

ANÁLISIS DEL ALGORITMO RED CON LA TEORÍA DE CONTROL MODERNA

CLAUDIO PARRA

Universidad Central de Venezuela. Facultad de Ingeniería. Escuela de Ingeniería Eléctrica.
email:claudio942002@yahoo.com

Recibido: septiembre de 2009

Recibido en forma final revisado: enero de 2010

RESUMEN

En este trabajo se presenta un estudio de la estabilidad del sistema formado por el algoritmo de detección temprana aleatoria (RED) en conjunto con el protocolo de control de transmisión (TCP) en función de los parámetros del sistema utilizando la teoría de control moderna. Se deduce la ecuación linealizada del sistema mediante la cual se hace un estudio de la estabilidad en función de la posición de los polos de la matriz del sistema linealizado. Se observa que a medida que crece la cantidad N de conexiones TCP, disminuye la estabilidad. Mediante simulaciones, se comprobaron los resultados obtenidos del análisis teórico. Los experimentos incluyeron diferentes condiciones de operación del sistema con varios valores de máxima probabilidad de marcado de los paquetes, observándose que para valores grandes de probabilidad, la estabilidad relativa también disminuye.

Palabras clave: TCP, RED, Congestión, Simulación, Matriz de estado.

ANALYSIS OF THE RED ALGORITHM WITH MODERN CONTROL THEORY

ABSTRACT

This paper presents a study of the stability of the system of the RED (Random Early Detection) algorithm in conjunction with TCP (Transmission Control Protocol) based on the parameters of the system using modern control theory. We deduce the linearized equation of the system through which a study of stability depending on the position of the poles of the linearized system matrix is made. It was noted that as the number (N) of TCP connections grows, stability reduces. Through simulations, the expected results were verified by theoretical analysis. The experiments include different system operating conditions with various values of highest probability of marking packets, noting that for high probability values, the relative stability also decreases.

Keywords: TCP, RED, Congestion, Simulation, State matrix.

INTRODUCCIÓN

El control de congestión en la redes de transmisión de datos requiere de una especial atención debido al crecimiento de demandas de recursos. Se han escrito diferentes trabajos sobre el algoritmo RED y se ha estudiado brevemente el problema del modelaje de la congestión de una red a la luz de la teoría de control clásico. En este trabajo se describe el algoritmo de detección temprana aleatoria como punto de partida en la aplicación de la teoría de control moderna, para estudiar el comportamiento según la variación de la probabilidad de descarte del mismo algoritmo. Para esto se deducen las ecuaciones del sistema basadas en el algoritmo

de detección temprana aleatoria y el protocolo de control de transmisión, y luego se derivan las ecuaciones promedio de transición de estado. Luego se genera la matriz de la ecuación de estado linealizada, la cual será utilizada para los experimentos. Dada la complejidad del estudio y su longitud, sólo se estudio la variación de la probabilidad de descarte y el número de fuentes TCP. Los resultados obtenidos concuerdan con los esperados por la teoría.

MARCO TEÓRICO

Uno de los mecanismos más importantes y más utilizado para lograr el control de congestión en redes IP, es el algo-

ritmo RED. La idea de usar un servidor con el algoritmo RED programado, es para mantener un promedio de longitud de la cola de paquetes baja (esto se observa en la memoria buffer a la entrada del servidor). El algoritmo RED descarta aleatoriamente los paquetes de entrada con una probabilidad proporcional al promedio de la longitud de cola. Según los creadores del algoritmo RED (Floyd *et al.* 1993) éste tiene ventajas sobre otros métodos como Drop Tail, las cuales podemos resumir así:

1. El promedio de la longitud de cola es mantenido bajo, por lo que el retardo de extremo a extremo de una conexión TCP es bajo también.
2. RED no tiene desventajas para el tráfico en ráfagas.
3. La sincronización global de conexiones TCP se toma en cuenta.

