulty for each problem. Using the ranked difficulty measure, Ashcraft et al. found that problem difficulty was a better predictor of RT than was the MIN structural variable for samples of subjects from first grade through college and led to very similar equations for both verification and production formats of problem presentation.

A third nontabular model was recently proposed by Hamann and Ashcraft (1986). Hamann and Ashcraft reasoned that the problem size effect may arise because problems with larger addends are presented less *frequently* and later in *order* during the grades when children are learning addition. The inverse relation between problem size and frequency of presentation may lead to lower memory strength, and therefore longer retrieval times, for problems as a function of problem size. Using textbooks designed for use at each grade from kindergarten through third grade, Hamann and Ashcraft obtained indices of the frequency of presentation and the order of presentation of each of the 100 basic addition problems. Hamann and Ashcraft reported relatively large correlations for both the frequency and order of presentation with RT measures from their earlier studies for subjects ranging from first grade through college. Interestingly, the Hamann and Ashcraft study replicated the correlation between problem size and frequency of textbook presentation reported by Thorndike (1922).

One advantage of the preceding nontabular models is that the models appear to be more similar to, or more compatible with, models for nonarithmetic types of memory network (Siegler & Taraban, 1986), for example, models for semantic memory. A second advantage is that such models are capable of representing the basis for the exceptional RT results associated with tie problems; tie problems may have special appeal to children because they represent the addition of like sets (Arthur Baroody, personal communication, October, 1987), leading to rather lower levels of difficulty than would be expected from the magnitude of the addends in the problems. A third advantage of nontabular models is that such models could reflect the true parameters of network access that lead table-related structural variables to be highly related to RT. Some form of memory network storage of arithmetic facts is supported by recent research on priming and interference effects with mental multiplication (Campbell, 1987a, 1987b; Campbell & Graham, 1985; LeFevre, Bisanz, & Mrkonjic, 1988). Ultimately, the choice between tabular and nontabular models must be determined on bases such as which type of model leads to maximal goodness of fit with empirical data and to the confirmation of the greatest number of unique, testable hypotheses. The present study will allow evaluations of the relative goodness of fit of tabular and nontabular models.

Rule-Based, Procedural Processes

In the preceding section on memory network retrieval processes, retrieval of declarative knowledge was presumed to underlie addition performance. Declarative knowledge refers to stored knowledge *of* addition facts—for example, facts such as 2 + 3 = 5. Procedural knowledge, on the other hand, refers to stored knowledge *about* arithmetic. Procedural knowledge subsumes knowledge about procedural algorithms for arriving at a correct sum, procedural rules or heuristics for obtaining sums, and the like. As a function of systematic instruction in simple numerical operations that begins in first grade, Ashcraft (1982) hypothesized that the strength of both declarative and procedural knowledge increases with schooling. In early grades, children likely need to access procedural knowledge in order to recalculate the correct sum when presented with an addition problem because the memory trace representing declarative knowledge of the correct answer is not sufficiently strong to enable its retrieval. Upon successful solution of the addition problem and the resultant association of the correct answer with the two addends comprising the addition problem, the strength of the memory trace associations for the given problem is enhanced. Eventually, with practice and the strengthening of memory traces for the declarative knowledge of addends and their sums, retrieval of the correct answer for a given problem becomes the more efficient way of solving simple addition problems. Ashcraft (1982; Ashcraft & Fierman, 1982) documented just such a developmental progression from use of procedural (e.g., digital) processes to use of declarative, retrieval processes, a trend noted in earlier research (e.g., Brownell, 1935; Ilg & Ames, 1951).

However, in a number of recent articles, Baroody (1983, 1985, 1987) argued that procedural knowledge, rather than declarative knowledge, may frequently be used by highly proficient persons to solve addition problems. For example, frequency of practice cannot account for acquisition of all number facts (Baroody & Ginsburg, 1986) because acquisition and representation of addition combinations involving zero appear to involve a rule (Baroody, 1983), and there is clear, swift transfer to unpracticed problems once combinations involving zero are mastered (Baroody, 1985, 1987). The learning of "plus one" combinations, of the form "$x + 1$" and "$1 + x$," also appears to involve a rule relating the addition operation and knowledge of the basic number sequence (Baroody, 1985, 1987). Baroody reviewed these and other findings to support his argument that rules and procedures may underlie much addition performance. Incorporating such processes into our componential model, discussed below, is beyond the scope of the present article, but further research on such efforts appears strongly merited.

Components Required in a General Model for Mental Addition

In the preceding section, five general classes of process were discussed as alternative representations of the central elementary information process that is invoked when a person retrieves or recomputes the correct sum for an addition problem. There remains the task of placing the search/compute process within an adequate, comprehensive processing model for mental addition that specifies all of the important elementary information processes involved in problem solution. In the present section, a general model, capable of representing addition problems of any degree of complexity, will be proposed and discussed.

A Model for Simple Addition

The simple flow diagram for mental addition presented in Figure 2 provided an adequate conceptualization for most previous research on simple addition. However, when investigators (e.g., Ashcraft & Stazyk, 1981) introduced more complex forms of addition, certain parameters were estimated, such as time taken to carry from the units to the tens column, though it was unclear how such processes conformed to the simple flow diagram. Further, in several instances (see Ashcraft & Stazyk, 1981; Tables 2 & 4), two structural variables corresponding to alternative processes in the search/compute stage entered into the same equation. Ashcraft and Stazyk (1981) did not attempt to explain the latter findings, and it appears that there is no simple way in which to reconcile such findings with the model in Figure 2. There is a need, therefore, to develop a processing model that explicitly includes all elementary information processes invoked in solving addition problems.

A General Model for Simple and Complex Addition

The flow diagram for a general model for verification-task performance on mental addition is presented in Figure 3, which builds on the simple model in Figure 2. The general model is able to represent simple addition problems, as well as complex problems with any number of addends and any

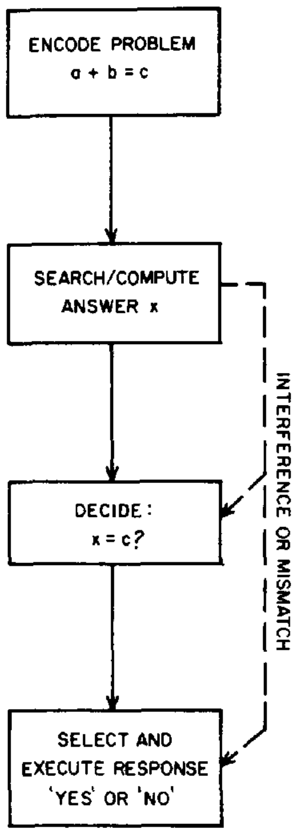

Figure 2. Flow diagram of simple model for verification task performance on addition (after Ashcraft, 1982). (Flow diagram adapted from Figure 3 in "The Development of Mental Arithmetic: A Chronometric Approach" by Mark H. Ashcraft, 1982; *Developmental Review, 2,* p. 229. Copyright 1982 by Academic Press, Inc. Adapted by permission.)

number of digits per addend. Given space constraints, only the components of the model in Figure 3 will be discussed below; this is not problematic because all components in the simple model in Figure 2 are contained within the model in Figure 3. Note here the following figural conventions used in Figure 3: Model components enclosed in boxes represent more controlled processes, processes assumed to require nontrivial durations that may be separately estimated; diamonds enclose branching operators that direct the flow of processing but that, being quite overlearned and therefore rather automatic, likely require trivial amounts of time for execution; and circles enclose operation of a counter that keeps track of the column of current processing, for example, units column.

The first stage depicted in the general model in Figure 3 involves encoding the type of addition problem presented. If problems are presented in homogeneous sets (e.g., a block of trials containing only simple addition problems), the first stage would be unnecessary. However, if problems of several types are presented in an intermingled, random manner, the person may need to initialize, or preset, certain counters or branching operators, given the number of addends and number of digits per addend, to allow optimally efficient processing of the given problem. As an example of the initialization of counters, we assume that persons will solve most simple and complex addition problems in columnwise fashion, beginning with the units column. Therefore, the column indicator is initialized with a setting to the first, or units, column, that is, $c = 1$.

The second step of processing involves a branching operator, the operation of which is determined by the number of digits in column c to be summed. The units column of all simple and complex addition problems must contain at least 2 digits, so the initiation of processing of an addition problem will always result in selection of the "> 1" branch the first time the present branching operator is invoked. However, in complex problems with more than one digit per addend, there may be only a single digit in the final, or leftmost, column of digits in the problem. In the latter case and if no carry operation to the final column is required, the "1" branch may be taken, as no summing of digits is required.

In the next stage of processing, two digits from column c are encoded. The encoding stage in Figure 3 assumes encoding of only two digits in column c. Previous conceptual models of addition, discussed above, presume that the elementary information process for addition obtains the correct sum of two numbers, and this appears to be a reasonable assumption. As a result, regardless of the number of digits in column c, we assume that the solving of the given addition problem will commence with the encoding of two, and only two, digits in column c. If there are more than two digits to be summed in column c, when encoding the first two digits the person must maintain in short-term memory an index variable indicating which digits are being encoded and which digits remain to be encoded and processed further. We assume that the encoding of the addends of an addition problem is likely to be a very rapid and overlearned process (cf. Poltrock & Schwartz, 1984), given the frequency with which individuals in our culture encounter numerical stimuli. However, encoding an addend probably requires retrieval from a long-term memory (LTM) store of certain attributes of the addend, such as the number of units represented by the addend. Such retrieval should take a consistent but *estimable* amount of time, perhaps on an order comparable to that found for access to name codes of letters (Posner et al., 1969; Hunt, Lunneborg, & Lewis, 1975).

The following stage is the search/compute stage, during which the correct sum of the two encoded digits is obtained. Once again, the search/compute stage in the present model is identical, in terms of both componential composition and temporal operating characteristics, to the like-named stage in the simple model in Figure 2. Thus, the correct sum of the two encoded digits is either retrieved from a memory store or calculated anew, with a temporal duration determined by problem size. The five major classes of processes presumed to underlie processing during the search/compute stage were discussed in detail above.

