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.

In distributed computing, a large number of processors jointly and distributively solve a task, without any of the processors getting all of the inputs or observing all of the outputs (4). All large-scale computing efforts, from web search to airplane control systems, use distributed computing algorithms to reach agreement, overcome failures, and decrease response times. Biological processes are also distributed. Parallel pathways are used to transform environmental signals to gene expression programs, and several tasks are jointly performed by independent cells without clear central control.

A long-standing distributed computing problem is that of electing a set of local leaders [the maximal independent set (MIS)] in a network of connected processors (4). The MIS is used to determine a backbone for wireless networks, for routing, and in several other network protocols (5). Formally, a MIS is defined as a set of processors (nodes) A so that every node in the network is either in A or directly connected to a node in A, and no two nodes in A are connected (Fig. 1A). Distributively electing a MIS has been considered a challenging problem for three decades (6). In particular, when all nodes are initially identical constructing a MIS by using deterministic algorithms is impossible (7), necessitating probabilistic approaches. Luby (8) and Alon et al. (9) presented fast probabilistic algorithms for electing a MIS. In these algorithms,

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 proposed that partially remove either of these assumptions (10, 11), but to date, no method has been able to efficiently reduce message complexity without assuming knowledge of the number of neighbors. These are important requirements for deployment of large, ad hoc sensor networks.

The selection of neural precursors during the development of the nervous system resembles the MIS election problem. The precursors of the fly's sensory bristles [sensory organ precursors (SOPs)] are selected during larvae and pupae development from clusters of equivalent

cells. The selection of the small bristles precursors (microchaetes) (Fig. 1B) is initiated 8 to 10 hours after pupae formation, when several elongated clusters of proneural cells, containing between 20 and 30 cells each, appear at specific positions in the imaginal discs, which will later become the fly's wings and notum. During the next 3 hours, SOPs begin to appear within these clusters. A cell that is selected as a SOP inhibits its neighbors by expressing high levels of the membrane-bound protein Delta, which binds and activates the transmembrane receptor protein Notch on adjacent cells (12). This lateral-inhibition process is highly accurate (13), resulting in a regularly spaced pattern in which each cell is either selected as SOP or is inhibited by a neighboring SOP (Fig. 1C). Thus, as in the MIS problem every proneural cluster must elect a set of SOPs (A) so that every cell in the cluster is either in A or connected to a SOP in A, and no two SOPs in A are adjacent.

Extensive studies and mathematical modeling were used to define the molecular components mediating SOP selection and the mechanism underlying selection. These studies suggest several similarities between the mechanism underlying SOP selection and current algorithms for MIS election (14). First, the selection of a particular cell as a SOP is a random event governed by an underlying stochastic process (15, 16). Second, similar to computational requirements SOP selection is probably constrained in time because the default of all cluster cells is to become SOPs unless they are inhibited (17). Lastly, in computational algorithms (8, 9) processors send messages only when they propose their
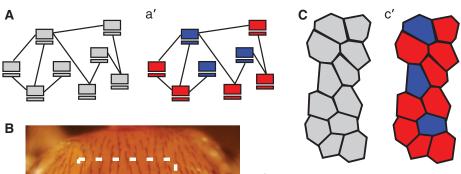

Fig. 1. Computational and biological overview. (**A**) Illustration of a MIS. Edges represent communication channels. (Left) Processors in a network are initially identical. (Right) Following a MIS selection algorithm, a set of local leaders (blue computers) is elected so that each computer is either a local leader or connected to a local leader. No two local leaders can be neighbors in the network. (**B**) The notum of an adult fly, presenting the typical pattern of small and large bristles (microchaetes and macrochaetes, respectively). Microchaetes are surrounded by a dashed line. (**C**) Illustration of SOPs in flies. (Left) Cells in a cluster are initially equivalent. (Right) Following a SOP selection process, selected SOPs (blue cells) inhibit their physical neighbors (red cells), and so for the cluster depicted in this figure, no more SOPs can be selected.

[1]Blavatnik School of Computer Science and Sackler School of Mathematics, Tel Aviv University, Tel Aviv 69978, Israel. [2]Institute for Advanced Study, Princeton, NJ 08544, USA. [3]Department of Molecular Genetics, Weizmann Institute of Science, Rehovot 76100, Israel. [4]School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA.

