Automatic Number Processing is Both Memory-Based and Algorithm-Based

Filip Van Opstal (filip.vanopstal@ugent.be)
Tom Verguts (tom.verguts@ugent.be)
Wim Fias (wim.fias@ugent.be)

Department of Psychology, Ghent University
9000 Ghent, Belgium

Keywords
Automaticity, number processing
Memory-based versus algorithm-based processing

While the automatic processing of numbers is widely accepted, the underlying mechanisms are still under discussion. The memory-based view states that automatic processing involves the retrieval of a single piece of information from memory (Logan, 1988). On the other hand, the algorithm-based view states that unconscious algorithms can be automatically executed (e.g. in a number comparison experiment; Dehaene et al., 1998). In a series of three experiments, it is shown that each view separately is insufficient to explain the obtained data pattern. We suggest that automatic number processing involves both memory-based and algorithm-based components.

Experiment 1
Subjects had to classify numbers as being smaller or larger than a fixed standard number. A number prime was shown subliminally before each target. In the first phase, a short range was used. Size of the target numbers was varied between subjects (Low group: targets 1, 2, 4, 5; primes 4 or neutral, standard 3; High group: targets 5, 6, 8, 9; primes 6 or neutral, standard 7). In the second phase, the same long range was used in both groups (targets 2, 3, 7, 8; same primes, standard 5).

Results
Results showed a significant interaction between prime type and range (see figure). For example, for target number 2 in the Low group (left panel), the prime 4 was incongruent in the range 1-5 (with standard 3), but congruent in the range 2-8 (with standard 5). This shows that even when presented subliminally, the primes are compared to the standard, as predicted by the algorithm-based view on automaticity.

Experiment 2
In this experiment, the standard was not fixed for a complete block of trials, but rather changed every four trials. The prime was always either 5 or a neutral prime.

Results
When prime and target were incongruent (e.g., prime 2, target 4, standard 3), RTs were slower than for the same target-standard pair with a neutral prime. However, this was only observed when the standard was smaller than the prime (5); when the standard was larger than 5, no congruency effect was observed. This pattern of data can be explained by a combined memory-based and algorithm-based mechanism. Suppose that number 5 has a label “large” attached to it. This will work against the (algorithm-based) congruency effect for standards larger than 5, but it will work with the congruency effect for standards smaller than 5. An additional prediction of this account is tested in the next Experiment.

Experiment 3
The design of Experiment 3 was similar to that of Experiment 2, except that the prime was always either 3 or neutral. If there are both algorithm- and memory-based components of automaticity, and if 3 has a label “small” attached to it, the congruency effect should now be observed only for standards larger than the prime (3).

Results
A significant congruency effect was found when the standard was larger than the prime, but not when it was smaller.

Discussion
Both memory-based and algorithm-based accounts are needed to explain the results on automatic number processing.

References