Immediately after the correct sum of the two digits has been obtained, a branching operator is encountered. If there are more digits in column c to be summed, the current sum is only provisional. Therefore, the current sum must be held in short-term memory while the *yes* branch is taken, which leads to the encoding of one more digit and the subsequent obtaining of a new sum. The preceding series of "encoding then summing" of digits continues until all digits in column

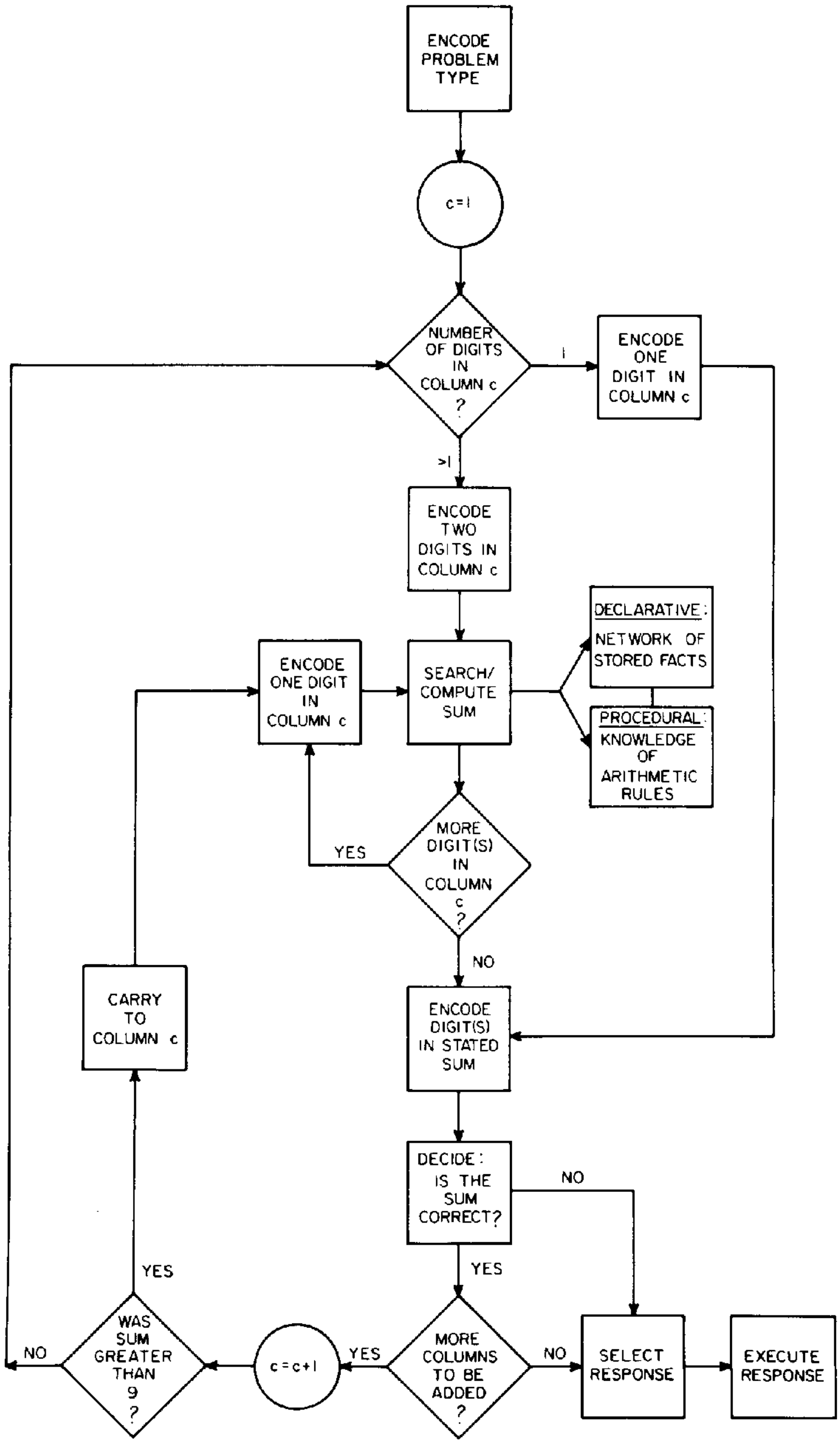

Figure 3. Flow diagram of general model for verification-task performance on mental addition.

c have been summed; at this point, the *no* branch is taken, as a correct column sum has been determined. One important prediction may be proposed regarding the form of the search/compute sequence. All conceptual models developed for a memory search process have assumed the existence of a long-term memory store for simple addition facts, that is, facts involving two single-digit addends. Given the preceding assumption, the first time the search/compute process is invoked in column c in a problem with more than two addends, the correct sum of any two of the digits in column c may be obtained via the efficient, automatized memory search process. The provisional sum, which serves as one addend as further digits in column c are summed, will soon be greater than 9, exceeding the bounds of the hypothesized memory network for simple addition facts. Hence, any subsequent invocation of the search/compute stage for column c should always result in the use of a slower, back-up process, of the counting or analog sort, because the correct sum would not reside in the stored network of facts.

Upon completion of the summing of digits in column c, the digits in the stated sum must be encoded so that the obtained and stated sums may be compared. If column c is *not* the final column of digits to be summed, only the single digit in column c of the stated sum need be encoded for comparison with the units value of the obtained sum. On the other hand, if column c is the final column in the given problem, then all remaining digits in columns c, $c + 1$, and so forth, of the stated sum must be encoded for comparison with the stated sum.

The decision stage comprises the comparison of the obtained and stated sums. The time required for this comparison may be an inverse function of the split, or difference, between stated and true sums (e.g., Ashcraft & Stazyk, 1981).

If the obtained and stated sums are unequal, the *no* branch from the decision stage is taken, and the response signifying "incorrect" is selected and then executed. If, conversely, the obtained and stated sums are identical, the *yes* branch is taken, leading to another branching operator. If there are no further columns of digits to be summed, the *no* branch is taken, and the response signifying that the stated sum is correct is selected and then executed. However, if there remain one or more columns of digits to be summed, the *yes* branch is taken. Processing then continues with a unit increment of the column counter, shifting the locus of processing to the next column of digits.

The final branching operator in our model governs whether the carry operation is performed. If the most recently obtained column sum was 9 or less, the *no* branch is followed, and processing of the given problem resumes with the encoding of one or two digits in column c. But, if the column sum in question was greater than 9, the *yes* branch must be taken, leading to the carrying to column c of information regarding the number of provisional tens. The latter information must be summed with one of the digits in column c before further digits from column c are encoded and summed.

Columnwise Versus Noncolumnwise Processing of Addition Problems

The design of the general model presented in Figure 3 was based on the assumption of columnwise processing of addition problems, an implicit assumption of most previous conceptual models for mental addition. However, both the analog model proposed by Restle (1970) and the procedural processes of Baroody (1987) presume that addition of numbers may be performed "of a piece," rather than columnwise. To represent such a conceptual model would require some modifications of the flow diagram in Figure 3. Specifically, all column counters and branching operators relating to column of processing and digits to be summed would be deleted as might the carry operator. Also, rather than encoding and summing only digits in column c, encoding of all digits in two multicolumn addends would be followed by the summing of the addends. The branching operator allowing looping if there were more than two addends would remain, as would the encoding of the stated sum, the comparison of the obtained and stated sums, and the selection and execution of a response. It appears that the flow diagram for noncolumnwise processing may require only modification and simplification of the diagram in Figure 3, but explicit, convincing tests of the relative fit of columnwise and noncolumnwise processing models could not be pursued within the scope of the present study.

The Present Study

The aim of the present study was to evaluate several hypotheses generated from consideration of the general model for mental addition presented in Figure 3. The first hypothesis was that the general model will allow the specification of one or more statistical models that well describe RT to simple and to several types of more complex addition problems. One supplement to the preceding general hypothesis is the hypothesis that one form of search/compute process would emerge as the best representation of RT data across types of addition problem, regardless of complexity of the problems.

The second hypothesis was that well-conditioned estimates for several elementary information processes underlying addition performance are estimable. Specifically, we attempted to estimate time associated with encoding of digits, the search for or computing of the sum, and the carry operation.[1]

The third, and final, hypothesis was that the temporal operating characteristics associated with the elementary information processes will either remain unchanged or change in a consistent, interpretable fashion as a function of problem

[1] Because we were interested in studying addition problems representative of those encountered outside the laboratory, we included in the present study only problems with correct or with "reasonably incorrect" stated sums. Exclusion of problems with "unreasonably incorrect" stated sums obviously led to nonidentifiability of a split effect parameter. Although the split effect, demonstrated by Ashcraft and Stazyk (1981), is of great importance when attempting to understand and model efficient verification-task performance on addition, the split structural variable appears only in models for verification-task performance and, therefore, is likely not representative of typical, everyday addition performance. Although the preceding argument appears to cast some doubt on the appropriateness of the verification-task format for the study of mental addition, Ashcraft, Fierman, and Bartolotta (1984) showed that use of production-task and verification-task formats led to nonsignificant differences in regression estimates for important structural variables common to the two formats.

complexity. The preceding hypothesis covers two types of effect: (a) instances in which the same process—for example, the search/compute process—is invoked more than once in the solving of a given problem—for example, in the units and tens columns, and (b) instances in which the same process appears in the identical position in the flow of processing in problems differing in overall complexity.

Method

Subjects

The subjects in the study were 23 undergraduate students at the University of California at Riverside who received course credit in introductory psychology for participation in the experiment. The mean age of subjects was 19.6 years (range: 16–29 years); there were 9 males and 14 females.

Stimuli

A set of 800 addition problems was used for the present study. The global set of problems comprised four types of problem, 200 problems of each type.

Type I: Two single-digit addends. The Type I problems were the basic 100 simple addition problems used in many previous studies (e.g., Parkman & Groen, 1971). The basic 100 problems result from the Cartesian product of using the digits 0 through 9 as first addend and the same set of digits as second addend. Presenting each problem once with the true sum and once with an incorrect sum, differing from the true sum by ± 1 or ± 2, resulted in the 200 Type I problems.

Type II: One single-digit and one double-digit addend. Each Type II problem contained one single-digit and one double-digit addend. The problems included in the study consisted of a random sample of 100 of the 900 problems formed by the Cartesian product of the 10 single-digit numbers 0 through 9 with the 90 double-digit numbers 10 through 99. For double-digit numbers, constraints invoked ensured that the numbers 0 through 9 appeared equally often in the units column and the numbers 1 through 9 appeared equally often in the tens column across the 100 problems. In 50 problems, the double-digit number appeared as the first addend; in the remaining 50 problems, the double-digit number was the second addend. Across the 100 problems, the numbers 0 through 9 appeared equally often as the single-digit addend, and the unit digit of the double-digit addend never equaled the single-digit addend; that is, no tie problems were allowed.