*These authors contributed equally to this work.
†To whom correspondence should be addressed. E-mail: naama.barkai@weizmann.ac.il (N.B.); zivbj@cs.cmu.edu (Z.B.-J.).
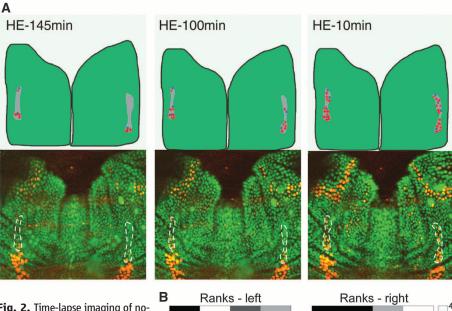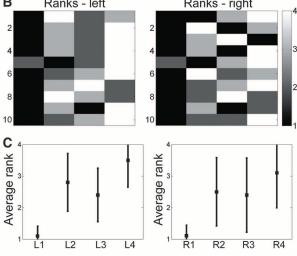
Fig. 2. Time-lapse imaging of notum microchaetes SOPs selection in a live pupa. (**A**) (Bottom) Time-lapse imaging of notum microchaetes SOPs selection in a live pupa of *hsflp; mα-dsRED;FRT80B, ubi-NLS-GFP* strain after the selection of the microchaetes SOPs at the fifth row of the left and right clusters (area surrounded by a dashed white curve). (Top) Annotated image highlighting the proneural, inhibited, and selected cells in the fifth row of the bottom panels. Proneural clusters are marked with gray, SOPs with blue, and non-SOPs with red. SOPs are identified by the up-regulation of *mα-dsRED* in adjacent cells. We followed the selection process from 145 min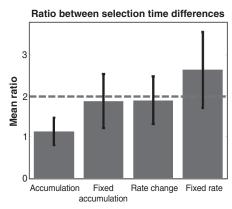 before head eversion (HE) to HE, corresponding to ~9.5 to 12 hours after pupa formation. (**B** and **C**) Statistics of SOP selection order from time-lapse imaging of ten pupae. The *y* axis represents the movie number. The *x* axis corresponds to the four SOPs selected in each row on the left and right sides (L1 to L4 and R1 to R4) ordered from bottom to top. Color in each (*x*, *y*) coordinate represents the order (1 to 4) in which this SOP was selected (see color legend on the right) (C) Average and SD of SOP selection order for L1 to L4 and R1 to R4; *x* axis is the same as in (B).

Fig. 3. Comparison of experimental and simulated results. We computed the average ratio for the difference between the selection times of SOPs in the movies we analyzed. The dashed line represents the average observed in our experiments (1.98). We simulated four stochastic models. In the accumulation model, cells accumulate random amounts of Delta at each step until reaching a threshold. The fixed accumulation and rate change models are described in the paper. In the fixed rate model, cells use the same burst distribution throughout the process (*14*). Values are based on 20,000 runs for each model. Error bars represent SD.

candidacy to become leaders, thus reducing communication complexity. Mathematical modeling by us and others suggests that this might also be the case during SOP selection: Before being selected, the ability of Delta to activate Notch on adjacent cells is inhibited by their interaction within the same cell, enabling communication between cells only after selection (*18–20*).

Although similar, the biological solution differs from computational algorithms in at least two aspects. First, SOP selection is probably performed without relying on knowledge of the number of neighbors that are not yet selected. Second, mathematical analysis demonstrated that SOP selection requires nonlinear inhibition that in effect reduces communication

to the simplest set of possible messages (binary) (*21, 22*).

These last two aspects of the biological process are attractive because they could greatly simplify applications of MIS selection. We thus examined SOP selection more closely in order to determine whether better understanding of this process can lead to an efficient algorithm for MIS selection. To verify the stochastic nature of this process, we first monitored the in vivo selection using a fluorescence reporter for Notch activity (*17*). We focused on the selection of SOPs in two symmetrical rows of pro-neural clusters [the fifth rows of notum microchaetes (Fig. 2A)], which occur before head eversion. These clusters consist of 10 rows with two cells each, giving rise to four SOPs per cluster. Mea-

suring the SOP selection times in 10 different pupae (20 distinct clusters) revealed a bias for early selection of the lowest SOP, probably reflecting an earlier initiation of this part of the cluster. However, the upper three SOPs appeared at seemingly random order (Fig. 2B), supporting previous evidences for stochastic selection (*23*).

