



Control de la congestión en TCP: reacción ante las pérdidas de segmentos

Apellidos, nombre	Baydal Cardona, María Elvira (mebaydal@disca.upv.es)
Departamento	Informática de Sistemas y Computadores
Centro	Escola Tècnica Superior d'Enginyeria Informàtica



1 Resumen de las ideas clave

En este artículo vamos a explicar como el protocolo TCP (*Transmission Control Protocol*) estima si existe o no congestión en una red de computadores. Así como las medidas que TCP aplicará cuando ha detectado que puede haber congestión en la red. Como veremos estas medidas son de distinto grado dependiendo de la gravedad de los indicios observados.

2 Objetivos

El objetivo de este artículo es explicar dos aspectos clave del control de la congestión en el protocolo TCP:

1. Cómo determina TCP que existe congestión en la red y si su grado es más o menos grave.
2. Cómo debe variar la tasa de transmisión de paquetes a la red en función de la congestión que percibe. Es decir, cuáles son las medidas que TCP debe aplicar cuando asume que existen posibilidades de que la red esté congestionada o en riesgo de llegar a estarlo.

3 Introducción

La congestión es uno de los problemas más críticos que pueden ocurrir en una red de computadores. De hecho en sus inicios, antes de incorporar medidas para evitar la congestión en la red, Internet llegó a sufrir lo que se conoce como colapso por congestión, debido al descarte masivo de paquetes en algunos routers y las consiguientes retransmisiones de esos paquetes descartados que contribuyeron a agravar el problema.

Uno de los principales objetivos del protocolo TCP es resolver este desafío, ya que la congestión provocará incrementos de la latencia (tiempo que tardan los paquetes en alcanzar su destino) y pérdida de paquetes (al ser descartados en los routers), pudiendo, en el peor de los casos, llegar a impedir totalmente el funcionamiento de la red. Además, TCP debe ser capaz de evitar la congestión utilizando sólo los indicios que pueda detectar localmente el emisor, sin que por la red circule información específica de control destinada a este fin. Adicionalmente, el reparto del ancho de banda de la red entre todos los flujos TCP presentes en la red debe ser equitativo. Un factor adicional que complica este reparto es que el tráfico en Internet varía con mucha rapidez, a veces en varios órdenes de magnitud en muy poco tiempo. TCP no solo debe evitar la congestión sino además intentar aprovechar en cada momento la máxima capacidad de transmisión disponible, ajustando el número de paquetes que la estación tiene en tránsito.

Para evitar la congestión TCP limitará la inyección de nuevos paquetes en la red desde los dispositivos conectados a ella, intentando que esta limitación afecte a todos los dispositivos cuyos paquetes transitan la misma área de la red, por igual. No se puede penalizar el tráfico de unos dispositivos para favorecer a otros.

4 Soluciones en TCP para el problema de la congestión

El control de la congestión TCP se ha estructurado en torno a la solución de tres problemas en el emisor:

- El primero, cómo limitar el envío de tráfico de datos, aunque no se detecten problemas.
- El segundo, cómo detectar sólo con información local que existe congestión en la red
- Por último, si se ha detectado congestión, cómo reducir la tasa de transmisión de datos en función de la congestión percibida

De los tres problemas que el emisor debe resolver, suponemos ya conocidas las estrategias que se utilizan para resolver el primero de los problemas mediante los algoritmos de arranque lento y evitación de la congestión. En este artículo nos centraremos en los dos últimos apartados: qué indicios utiliza TCP para detectar la posible congestión de la red y qué medidas aplica cuando detecta congestión. Además, también veremos cómo las medidas aplicadas intentan ajustarse al grado de gravedad del problema.