Presenting the 100 Type II problems once with the true sum and once with an incorrect sum led to the set of 200 Type II problems. Of the 100 incorrect problems, one half had sums differing from the correct sum by ± 1 or ± 2, while the remaining incorrect problems had sums differing from the correct sum by ± 10 (i.e., ± 1 in the tens column). These values were used to ensure that all Type II problems had either correct sums or sums that were not obviously incorrect.

Type III: Two double-digit addends. A constrained random sample of 100 of the 8,100 addition problems defined by the Cartesian product of the numbers 10 through 99 as the first addend and the same array of numbers as the second addend was chosen. The constraints resulted in each of the 90 numbers 10–99 appearing at least once as the first addend and at least once as the second addend, the digits 0 through 9 appearing equally often in each of the four positions (i.e., units or tens column of the first or second addend) and no ties allowed in either the units or tens columns. As with previous types of problem, the 100 problems with two double-digit addends were presented once with the true sum and once with an incorrect sum, generating the set of 200 Type III problems. For the incorrect problems, constraints with regard to the position and magnitude of the error in the stated sum identical to those for Type II problems were used.

Type IV: Three single-digit addends. The set of Type IV problems comprised a random sample of 100 of the 1,000 problems formed by the Cartesian product of using the single digits 0 through 9 as the first, second, and third addend in three-addend problems. Each of the digits 0 through 9 appeared equally often in each of the three positions across the 100 problems, and the same digit did not appear in consecutive positions within a given problem. The 100 Type IV problems were each presented once with the true sum and once with an incorrect sum (differing from the true sum by ± 1 or ± 2) to generate the 200 Type IV problems.

Problem sets. The 800 addition problems were randomly assigned to 200 quartets of four problems so that each quartet contained, in random order, one of each of the four types of problem. Consistent with previous research, no more than five consecutive correct or incorrect problems were allowed. In addition, a particular combination of digits was never allowed to occupy the same column in consecutive problems, to eliminate priming effects.

The first 100 quartets of problems were termed Set A, and the second 100 quartets were termed Set B. Twenty practice problems, five of each of the four types, were developed for each set of 400 problems.

Apparatus

The addition problems were presented at the center of a 30-cm × 30-cm video screen controlled by an Apple II Plus microcomputer. A Cognitive Testing Station clocking mechanism ensured the collection of RTs with ± 1-ms accuracy. Subjects were seated approximately 40 cm from the video screen and responded by depressing one of two response buttons located on a board directly in front them. Subjects were instructed to indicate that the stated sum for a problem was correct by depressing one of the response buttons and to depress the other response button if the stated sum was incorrect.

For each problem, a READY prompt appeared at the center of the video screen for a 500-ms duration, followed by a 1,000-ms period during which the screen was blank. Then, an addition problem appeared on the screen and remained until the subject responded, at which time the problem was removed. If the subject responded correctly, the screen was blank for 1,000 ms, and the READY prompt for the next problem then appeared. If the subject responded incorrectly, a WRONG prompt with a 1,000-ms duration followed the removal of the stimulus and preceded the 1,000-ms blank period.

Procedure

Subjects were tested individually in a darkened room in two sessions separated by at most 1 week. Twelve of the subjects received Set A during the first session and Set B during the second, and the remaining 11 received the sets in the B-A order. Subjects were told that they would see a variety of types of addition problem and were to indicate whether a problem was correct or incorrect by depressing the appropriate button. Speed and accuracy of responding were emphasized equally. After responding to the 20 practice problems, the speed/accuracy instructions were briefly repeated, and the subjects responded to the 400 problems in the set. Each session lasted approximately 45–50 min.

Results and Discussion

For clarity of presentation, the results from the present study will be presented and discussed in two major sections, followed by a general discussion of the results and their

Table 2
Summary of Regression Analyses for Type I Addition Problems

Equation	R^2	*df*	*F*	RMS_e
True, nontie problems				
RT = 926 + 7.91 (PROD)	.671	1, 88	179.28	106.1
RT = 879 + 2.09 (SUM^2)	.657	1, 88	168.21	108.3
RT = 908 + 64.1 (MIN)	.601	1, 88	132.53	116.8
False, nontie problems				
RT = 1057 + 10.53 (PROD)	.640	1, 88	156.70	151.1
RT = 998 + 2.74 (SUM^2)	.609	1, 88	136.94	157.6
RT = 1027 + 87.3 (MIN)	.600	1, 88	132.04	159.4
Combined set, nontie problems				
RT = 926 + 7.91 (PROD) + 131 (TRUTH) + 2.62 (PROD × TRUTH) Partial *F*s: 118.36, 22.17, 6.52	.702	3, 176	138.50	130.6
RT = 682 + 80 (ENCODE) + 6.25 (PROD) + 133 (TRUTH) + 2.72 (PROD × TRUTH) Partial *F*s: 6.89, 43.01, 23.54, 7.20	.714	4, 175	109.08	128.4
RT = 712 + 66 (ENCODE) + 7.26 (PROD) + 93 (TRUTH) + 66 (REENC) Partial *F*s: 13.42, 97.32, 8.99, 13.42	.713	3, 176	145.96	128.2
Combined set, tie problems				
RT = 971 + 121 (TRUTH)	.240	1, 18	5.68	113.3
Combined set, tie and nontie problems				
RT = 726 + 68 (ENCODE) + 5.32 (PROD) + 81 (TRUTH) + 68 (REENC)	.561	3, 196	83.81	154.8
RT = 760 + 50 (ENCODE) + 7.72 (PROD*) + 106 (TRUTH) + 50 (REENC)	.702	3, 196	153.88	127.7

Note. Mean RT = 1,169.40 ms. PROD stands for product of addends; PROD* is same as PROD except that values for tie problems are set to zero; SUM^2 = squared sum of addends; MIN = smaller addend; TRUTH = correct (0) or incorrect (1) stated sum; ENCODE = number of items encoded (3 or 4); and REENC = reencoding of digits in incorrect stated sums (1 or 2).

implications. In the first major section, the results of analyses demonstrating the intensive validation of the proposed general model for each type of problem will be presented. In the second major section, results demonstrating the extensive validation of the general model will be reported. Following previous research (e.g., Ashcraft & Battaglia, 1978), the average RT across subjects was computed for each problem and served as dependent variable.

Intensive Validation

Type I: Simple Addition

A total of 200 simple addition problems were included in the study, leading to 4,600 reaction times across the 23 subjects. Overall error rate was 2.52%, and an additional 0.1% of responses were eliminated on the basis of a test for outliers (Dixon's test; Wike, 1971). Thus, a total of 2.63% of the responses, or 121 of the 4,600 responses, by subjects were excluded from analyses, an error rate that is relatively low and comparable to that in previous studies (e.g., Ashcraft & Battaglia, 1978).

True, nontie problems. As in previous studies of mental addition, a wide variety of potential predictors of RT to the 90 true, nontie problems was entertained. Specifically, all 12 structural variables listed in Table 1 were tested individually as were 8 additional structural variables proposed by Ashcraft and Battaglia (1978) and Ashcraft and Stazyk (1981). The latter set of structural variables included such variables as whether the first addend was even or odd and whether the true sum was a single- or double-digit number.[2]

A summary of the results of the regression analyses is presented in the first section of Table 2, where equations for the best three predictors are given. As shown in Table 2, both the MIN (R^2 = .601) and the sum-squared (R^2 = .657) structural variables provided good representations of RT to the true, nontie problems. However, the strongest predictor of RT was the product of the addends, which explained over 67% of RT variance, $F(1, 88) = 179.28$, $p < .0001$. The analyses of the true, nontie problems thus provided strong support for some type of network retrieval conceptual model over any of the counting models because two structural variables consistent with network retrieval models—the sum-squared and the product—were clearly the strongest predictors of RT. The best nontabular retrieval model included the

[2] Because of space limitations, a table providing the intercorrelations among all potential predictors and RT for true and false problems was not included. This table is available from the first author on request.

Wheeler difficulty measure[3] as predictor and had an R^2 of only .620.

Because there have been no previously published results using the product as predictor of *raw* RT to addition problems,[4] it is not possible to determine the comparability of the obtained results for the product variable with any previous findings. However, the RT data reported in Table 2 appear quite representative of certain previous results. For example, Ashcraft and his associates (Ashcraft & Battaglia, 1978; Ashcraft & Stazyk, 1981) reported regression weights for the sum-squared structural variable, across several experiments, that ranged from 1.15 to 2.60 ms per unit increase in the sum squared.[5]

False, nontie problems. The set of structural variables for predicting RT to simple addition problems was applied next to the false, nontie problems. The results for the best three equations are presented in the second section of Table 2. The best structural variable consistent with a counting model was once again the MIN (R^2 = .60), and the sum-squared again outperformed all counting and nontabular retrieval models (R^2 = .61). However, the product structural variable was clearly the strongest predictor of RT to the false, nontie problems (R^2 = .64), repeating its success in representing RT to true, nontie problems.

The regression equation for the product structural variable, RT = 1,057 + 10.53 (product), offered interesting contrasts with the corresponding equation for true, nontie problems: The intercept for the false problems was about 130 ms larger than that for true problems, and the regression slope for the product variable was approximately 33% larger for false problems. The differences in intercept between true and false problems probably reflect a difference in the selection and execution of responses for the two types of problems. Furthermore, the intercept difference is similar to intercept differences reported for a variety of tasks, from scanning in short-term memory (Sternberg, 1966) to spatial rotation (Cooper & Shepard, 1973).

The difference between true and false problems in the regression slope for the product structural variable is, however, more problematic than the intercept difference. According to additive factors methodology (Sternberg, 1969), the interaction of a structural variable, such as the product, with a second variable, such as truth versus falsity of the stated sum, implies some form of departure from strict serial processing of the associated stages; this problem will be discussed in more detail below. However, the slope difference reported in Table 2 is representative of previous research findings. For example, Ashcraft and Battaglia (1978) reported regression slope estimates for the sum-squared variable that were approximately 50%–70% larger for false problems than for true problems, and similarly disparate weights for true and false problems were reported by Ashcraft and Stazyk (1981). To our knowledge, in no previous research on mental addition has the apparent difference between true and false problems in processing rates for important structural variables been stressed; rarely, if ever, has the difference in slope estimates for true and false problems been tested for significance.

Combined set of nontie problems. In order to represent and test most efficiently the differences between true and false problems, analyses of the combined set of 180 nontie problems were undertaken. As discussed in texts on regression analysis (e.g., Cohen & Cohen, 1983), a simple way to test differences between regression equations is to place all observations in a single data set and then to employ properly specified pseudovariates, both alone and multiplied by important structural variables, to test intercept and slope differences for groups of observations. Here, one pseudovariate, coded 0 for true problems and 1 for false problems, was used to represent intercept differences between true and false problems, and the product of the preceding pseudovariate and the product structural variable was used to test the difference in slope estimates for true and false problems.

