

# **Structure of Representations of Commutative Problems in Mental Arithmetic: An Examination by Priming Paradigm**

Hideaki Shimada

(Doctoral Program in Psychology, University of Tsukuba, Japan)

Yoshitaka Iwata

(College of Human Science, University of Tsukuba, Japan)

# Purpose

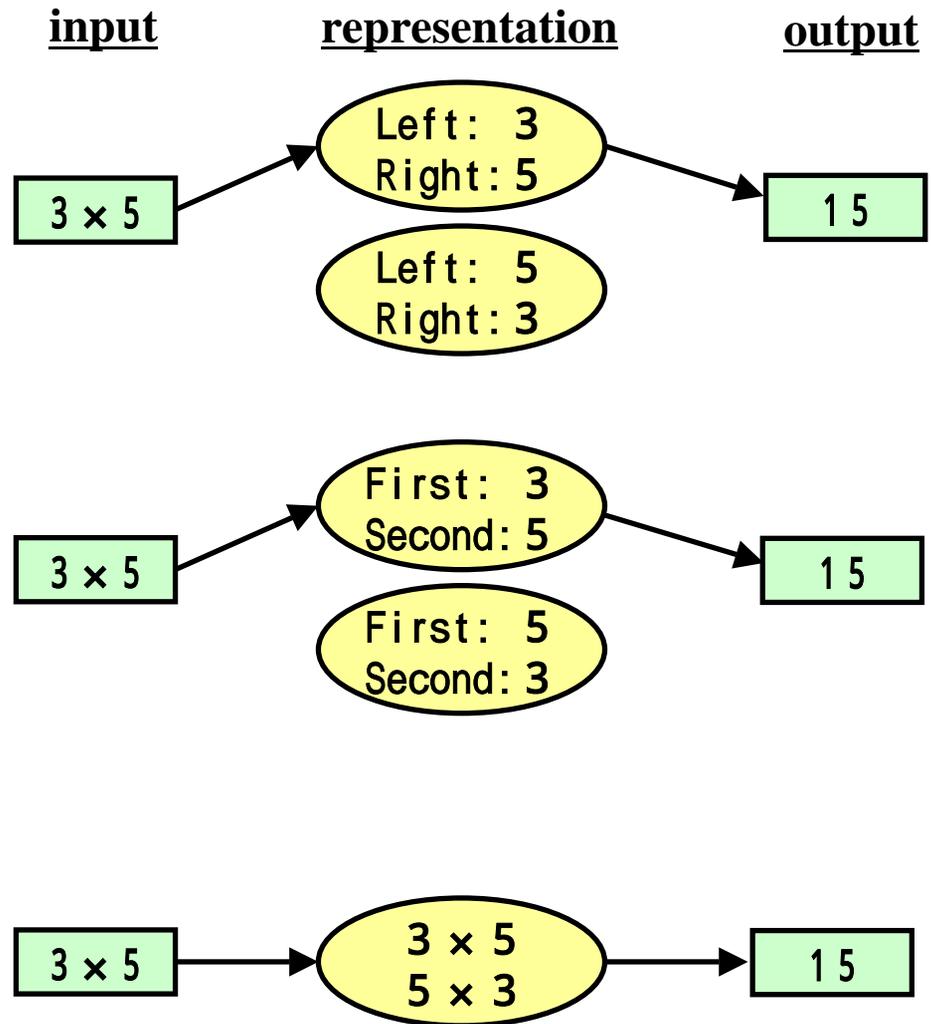
- Purpose of this study was to examine structure of representations of commutative problems in mental arithmetic.
  - Commutative means that  $a*b=b*a$ , where “\*” is a operation ,and “a” and “b” are operand, for example  $3 \times 5=5 \times 3$  .
  - Some studies indicate that representations of arithmetic fact of commutative problems are *separate* (e.g. Experimental studies: Arbuthnott & Campbell, 1996; Phenix & Campbell, 2000; Computational models: Campbell, 1995: Shimada, submitted - The idea of this model was in Appendix of this poster).
  - What kind of information differentiates commutative problems, or  $3 \times 5$  and  $5 \times 3$ ? Following two information: **position and order are suggested, and we examined which information differentiate commutative problems.**
  - In addition to discrete models, identical element model (e.g. Rickard & Bourne, 1994), in which commutative problems are represented *identically*, is also examined with two discrete models.
- 
- Is very similar pair of items like  $3 \times 5$  and  $5 \times 3$  represented separately in human memory as a result of life-span learning? This study contributes to not only specific domain, but also general human memory research.