4.1 Cómo detectar que existe congestión

Para detectar si existe riesgo de congestión en la red, TCP utilizará dos tipos de indicios en el emisor. Uno de carácter negativo y otro positivo:

- Como indicio negativo utilizará la retransmisión de segmentos de datos. Recordemos que la retransmisión de un segmento de datos se puede producir por dos motivos, porque no se recibe el reconocimiento del segmento a tiempo, antes de que venza su temporizador de retransmisión, o bien porque se reciben tres reconocimientos duplicados, que reconocen el último segmento recibido correctamente y en orden. La realidad es que cuando vence un temporizador de retransmisión puede haberse perdido o retrasado el propio segmento o bien su reconocimiento asociado (Figura 1), pero esta distinción no tiene importancia para TCP. En cualquier caso, la red tiene problemas de tráfico y TCP reaccionará reduciendo la tasa de envío de segmentos de datos.

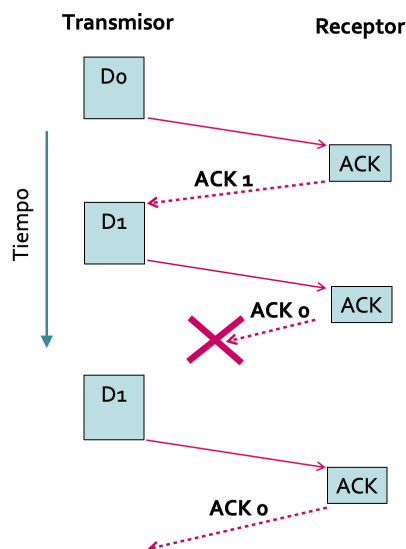


Figura 1. Retransmisión de un segmento TCP debida a la pérdida de su reconocimiento.

- Por el contrario, la recepción de un reconocimiento que reconoce datos nuevos, no reconocidos previamente, se interpretará como un signo positivo que permite incrementar la tasa de envío. Al menos en cuanto a control de la congestión se refiere, ya que habrá que tener en cuenta también si el control de flujo del receptor lo permite.

Analicemos con más detalle el indicio negativo. TCP supone que la retransmisión de un segmento está causada siempre por un problema de congestión, que ha provocado la pérdida del segmento o bien de su reconocimiento asociado. Se trata de una asunción bastante realista, ya que las pérdidas debidas a errores de transmisión son escasas en enlaces cableados. Mientras que en los enlaces inalámbricos, aunque son más frecuentes suelen ser solucionadas por el protocolo de nivel de enlace de datos. Por lo tanto, TCP puede asumir sin faltar demasiado a la realidad que si hay que volver transmitir un segmento de datos ya enviado previamente, se debe a que el segmento o su reconocimiento asociado han sido descartados por un router cuya cola ha sobrepasado el nivel máximo debido al exceso de paquetes recibidos.

Ahora bien, un segmento se puede retransmitir por dos motivos distintos:

- Que venza el temporizador de retransmisión asociado al segmento.
- O que se reciban tres reconocimientos duplicados. Se entiende por **reconocimiento duplicado** un reconocimiento que no lleva datos, no tiene activados los bits de SYN ni de FIN, y reconoce el último segmento de datos ya reconocido previamente, aunque el receptor del reconocimiento sigue teniendo aún otros datos pendientes de reconocerse. Es decir, deben recibirse, cuatro segmentos consecutivos en total, **que no lleven datos** y con el mismo valor en el campo de reconocimiento. El hecho de que no lleven datos implica que el otro extremo los ha enviado expresamente para avisar de un problema de recepción de segmentos desordenados.

Piensa un momento si las dos situaciones tienen la misma gravedad a efectos de congestión. Las dos imágenes mostradas en la Figura 2 pueden servirte de ayuda. Para ahorrar espacio y simplificar la situación hemos evitado utilizar reconocimientos retrasados en los dos segmentos iniciales que se han recibido en orden. En el caso de los segmentos recibidos fuera de orden, los reconocimientos **no** pueden retrasarse para avisar al otro extremo del problema con la mayor brevedad.

