

Nâng cao hiệu năng giao thức TCP

Improving Performance of TCP Protocol

Phạm Đạo, Lê Xuân Vinh

Abstract: The Internet is composed of many heterogenous networking technologies and efficiently supports the decentralized multiplexing of these resources; this ability has been instrumental of its success. Among the traffic crisscrosses the Internet, TCP/IP traffic takes a large share due to the majority of Internet applications, such as Web surfing or file transfer, is TCP/IP based. As a result, the performance of TCP/IP greatly affects the performance of the network, and has a direct impact on the overall quality of service. Studies on improving TCP performance remains a hot topic despite TCP is an invention of the 80s. This paper studies congestion control mechanism, an essential part in the protocol operation and point out several criteria in evaluating and testing TCP performance.

I. GIỚI THIỆU

Mạng Internet là mạng máy tính rộng lớn sử dụng nhiều công nghệ mạng khác nhau và cho phép chia sẻ tài nguyên mạng một cách phân tán. Chính khả năng này là nguyên nhân chính đem lại sự thành công to lớn của công nghệ Internet. Trong phần lớn lưu lượng trên mạng Internet, lưu lượng TCP/IP đóng góp một phần đáng kể vì phần lớn ứng dụng trên mạng Internet, ví dụ như lướt Web và truyền file, đều sử dụng giao thức TCP/IP. Do vậy, có thể thấy rằng hiệu năng của TCP/IP sẽ có ảnh hưởng lớn đến hiệu năng của mạng và trực tiếp ảnh hưởng đến chất lượng dịch vụ của mạng. Việc nghiên cứu các phương thức giúp nâng cao hiệu năng giao thức TCP/IP vẫn còn hết sức nóng hổi mặc dù giao thức TCP/IP đã được phát minh từ đầu thập niên 80. Bài báo này sẽ xem xét cơ chế điều khiển tắc nghẽn, một thành phần quan trọng trong hoạt động của giao thức TCP/IP và đề ra một số tiêu chí để đánh giá và đo kiểm hiệu năng TCP/IP.

Bài báo được chia làm 5 phần. Phần đầu tiên sẽ mô tả sơ lược hoạt động của cơ chế điều khiển tắc nghẽn trong giao thức TCP. Sau đó, phần 2 phân tích các yếu

tố ảnh hưởng đến hoạt động điều khiển tắc nghẽn của giao thức này. Phần 3 xem xét các dạng biến thể của TCP và phân tích ảnh hưởng của chúng. Một số tiêu chí để đánh giá và đo kiểm hiệu năng của giao thức TCP sẽ được trình bày trong phần 4. Phần cuối cùng tóm tắt nội dung của bài báo.

II. CƠ CHẾ ĐIỀU KHIỂN TẮC NGHẼN

Tắc nghẽn xảy ra trong mạng thông tin khi lưu lượng gửi vào mạng vượt quá dung lượng truyền dẫn. Cơ chế điều khiển tắc nghẽn TCP [1] được phát triển nhằm giải quyết hiện tượng tắc nghẽn trong mạng Internet. Nhờ vào cơ chế điều khiển tắc nghẽn, tài nguyên mạng được chia sẻ giữa các luồng tin trong thời gian xảy ra tắc nghẽn. Việc chia sẻ này cho phép các luồng tin có thời gian trễ RTT (Round trip Time) giống nhau đạt được thông lượng như nhau và tránh việc một luồng tin bị tắc nghẽn hoàn toàn. TCP đã rất thành công trong việc chia sẻ băng thông trong khi vẫn tận dụng tối đa dung lượng hiện có trong nhiều điều kiện tải khác nhau.