# Three models on commutative problems

(1) *Discrete elements model on position information*:  $3 \times 5$  accesses “Left 3, Right 5” representation, and 15 is output. **Position is critical** information in this model.

(2) *Discrete elements model on order information*:  $3 \times 5$  accesses “First:3, Second:5” representation, and 15 is output. **Order is critical** information in this model. It is assumed that  $3 \times 5$  is processed from 3 to 5 temporally.

(3) *Identical element model*:  $3 \times 5$  accesses common representation with  $5 \times 3$ , and 15 is output. **Neither position nor order is critical** information in this model.



**Figure 1:** Models of arithmetic fact retrieval for commutative problems.

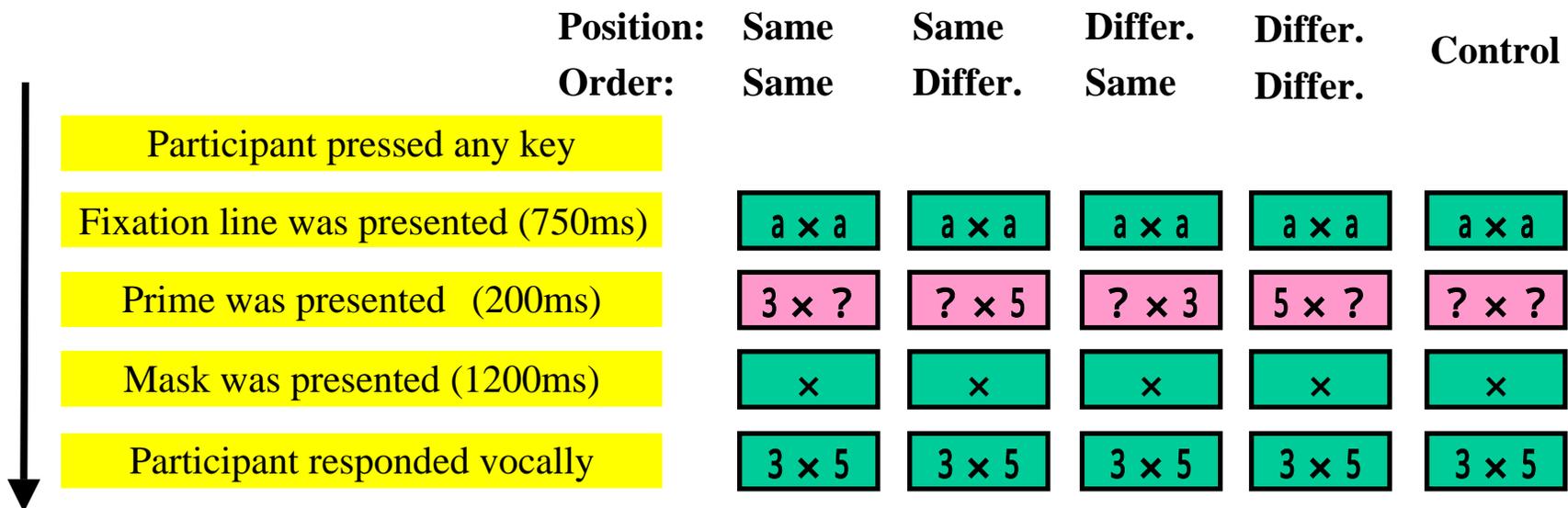
# Method

Participants 24 adults participated in this study.

Materials 56 multiplication problems, in which tie-problems like  $3 \times 3$  were excepted, were used. We used a priming task in which **one operand within current problem was presented prior to the problem.**

Design Two variables were manipulated. One was *order* (Same/Different), and the other was *position* (Same/Different), and these conditions were manipulated within a participant. In addition to these conditions, control condition was prepared.

Procedure Each participant solved 280 multiplication problems which consisted of five conditions of 56 problems. The time course of one trial was following form (Figure2). Participants were required to solve problems as fast and accurately as possible. RTs from presentation of problem to response and participant's answers were recorded for each trial.