El modelo analítico (Comer, 1997; McCabe, 2003) de una red de datos se muestra en la figura 1:

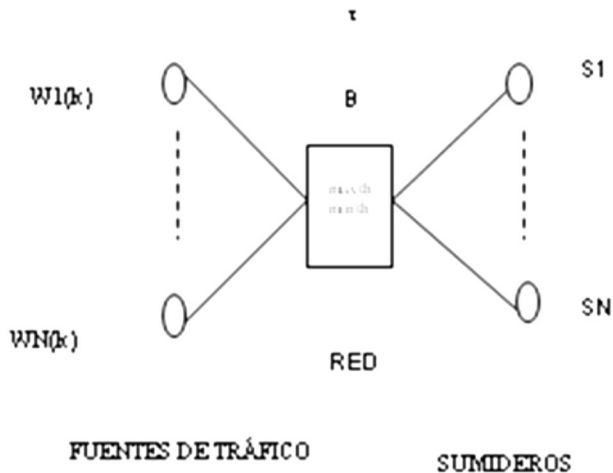


Figura 1. Modelo analítico.

donde:

N número de conexiones TCP, τ la latencia, B la velocidad de procesamiento, $W_n(k)$ el tamaño de la ventana del host n (fuente) y la ranura k (una ranura (slot) es el tiempo RTT (tiempo de ida y vuelta) de la conexión TCP. La longitud de la ranura cambia constantemente debido a los retardos de la cola en el servidor que ejecuta el algoritmo RED), S1-SN son los sumideros, y TCP el protocolo de control de transmisión.

Una dificultad importante para la aplicación del algoritmo RED consiste en que sus parámetros son difíciles de ajustar en forma óptima en las redes reales (Christiansen *et al.* 2001). Cada vez que un paquete nuevo llega a la cola del

enrutador (servidor con RED), se estima el tamaño promedio de la cola usando un proceso EWMA (Promedio de movimiento de peso exponencial), utilizando la siguiente ecuación:

$$\text{avg}(\bar{q}) = (1-w_q) * \text{avg} + w_q * q \quad (1)$$

donde:

avg es la longitud promedio de cola estimada, q la longitud instantánea de cola, w_q el factor de peso, y \bar{q} la longitud promedio de cola.

Este valor de avg es comparado por dos umbrales min_{th} y max_{th} (figura 2).

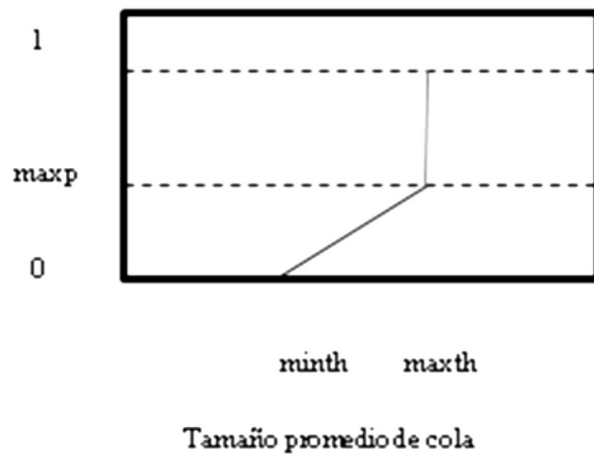


Figura 2. Algoritmo RED.

Si el valor promedio es menor que min_{th} , entonces no pasa nada, pero si es más grande que max_{th} el paquete se desecha. Los valores intermedios son marcados con un valor entre 0 y max_p , donde max_p es un valor probabilístico que se asocia a los paquetes marcados y es directamente proporcional al promedio de la longitud de la cola.

Para nuestro estudio necesitamos definir un modelo analítico donde se tiene que:

1. $q(k)$ es el valor actual de la cola (medido en paquetes).
2. $\bar{q}(k)$ es el tamaño de cola promedio en la ranura k (medido en paquetes).

Se asume que q y \bar{q} no cambian durante cada ranura. Esto ha sido confirmado para valores pequeños de w_q ya que se usa un filtro pasa bajos para calcular $\bar{q}(k)$ (ecuación (1)) (Floyd *et al.* 1993).

El servidor que ejecuta el algoritmo RED (servidor RED),

calcula una probabilidad de marcado de paquetes p_b usando el promedio de longitud de cola para los paquetes que llegan. A continuación se muestra el mecanismo de marcado:

$$p_b \text{ es } 0 \text{ si } \bar{q}(k) < \min_{th} \quad (2)$$

$$\text{es } 1 \text{ si } \bar{q}(k) = \delta > \max_{th}, \quad (3)$$

en otro caso:

$$p_b = \max_p(\bar{q} - \min_{th}) / (\max_{th} - \min_{th}) \quad (4)$$

Luego, el servidor RED aleatoriamente descarta cada paquete de entrada con probabilidad p_a , según la siguiente ecuación:

$$p_a = p_b / (1 - \text{contador} * p_b) \quad (5)$$

El contador indica el número de paquetes no marcados que han llegado desde el último paquete marcado. El mecanismo de marcado de paquetes no es por flujo sino por el número de paquetes. Para encontrar más información sobre RED se puede consultar (Floyd *et al.* 1993), donde ellos recomiendan valores para el uso del algoritmo RED (tabla 1).