Analyses of the combined set of simple, nontie addition problems are presented in the third section of Table 2; analyses were restricted to use of the product structural variable because of the superior performance of the product variable as predictor of RT for both true and false problems. The first equation included the product variable, the truth variable, and the product of the preceding two variables. The first regression equation provided a strong representation of the RT data, R^2 = .702, $F(3, 176) = 138.50$, $p < .0001$. Because the truth variable was zero for all true problems, the truth and Truth × Product structural variables may be dropped from the equation for true problems; this leaves an equation, RT = 926 + 7.91 (product), that was identical to that derived from analyses of true problems alone. In a similar manner, the equation for false problems, for which the truth variable takes on a value of unity, is obtained as RT = (926 + 131) + (7.91 + 2.62) (product) = 1057 + 10.53 (product), identical to the equation from analyses of false problems alone. Most importantly, the combined analyses allow a direct estimate and test of the intercept difference between true and false problems, b = 131 ms, $F(1, 176) = 22.17$, $p < .0001$, and of the slope difference for the product structural variable, b = 2.62 ms, $F(1,176) = 6.52$, $p < .01$.

Although the preceding tests imply that the slope estimates for the product structural variable differ significantly across true and false problems, there remains the possibility that the

[3] The Wheeler difficulty measure employed in our analyses was the percentage of students in the Wheeler study who failed to master each simple addition problem. Because Wheeler (1939) reported percent mastery for each problem, we simply calculated (1 − percent mastery) to arrive at a measure of problem difficulty. This percentage measure consistently explained more variance than did the ranked difficulty index provided by Wheeler.

[4] Miller, Perlmutter, and Keating (1984) used the product as structural variable for predicting RT to simple addition problems but did not use raw RT as the dependent variable. Rather, Miller et al. used an adjusted RT as dependent variable, adjusted RT computed as RT to simple addition problems minus time taken to name the correct sum, estimated on the basis of a number naming task.

[5] Because Baroody (1983, 1987) discussed the exceptional nature of problems involving zero, we recomputed all analyses reported in Table 2 excluding all problems involving zero. The results of these supplementary analyses were virtually identical to those reported in Table 2, both with respect to parameter estimates and to the ranking of models for their predictive efficiency. A table reporting these supplementary analyses is available from the first author on request.

test of the interaction is biased because of model misspecification, which may lead to bias in regression weight estimates and their associated tests of significance (Cohen & Cohen, 1983). A common type of model misspecification is failure to include a structural variable that is required to represent the underlying processes generating the data. According to the general model in Figure 3, time for encoding digits is a parameter that could be estimated, and the failure to estimate encoding time could have biased the previous regression results. Therefore, another structural variable, ENCODE, was specified that represented the number of items, or digits, encoded during processing of a given addition problem, and took on values of 3 and 4 for problems with one- and two-digit stated sums, respectively. The ENCODE variable was included in the second equation for the combined set of problems (see Table 2). The time taken to encode each digit was estimated to be 80 ms, a value rather similar to that reported for retrieving name codes for letters (Hunt et al., 1975). Incorporating the ENCODE variable in the regression equation led to a significant increase in explained variance, $\Delta R^2 = .011$, $F(1, 175) = 6.89$, $p < .01$. But, although introduction of the ENCODE variable appropriately altered certain of the regression estimates, the regression weight estimates and associated levels of statistical significance for the truth main effect and the Truth × Product interaction remained largely unchanged.

The final form of model misspecification that was considered involved a possible difference in the processing of stated sums for true and false problems. There is evidence for multiplication facts that presenting an incorrect answer for a problem increases RT, presumably due to interference effects between the true and stated answers (Campbell, 1987b). The interference probably leads the subject to engage in longer or more involved encoding of the incorrect answer; we have termed this effect a "reencoding" of the digits in the incorrect sum.

The effect that the encoding of digits and the reencoding of digits in incorrect stated sums might have on RT is illustrated in Figure 4 with hypothetical data. The bottom two solid lines represent RT for true problems, and the top two solid lines represent RT for false problems. The solid lines drawn through unfilled data points represent RT to problems with single-digit sums, while solid lines drawn through filled data points represent RT to problems with double-digit sums. All solid lines in Figure 4 are parallel, reflecting identical memory search rates for simple addition problems regardless of the number of digits in the stated sum and the correctness of the stated sum. The vertical separation between the two solid lines for true problems represents the difference in time taken to encode one digit; three digits are encoded in true problems with single-digit sums, and four digits are encoded when sums have two digits. The dashed line drawn as a single best-fit line for true problems provides a graphical demonstration of model misspecification: If the difference in number of digits encoded for true problems with one- and two-digit sums is not accounted for, the regression weight for the product variable based on RT to true problems will be positively biased and, hence, overestimated.

Turning to false problems, the effects of model misspecification are again apparent in the patterns in data portrayed in Figure 4. Assuming that subjects spend more time encoding digits in stated sums of false problems, the vertical separation of the two solid lines for false problems should be twice that for true problems, reflecting the difference in RT for encoding two digits. In false problems with single-digit stated sums, subjects would encode three digits and then reencode the single-digit sum, a total of four digits; in problems with two-digit sums, subjects would encode four digits and then reencode the two-digit sum, a total of six digits. If subjects perform such reencoding of digits in incorrect stated sums, the failure to include a reencoding parameter in the model would lead to positive bias in the estimated search rate for false problems (see the dashed line for false problems in Figure 4); more importantly, the failure to include a reencoding parameter would lead to different estimated search rates for true and false problems, represented by the two dashed lines in Figure 4, even though the true search rates for all problems were *identical.*

To test the plausibility of the reencoding hypothesis, we specified a reencoding structural variable (REENC) with a value of 0 for all true problems and values of 1 and 2 for false problems with single- and double-digit sums, respectively. With the procedures outlined by Rindskopf (1984), the estimated regression weights for the ENCODE and REENC variables were forced to equality, embodying the a priori hypothesis that encoding and reencoding have identical temporal durations per digit. The resulting equation is presented as the third equation based on the combined set of nontie problems. In the third equation, encoding is estimated to take 66 ms per digit, reencoding is estimated to take an additional 66 ms per digit in false stated sums, the search rate parameter for the product structural variable is 7.26 ms per unit increase in the product, and the residual difference in RT between true and false problems is 93 ms. The third equation, which utilized three regression parameter estimates, had a level of fit ($R^2 = .713$, $SE = 128.2$ ms) that was essentially equal to that of the

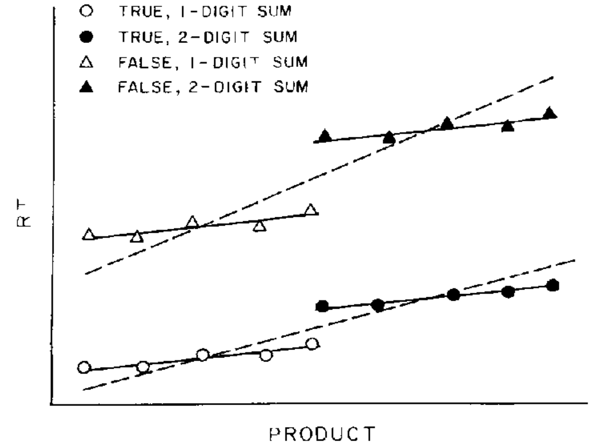

Figure 4. Hypothetical data embodying effects of the product structural variable, the correctness of the stated sum, and the encoding and reencoding of digits on RT to simple addition problems.

second equation ($R^2 = .713$, $SE = 128.4$ ms), which required four parameter estimates; the third equation was therefore preferable on grounds of parsimony—equal levels of fit based on one fewer parameter estimate.

Two further results support acceptance of the third equation over the second. First, relaxing the equality constraint on the regression weights for the ENCODE and REENC variables led to a nonsignificant increment in explained variance ($F < 1.0$), favoring the a priori equality constraint. Second, and more important, the Product × Truth variable, representing the interaction of search rate and correctness of the stated sum, was nonsignificant, $F(1, 175) = 2.09$, $p > .15$. Therefore, a model positing identical memory search rates for true and false problems cannot be rejected if the reencoding parameter is included in the model. The two effects of proper model specification represented in the hypothetical data in Figure 4 were, thus, borne out in analyses of data from the combined set of problems: (a) the estimated search rate parameter from the third combined set equation ($b = 7.26$ ms) was smaller, and presumably less biased, than the corresponding parameter estimate based on analyses of either true ($b = 7.91$ ms) or false ($b = 10.53$ ms) problems alone, and (b) the search rate parameter did not differ significantly across true and false problems.

Tie problems. Regression analyses were performed on true tie problems, false tie problems, and the combined set of tie problems. In none of the analyses did any structural variable associated with the search/compute stage, whether reflecting a counting or memory search conceptual model, emerge as a significant predictor of RT. The only structural variable significantly related to tie problem RT was the truth variable. As shown in the fourth section of Table 2, the truth variable explained 24% of RT variance for the combined set of 20 true and false tie problems, $F(1, 18) = 5.68$, $p < .01$. As shown by the equation, true tie problems had a mean RT of 971 ms, while false problems had a mean RT of 1,092 ms. The difference between true and false tie problems, $b = 121$ ms, was approximately the same order of magnitude as that estimated on the basis of nontie problems (see Table 2).

Combined set of tie and nontie problems. A final set of analyses, based on the total set of 200 tie and nontie problems, was undertaken to test a central assumption associated with the nontabular retrieval models: that nontabular models could more easily account for the exceptional RT patterns for tie problems than could any tabular retrieval model. In the fifth section of Table 2, two equations are presented that have the product of addends as the retrieval structural variable. In the first equation, the product of addends was used for all problems; in the second equation, the value on the product variable (PROD*) for each tie problem was set to zero, representing the lack of a problem size effect for tie problems. The second equation had a rather higher level of fit, $R^2 = .702$, than did the first equation, $R^2 = .561$, supporting the hypothesis that tie problems have a flat problem size effect. More important, replacing the product variable in the first equation with any of the nontabular retrieval structural variables invariably led to rather lower levels of fit. The best equation involved the Wheeler difficulty measure, an equation that had an R^2 of only .550. These results clearly support the product of addends as the structural variable, among those considered in this study, most strongly reflecting the temporal course of retrieval of addition facts from an LTM store.