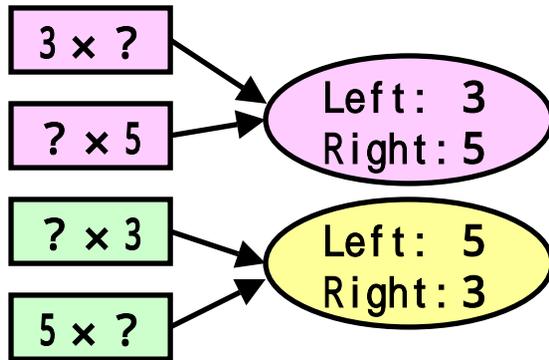


**Figure 2:** Procedure and conditions.

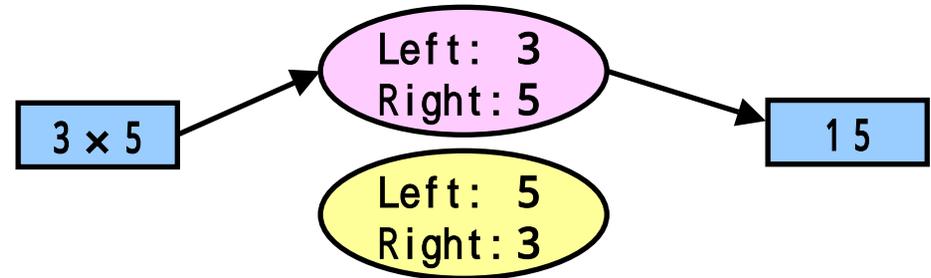
# Predictions based on three models

(1) **Discrete elements model on position information:**  $3 \times 5$  is solved faster in primed by  $3 \times ?$  and  $? \times 5$  than  $? \times 3$  and  $5 \times ?$  because of position compatibility.

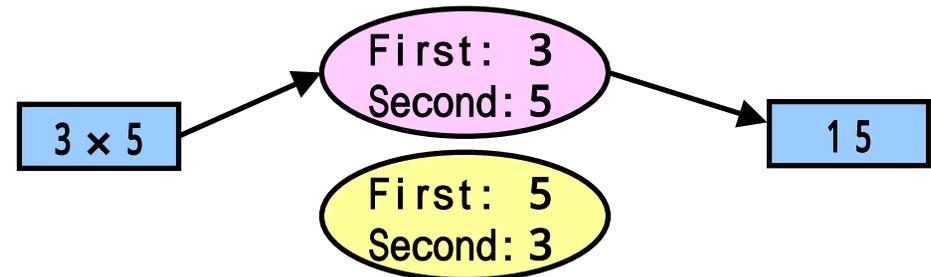
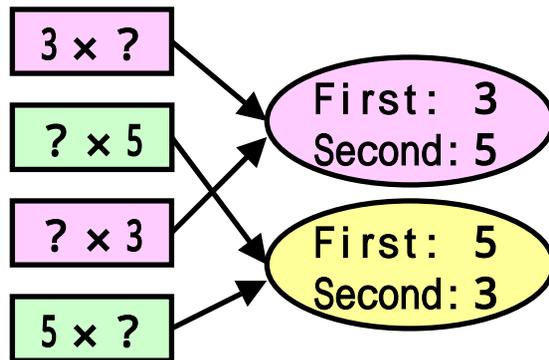
## Prime phase