Tabla 1. Valores recomendados para el algoritmo RED.

Minth	5,00
Maxth	15,00
maxp	0,1
Wq	0,002

MARCO METODOLÓGICO

Análisis del estado estable

Para deducir las ecuaciones del sistema que predicen el comportamiento del estado estable, se necesita hallar las ecuaciones de transición. Estas ecuaciones se derivan del control de congestión en TCP y la que rige al algoritmo RED (Floyd *et al.* 1993).

Para esto se determina la probabilidad de desechar un paquete en la ranura k . En la primera ranura k el transmisor n envía $w_n(k)$ paquetes hacia el sumidero. Como ya sabemos, el servidor RED marca cada paquete basado en el promedio de la longitud de la cola (p_b). Se asume que \bar{q} es fijo en una ranura por lo que p_b es fijo también. Donde p_b se determina por:

$$p_b(k) = \max_p(\bar{q}(k) - \min_{th}) / (\max_{th} - \min_{th}) \quad (6)$$

Según el algoritmo RED, la probabilidad de marcado definitiva es:

$$p_a(k) = p_b(k) / (1 - \text{contador} * p_b(k)) \quad (7)$$

Luego definimos \bar{X} como el número de paquetes no marcados entre dos consecutivos marcados. \bar{X} es una variable aleatoria uniformemente distribuida: P_k que toma valores $\{1, 2, \dots, 1/p_b(k)\}$ y es:

$$P_k[X = n] = \{p_b(k) \text{ si } 1 \leq n \leq 1/p_b(k); 0 \text{ en otro caso}\} \quad (8)$$

donde:

\bar{X} representa el número de paquetes no marcados entre dos marcados consecutivamente en la ranura k .

Entonces:

$$\bar{X}_k = \sum_{n=1}^{\infty} n P_k[X = n] = (1/p_b(k) + 1) / 2 \quad (9)$$

Para este trabajo usaremos la versión de TCP básica, en la que si al menos un paquete es marcado con una probabilidad, el ancho de la ventana en la ranura $k+1$ será:

$$w(k+1) = \{w(k)/2 \text{ con probabilidad } \bar{p}, w(k)+1 \text{ en otro caso}\}$$

Y \bar{p} es:

$$\bar{p} \text{ es el mínimo del intervalo } [(w(k)/(1/p_b(k)), 1]$$

Ahora $q(k+1)$ es igual al estado actual $q(k)$ más los anchos de la ventana de todas las conexiones TCP. Si se incluye la velocidad de procesamiento B y la latencia τ , se tiene:

$$\begin{aligned} q(k+1) &= q(k) + \sum_{n=1}^N w_n(k) - B(\tau + q(k)/B) = \\ &= \sum_{n=1}^N w_n(k) - B\tau \end{aligned} \quad (10)$$

Tomando en cuenta las siguientes condiciones sobre el algoritmo RED:

1. El servidor RED actualiza el promedio de longitud de cola con cada paquete que llega.
2. Esta actualización es $\alpha = \sum_{n=1}^N w_n(k)$ veces en la ranura k .
3. $q(k)$ es fijo en cada ranura

Luego podemos decir que la relación entre $\bar{q}(k+1)$ y $\bar{q}(k)$ está dada por la siguiente ecuación:

$$\bar{q}(k+1) = (1 - w_q)^{\alpha} \bar{q}(k) + \{1 - (1 - w_q)^{\alpha}\} q(k) \quad (11)$$

Derivación de las ecuaciones promedio de transición de estado

Ohsaki *et al.* (2002) en su estudio introducen la idea de tratar una red de transmisión de datos como un sistema en tiempo discreto en donde se manejan secuencias de datos y a cada secuencia corresponden varias ranuras (ésta es una visión más amplia que la de considerar solamente ranuras individuales). Se define \bar{s}_k como el número promedio de ranuras en una secuencia determinada, que comienza en la k -ésima ranura.