Type II: Addition of One Single-Digit and One Double-Digit Addend

A total of 200 Type II addition problems were included in the study, resulting in 4,600 RTs to Type II problems across the 23 subjects. Overall error rate was 3.48%, and an additional 0.3% of responses were eliminated on the basis of a test for outliers (Dixon's test; Wike, 1971). Thus, a total of 3.78% of the responses, or 174 of the 4,600 responses, by subjects to Type II problems were excluded from analyses. The error rate for the Type II problems, although somewhat higher than that for Type I problems, was still quite low. The somewhat higher error rate was most probably due to the greater complexity of the Type II problems. Given the successful combined set analyses of Type I problems and the subsequent ability to test interactions of structural variables with the truth variable, only combined set analyses were performed on Type II problems. Also, the constraints invoked during construction of Type II problems eliminated tie problems, simplifying the types of analyses performed.

Under columnwise processing of Type II problems, the effects of several elementary components should be identifiable. Given the general model in Figure 3, the identifiable components include the following: the encoding of digits in the addends and stated sum, the searching for/computing of the correct sum in the units column, the presence of a carry from the units to the tens column, the searching for/computing of the correct sum in the tens column if a carry is undertaken, a residual intercept difference between true and false problems, and a possible reencoding of digits in incorrect stated sums.

To identify the operation of the preceding processes, structural variables for each had to be specified. To reflect columnwise processing, structural variables associated with the search/compute stage were specified separately for the units and tens columns. Note that the search for, or computing of, a sum in the tens column was required only if a carry from the units column occurred. To represent self-terminating processing, the encoding structural variable, ENCODE, took on a value of 3 for false problems with an error in the units column, and values of 5 or 6 for the remaining problems, depending on whether the stated sum had 2 or 3 digits, respectively. The reencoding variable had a value of 0 for true problems, 1 for false problems with an error in the units column, and 1 or 2 for false problems with single- or double-digit sums in the tens column, respectively. The carry structural variable had values of 0 for absence and 1 for presence of a carry, and the truth variable was coded 0 for true problems and 1 for false problems. Finally, the structural variables for the search/compute process for the tens column and the carry were coded 0, indicating self-termination and hence absence of process execution, for false problems with an error in the units column of the stated sum.

Table 3
Summary of Regression Analyses for Type II Addition Problems

Equation	R^2	*df*	*F*	RMS_e
RT = 776 + 108 (ENCODE) + 11.60 (UNITPROD) + 150 (CARRY) + 11.60 (TENPROD) + 111 (TRUTH) + 108 (REENC) Partial *F*s: 38.03, 142.43, 9.61, 142.43, 13.68, 38.03	.743	4, 195	140.93	200.0
RT = 724 + 118 (ENCODE) + 11.66 (UNITPROD) + 147 (CARRY) + 11.66 (TENPROD) + 236 (TRUTH) Partial *F*s: 34.66, 140.62, 8.93, 140.62, 49.30	.740	4, 195	138.19	201.5
RT = 749 + 104 (ENCODE) + 96.3 (UNITMIN) + 142 (CARRY) + 96.3 (TENMIN) + 117 (TRUTH) + 104 (REENC) Partial *F*s: 35.31, 141.77, 8.38, 141.77, 15.30, 35.31	.741	4, 195	140.56	200.6

Note. Mean RT = 1,682.96 ms. ENCODE stands for number of items (digits) encoded (either 3, 5, or 6 in this set of problems); UNITMIN and TENMIN for minimum addend in units and tens columns, respectively; UNITPROD and TENPROD for product of addends in units and tens columns, respectively; CARRY for presence (1) or absence (0) of a carry from the units to tens column; TRUTH for correct (0) or incorrect (1) stated sum; and REENC for the reencoding of digits in incorrect stated sums (1 or 2).

All structural variables for the search/compute stage listed in Table 1 were used in separate equations, once with and once without the reencoding process, to model RT to Type II problems. In Table 3, the three equations providing the best fit to RT are presented. The first equation had the highest level of fit, $R^2 = .743$; this equation included the columnwise product of addends as structural variable for the search/compute stage as well as the reencoding process. Two a priori equality constraints were invoked: one for the search rate estimate in the units and tens columns and the other for the encoding and reencoding processes. Both equality constraints were justifiable statistically because relaxing either constraint led to nonsignificant increments in R^2 ($p > .25$). Interactions of truth with encoding, product, and carry structural variables were all nonsignificant ($p > .15$). The parameter estimates for the intercept and truth variables were approximately 10% higher than those from equations for simple addition, while estimates for the encoding, reencoding, and search rate variables were approximately 60% higher; the reliability and interpretation of these differences between estimates from simple and complex addition problems are discussed later.

The second and third equations are presented for purposes of comparison. The second equation is identical to the first except for the deletion of the reencoding process. Because of the equality constraint on encoding and reencoding processes, the first equation attained a somewhat higher level of fit than did the second equation, $R^2 = .743$ versus .740, for the same number of parameter estimates, supporting acceptance of the first equation. The third equation, which included a structural variable consistent with a counting process, gave the next highest level of fit. The level of fit for the best model incorporating the columnwise sum-squared structural variable was lower still, $R^2 = .722$.

Type III: Addition of Two Double-Digit Addends

Across the 23 subjects, a total of 4,600 RTs to Type III problems were obtained, given the 200 Type III problems included in the study. Overall error rate was 6.72%, and an additional 0.11% of the RTs was excluded on the basis of a test for outliers (Wike, 1971). Thus, a total of 6.83% of the responses, or 314 of the 4,600 responses, by subjects to Type III problems were excluded from analyses. Although the overall error rate was somewhat higher than the error rates for Type I and Type II problems, the greater complexity of the Type III problems is likely responsible for the higher error rate. In spite of this, the error rate is still relatively low and should not affect modeling of effects on RT. The analyses of Type III problems were very similar in form to those performed on Type II problems. The only important difference involved model specification of the structural variable for the search/compute process in the tens column. In Type III problems, the search/compute process was executed regardless of the presence of a carry from the units column because there were always two tens-column digits to be summed. In all modeling of Type III data, if a carry occurred, we assumed that the carried unit value was added to the first digit in the tens column by the same process and the same search rate as occurred for the summing of the two tens-place digits.

The three equations providing the best fit to Type III problem RT data are presented in Table 4. The first equation includes separate columnwise search rate estimates and includes the reencoding process. An a priori equality constraint for the encoding and reencoding processes resulted in poorer fit and a negative regression weight for the truth variable that would be difficult to interpret. The estimated regression weight for the ENCODE variable for Type III problems was rather larger than that estimated for previous problem types; it seemed reasonable to assume that forcing the reencoding regression weight to equal the relatively large regression weight for the ENCODE variable might have produced the unacceptable set of estimates. Freely estimating the reencoding rate led to a much lower estimate, but the standard error for the estimate was rather large. Hence, to produce a statistically well-conditioned, appropriate estimate, the regression weight for the reencoding variable was constrained to equal one-third

Table 4
Summary of Regression Analyses for Type III Addition Problems

Equation	R^2	df	F	RMS_e
RT = 798 + 158 (ENCODE) + 10.71 (UNITPROD) + 245 (CARRY) + 7.52 (TENPROD) + 113 (TRUTH) + 53 (REENC) Partial *F*s: 104.26, 84.93, 24.37, 53.11, 10.39, 104.26	.841	5, 194	205.10	224.3
RT = 893 + 141 (ENCODE) + 8.92 (UNITPROD) + 287 (CARRY) + 8.92 (TENPROD) + 117 (TRUTH) + 47 (REENC) Partial *F*s: 116.97, 129.24, 39.27, 129.24, 10.91, 116.97	.837	4, 195	251.04	226.2
RT = 791 + 150 (ENCODE) + 76.4 (UNITMIN) + 352 (CARRY) + 62.6 (TENMIN) + 134 (TRUTH) + 50 (REENC) Partial *F*s: 68.81, 61.51, 50.75, 31.89, 12.96, 68.811	.821	5, 194	177.97	237.9

Note. Mean RT = 2,254.43 ms. ENCODE stands for number of items (digits) encoded (either 3, 6, or 7 in this set of problems); UNITMIN and TENMIN for minimum addend in units and tens columns, respectively; UNITPROD and TENPROD for product of addends in units and tens columns, respectively; CARRY for presence (1) or absence (0) of a carry from the units to tens column; TRUTH for correct (0) or incorrect (1) stated sum; and REENC for the reencoding of digits in incorrect stated sums (1 or 2).

the weight for the ENCODE variable so that the reencoding rate would be of similar magnitude to that found for Type I problems. The resulting equation is reported as the first equation in Table 5; the equation has a very high level of fit, $R^2 = .835$, and reasonable regression weights for all structural variables. The weights for the encoding (ENCODE) and carry variables for Type III problems are rather larger than corresponding estimates for Type II problems, but the remaining estimates were either fairly stable or decreased somewhat when compared with estimates from Type II problems. No interactions of structural variables with truth approached significance (all $ps > .40$).

The second equation in Table 4 is identical to the first except for imposition of an equality constraint on search rate in the units and tens columns. The equality constraint led to a small decrease in explained variance, $\Delta R^2 = .004$, that was of borderline significance, $F(1, 194) = 2.30$, $p < .08$. Because there are benefits associated with each of the first two equations, the two equations are presented as alternative representations of the Type III data; however, parsimony likely resides on the side of the second equation. The third equation, which incorporates a counting process structural variable, provided the next best, but rather poorer, fit to the Type III RT data. As with the first equation, in neither the second nor the third equation did structural parameters differ significantly for true and false problems (all $ps > .40$).

Type IV: Addition of Three Single-Digit Addends

Given the 200 Type IV problems included in the study, 4,600 RTs were obtained across the 23 subjects. Overall error rate was 3.63%, and an additional 0.17% of the RTs were excluded from analyses on the basis of a test for outliers (Wike, 1971). Thus, a total of 3.80% of the responses, or 175

Table 5
Summary of Regression Analyses for Type IV Addition Problems

Equation	R^2	df	F	RMS_e
RT = 675 + 114 (ENCODE) + 7.42 (LARGEPROD) + 239 (MIN) + 114 (REENC) Partial *F*s: 36.26, 28.68, 238.87, 36.26	.827	3, 196	311.99	271.6
RT = 1,154 + 9.68 (LARGEPROD) + 242 (MIN) + 168 (TRUTH) Partial *F*s: 48.57, 225.78, 17.52	.812	3, 196	281.55	283.3
RT = 843 + 54.7 (LARGESUM) + 246 (MIN) + 169 (TRUTH) Partial *F*s: 55.18, 266.31, 18.24	.817	3, 196	290.92	279.5

Note. Mean RT = 2,027.19 ms. ENCODE stands for number of items (digits) encoded (either 4 or 5 in this set of problems); LARGEPROD for the product of the two largest addends; MIN for the smallest of the three addends; LARGESUM for the sum of the two largest addends; TRUTH for correct (0) or incorrect (1) stated sum; and REENC for reencoding of digits (1 or 2) in incorrect stated sums.

of the 4,600 responses, by subjects to Type IV problems were excluded from analyses, an error rate that was rather low and approximately equal to the error rate for Type II problems.

