*** 北米連携センター定期報告***

2014 年 7 月 17 日

●マサチューセッツ工科大学（MIT）研究チーム、データセンターの伝送遅延緩和する新システム開発

【MIT, 2014/07/17】

大手ウェブサイトが持つデータセンターは数万あるいは数 10 万台のサーバで構築されており、大規模分散型ネットワークであることから混雑が起きやすいのは宿命。

マサチューセッツ工科大学（MIT）の研究チームは 8 月に開かれる「CM Special Interest Group on Data Communication」の年次カンファレンスで、この問題を緩和する新しいネットワーク管理システムを発表する予定。

同システムは、実験ではフェイスブックのデータセンターで使われるルータの平均キュー時間を 99.6%短縮することに成功。ネットワーク・トラフィックが多い時の平均レイテンシーは、3.56 ミリ秒から 0.23 ミリ秒に縮んだ。

「Fastpass」と名付けられたこのシステムでは、従来の分散型システムではそれぞれのノードが隣接するどのノードにいつデータを送信できるかを決めるのに対して、これを総合的に判断する「Arbiter」という中央サーバを置いている。

データを発信するノードは、まずこの「Arbiter」にリクエストを出し、転送経路の割当てを受け取る仕組み。これまでの実験では、8 コアの「Arbiter」で毎秒 2.2 テラビットのデータを伝送するネットワークに対応できるという。これは、サーバ 2000 台のデータセンターを 1Gps で接続し、常時フル稼働させることに相当する。

（参考）本件報道記事

No-wait data centers

New system could reduce data-transmission delays across server farms by 99.6 percent

Big websites usually maintain their own "data centers," banks of tens or even hundreds of thousands of servers, all passing data back and forth to field users' requests. Like any big, decentralized network, data centers are prone to congestion: Packets of data arriving at the same router at the same time are put in a queue, and if the queues get too long, packets can be delayed.

At the annual conference of the ACM Special Interest Group on Data Communication, in August, MIT researchers will present a new network-management system that, in experiments, reduced the average queue length of routers in a Facebook data center by 99.6 percent — virtually doing away with queues. When network traffic was heavy, the average latency — the delay between the request for an item of information and its arrival — shrank nearly as much, from 3.56 microseconds to 0.23 microseconds.

Like the Internet, most data centers use decentralized communication protocols: Each node in the network decides, based on its own limited observations, how rapidly to send data and which adjacent node to send it to. Decentralized protocols have the advantage of an ability to handle communication over large networks with little administrative oversight.

The MIT system, dubbed Fastpass, instead relies on a central server called an "arbiter" to decide which nodes in the network may send data to which others during which periods of time. "It's not obvious that this is a good idea," says Hari Balakrishnan, the Fujitsu Professor in Electrical Engineering and Computer Science and one of the paper's coauthors.

With Fastpass, a node that wishes to transmit data first issues a request to the arbiter and receives a routing assignment in return. "If you have to pay these maybe 40 microseconds to go to the arbiter, can you really gain much from the whole scheme?" says Jonathan Perry, a graduate student in electrical engineering and computer science (EECS) and another of the paper's authors. "Surprisingly, you can."

Division of labor

Balakrishnan and Perry are joined on the paper by Amy Ousterhout, another graduate student in EECS; Devavrat Shah, the Jamieson Associate Professor of Electrical Engineering and Computer Science; and Hans Fugal of Facebook.

The researchers' experiments indicate that an arbiter with eight cores, or processing units, can keep up with a network transmitting 2.2 terabits of data per second. That's the equivalent of a 2,000-server data center with gigabit-per-second connections transmitting at full bore all the time.

"This paper is not intended to show that you can build this in the world's largest data centers today," Balakrishnan says. "But the question as to whether a more scalable centralized system can be built, we think the answer is yes."

Moreover, "the fact that it's two terabits per second on an eight-core machine is remarkable," Balakrishnan says. "That could have been 200 gigabits per second without the cleverness of the engineering."

The key to Fastpass's efficiency is a technique for splitting up the task of assigning transmission times so that it can be performed in parallel on separate cores. The problem, Balakrishnan says, is one of matching source and destination servers for each time slot.

"If you were asked to parallelize the problem of constructing these matchings," he says, "you would normally try to divide the source-destination pairs into different groups and put this group on one core, this group on another core, and come up with these iterative rounds. This system doesn't do any of that."

Instead, Fastpass assigns each core its own time slot, and the core with the first slot scrolls through the complete list of pending transmission requests. Each time it comes across a pair of servers, neither of which has received an assignment, it schedules them for its slot. All other requests involving either the source or the destination are simply passed on to the next core, which repeats the process with the next time slot. Each core thus receives a slightly attenuated version of the list the previous core analyzed.

Bottom line

Today, to avoid latencies in their networks, most data center operators simply sink more money into them. Fastpass "would reduce the administrative cost and equipment costs and pain and suffering to provide good service to the users," Balakrishnan says. "That allows you to satisfy many more users with the money you would have spent otherwise."

Networks are typically evaluated according to two measures: latency, or the time a single packet of data takes to traverse the network, and throughput, or

the total amount of data that can pass through the network in a given interval.

Source: http://newsoffice.mit.edu/2014/no-wait-data-centers-0717

以　上