Para efectos de este trabajo utilizamos las 3 ecuaciones (13), (14) y (15), propuestas por Ohsaki *et al.* (2002), para determinar los valores de equilibrio de un sistema de transmisión de datos. Estas ecuaciones se basan en el algoritmo RED y en el protocolo TCP, y el análisis de ranuras para este proceso discreto. Asumiendo que \bar{s}_k es el número de ranuras en una secuencia que guarda la siguiente relación con la ventana $w(k)$:

$$\bar{s}_k = 0.5 - w(k) + (\sqrt{N^2(1 - 2w(k))^2 + 8NX_k}) / 2N \quad (12)$$

donde:

N es el número de fuentes TCP

Las 3 ecuaciones de $w(k + \bar{s}_k)$, $q(k + \bar{s}_k)$ y $\bar{q}(k + \bar{s}_k)$ describen las ecuaciones de transición de la ranura k a $k + \bar{s}_k$, y son:

$$w(k + \bar{s}_k) = (w(k) + \bar{s}_k - 1) / 2 \quad (13)$$

$$q(k + \bar{s}_k) = N(w(k) + \bar{s}_k - 1) - B\tau \quad (14)$$

$$\bar{q}(k + \bar{s}_k) = (1 - w_q) \bar{X}_k \bar{q}(k) + \{1 - (1 - w_q) \bar{X}_k\} q(k) \quad (15)$$

Donde estas ecuaciones de transición describen el comportamiento promedio del tamaño de la ventana, la longitud de la cola actual y el promedio de la longitud de la cola.

Cuando los tamaños de las ventanas de los hosts son idénticos podemos decir que:

$$\bar{X}_k = N \{ \bar{s}_k w(k) + \bar{s}_k (\bar{s}_k - 1) / 2 \} \quad (16)$$

En el estado estable se pueden obtener las ecuaciones de equilibrio según las siguientes igualdades. (Para más detalles véase Ohsaki *et al.* 2002):

$$w(k) = w(k + \bar{s}_k) \quad (17)$$

$$q(k) = q(k + \bar{s}_k) \quad (18)$$

$$\bar{q}(k) = \bar{q}(k + \bar{s}_k) \quad (19)$$

resultando:

$$w^* = \frac{\sqrt{0.25 + (1/3N) * ((\max_{th} - \min_{th}) / (\max p(\bar{q}^* - \min_{th}) + 1))}}{\quad} \quad (20)$$

$$q^* = Nw^* - B\tau \quad (21)$$

$$\bar{q}^* = q^* \quad (22)$$

donde:

w^* representa el mínimo valor esperado de la ventana al comienzo de la secuencia, para obtener el promedio del tamaño de la ventana consultar en Ohsaki *et al.* (2002).

Análisis de estabilidad y la teoría de control moderno

En esta parte se estudia la estabilidad del sistema (servidor RED). El sistema en cuestión es no lineal, por lo que se linealiza y se estudia su comportamiento alrededor de un punto de equilibrio. Las ecuaciones (13), (14) y (15) definen el comportamiento del sistema TCP y RED.

Dado un vector de estado $x(k)$ tenemos que $\delta x(k)$ es la diferencia entre el vector en un instante k y el punto de equilibrio:

$$\delta x(k) = [w(k) - w^*, q(k) - q^*, \bar{q}(k) - \bar{q}^*]^T \quad (23)$$

Linealizando alrededor de (*) $w(k)$, $q(k)$ y $\bar{q}(k)$ de las ecuaciones (13), (14) y (15), obtenemos:

$$\delta x(k + \bar{s}) = A \delta x(k) \quad (24)$$

donde:

A es la matriz de la ecuación de estado linealizada:

$$A = \begin{pmatrix} \partial w(k) / \partial w & \partial w(k) / \partial q & \partial w(k) / \partial \bar{q} \\ \partial q(k) / \partial w & \partial q(k) / \partial q & \partial q(k) / \partial \bar{q} \\ \partial \bar{q}(k) / \partial w & \partial \bar{q}(k) / \partial q & \partial \bar{q}(k) / \partial \bar{q} \end{pmatrix} / * \quad (25)$$

donde para nuestro caso específico se obtiene:

