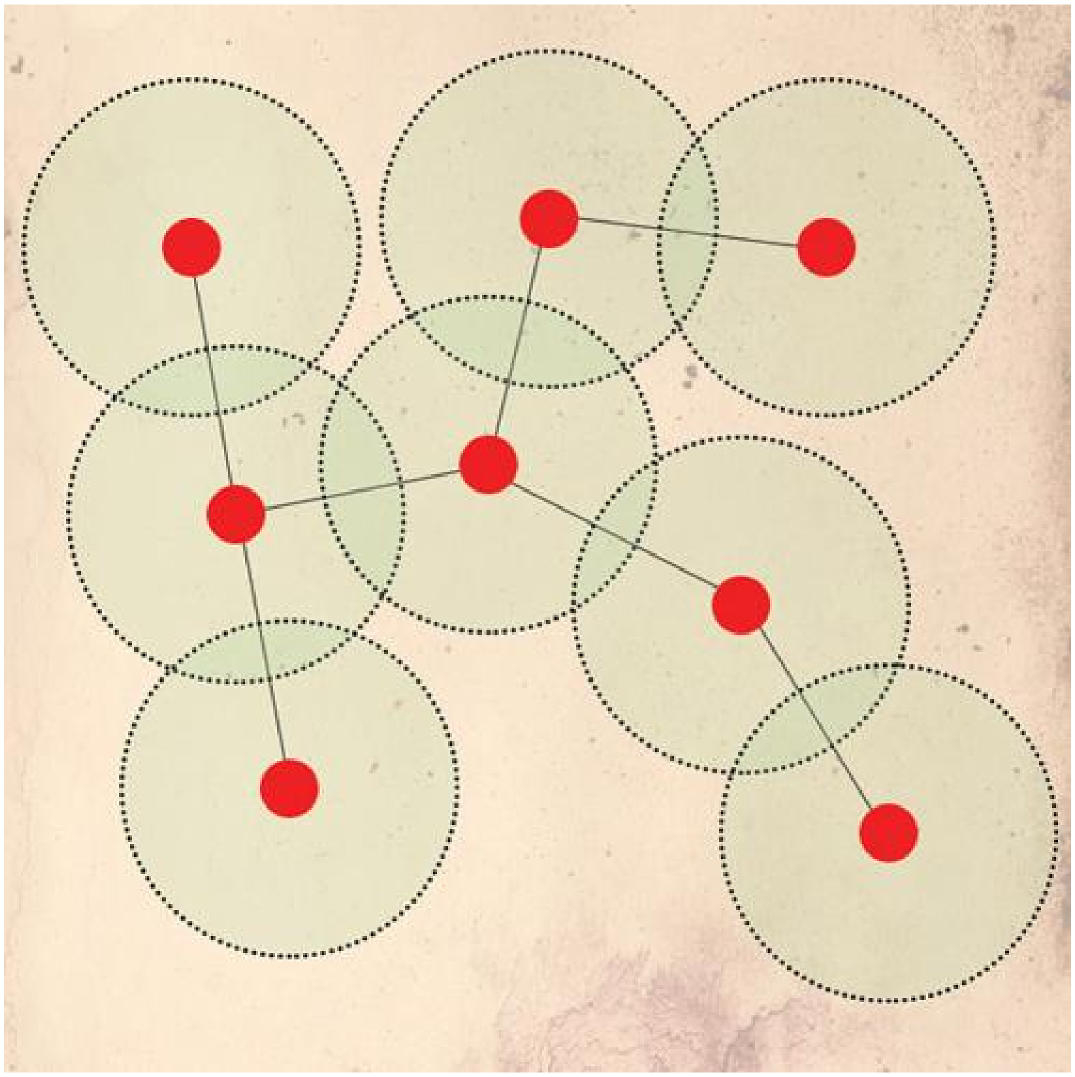
New algorithm for message dissemination in decentralized networks faster than its predecessors and guarantees delivery

January 9 2013, by Larry Hardesty



Credit: Christine Daniloff/News Office

Ad hoc networks—communication networks set up on the fly by mobile sensors—pose problems that ordinary office networks don't. Ad hoc networks are usually decentralized, meaning that no one node knows what the network as a whole looks like.