Thuật toán điều khiển luồng TCP sử dụng cơ chế cửa sổ và thông báo đã nhận (ACK) đầu cuối tới đầu cuối để đảm bảo truyền tin một cách tin cậy trong mạng. Bên gửi duy trì một cửa sổ tắc nghẽn, c_{wnd} , đây là số gói tin có thể được máy gửi đi trong khi chờ bản tin ACK từ bên nhận. Sau khi nhận được gói tin số liệu, bên nhận chuẩn bị một bản tin ACK để thông báo cho bên gửi về các bản tin đã nhận được. Bên nhận cũng thông báo cho bên gửi một cửa sổ nhận, r_{wnd} , là kích thước của bộ đệm dành cho kết nối này. Bên gửi không được nhiều hơn $\min\{c_{wnd}, r_{wnd}\}$ gói tin trong khi chờ nhận bản tin ACK. Cửa sổ nhận cho phép bên nhận điều khiển luồng vì nếu bên nhận không thể xử lý kịp thông tin với tốc độ hiện thời, cửa sổ nhận sẽ được làm nhỏ đi, và bên gửi sẽ giảm tốc độ

gửi của mình. Cửa sổ tắc nghẽn có nhiệm vụ điều khiển luồng trong điều kiện tắc nghẽn. Việc mất gói tin được phát hiện thông qua timeout của các gói tin chưa được thông báo nhận hoặc nhận được nhiều bản tin ACK giống nhau, hoặc nhận được bản tin SACK từ bên nhận [2]. Với giao thức TCP, việc mất gói tin đồng nghĩa với tắc nghẽn trong mạng vì mất gói tin được giả thiết là do tràn bộ đệm do lưu lượng vượt quá dung lượng trên tuyến kết nối đầu cuối tới đầu cuối. Bên gửi cập nhật kích thước cửa sổ tắc nghẽn khi nhận được bản tin ACK cho các gói tin gửi đi và khi phát hiện tắc nghẽn. Sau một RTT không có tắc nghẽn, cho mỗi bản tin ACK nhận được, cửa sổ tắc nghẽn được cập nhật theo công thức:

$$c_{wnd} = c_{wnd} + \frac{1}{c_{wnd}} \quad (1)$$

và khi phát hiện tắc nghẽn trong một RTT:

$$c_{wnd} = \frac{c_{wnd}}{2} \quad (2)$$

Quy trình tăng giảm kích thước cửa sổ tắc nghẽn cho phép TCP tận dụng băng thông hiện có trên một tuyến kết nối đầu cuối tới đầu cuối.

Mức độ “linh hoạt” của thuật toán điều chỉnh kích thước cửa sổ có thể được đánh giá dựa trên việc xem xét thời gian cần thiết để cửa sổ tắc nghẽn có kích thước như cũ sau một lần tắc nghẽn. Giả sử kết nối có RTT là 200 ms và kích thước gói tin là 1500 byte. Băng thông 1Gb/s sẽ tương ứng với cửa sổ tắc nghẽn có kích thước tương ứng với 17000 gói tin. Sau một lần tắc nghẽn, c_{wnd} được điều chỉnh xuống còn 8500 gói tin, tương ứng với thông lượng 500 Mb/s. Để đạt được tốc độ gửi 1Gb/s, sẽ cần tới 8500 RTT, tức là 28 phút. Điều này dẫn tới hiệu suất sử dụng thấp đối với các kết nối tốc độ cao.

III. CÁC YẾU TỐ ẢNH HƯỞNG

Xem xét thuật toán điều khiển tắc nghẽn TCP, chúng ta thấy kích thước cửa sổ tắc nghẽn và cửa sổ nhận sẽ quyết định hiệu năng (thông lượng) của TCP. Do số lượng gói tin bên gửi có thể gửi đi trước khi nhận bản tin ACK bằng $\min\{c_{wnd}, r_{wnd}\}$ nên c_{wnd} và r_{wnd} càng lớn thì bên gửi có thể gửi càng nhiều gói tin, tức là thông lượng kết nối sẽ càng lớn. Các yếu tố quyết định kích thước và tốc độ thay đổi của cửa sổ

tắc nghẽn là:

1. RTT
2. Xác suất gói bị lỗi hay mất gói
3. Băng thông đầu cuối tới đầu cuối

Kích thước của cửa sổ tắc nghẽn tỷ lệ nghịch với giá trị của cả 3 đại lượng trên. RTT càng lớn thì thời gian kích thước cửa sổ tăng lên 1 đơn vị sẽ tăng. Xác suất lỗi hay mất gói lớn, thì xác suất không nhận được bản tin ACK hoặc nhận được bản tin ACK lặp lại sẽ lớn, đồng nghĩa với việc kích thước cửa sổ bị giảm đi. Băng thông đầu cuối tới đầu cuối càng lớn thì thời gian để cửa sổ tắc nghẽn, sau khi tắc nghẽn xảy ra, có được kích thước như cũ (còn được gọi là thời gian phục hồi) sẽ tăng lên.

Ta có thể thấy rằng, cả 3 đại lượng trên đều phụ thuộc vào mạng kết nối bên gửi và bên nhận chứ không phụ thuộc vào bên gửi. Như vậy, để tăng thông lượng của TCP, ngoài việc nâng cao chất lượng mạng (giảm RTT và xác suất lỗi), ta có thể thay đổi thuật toán cập nhật cửa sổ tắc nghẽn ở bên gửi.

IV. CÁC BIẾN THỂ CỦA TCP

Phần lớn các biến thể TCP đều tập trung vào việc nâng cao thông lượng kết nối TCP trong môi trường mạng tốc độ cao thông qua việc thay đổi cách thức cập nhật kích thước cửa sổ tắc nghẽn. Như ta đã thấy ở phần trên, khi băng thông đầu cuối tới đầu cuối lớn, thuật toán điều khiển tắc nghẽn tăng kích thước cửa sổ rất chậm. Bảng 1 cho thấy ảnh hưởng của băng thông lên thời gian phục hồi của kết nối TCP có RTT là 200 ms.

Biến thể đầu tiên là TCP Vegas [3-5]. TCP Vegas coi trễ bộ đệm trong mạng (chứ không phải là mất gói như TCP truyền thống) là tín hiệu của sự tắc nghẽn mạng. Do vậy, nếu trễ bộ đệm trong mạng có thể được điều khiển và sử dụng như một cơ chế báo hiệu tắc nghẽn thì có thể đạt được hiệu suất sử dụng mạng rất cao. Thuật toán cập nhật cửa sổ tránh tắc nghẽn được thực hiện như phương thức thông thường (đã trình bày ở trên). Cách tiếp cận này có thể tốt nhưng rất khó thực hiện được. Để đạt được hiệu quả cao, mã TCP Vegas phải hoạt động ổn định trong môi trường có nhiều yếu tố ảnh hưởng đến việc ước lượng trễ mạng như cơ chế bộ đệm khác nhau, cơ chế định thời của hệ điều hành, hoạt động của firewall mạng, và các lưu

lượng không có cơ chế điều khiển trễ bộ đệm như TCP và UDP cổ điển.

Bảng 1. Ảnh hưởng của băng thông đến thời gian phục hồi của thuật toán điều khiển tắc nghẽn truyền thống.

Băng thông	Kích thước cửa sổ (gói)	Thời gian phục hồi
1Mbit/s	17	1.7 s
10Mbit/s	170	17s
100Mbit/s	1700	2m 50s
1 Gbit/s	17000	28 m
10 Gbit/s	170000	4h 43m

Cơ chế tạo một kết nối logic TCP từ nhiều kết nối TCP cũng có thể tăng thông lượng của kết nối. Cơ chế này có thể được thực hiện ở lớp truyền tải [6] hoặc ở lớp ứng dụng. Mỗi luồng kết nối TCP sẽ sử dụng cơ chế cập nhật cửa sổ như thông thường nhưng do sử dụng cùng lúc nhiều kết nối và do tính chất “hấu ăn” của thuật toán cập nhật cửa sổ nên thông lượng tăng hơn so với chỉ sử dụng một luồng kết nối TCP duy nhất. Tuy nhiên, sẽ rất khó lựa chọn các tham số hoạt động phù hợp để đảm bảo giao thức hoạt động tốt mà không làm ảnh hưởng đến lưu lượng hiện có trong trường hợp tắc nghẽn xảy ra.