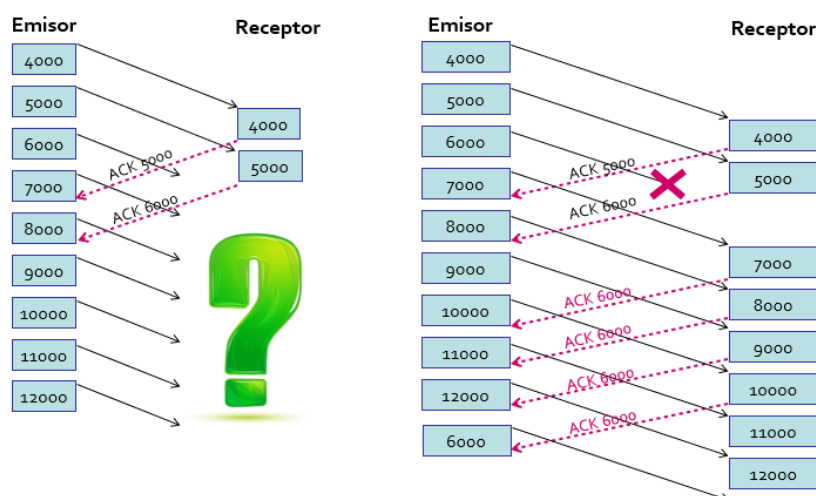


Figura 2. Ejemplos de pérdida de segmentos de datos en TCP.



En la imagen de la izquierda el emisor ha enviado una tanda de nueve segmentos de datos, ha recibido el reconocimiento de los dos primeros y después no ha llegado ningún reconocimiento más, no ha obtenido ningún tipo de realimentación. En la imagen de la derecha, los dos primeros segmentos también se han recibido correctamente, y el emisor ha recibido los reconocimientos de esos segmentos. Sin embargo, el cuarto segmento, que lleva el número de secuencia 6000, se ha perdido, posiblemente descartado por un router. Esto provoca que los segmentos siguientes, aunque han llegado correctamente al receptor, estén desordenados y no puedan reconocerse. Tampoco se podrán pasar los datos de esos segmentos a la capa de aplicación. Hay que tener en cuenta que los reconocimientos en TCP son globales, es decir, al reconocer un segmento se reconocen también implícitamente los que tienen números de secuencia anteriores a él. Por este motivo, a partir del segmento 7000 el receptor envía un reconocimiento duplicado por cada segmento que recibe con número de secuencia superior al 6000, que es el segmento que el receptor está esperando. Es su forma de avisar al emisor de que le falta recibir el segmento 6000. Tras recibir el tercer reconocimiento duplicado, el emisor retransmitirá el segmento 6000 sin esperar a que venza su temporizador de retransmisión (mecanismo de retransmisión rápida).

Ahora piensa que situación parece más grave, en cuál de los dos casos parece que la red esté más congestionada. Cuando se han enviado una tanda de segmentos y vence el temporizador asociado a uno de ellos o cuando se reciben tres reconocimientos duplicados indicando que han llegado tres segmentos fuera de orden porque falta alguno de los segmentos anteriores (o varios).

Como habrás concluido, el vencimiento del temporizador se considera un evento más grave porque, o bien, todos los segmentos enviados han sufrido incrementos muy grandes en la latencia esperada, posiblemente debido a la congestión de la red, o bien, han sido descartados en un router que está recibiendo más tráfico del que puede transmitir. Por el contrario, si se reciben duplicados parece indicar que la red ha descartado algún segmento, pero ha sido capaz de entregar los restantes a su destino. Por ello, las medidas de reducción que se apliquen serán más agresivas cuando venza un temporizador de retransmisión.