The Type IV problems presented some unique possibilities and difficulties for the modeling of RT. Translating previous types of statistical models, developed for addition of two addends, to represent accurately the addition of three addends led to a geometrically increased array of models to be considered. According to our general model in Figure 3, addition always proceeds with the summing of two digits, and a third digit could then be added onto the provisional sum. Formulating models representing analog or digital processing led to a large set of potential models. For example, one model that represented an extension of the MIN model to Type IV problems was this: Set the counter to the largest addend, then increment the midsized and smallest addends.

A similarly large array of memory network retrieval models was specified. An example of network retrieval models was as follows: Search the memory network for the sum of the two largest addends and then either search the network for the sum of the provisional sum and the smallest addend or increment the smallest addend onto the provisional sum. The former option might be taken if the sum of the two largest addends was less than 10 because the provisional sum could be relabeled mentally as a single-digit addend; incrementing might occur only if the provisional sum was greater than 9 because the provisional sum exceeded the bounds of the memory network. Or, the latter, incrementing option might always be taken; once a sum is retrieved from the memory network, the subject may automatically switch to a back-up process because of the need to expend mental resources on monitoring the process of problem solution. Thus, because he or she must keep track of the digits already summed and the digit that remains to be summed, the subject may always revert to a slower, back-up incrementing process after obtaining the provisional sum in the most efficient manner. The structural variable representing the memory search process was not specified in the preceding example; in fact, any of the network retrieval structural variables listed in Table 1 could be used. As a result, separate models were fit to the data by using each of the retrieval structural variables, in turn, as the representation of the retrieval process.

Finally, several models, which might be termed *three-sum* models were considered. An example of the three-sum models was one incorporating as a structural variable the product of all three addends. The product of the three addends is a direct function of the volume of a three-dimensional network that must be traversed to arrive at the intersection of the three nodal values, a direct function of volume, assuming equally spaced nodal values 0 through 9 in a network with three orthogonal dimensions. No such three-dimensional network has been proposed in previous research on mental addition. Furthermore, the preceding three-sum model reflects simultaneous summing of all three addends, rather than summing of two addends at a time as embodied in our general model. Thus, none of the three-sum models was expected to describe well RT to Type IV problems. However, several types of three-sum models were fit to the data to ensure that formulation of our general model in Figure 3 had not led to our failure to consider other potentially appropriate models.

The three regression models providing the best fit to RT data for Type IV problems are presented in Table 5. The best equation, listed as the first equation in Table 5, included the product of the two largest addends as structural variable for the memory search process. Interestingly, the estimated regression weight for the product variable (b = 7.42 ms) was very similar to the corresponding regression weight derived on the basis of other problem types (e.g., b = 7.26 ms for Type I problems). The first regression model also included digit encoding and incorrect sum digit reencoding processes. The estimated time for encoding digits, b = 114 ms, was very similar to estimates based on Type II problems. When the encoding and reencoding processes were added to the equation, the truth variable had a negligible, nonsignificant regression weight and was therefore dropped from the equation. The estimated intercept for the model, a = 675 ms, was somewhat lower than, but still quite similar to, estimates based on previous problem types.

The first regression model also included the smallest addend, MIN, as a structural variable, representing rate of incrementing onto the provisional sum. The estimated regression weight for the MIN variable was b = 239 ms, a value that was several times larger than regression weights estimated for the MIN variable from previous types of problems. An explanation of the magnitude of this effect is provided by an experiment by Landauer (1962). Among other conditions, Landauer (1962) had adult subjects recite, either aloud or implicitly, the 10 numbers from 11 to 20. The preceding task, whether numbers were recited aloud or implicitly, took between 2 and 2.5 s, yielding a recitation rate of 200–250 ms per number. The estimate from Landauer (1962) of 200–250 ms per number agrees quite well with the estimated incrementing rate of 239 ms, suggesting that subjects increment the smallest addend onto the provisional sum in a unit-by-unit fashion via a relatively slow, implicit speech process.

The second and third equations are presented in Table 5 for purposes of comparison with the first equation. If the reencoding process is dropped from the first equation, the truth structural variable becomes a significant predictor of RT, resulting in the second equation in Table 5, which had a lower level of fit than the first equation. The third equation replaced the retrieval variable with a digital process variable, but this equation also had a lower level of fit. These equations support inclusion of the retrieval and reencoding processes, even at the expense of the truth parameter.

To summarize the remaining, unreported analyses, the best model including the sum-squared structural variable had a lower level of fit than did the models listed in Table 5, R^2 = .809. Alternative counting models also had lower levels of fit, R^2s < .800. Finally, the best three-sum equation, utilizing the product of all three addends, had a still worse level of fit, R^2 = .727.

Extensive Validation

Analyses of Problems with Two Addends: Types I, II, and III

The processing models for Type I, II, and III problems were fairly similar, including the summing of at most two digits per column. In order to determine whether a smaller, constrained set of estimates would adequately represent RT for three types of problem, the 600 problems in Types I, II, and III were combined into a single data set. The 20 Type I tie problems were included in all combined analyses. Because RT to tie problems did not vary as a function of problem size, tie problems were given values of 0 on the product structural variable. Values for tie problems on other structural variables—for example, the encoding structural variable—were set at appropriate levels.

The first model specified was termed the *baseline* model. The baseline model comprised a fully parameterized regression model that had separate regression estimates for each type of problem for the intercept and truth variables, for speed of encoding and reencoding processes, for memory network search rates in the units and tens columns, and for speed of executing the carry operation. The baseline model resulted in identical parameter estimates and levels of fit for each type of problem as were reported in Tables 2, 3, and 4. Across the 600 problems, the baseline model required 18 parameter estimates and had a very high level of fit, $R^2 = .9075$, $F(18, 581) = 316.48$, $p < .0001$, $SE = 188.6$.

Next, a series of nested models was specified, each member of which constrained to equality across all problem types one set of parameters (e.g., the parameter estimate for the carry operation). The fit of each of these models was compared with the baseline model to determine whether a significant decrease in fit accompanied imposition of the particular equality constraint. For example, forcing the intercept terms to be equal across the three problem types left the R^2 unchanged to four decimal places, $F(2, 581) < 1$, *ns*, indicating that the intercept terms did not differ significantly across problem types.

Only two of the initial set of nested models led to significant decrements in fit, the models involving equality constraints on the memory search rate and encoding parameters. Given the lack of equality of memory search rate estimates, $\Delta R^2 = .004$, $F(4, 581) = 5.85$, $p < .0002$, two possibilities remained for determining a smaller, constrained set of parameters. When results of analyses reported in Tables 2, 3, and 4 are compared, it appears that either of two conditions may hold: Memory search rate may be slower in Type II problems, or memory search rate may be slower in the units column of two-column problems. The results supported the latter option because statistical models embodying the latter type of model led to higher levels of fit, on the basis of the same number of parameter estimates, than did models allowing a slower search rate for Type II problems.

Turning to speed of encoding digits, constraining encoding speed estimates to equality across problem types led to a significant decrease in fit, $\Delta R^2 = .003$, $F(2, 581) = 8.93$, $p < .0002$. The estimates from the baseline suggested that time for encoding each digit increased linearly across problem Types I, II, and III, whereas reencoding rate remained constant. Therefore, a basic encoding rate parameter, ENCODE_{int} for intercept of the ENCODE function, was specified with values equal to the number of digits encoded. Then, an encoding rate increase variable, ENCODE_{lin} for linear increase in encoding time, was specified that contained values of the number of digits encoded multiplied by 0, 1, and 2 for problem Types I, II, and III, respectively. The ENCODE_{lin} variable represented the linear increase in encoding time per digit across the three types of problem. Substitution of the ENCODE_{int} and ENCODE_{lin} structural variables in place of the three separate ENCODE variables resulted in a very nonsignificant decrease in fit from the baseline model, $F(1, 581) < 1$, *ns*. The need to specify a linear increase in encoding time across problem type stands in stark contrast to the constraints imposed on reencoding time. An equality constraint across problem type for the reencoding parameter was quite satisfactory, $\Delta R^2 = .0001$, $F(2, 581) < 1$, *ns*.

The final, highly constrained equation for Type I, II, and III problems is presented in the top half of Table 6. The constrained equation had a very high level of fit, $R^2 = .9048$, $F(6, 593) = 939.71$, $p < .0001$, and each of the individual parameter estimates was highly significant (all $ps < .0001$). Only six regression weights were estimated in the constrained equation because the regression weight for the reencoding structural variable was constrained to equal the weight for the basic encoding process, ENCODE_{int}, an equality constraint that was reasonable a priori and the relaxing of which led to a very nonsignificant increase in model fit, $F(1, 592) < 1$, *ns*. When compared with the fit of the baseline model, the constrained model, which had 12 fewer parameter estimates, explained only slightly less variance, $\Delta R^2 = .0027$, a decrease in R^2 that was statistically nonsignificant, $F(12, 581) = 1.39$, $p > .15$. Given both the notable economy of estimates in the constrained equation (6 vs. 18 estimates) and the quite negligible decrease in goodness of fit, the constrained equation provides a concise, powerful summary of RT to Type I, II, and III problems.

In addition to the overall fit of the constrained equation, a complementary way to demonstrate the adequacy of the regression model with contrained estimates is to compare the fit for each type of problem under constrained estimation with the fit for each type under unconstrained, best fit estimation. In the first row of Table 7, two measures of fit, R^2 and *SE*, are given for each type of problem under unconstrained estimation of parameters. In the second and third rows of Table 7, comparable measures of fit for each type of problem based on the regression model with contrained estimates are presented. Inspection of values in Table 7 indicated that the constrained model led to only slightly poorer fit for each type of problem than did an unconstrained model, with the largest drop in R^2 being .011 for Type II problems. These results provided further evidence of the utility and adequacy of the constrained model.