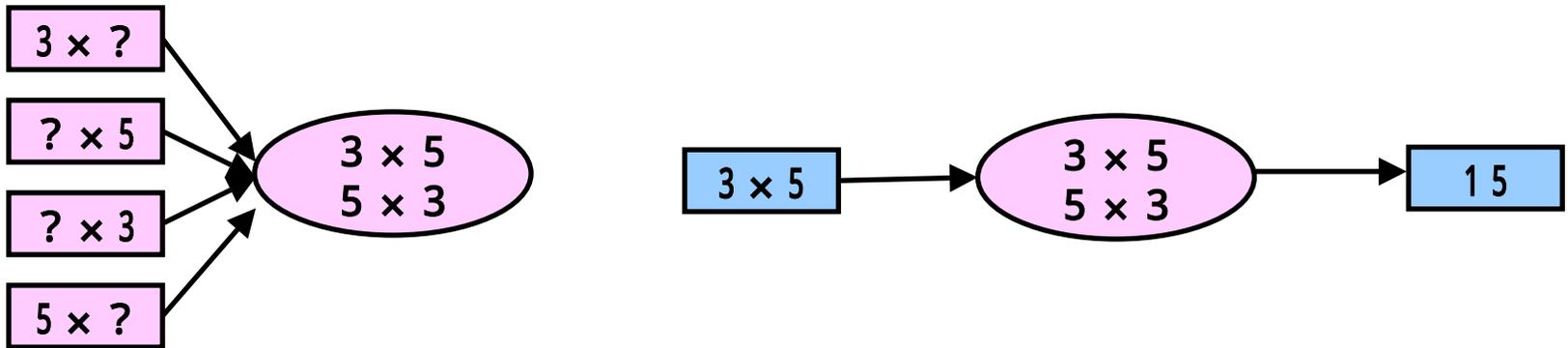
## Probe phase



(2) **Discrete elements model on order information:**  $3 \times 5$  is solved faster in primed by  $3 \times ?$  and  $? \times 3$  than  $? \times 5$  and  $5 \times ?$  because of order compatibility.



(3) **Identical element model**:  $3 \times 5$  is solved as fast as for any prime conditions because all primes facilitate identical representation of “ $3 \times 5$ ” and “ $5 \times 3$ ”.



**Table 1:** Summary of predictions.

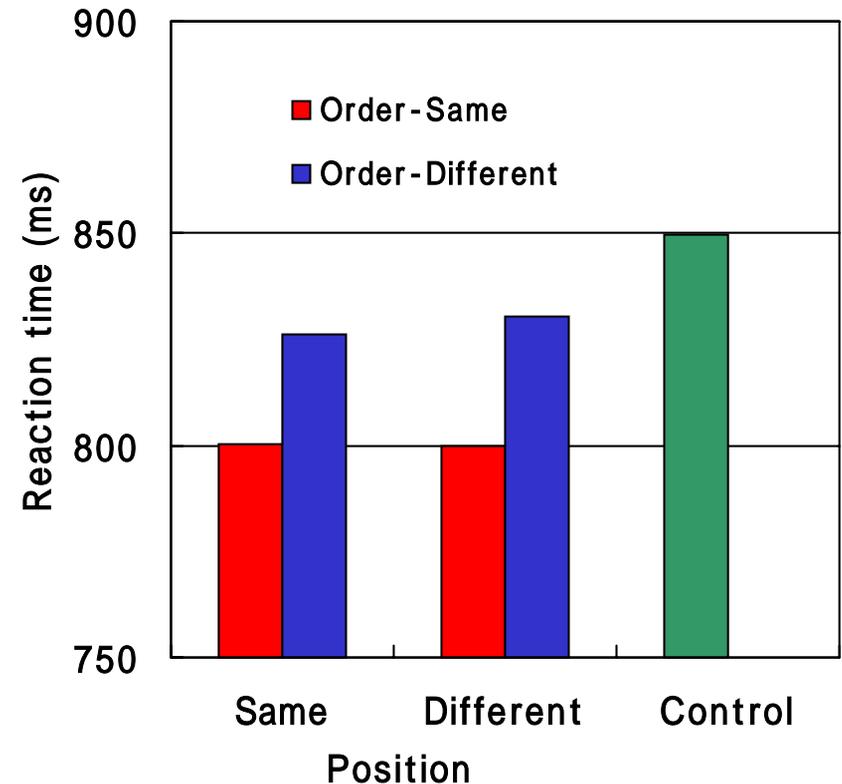
|   | Effect of Position | Effect of Order |
|---|--------------------|-----------------|
| Discrete model on <i>position</i> information |                    | ×               |
| Discrete model on <i>order</i> information    | ×                  |                 |
| Identical elements                            | ×                  | ×               |

## Result and Conclusion

■RTs in order-same conditions were significantly faster than order-different conditions ( $p < .01$ ), and there was no effect of position as a result of  $2 \times 2$  ANOVA (Order  $\times$  Position) for four experimental conditions.

➤ **This result suggests that discrete model on *order* information is valid for representations of commutative problems in arithmetic fact retrieval.**

■RTs in three experimental conditions except position-different/order-different condition were significantly faster than control ( $p < .05$ ), but this condition was marginally significant ( $p < .10$ ) as a result of planned comparison by Dunnett's t-test.



**Figure 3:** Reaction times for five conditions.

## Appendix

■Outline of two factor theory in arithmetic fact retrieval (Shimada, submitted) is following.

■In this model, RT for each of addition and multiplication problem in production ( $3+5=?$ ) and verification ( $3+5=9$ , true or false?) is explained by following two factor.

■ Association strength (AS): The strength of association between problem and answer. The stronger this value is, the faster RT is for both production and verification task.