Một dạng biến thể khác đề xuất một công thức cập nhật kích thước cửa sổ mới, giúp thời gian phục hồi của kết nối không phụ thuộc vào băng thông, do vậy thích hợp với mạng tốc độ cao hơn. Scalable TCP [7] tăng giảm kích thước cửa sổ theo công thức sau:

$$c_{wnd} = c_{wnd} + a$$

$$c_{wnd} = c_{wnd} - \lceil b \cdot c_{wnd} \rceil$$

trong đó, a và b là hai hằng số có giá trị thích hợp trong khoảng (0,1). Nếu thời gian phục hồi của TCP cổ điển là $C/2 \text{ RTT}$ thì Scalable TCP có thời gian phục hồi là $\log(1-b)/\log(1+a)$. Như vậy, thời gian phục hồi của Scalable TCP luôn không đổi với bất kỳ băng thông nào, trong khi thời gian phục hồi của TCP cổ điển tăng theo độ lớn của băng thông.

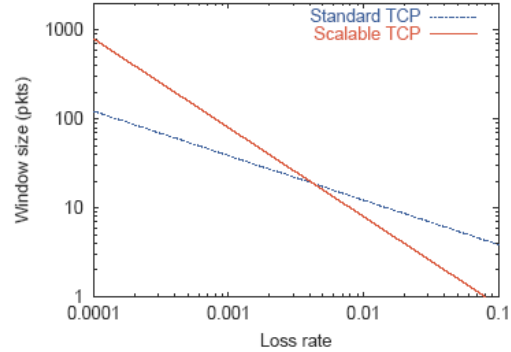
V. TIÊU CHÍ ĐÁNH GIÁ VÀ ĐO KIỂM

Với bất kỳ biến thể nào của TCP, chúng ta cần xem xét đánh giá trên lý thuyết các đặc tính sau:

1. Đường cong đáp ứng và phân chia băng thông.
2. Biến thiên tốc độ tức thời.
3. Tốc độ hội tụ.

4. Độ ổn định.

Đường cong đáp ứng cho thấy mối quan hệ giữa kích thước cửa sổ tắc nghẽn và tốc độ báo hiệu đầu cuối tới đầu cuối. Hình 1 minh họa đường cong đáp ứng của TCP cổ điển và Scalable TCP [7]. Đường cong đáp ứng của giao thức càng nằm trên cao thì kích thước cửa sổ tắc nghẽn của kết nối TCP sử dụng giao thức đó càng lớn và băng thông dành cho kết nối đó sẽ cao hơn các kết nối TCP dùng giao thức khác.



Hình 1. Đường cong đáp ứng của TCP cổ điển và Scalable TCP.

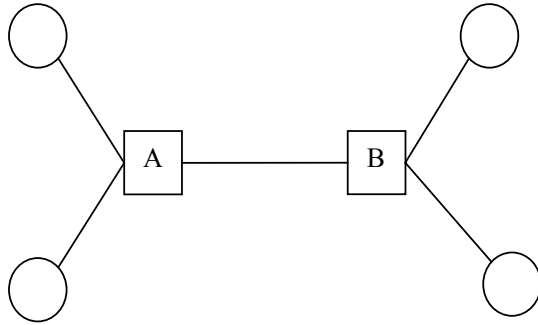
Tốc độ tức thời của một kết nối TCP biến thiên xung quanh giá trị trung bình của nó. Biến thiên tốc độ tức thời càng nhỏ càng tốt vì biến thiên tốc độ tức thời nhỏ đồng nghĩa với sự thay đổi kích thước cửa sổ quanh giá trị trung bình nhỏ.