Turning to the regression weight estimates in the constrained equation, given in Table 6, the intercept of 757 ms represents a combined estimate of the following processes: encoding the problem type, decision time, and selection and execution of response. With regard to encoding of digits, encoding was estimated to require 53 ms per digit for Type I problems, increasing to $[53 + (1 \times 57)]$ ms, or 110 ms, per digit for Type II problems and to $[53 + (2 \times 57)]$ ms, or 167 ms, per digit for Type III problems. Because of the equality constraint, reencoding of incorrect sums took an additional 53 ms per digit. The regression weight for the truth variable revealed that responses to false problems took 131 ms longer than responses to true problems.

Table 6
Summary of Regression Analyses on Combined Sets of Addition Problems

Equation	R^2	*df*	*F*	RMS_e
Types I, II, and III				
RT = 757 + 53 (ENCODE$_{int}$) + 57 (ENCODE$_{lin}$) + 7.17 (UNITPROD) + 4.36 (UNITPROD1) + 205 (CARRY) + 7.17 (TENPROD) + 131 (TRUTH) + 53 (REENC) Partial *F*s: 26. 45, 403.07, 209.34, 32.75, 50.80, 209.34, 50.63, 26.45	.9048	6, 593	939.71	189.3
Types I, II, III and IV				
RT = 744 + 55 (ENCODE$_{int}$) + 56 (ENCODE$_{lin}$) + 7.10 (UNITPROD) + 4.91 (UNITPROD1) + 201 (CARRY) + 7.10 (TENPROD) + 114 (TRUTH) + 55 (REENC) + 228 (MIN IV) Partial *F*s: 26.99, 332.76, 198.03, 35.08, 38.94, 198.03, 33.61, 26.99, 850.19	.8873	7, 792	890.34	214.4

Note. Mean RT (Types I–III) = 1,697.67 ms. Mean RT (Types I–IV) = 1,780.1 ms. UNITPROD and TENPROD stand for the columnwise product of addends in the units and tens columns of all problems, respectively; UNITPROD1 for additional retrieval time as a function of columnwise product of addends in units column of Type II and III problems; ENCODE$_{int}$ for a basic digit-encoding process for all problems; ENCODE$_{lin}$ for the linear increase in encoding time per digit from Type I to Types II and IV (+1) and then to Type III (+2); CARRY for presence (1) or absence (0) of a carry from the units to the tens column; TRUTH for correct (0) or incorrect (1) stated sum; REENC for reencoding of digits in incorrect stated sums; and MIN IV for the smallest addend in Type IV problems.

The remaining two processes, rate of memory search for the correct sum and the carry operation, are central to models of mental addition. The latter, carry process required an estimated 205 ms of execution time. The former, memory search rate process proceeds at an estimated 7.17 ms per unit increase in the columnwise product structural variable for all columns of processing. Search rate in the units column of multicolumn problems required an estimated additional 4.36 ms per unit increase in the structural variable, for an estimated overall search rate of 11.53 ms per unit increase in the product structural variable in the units column of Type II and III problems.

Combined Analyses of All Four Types of Problems

Given the successful fitting of a constrained model for the first three types of problem, a simultaneous analysis of all 800 problems was undertaken. Here, the baseline model had 24

Table 7
Indices of Fit of Alternative Regression Models for All Types of Addition Problems

	Problem type							
	I		II		III		IV	
Regression equation	R^2	MSE_e	R^2	RMS_e	R^2	RMS_e	R^2	RMS_e
Least squares best fit	.702	127.7	.743	200.0	.841	224.3	.827	271.6
Constrained								
Types I–III								
$\Sigma e \neq 0$ per type	.685	131.2	.732	204.4	.833	229.9		
$\Sigma e = 0$ per type	.687	130.9	.733	203.8	.833	229.2		
Constrained								
Types I–IV								
$\Sigma e \neq 0$ per type	.697	128.6	.727	206.3	.833	229.7	.819	277.4
$\Sigma e = 0$ per type	.687	130.9	.733	203.8	.833	229.7	.821	275.9

Note. In regression analyses, errors of prediction (e) are defined so that the sum of errors, Σe, equals zero. But, when invoking constraints across types of problem, although the sum of errors across all problems must equal zero, the sum of errors per type may deviate from zero. In each set of constrained analyses, the first line (labeled "$\Sigma e \neq 0$ per type") reports statistics consistent with Formula 1 from Kvålseth (1985), in which the mean residual per type is not constrained to equal zero. The second line (labeled "$\Sigma e = 0$ per type") reports statistics consistent with Formula 4 from Kvålseth, in which the mean error per type is subtracted from each residual. The results from Formula 1 reflect equal or worse fit for the regression model relative to Formula 4 because deviation from zero of the mean residual contributes to error sums of squares.

parameter estimates and an overall level of fit of $R^2 = .8922$, $F(24, 775) = 267.22$, $p < .0001$, $SE = 212.0$. Procedures for nested model specification and testing identical to those outlined above were followed in the model fitting on the entire set of 800 problems.

The resulting highly constrained regression model is presented in the bottom half of Table 6. The fit of the constrained model was very high, $R^2 = .8873$, $F(7, 792) = 890.34$, $p < .0001$, and each of the individual parameters was highly significant (all *p*s < .0001). The difference between the baseline and constrained models was statistically significant, $\Delta R^2 = .0049$, $F(17, 775) = 2.07$, $.007 < p < .01$. However, the differences between the two models in levels of practical fit—for example, R^2 and *SE*—were very small and not of any practical import, and there is decidedly greater parsimony associated with the constrained model.

The differences in fit associated with the constrained model for all types of problem are also shown by comparing fit under unconstrained estimation (line 1) with fit under the constrained model (lines 4 and 5), indices presented in Table 7. Once again, the differences in fit of constrained and unconstrained models were not large, though the decrement in fit for Type II problems was slightly greater than that for the other three types of problem.

Finally, the regression estimates in the second equation in Table 6 were very similar to those given in the first constrained equation. Aside from the rather larger number of problems included, the major addition to the second equation was the inclusion of a structural variable, MIN IV, for the smallest addend in Type IV problems. The estimated regression weight for the MIN IV variable, $b = 228$ ms, is a value still well within the 200–250-ms range for incrementing the mental counter by using a process based on implicit speech (Landauer, 1962).

General Discussion

The two major goals of the present study were to provide further clarity regarding the search/compute process for mental addition and to specify and test a general model for mental addition across an array of simple and complex forms of addition problem. Our results provided strong support for the general model and indicated that subjects (a) tended to rely primarily on a memory network retrieval process, rather than analog or digital processes, to arrive at correct sums and (b) tended to utilize additional elementary information processes that require estimable, nontrivial amounts of execution time. In the following, the implications of each of these conclusions will be discussed in turn.

Memory Network Retrieval

With regard to the issue of the process by which subjects determine the correct sum of two digits, the results strongly supported the use of some form of memory network retrieval. When each of the four types of problems was analyzed separately, a model that contained the columnwise product of addends provided the best representation of RT data for each problem type. For two of the problem types (Types I and III), a model incorporating the product structural variable was clearly the best representation of data; for the remaining two types of problem (Types II and IV), a product model was somewhat better than models containing other sorts of structural variables for the search/compute stage. Thus, the product variable consistently led to better levels of fit than did any other structural variable. Thus, the superiority of the product structural variable in the present study replicated results reported by Miller et al. (1984) for simple addition and extended the applicability of the product variable to three types of more complex forms of addition. Importantly, the sum squared structural variable, proposed by Ashcraft and his associates, was a serious competitor to the product variable only for Type I, or simple addition, problems. On other, more complex types of problem, the strongest alternative models involved structural variables consistent with counting processes.

One empirical finding related to the product variable will require additional research to resolve. Judging from combined set analyses, the estimated search rate in the units column of multicolumn problems was rather slower than that in the tens column of such problems or the units column of Type I and IV problems. Only further research with yet more complex types of addition problem will determine whether the column search-rate effect is replicable and, if so, whether even further slowing of the search rate will occur in the units column of problems having addends with three or more digits.

The product structural variable is consistent with two relatively simple conceptual models of the network retrieval process. Under the tabular model, retrieval time is related to the area of a tablelike network that must be traversed in order to obtain the correct sum of two single-digit addends. Given the conception of a printed table of addition facts with nodes corresponding to the single-digit addends and with sums stored at the intersections of nodal values, the product structural variable presumes a table that is square, symmetric, and has equally spaced nodal values. The latter three properties are desirable for their simplicity and parsimony, yet Ashcraft and Battaglia (1978) were forced to drop the property of equal spacing of nodal values as inconsistent with the sum squared structural variable. As detailed earlier, the sum squared structural variable is also inconsistent with the remaining two properties, squareness and symmetry. The second conceptual model that may be conformable with the product structural variable is the nontabular "distribution of associations" model (Siegler & Shrager, 1984), if the value of the product structural variable is inversely related to the peakedness of the distributions of associations for addition problems. Both of the preceding conceptual models entail processes involving time-consuming search for and retrieval of addition facts from long-term memory storage, in contrast to models invoking analog or digital processes. In sum, considering both empirical and theoretical advantages, the product structural variable and its associated conceptual models appear preferable to alternatives proposed to date.

The results based on Type IV problems have implications for theories about the mental representations on which the memory retrieval process operates. For Type IV problems, the best fitting model implied that subjects first obtained a provisional sum of the two largest addends, via the very fast and efficient memory network retrieval process, and then incremented the smallest addend onto the provisional sum. This finding has at least three important implications. First, the existence of a three-dimensional memory network, a logical extension of previous models presuming a two-dimensional network, does not appear to be a reasonable hypothesis because the fit of the best three-sum model was far worse than the most acceptable model. Second, the failure of subjects to search for the sum of the provisional sum and the smallest addend provides unique support for the position that the memory network is bounded by nodal values from 0 through 9 and does not include larger nodal values. Third, the similarity of the estimated temporal operating characteristics of the process for incrementing onto the provisional sum with the findings by Landauer (1962) is consistent with the hypothesis that the incrementing in Type IV problems was effected by use of implicit or subvocal speech processes.

Response Constant

A parameter common to all equations representing RT to addition problems was the *intercept*, or response constant. The intercept represents a conglomerate estimate of the time taken to execute all processes not explicitly modeled by structural variables in the equation and includes such processes as encoding problem type, decision time, and selection and execution of a response. The invariance of the intercept parameter estimate across all problem types is of signal importance because this stability implies that all sources of variance among the four types of problem are captured adequately by the structural variables included in the equations. Thus, the processes encompassed by the intercept require a constant amount of time regardless of problem type. Further, the large, systematic differences in RT across problem type are accounted for by structural variables currently included in the equations, and no additional structural variables appear to be required to represent more accurately these differences.

