

Honeybees and the internet

The internet

Bees can teach computer programmers a lesson

Apr 15th 2004 | [From the print edition](#)

HONEYBEES want to make honey, while internet host providers want to make money. And the rhyme is not the only thing to link honeybees and internet host providers, who operate vast warehouses full of server computers on behalf of their customers. Sunil Nakrani of Oxford University and Craig Tovey of the Georgia Institute of Technology, in Atlanta, have recently developed what they believe is an efficient way of organising internet servers, by mimicking the behaviour of honeybee colonies.

The unpredictability of internet traffic (sudden surges of shoppers or stockmarket trading; the ups and downs of an internet auction) gives firms which run servers quite a headache. To maximise their profit, they have to juggle their computers between different applications in order to adapt to changing levels of demand. Only one web-application can be loaded on to a computer at one time, and switching between applications incurs a penalty of five to seven minutes of downtime while the computer is reconfigured.

Honeybees have a similar problem. Patches of flowers vary in quality, so a colony needs to “decide” how many bees will forage at each patch and how long they will forage, in order to maximise the rate of nectar collection.

Millions of years of evolution have provided the bees with a pretty good solution. Around one-fifth of the bees in a hive are employed as nectar collectors. Their job is to zip back and forth to flower patches, gathering batches of nectar. On returning to the hive, they transfer their takings to one of the stay-at-home food-storer bees, who then stash it in a honeycomb.

A nectar-collecting bee judges how good its flower patch is, relative to the patches being visited by its hive-mates, by seeing how long it takes to find an unemployed food-storer bee. If it takes ages, then the forager concludes that its patch is nothing special, and that most of the other forager bees must have had successful runs, too. But if there is a plethora of food-storer bees ready to take the nectar, then the forager realises that it has struck lucky.

Using this information, the forager decides whether its flower patch is worthwhile. If so, it signals to others to follow it back by doing the famous waggle dance. The length of this dance indicates how profitable a flower patch is likely to be.

Dr Nakrani and Dr Tovey have exploited the honeybees' strategy and applied it to the problems of internet hosts. By comparing individual servers to foraging bees, and customer requests to flower patches, the two researchers have developed a “honeybee” algorithm for internet-server “hives”. Instead of waggle-dancing, a server produces an “advert”, which it sends to the other servers in the hive. The duration of this advert reflects the importance and profitability of that server's customers. Other servers that read the advert act like worker bees following a waggle dance—judging on the basis of the advert, and of their own recent experience, whether to switch from the customers they are currently serving to the new ones being served by the server that produced the advert.

Honeybees and internet servers share similar problems right down to the finest level of detail. For example, the downtime penalty incurred when a computer is switched from one web-application to another can be compared to the trade-off for a honeybee that is switching flower patches—an activity that usually requires several attempts before a profitable new patch is located.

To see whether their analogy is useful, Dr Nakrani and Dr Tovey tested the honeybee algorithm against the so-called greedy algorithm currently used as the basis for the allocation of servers by most internet host providers. A greedy algorithm is backward looking. It divides time into fixed periods and allocates servers to customers for a period by working out what would have been the most profitable arrangement in the preceding one.

When internet traffic was highly variable, the honeybee algorithm outperformed greedy by as much as 20%. If the traffic became more uniform, then the greedy algorithm began to outperform honeybee. However, most internet traffic is, indeed, highly variable. That suggests Dr Nakrani and Dr Tovey might be on to something. As is often the case with human inventions, nature got there first.

[From the print edition: Science and technology](#)