Tốc độ hội tụ của giao thức có ý nghĩa hết sức quan trọng vì nó ảnh hưởng đến thời gian thích nghi của giao thức. Trên lý thuyết, sự hội tụ phải xảy ra tức thì. Tuy nhiên, trên thực tế, điều này rất khó thực hiện do việc báo hiệu tắc nghẽn lại thông qua mất gói tin và phải đảm bảo được sự tương thích giữa các biến thể TCP cùng hoạt động trên mạng. Tốc độ hội tụ được tính bằng thời gian (tính theo RTT) cần thiết để TCP giảm tốc độ xuống $\frac{1}{2}$ sau khi có tắc nghẽn và để tăng tốc độ lên gấp đôi sau khi hết tắc nghẽn.

Độ ổn định của giao thức phải được chứng minh trên lý thuyết. Lý thuyết điều khiển thường được sử dụng để đánh giá độ ổn định của giao thức xung quanh điểm cân bằng của nó. Các tham số của giao thức phải được lựa chọn sao cho độ ổn định phải được đảm bảo với các giá trị RTT và topo mạng khác nhau.

Thông lượng của kết nối TCP sử dụng các biến thể TCP khác nhau có thể được kiểm tra trên cấu hình

mạng trên Hình 2.



Hình 2. Cấu hình đo kiểm.

Trong cấu hình thử nghiệm này, các hình tròn là các server dùng giao thức TCP, A và B là hai router. Các đường thẳng liền nét giữa máy tính và router là các kết nối Ethernet tốc độ cao (để đảm bảo băng thông và trễ trên kết nối này tốt hơn mức cần thiết), và giữa hai router A và B là kết nối có băng thông và trễ RTT theo yêu cầu thử nghiệm (băng thông và trễ RTT trên kết nối này sẽ được cấu hình theo yêu cầu của bài đo).

Trong bước kiểm tra đầu tiên, cấu hình n cặp server bên A và máy nhận bên B trao đổi file dữ liệu kích thước 2GB, cấu hình bộ đệm bên gửi và bên nhận theo yêu cầu sao cho một luồng tin sẽ sử dụng hết băng thông hiện có. Việc truyền file được thực hiện liên tục giữa server. Thời gian truyền file được ghi nhận. Giao thức TCP có số lần truyền file thành công trong khoảng thời gian nhỏ hơn một giá trị T cho trước nhiều nhất là giao thức cho thông lượng lớn nhất.

Bước kiểm tra thứ hai đánh giá ảnh hưởng (tiêu cực) của biến thể TCP tới các lưu lượng TCP cổ điển. Ở đây, m cặp server và bên nhận được cấu hình để tạo lưu lượng Web. Một cặp server và máy nhận được cấu hình để truyền 8 cuộc truyền file kích thước 2 GB một lúc. Mọi sự thay đổi thông lượng của lưu lượng Web được ghi nhận. Thông lượng Web không thay đổi chứng tỏ biến thể TCP không làm ảnh hưởng đến hoạt động của TCP truyền thống.

VI. THỬ NGHIỆM

Chúng tôi dùng công cụ mô phỏng ns-2 đánh giá hiệu năng của Scalable TCP theo các tiêu chí đánh giá trình bày ở trên với cấu hình mạng mô phỏng sau. Mỗi đầu gồm 06 server chạy Scalable-TCP kết nối với router qua kết nối Gigabit Ethernet. Hai router được

nối với nhau qua kết nối 2.4Gbit/s với RTT là 120ms. Kết quả thu được được trình bày trong bảng 2 dưới đây.