Additional Elementary Components

The importance of several additional elementary information processes for mental addition was supported by the reported results. The importance of the *carry* operation replicates results in the studies by Ashcraft and Stazyk (1981) and Hamann and Ashcraft (1985). The results with regard to the carry operation in the present study are buttressed by the placement of the carry operation within a general model for mental addition. Given the inclusion of structural variables for other processes underlying mental addition, the estimate of 210–220 ms for execution of a carry, based on combined set analyses, is likely a more accurate, less biased estimate of the temporal duration of a carry operation than are estimates from previous studies.

A second elementary component isolated in analyses was the *encoding* of digits in an addition problem. Following previous studies assessing encoding of terms in analogy problems (e.g., Sternberg, 1977) or of access to name codes of letters (e.g., Posner et al., 1969), we assumed that encoding a digit would include retrieval of the cardinal value of the digit and may also include activation of the nodal value corresponding to the digit. The basic digit encoding parameter estimated from RT to simple addition problems was in the same range as that for access to name codes of letters (Posner et al., 1969; Hunt et al., 1975), 50–80 ms per digit. This is quite interesting because letters and digits are the two sets of highly overlearned symbols for persons in our culture.

Analyses of the combined sets of problems indicated that encoding time per digit increased from Type I to Type II and IV problems and then increased again for Type III problems. One explanation for this finding is that encoding time is lengthened as a function of the number of digits in the addends of a problem and, thus, to the visual "clutter" from which the addend digits must be extracted. An alternative explanation is that different types of processes are causing the increased digit encoding times in Type II and IV problems. Thus, digit encoding time may be increased when having to extract the units digit from a two-digit addend; this would lead to a linear increase from Type I problems to Type II problems and then again to Type III problems because the number of two-digit addends increases linearly across these problem types. The increased time for digit encoding in Type IV problems might be due to a different cause: the need to determine which two of the addends were larger while keeping track mentally of the ordinal position and value of the smaller addend. The increased load on short-term memory processes may then lead to slower encoding of digits in the Type IV problems. Further research should be pursued to offer tests of these alternatives because the unexplained nature of this effect represents a potential challenge to the validity of our theoretical and empirical analyses.

The third, and last, elementary process identified was the *reencoding* of incorrect digits in stated sums. As noted earlier, this effect may, in fact, be due to interference between the true and stated sums (cf. Campbell, 1987b). Presuming that this effect is related to encoding processes, the estimate of temporal duration for this process appeared reasonable and interpretable; only further research will allow a determination of the replicability of the effect and its true basis.

Finally, we trust that the general model, presented in Figure 3, will be an impetus for further research on mental addition, requiring modification as a broader scope of effects on mental addition is studied. Topics that merit investigation include (a) the manner in which the componential model could be modified to allow the representation of the use of rule-based, procedural processes to solve addition problems, (b) the basis of the increase in encoding time per digit across problem types, (c) whether the so-called reencoding of incorrect stated sums is the result of interference or some other effects, (d) how to incorporate and test production-task data within the context of the general verification-task model presented in Figure 3, and (e) how to incorporate important, recently stated concerns by Siegler (1987) regarding effects of inter- and intrapersonal variation in strategy choice across problems. More broadly, the general model provides a framework for the investigation of other types of simple numerical operations, such as multiplication and subtraction. We hope, therefore, that our general model will be one more step in the direction of the development of a comprehensive model for numerical facility and, more generally, mathematical ability.

References

Ashcraft, M. H. (1982). The development of mental arithmetic: A chronometric approach. *Developmental Review, 2,* 213–236.

Ashcraft, M. H., & Battaglia, J. (1978). Cognitive arithmetic: Evidence for retrieval and decision processes in mental addition. *Journal of Experimental Psychology: Human Learning and Memory, 4,* 527–538.

Ashcraft, M. H., & Fierman, B. A. (1982). Mental addition in third, fourth, and sixth graders. *Journal of Experimental Child Psychology, 33,* 216–234.

Ashcraft, M. H., Fierman, B. A., & Bartolotta, R. (1984). The production and verification tasks in mental addition: An empirical comparison. *Developmental Review, 4,* 157–170.

Ashcraft, M. H., & Stazyk, E. H. (1981). Mental addition: A test of three verification models. *Memory & Cognition, 9,* 185–196.

Baroody, A. J. (1983). The development of procedural knowledge: An alternative explanation for chronometric trends of mental arithmetic. *Developmental Review, 3,* 225–230.

Baroody, A. J. (1985). Mastery of basic number combinations: Internalization of relationships or facts? *Journal for Research in Mathematics Education, 16,* 83–98.

Baroody, A. J. (1987). *Children's mathematical thinking: A developmental framework for preschool, primary, and special education teachers.* New York: Teachers College Press.

Baroody, A. J., & Ginsburg, H. P. (1986). The relationship between initial meaningful and mechanical knowledge of arithmetic. In J. Hiebert (Ed.), *Conceptual and procedural knowledge: The case of mathematics* (pp. 75–112). Hillsdale, NJ: Erlbaum.

Browne, C. E. (1906). The psychology of the simple arithmetical processes: A study of certain habits of attention and association. *American Journal of Psychology, 17,* 1–37.

Brownell, W. A. (1935). Psycological considerations in the learning and the teaching of arithmetic. In *The teaching of arithmetic* (10th Yearbook of the National Council of Teachers of Mathematics, pp. 1–31). New York: Columbia University, Teachers College, Bureau of Publications.

Campbell, J. I. D. (1987a). Network interference and mental multiplication. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 13,* 109–123.

Campbell, J. I. D. (1987b). Production, verification, and priming of multiplication facts. *Memory & Cognition, 15,* 349–364.

Campbell, J. I. D., & Graham, D. J. (1985). Mental multiplication skill: Structure, process, and acquisition. *Canadian Journal of Psychology, 39,* 338–366.

Cohen, J., & Cohen, P. (1983). *Applied multiple regression/correlation analysis for the behavioral sciences* (2nd ed.). Hillsdale, NJ: Erlbaum.

Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological Review, 82,* 407–428.

Cooper, L. A., & Shephard, R. N. (1973). Chronometric studies of the rotation of mental images. In Chase, W. G. (Ed.), *Visual information processing* (pp. 75–176). New York: Academic.

Groen, G. J., & Parkman, J. M. (1972). A chronometric analysis of simple addition. *Psychological Review, 79,* 329–343.

Groen, G. J., & Resnick, L. B. (1977). Can preschool children invent addition algorithms? *Journal of Educational Psychology, 69,* 645–652.

Hamann, M. S., & Ashcraft, M. H. (1985). Simple and complex mental addition across development. *Journal of Experimental Child Psychology, 40,* 49–72.

Hamann, M. S., & Ashcraft, M. H. (1986). Textbook presentations of the basic addition facts. *Cognition and Instruction, 3,* 173–192.

Hunt, E. (1978). Mechanics of verbal ability. *Psychological Review, 85,* 109–130.

Hunt, E., Lunneborg, C., & Lewis, J. (1975). What does it mean to be high verbal? *Cognitive Psychology, 7,* 194–227.

Ilg, F., & Ames, L. B. (1951). Developmental trends in arithmetic. *Journal of Genetic Psychology, 79,* 3–28.

Kvålseth, T. O. (1985). Cautionary note about R^2. *American Statistician, 39,* 279–285.

Landauer, T. K. (1962). Rate of implicit speech. *Perceptual and Motor Skills, 15,* 646.

LeFevre, J., Bisanz, J., & Mrkonjic, L. (1988). Cognitive arithmetic: Evidence for obligatory activation of arithmetic facts. *Memory & Cognition, 16,* 45–53.

Miller, K., Perlmutter, M., & Keating, D. (1984). Cognitive arithmetic: Comparison of operations. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 10,* 46–60.

Moyer, R. S., & Landauer, T. K. (1967). Time required for judgments of numerical inequality. *Nature, 215,* 1519–1520.

Parkman, J. M., & Groen, G. J. (1971). Temporal aspects of simple addition and comparison. *Journal of Experimental Psychology, 89,* 335–342.

Pellegrino, J. W., & Kail, R. (1982). Process analyses of spatial aptitude. In R. J. Sternberg (Ed.), *Advances in the psychology of human intelligence* (Vol. 1, pp. 311–365). Hillsdale, NJ: Lawrence Erlbaum.

Poltrock, S. E., & Schwartz, D. R. (1984). Comparative judgments of multidigit numbers. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 10,* 32–45.

Posner, M. I., Boies, S. J., Eichelman, W. H., & Taylor, R. L. (1969). Retention of visual and name codes of single letters. *Journal of Experimental Psychology* (Monograph), *79,* 1–16.

Restle, F. (1970). Speed of adding and comparing numbers. *Journal of Experimental Psychology, 85,* 274–278.

Rindskopf, D. (1984). Linear equality restrictions in regression and loglinear models. *Psychological Bulletin, 96,* 597–603.

Siegler, R. S. (1987). The perils of averaging over strategies: An example from children's addition. *Journal of Experimental Psychology: General, 116,* 250–264.

Siegler, R. S., & Shrager, J. (1984). Strategy choices in addition and subtraction: How do children know what to do? In C. Sophian (Ed.), *Origins of cognitive skills* (pp. 229–294). Hillsdale, NJ: Erlbaum.

Siegler, R. S., & Taraban, R. (1986). Conditions of applicability of a strategy choice model. *Cognitive Development, 1,* 31–51.

Stazyk, E. H., Ashcraft, M. H., & Hamann, M. S. (1982). A network approach to mental multiplication. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 8,* 320–335.

Sternberg, R. J. (1977). *Intelligence, information processing, and analogical reasoning: The componential analysis of human abilities.* Hillsdale, NJ: Erlbaum.

Sternberg, R. J. & Gardner, M. K. (1983). Unities in inductive reasoning. *Journal of Experimental Psychology: General, 112,* 80–116.

Sternberg, S. (1966). High speed scanning in human memory. *Science, 153,* 652–654.

Sternberg, S. (1969). The discovery of processing stages: Extension of Donders' method. *Acta Psychologica, 30,* 276–315.

Thorndike, E. L. (1922). *The psychology of arithmetic.* New York: Macmillan.

Wheeler, L. R. (1939). A comparative study of the difficulty of the 100 additional combinations. *Journal of Genetic Psychology, 54,* 295–312.

Wike, E. L. (1971). *Data analysis.* Chicago: Aldine-Atherton.

Received July 13, 1987
Revision received December 28, 1988
Accepted January 11, 1989 ■