A defining aspect of algorithms for MIS selection is the per-round probability that a node joins the MIS. Current algorithms (*8, 9*) optimize this rate by dynamically increasing it when the number of active neighbors a node has decreases. During SOP selection, cells do not know the number of nonselected neighbors. However, the temporal selection rate may still be optimized by cell-autonomous mechanisms, for example by stochastically accumulating a protein (such as Delta) until it passes some threshold. Characterizing the stochastic accumulation rate is thus a key for understanding the biological selection strategy. To determine this rate, we compared statistics derived from the observed SOP selection times with several in silico stochastic accumulation models. The models differ in the way by which stochasticity is introduced (*14*). Results of two of these models were consistent with our experimental data (Fig. 3). The first consistent model assumed a fixed rate of accumulation over time, and we concluded that it is not appropriate for computer networks (*14*). In contrast, the second model assumed a burst-like protein production in which the likelihood of bursting increases in time, resembling a com-

Table 1. MIS algorithm.

1. Algorithm: MIS (n, D) at node u
2. For $i = 0$: log D
    3. For $j = 0$: M log n // M is constant derived below
        4. * exchange 1*
        5. $v = 0$
        6. With probability $\frac{1}{2^{\log D - i}}$ broadcast **B** to neighbors and set $v = 1$ // **B** is one bit
        7. If received message from neighbor, then $v = 0$
        8. * exchange 2 *
        9. If $v = 1$ then
            10. Broadcast B; join MIS; exit the algorithm
        11. Else
            12. If received message B in this exchange, then mark node u inactive; exit the algorithm
        13. End
    14. End
15. End

putational algorithm for MIS in which the selection probability also increases in time as the number of active processors decreases. We thus asked whether we can develop an algorithm for MIS selection on the basis of this stochastic rate change model that would not require knowledge about the number of active neighbors and would only use threshold (binary) communication.

We assumed a collection of identical processors placed at nodes of an arbitrary synchronous communication network. Nodes can only broadcast one-bit messages. A message broadcasted by a node reaches all of its neighbors that are still active in the algorithm. In each round, a processor can only tell whether or not a message was sent to it. When a processor receives a message, it cannot tell which of its neighboring processors sent it, and it cannot count the number of messages received in a round. Hence, our model is appropriate for radio networks with collision detection. We assumed that nodes receive as input an upper bound on the number of nodes in the network (n) and an upper bound D, on the number of neighbors any node can have (if no such bound is known, we set D to n). We further assumed that no failures occur. The algorithm, presented in Table 1, is synchronously executed by all nodes.

The algorithm proceeds in $log D$ phases, each consisting of M log n steps, where M is a constant; its value is given in (14). Initially, all nodes are active. Each step in each phase i consists of two exchanges. In the first exchange, each active node broadcasts a message to its neighbors with probability p_i. Such as in the biological model, the probability p_i increases with i. In the second exchange, a node that has broadcasted a message in the first exchange joins the MIS if none of its neighbors had broadcasted at the first exchange. Such node broadcasts again a message to its neighbors, telling them to become inactive, and exit the algorithm.

We proved that when the algorithm terminates, no two neighboring nodes are in A (the

MIS set), and that every node that has become inactive has a neighbor in A [the proof can be found in (14)]. We thus conclude that the only way the algorithm may err is by terminating while leaving some nodes that are not in A and are also not connected to nodes in A. Next, we show that when the algorithm terminates all nodes are, with high probability, either in A or connected to a node in A, which solves the MIS problem.

The proof and the complete analysis are provided in (14). Briefly, the proof relies on an inductive argument to show that with high probability, by the time phase i ends (14) there are no active nodes with more than $\frac{D}{2^i}$ active neighbors. Thus, nodes with many neighbors either leave the algorithm (joining A or eliminating when a neighbor joins A) or lose many of their neighbors at each phase as these neighbors exit the algorithm. By the time the algorithm ends, i equals $log D$, and so all nodes that have not joined A are, with high probability, not connected to any active node (and are also not connected to any node in A) and thus can join A with no collisions.