Bảng 2. Tốc độ hội tụ và biến thiên tốc độ tức thời

B	a	$T1/2$ (ms)	$T2$ (ms)	Biến thiên
0,5	0,04	0,12	2,124	0,72
0,25	0,02	0,2892	4,2	0,502
0,125	0,01	0,6228	8,364	0,354
0,0625	0,005	1,2084	16,68	0,257

Độ ổn định của Scalable TCP đã được đánh giá trên lý thuyết trong [8].

VII. KẾT LUẬN

Bài báo này xem xét vấn đề nâng cao hiệu năng của giao thức TCP, cụ thể là các phương thức nâng cao thông lượng của kết nối TCP. Một số biến thể TCP cùng với các kỹ thuật giúp tăng thông lượng của kết nối đã được giới thiệu. Đặc biệt, các tiêu chí để đánh giá một biến thể TCP và đo kiểm hiệu năng của nó cũng đã được trình bày.

Ngoài các kỹ thuật cập nhật cửa sổ tắc nghẽn, còn nhiều phương pháp khác góp phần nâng cao hiệu năng của TCP như thay đổi cơ chế phát hiện lỗi, v.v... Bài toán tối ưu hoá TCP do vậy vẫn là một bài toán mở, cần được nghiên cứu thêm.

TÀI LIỆU THAM KHẢO

- [1] V. Jacobson. "Congestion Avoidance and Control." In SIGCOMM 1988.
- [2] M. Mathis, J. Mahdavi, S. Floyd, and A. Romanow. "TCP Selective Acknowledgment Options." Internet RFC 2018, October 1996.
- [3] L. S. Brakmo and L. L. Peterson. "TCP Vegas: End to End Congestion Avoidance on a Global Internet." IEEE Journal on Selected Areas in Communications, 13(8):1465–1480, October 1995.
- [4] E. Weigle and W. Feng. "A Case for TCP Vegas in High-Performance Computational Grids." In Proc. of the 9th IEEE International Symposium on High performance Distributed Computing (HPDC'01), San Francisco, CA, August 2001.
- [5] D. H. Choe and S. H. Low. "Stabilized Vegas." In Proc. of the 39th Annual Allerton Conference on

Communication, Control, and Computing, Monticello, IL, October 2002.

[6] J. Crowcroft and P. Oechsli. “*Differentiated End-to-End Internet Services using a Weighted Proportional Fair Sharing TCP.*” Computer Communication Review, 28(3), July 1998.

[7] T. Kelly. “*Scalable TCP: Improving Performance in Highspeed Wide Area Networks*”, Computer Communication Review 32(2), April 2003.

[8] G. Vinnicombe. *On the stability of networks operating TCP-like congestion control.* In Proc. of the 15th IFAC World Congress on Automatic Control, Barcelona, Spain, July 2002.

Ngày nhận bài: 19/09/2005

SƠ LƯỢC TÁC GIẢ

PHẠM ĐẠO

Tốt nghiệp kỹ sư tại Học viện Bưu điện Bắc kinh năm 1963. Được công nhận tiến sỹ kỹ thuật tại Đại học Kỹ thuật Budapest, Hungary. Được phong Phó Giáo sư năm 1992.

Nguyên Giám đốc Học viện Công nghệ Bưu chính Viễn thông.

Lĩnh vực quan tâm: Thông tin hữu tuyến, lý thuyết mạng.

LÊ XUÂN VINH

Sinh năm 1967.

Tốt nghiệp kỹ sư Vô tuyến điện tử – Thông tin tại trường Đại học Bách khoa – Hà nội, năm 1989. Đang là nghiên cứu sinh tại Học viện Công nghệ Bưu chính Viễn thông.

Hiện công tác tại Phòng Nghiên cứu Kỹ thuật Chuyển mạch - Viện Khoa học Kỹ thuật Bưu điện.

Lĩnh vực quan tâm: chuyển mạch và kỹ thuật đo lường.

Email: vinh.le@hn.vnn.vn