4.2 Reducción de la tasa de tasa de envío de datos

Una vez hemos visto cómo detectar que existe congestión en la red, pasemos a analizar qué medidas se aplican para intentar paliarla. Se actuará sobre las mismas variables TCP que emplean los algoritmos de arranque lento y evitación de la congestión: la ventana de congestión y el umbral. Recuerda que la **ventana de congestión** es una variable que utiliza el emisor y que se actualiza en base a la información local de congestión. La ventana de congestión muy importante porque limita el tamaño máximo que puede tener en un instante dado la ventana de transmisión del emisor. Este tamaño máximo vendrá dado por el mínimo entre la ventana de recepción que anuncia el otro extremo de la conexión TCP, y que envía en la cabecera de cada segmento TCP, y la ventana de congestión. Si llamamos E al emisor y R al receptor, la ventana de congestión del emisor vendrá dada por la expresión siguiente (Ecuación 1):

$$V_{\text{TransmisiónMáx}E} = \min (V_{\text{Congestión}E}, V_{\text{Recepción}R})$$

Ecuación 1. Cálculo de la ventana de transmisión máxima del emisor E.

La Figura 3 muestra un ejemplo de cómo la ventana de congestión del emisor puede establecer una restricción adicional sobre el tamaño del envío establecido por el receptor.

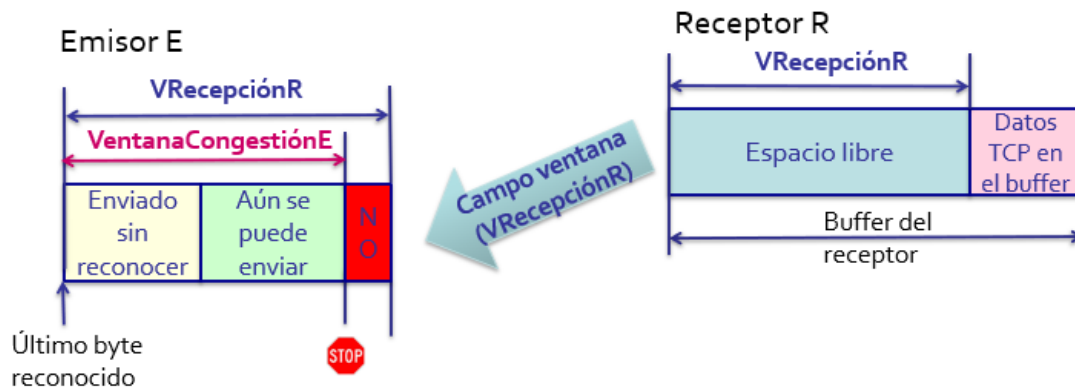


Figura 3. Limitación de la capacidad de transmisión debido a la ventana de congestión.

Aunque las ventana de congestión y de transmisión se gestionan en bytes, realizaremos la explicación siguiente en segmentos para facilitar la comprensión del funcionamiento. Para traducir los valores a bytes sería suficiente multiplicar por el valor del MSS (*Maximum Segment Size*).

El **umbral** se inicializa al comienzo de la conexión TCP con un valor muy grande (por ejemplo, el máximo valor permitido para la ventana de recepción). Después debe reducirse (o ampliarse) sólo como respuesta a los eventos de congestión.

Independientemente de cómo se haya detectado la pérdida del segmento, por el vencimiento de un temporizador o por la recepción de tres reconocimientos duplicados, si es la primera vez que el segmento va a retransmitirse, se tomarán medidas para reducir la inyección de tráfico a la red. En primer lugar, se reducirá el umbral a la mitad del valor de la ventana de transmisión en ese momento. Además, el valor mínimo para el umbral son dos segmentos (Ecuación 2).

$$Umbral = \max\left(\frac{VentanaTransmisión}{2}, 2 \text{ segmentos}\right)$$

Ecuación 2. Modificación de la variable umbral tras la primera retransmisión de un segmento TCP.