■ Interference strength (IS): The strength of association between problem and some integers *except answer of the problem*. In production task, the stronger this value is, the slower RT is, but in verification task, this value does not related to RT.

■ Detail of this model is following below.

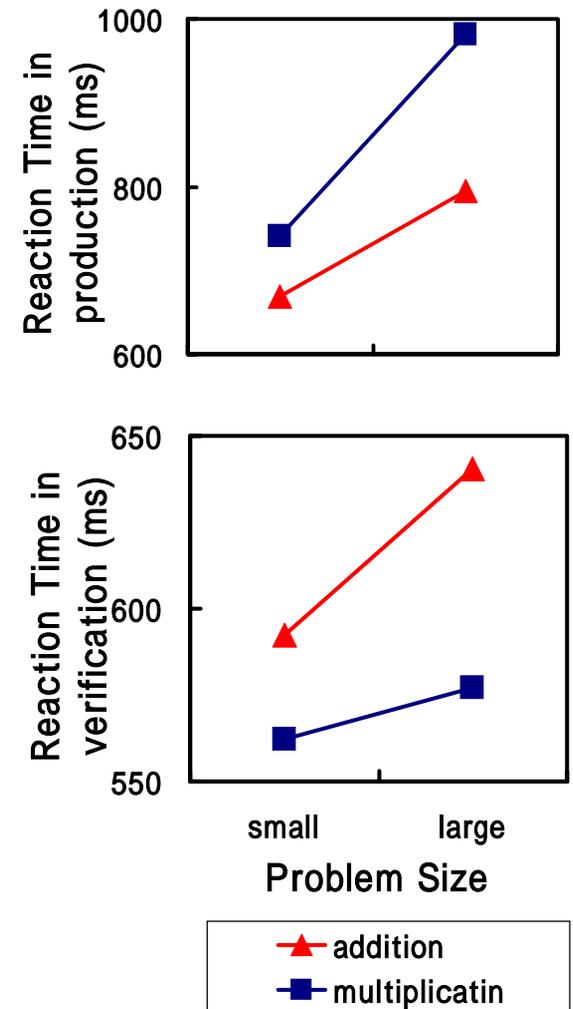
■ AS for each problem is calculated as a function of problem-based association, operand-based association, answer-fan effect, and magnitude similarity, and equation for AS is

$$\frac{dAss(a,b,r,c;t)}{dt} = a_p(1 - Ass) + a_o(1 - Ass) + a_f(0 - Ass) + a_s(0 - Ass)$$

where  $Ass(a,b,r,c;t)$  is value of AS,  $a$  and  $b$  are operands,  $r$  is operation,  $c$  is answer, and  $t$  is time (or frequency of learning for each problem). For example,  $Ass(3,5, \times, 15)$  is AS for  $3 \times 5=15$ . Four parameters (indexed by  $a_x$ ) were estimated utilizing some experiments. It is assumed that adults have sufficiently learned arithmetic facts, or  $t$  , and  $Ass(a,b,r,c)$ s are given.

■ IS for each problem is calculated using corresponding ASs and magnitude similarity. The most importance is the fact that IS in multiplication tend to be stronger than addition because multiplication facts have more numbers of interference fact. For example, “neighbors” of  $3 \times 5 = 15$  are  $2 \times 5$ ,  $4 \times 5$ ,  $3 \times 4$ ,  $3 \times 6$ , and “neighbor answers” are 10, 20, 12, 18 whereas neighbors of  $3+5$  are  $2+5$ ,  $4+5$ ,  $3+4$ ,  $3+6$ , but neighbor answers are only 7 and 9, thus  $3 \times 5$  have more IS than  $3+5$ . In particular, IS is strong as  $c$  increases, so IS is stronger for large multiplication fact.

- The most important finding is following below.
  - RTs of production and verification task were set out, findings are (figure 4):
    - In production task, problem size effect for addition is *smaller* than for multiplication.
    - In verification task, problem size effect for addition is *larger* than for multiplication.
  - Two factor theory explained this reversal. The larger problem size is, the stronger AS is in addition, but not in multiplication. So problem size effect for addition is large in verification because this task depend on only AS. But in production task, IS affects RTs, and IS is larger in addition than in multiplication, particularly in large-size multiplication. So this reversal is expected.
  - This reversal is not explained by other computational models (e.g. network retrieval model, Ashcraft, 1987).



**Figure 4:** RTs in production (top panel) and verification (bottom panel) as a function of problem size and operation.