The study of Drosophila faciliates the solution of a fundamental computing problem (notice the reference to the FLY – as if Drosophila were the one and only)

## Computer Scientists Learn from Flies

Designing distributed networks of computers that work together to solve a problem without any single processor receiving all of the inputs or observing all of the outputs represents a difficult problem. **Afek *et al.*** (p. 183) noted the similarity of this problem to the process of patterning of sensory bristles on the fruit fly. By studying the developmental process in the fly and modeling its mechanism, the authors derived an algorithm that works efficiently to solve the computer science problem of identifying what is known as a "maximal independent set" that may prove useful in the design of wireless networks.

---

# A Biological Solution to a Fundamental Distributed Computing Problem

Yehuda Afek,[1]* Noga Alon,[1,2]* Omer Barad,[3]* Eran Hornstein,[3] Naama Barkai,[3]† Ziv Bar-Joseph[4]†

Computational and biological systems are often distributed so that processors (cells) jointly solve a task, without any of them receiving all inputs or observing all outputs. Maximal independent set (MIS) selection is a fundamental distributed computing procedure that seeks to elect a set of local leaders in a network. A variant of this problem is solved during the development of the fly's nervous system, when sensory organ precursor (SOP) cells are chosen. By studying SOP selection, we derived a fast algorithm for MIS selection that combines two attractive features. First, processors do not need to know their degree; second, it has an optimal message complexity while only using one-bit messages. Our findings suggest that simple and efficient algorithms can be developed on the basis of biologically derived insights.

Computational and mathematical methods are extensively used to analyze and model biological systems (*1*–*3*). We provide an example of the reverse of this strategy, in which a biological process is used to derive a solution to a long-standing computational problem.

nodes change their probability of being elected based on the number of active neighbors they have (nodes that are not yet connected to nodes in *A*), and they require processors to send messages the sizes of which are a function of the number of nodes in the network. Recent methods were