Una vez modificado el umbral se actualiza la ventana de congestión, distinguiendo los dos casos que hemos comentado para intentar adaptarse a la gravedad de la situación. Si se han recibido tres reconocimientos duplicados, el caso considerado más leve, la ventana de congestión se igualará al nuevo valor del umbral (Ecuación 3), y a partir de aquí cuando vuelvan a recibirse nuevos reconocimientos se aplicará el algoritmo de evitación de la congestión. Esto provocará un incremento de la ventana de congestión que como máximo será lineal.

$$VentanaCongestión = Umbral$$

Ecuación 3. Modificación de la ventana de congestión tras la recepción de tres reconocimientos duplicados.

Sin embargo, en el caso de vencimiento de un temporizador de retransmisión (lo peor que puede pasar en términos de congestión), la venta de congestión pasará a valer un segmento (Ecuación 4). Es decir, sólo se podrá retransmitir el segmento cuyo temporizador acaba de vencer. Además, cuando se reciban reconocimientos que reconozcan nuevos datos, se

aplicará el algoritmo de arranque lento hasta que la ventana de congestión alcance el valor del umbral. Después de alcanzar el umbral, pasará a aplicarse el algoritmo de evitación de la congestión.

$$\text{VentanaCongestión} = 1 \text{ segmento}$$

Ecuación 4. Modificación de la ventana de congestión tras el vencimiento de un temporizador de retransmisión.

Por lo tanto, una consecuencia importante de la modificación del umbral es que determina el tipo de algoritmo de recuperación que se utilizará para ampliar la ventana de congestión cuando vuelvan a recibirse nuevos reconocimientos: arranque lento o evitación de la congestión.

De esta forma TCP intentará ajustarse a la capacidad de la red disponible en cada momento reduciendo el envío de datos cuando detecta señales de posible congestión e incrementándolo cuando las condiciones de la red mejoran (Figura 4).

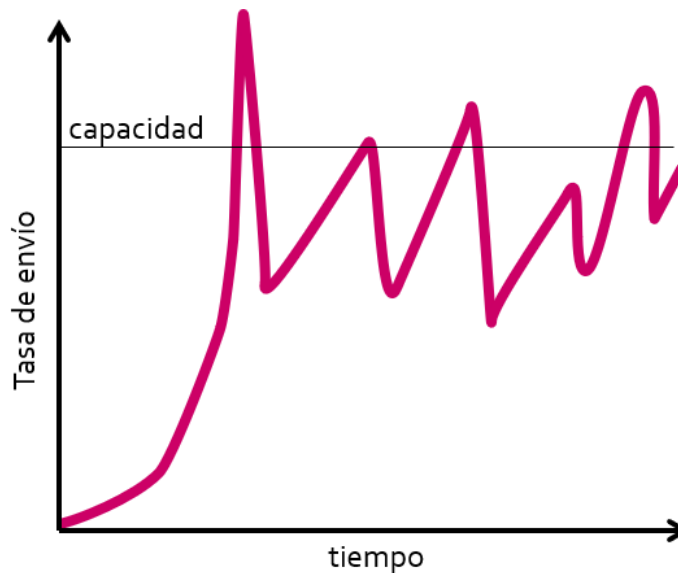


Figura 4. Adaptación de TCP a la capacidad de transmisión de la red.

5 Resumen

Para acabar recordemos que TCP interpreta que la red está congestionada cuando tiene que retransmitir un segmento de datos.

Al detectar congestión, TCP reducirá la capacidad de transmisión del emisor modificando el valor de dos variables: la ventana de congestión y el umbral. El valor de la variable umbral siempre se reduce a la mitad independientemente del evento que haya causado la retransmisión del segmento.

Sin embargo, la reducción de la ventana de congestión es distinta si el segmento se ha retransmitido por la recepción de tres reconocimientos duplicados o por el vencimiento de un temporizador de retransmisión.



6 Bibliografía

Peterson, L. L. and Davie, S. D.: "Computer Networks. A System Approach", en Ed. Elsevier, 6ª edición, 2022, págs. 481-493.

Allman, M.; Paxson, V.; Blanton, E. "RFC 5681. Tcp Congestion Control", 2009. Disponible en <https://www.rfc-editor.org/rfc/rfc5681>.