The running time of the algorithm is O($log n$ $log D$), which is the number of rounds required to execute the two nested loops. The worst-case running time is O($log^2 n$). All messages in the algorithm are one bit. We prove in (14) that the expected number of messages sent to active nodes in our algorithm is linear in the number of nodes of the network, which is optimal because each node is required to at least receive a message from its local leader.

Taken together, by studying a developmental process in flies we devised a solution to an important distributed computing problem. The new algorithm does not require knowledge of the degree of individual processors, uses one-bit messages, and has an optimal message complexity. These features are useful for many applications, including wireless communication systems and ad hoc sensor networks.

Biologists are increasingly relying on advanced modeling techniques. The other direction—using

insights from biology to advance computational systems—has mainly focused on optimization techniques inspired by biological observations, including neural networks, genetic algorithms, and routing (24). We have shown that areas of computer science that require strict, provable guarantees can also benefit from knowledge regarding how biological systems operate. Better understanding of these biological systems can lead to further improvement in the design of complex distributed computing systems.

References and Notes
1. P. Flicek, E. Birney, *Nat. Methods* **6**, (11s), S6 (2009).
2. S. F. Altschul *et al.*, *Nucleic Acids Res.* **25**, 3389 (1997).
3. U. Alon, *An Introduction to Systems Biology: Design Principles of Biological Circuits* (Chapman & Hall/CRC, Boca Raton, FL, 2006).
4. N. Lynch, *Distributed Algorithms* (Morgan Kaufmann, San Francisco, 1996).
5. P. J. Wan, K. M. Alzoubi, O. Frieder, *Mobile Netw. Appl.* **9**, 141 (2004).
6. L. G. Valiant, *Proceedings of the 7th IBM Symposium on Mathematical Foundations of Computer Science* (IBM, New York, 1982).
7. S. Cohen, D. Lehmann, A. Pnueli, *Theor. Comput. Sci.* **34**, 215 (1984).
8. M. Luby, *SIAM J. Comput.* **15**, 1036 (1986).
9. N. Alon, L. Babai, A. Itai, *J. Algorithms* **7**, 567 (1986).
10. Y. Métivier, J. M. Robson, N. Saheb-Djahromi, A. Zemmari, "An optimal bit complexity randomized distributed MIS algorithm," in *Proceedings of the 16th International Colloquium on Structural Information and Communication Complexity (SIROCCO)*, pp. 323–337 (2009).
11. T. Moscibroda, R. Wattenhofer, "Maximal independent sets in radio networks," in *Proceedings of the Twenty-Fourth Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing (PODC)*, pp. 148–157 (2005).
12. S. J. Bray, *Nat. Rev. Mol. Cell Biol.* **7**, 678 (2006).
13. J. M. Rendel, B. L. Sheldon, D. E. Finlay, *Genetics* **52**, 1137 (1965).
14. Materials and methods are available as supporting material on *Science* Online.
15. P. Simpson, *Curr. Opin. Genet. Dev.* **7**, 537 (1997).
16. M. E. Fortini, *Dev. Cell* **16**, 633 (2009).
17. B. Castro *et al.*, *Development* **132**, 3333 (2005).
18. A. C. Miller, E. L. Lyons, T. G. Herman, *Curr. Biol.* **19**, 1378 (2009).
19. O. Sprinzak *et al.*, *Nature* **465**, 86 (2010).
20. O. Barad, D. Rosin, E. Hornstein, N. Barkai, *Sci. Signal.* **3**, ra51 (2010).
21. J. R. Collier, N. A. Monk, P. K. Maini, J. H. Lewis, *J. Theor. Biol.* **183**, 429 (1996).
22. E. Plahte, *J. Math. Biol.* **43**, 411 (2001).
23. K. Usui, K.-i. Kimura, *Rouxs Arch. Dev. Biol.* **203**, 151 (1993).
24. A. Tero *et al.*, *Science* **327**, 439 (2010).
25. We thank J. W. Posakony for fly stock and D. Rosin for help in experiments. This work was supported by a European Research Council (ERC) Advanced grant, the Oswald Veblen Fund, and the Bell Companies Fellowship (N.A.), the Azrieli Foundation (O.B.), the Israel Science Foundation (E.H. and N.B.), ERC (IDEA), the Hellen and Martin Kimmel award for innovative investigations (N.B.), NIH grant 1RO1 GM085022, and NSF CAREER award 0448453 (Z.B.-J.).