One of the questions raised by decentralized networks is how to relay messages so that they will reliably reach all the nodes, even when the network's shape is unknown. At the ACM-SIAM Symposium on Discrete Algorithms this month, Bernhard Haeupler, a graduate student in MIT's Department of Electrical Engineering and Computer Science, won one of two best-student-paper prizes for a new algorithm that answers that question.

Haeupler's algorithm is faster than previous algorithms. But its real interest, he believes, is that it's deterministic, meaning that it will provably relay messages to every node in a network. Previous algorithms, he explains, were probabilistic: No matter how long they were allowed to run, there would always be some minuscule chance that a message didn't reach some nodes.

"In the distributed community, solving problems without randomization is often a completely different problem, and deterministic algorithms are often drastically slower," Haeupler explains. "I don't think that anyone really believed that you can do anything deterministically in this setting."

If one node in a network maintains a map of the network as a whole, then it's relatively easy to devise a deterministic message-passing algorithm. But that kind of centralization is impractical in ad hoc networks, for several reasons: Ad hoc networks are often deployed in harsh environments, such as battle zones or the ocean's depths; a supervisor node could simply get knocked offline. Many ad hoc networks—like networks of wirelessly connected cars, or networks of swarming robots investigating an oil leak—are constantly changing, so a network map would quickly become obsolete. Finally, devices that form ad hoc networks usually run on batteries, so power is at a premium: Even if a supervisor node could keep up with the network's changing shape, doing so might drain its battery—and those of the nodes feeding it information.

Collective action

So researchers who study ad hoc networks are generally looking for distributed algorithms, in which all the nodes perform the same simple, local task, from which the desired global behavior emerges.

As is typical in the field, Haeupler envisions communication in the network as a game that proceeds in a series of rounds, where a round is the time required for two adjacent nodes to exchange information. In the basic version of his algorithm, each node begins by selecting one neighbor and exchanging information with it for one round. Then it selects a second node, and it exchanges information with both of the nodes it's selected for two rounds each.

At this point, the node will have received information that was forwarded by its neighbors. So when it selects a third node, it chooses one from which it hasn't heard either directly or indirectly. Then it sends all its information to all three nodes, for three rounds each. The algorithm continues in this manner, selecting new nodes from which it hasn't heard and incrementing the number of rounds it uses to send information to each of them. It continues this process until it has heard, either directly or indirectly, from all of its neighbors.

Haeupler showed that the largest number of neighbors that a node will ever need to select is equivalent to the logarithm, base two, of the number of nodes in the network. And in fact, he developed an even more efficient version of the algorithm, which reduces the number of rounds that a node must spend communicating with each of its neighbors, provided it addresses them in a prescribed order. "There are some indications that this might be as fast as you can do it, but I couldn't completely prove it," Haeupler adds.

Clear picture

"[Haeupler's] analysis is way simpler and much more elegant than what has been done in the past," says Rajmohan Rajaraman, a professor of computer and information science at Northeastern University. "I think it greatly enhances the understanding of processes of the kind that he's studying."

In order to prove that probabilistic information-passing algorithms don't incur exorbitant delays, Rajaraman explains, computer scientists will usually offer "delay-sequence" arguments. "You would use nontrivial probabilistic arguments to show that the probability of this happening is very small, and OK, this will not happen, and for this event to happen, some other event has to happen, and that likelihood is small," he says. "So it will be a chain of arguments which is not deep but certainly is more involved."

In Haeupler's case, however, "that delay-sequence argument basically comes to saying, for there to be an excess of delays, there must exist this large graph structure, which is impossible," Rajaraman says. "It's much easier to look at this argument, believe it, and keep it in your head."

"There are still a couple of gaps, I believe," Rajaraman adds. "The problem is not yet completely settled. But this is a nice almost-end to this line of work."

research, innovation and teaching.

Provided by Massachusetts Institute of Technology

Citation: New algorithm for message dissemination in decentralized networks faster than its predecessors and guarantees delivery (2013, January 9) retrieved 25 April 2024 from https://phys.org/news/2013-01-algorithm-message-dissemination-decentralized-networks.html