$$A = \begin{pmatrix} f(N, w, \bar{X}_k) & 0 & f(N, w, \bar{X}_k) \\ f(N, w, \bar{X}_k) & 0 & f(N, w, \bar{X}_k) \\ 0 & 1 - (1 - w_q)^{\bar{X}_k} & (1 - w_q)^{\bar{X}_k} \end{pmatrix} / * \quad (26)$$

donde:

$$\frac{\partial w(k)}{\partial w} = f(N, w, \bar{X}_k) = \frac{((-1)N(2w(k)-1) / \sqrt{0.5*(N^2(1-2w(k))^2 + 8N\bar{X}_k)})*0.5}{(27)}$$

$$\frac{\partial w(k)}{\partial w} = f(q, N, w, \bar{X}_k) = \frac{-1/((q(k) - 5)^2 \sqrt{N^2(1 - 2w(k))^2 + 8N\bar{X}_k})}{(28)}$$

$$\frac{\partial q(k)}{\partial w} = f(N, w, \bar{X}_k) = \frac{(N^2(2w(k)-1) / \sqrt{0.5*(N^2(1-2w(k))^2 + 8N\bar{X}_k)})}{(29)}$$

$$\frac{\partial q(k)}{\partial w} = f(q, N, w, \bar{X}_k) = \frac{-N/((q(k) - 5)^2 \sqrt{N^2(1 - 2w(k))^2 + 8N\bar{X}_k})}{(30)}$$

TÉCNICAS EXPERIMENTALES

El propósito del trabajo es estudiar la estabilidad del algoritmo RED y el mecanismo de congestión más simple que usa TCP. Para esto fijamos las condiciones de operación en un punto estable. Además, se determinó la observabilidad y la controlabilidad del sistema TCP-RED.

Una vez determinado el punto de equilibrio (20), (21) y (22) se evaluó la matriz de estado en el punto estable:

$$A^* = f(w^*, q^*, q^*) \quad (31)$$

Luego, se tomaron los valores iniciales de las gráficas de los resultados de Ohsaki *et al.* (2002) para evaluar las ecuaciones (20), (21) y (22), obteniéndose resultados de q y w que permitiesen determinar la Matriz de Estado y su polinomio característico. Se corrieron varios casos con diferentes maxp. También se obtuvo la Matriz para otros casos de B y τ , lo cual es propósito de una segunda parte de este trabajo.

Se usaron los valores de la tabla 1 para la consideración de los parámetros del algoritmo RED. Se observó la estabilidad del sistema en cada caso mediante la magnitud de los polos de la Matriz A. Se usaron diferentes valores de la probabilidad de descarte.

RESULTADOS Y DISCUSIÓN

En la tabla 2 se resumen los casos estudiados mediante simulación. Cada uno de ellos comprende fuentes de tráfico en 1 y 20 (N=1,2,4,8,10,15 y 20).

Tabla 2. Valores iniciales de los experimentos y simulaciones.

N (1 a 20)	w supuesto	q supuesto	w cal-sup	maxp	min	max	q calculado	B
1	9.45	8.2	10.2347447	0.01	5	15	8.23474475	2
2	5.4	9.8	5.92780641	0.01	5	15	9.85561283	2
4	3.1	11.45	3.64049859	0.01	5	15	12.5619944	2
8	1.9	13.2	2.31796624	0.01	5	15	16.5437299	2
10	1.6	14.2	1.97649228	0.01	5	15	17.7649228	2
15	1.1	15.2	1.56552594	0.01	5	15	21.4828891	2
20	1	15.8	1.34531652	0.01	5	15	24.9063304	2
N(1 a 20)	w supuesto	q supuesto	w cal-sup	maxp	min	max	q calculado	B
1	7.2	6	8.20060973	0.05	5	15	6.20060973	2
2	3.9	6.65	4.54078043	0.05	5	15	7.08156085	2
4	2.1	7.45	2.67133944	0.05	5	15	8.68535777	2
8	1.2	8.2	1.70171482	0.05	5	15	11.6137186	2
10	1	8.6	1.46122729	0.05	5	15	12.6122729	2
15	0.6	9	1.17615192	0.05	5	15	15.6422788	2
20	0.4	9.4	1.01204863	0.05	5	15	18.2409725	2
N (1 a 20)	w supuesto	q supuesto	w cal-sup	maxp	min	max	q calculado	B
1	6.6	5.4	9.16060406	0.1	5	15	7.16060406	2
2	3.6	6	4.13319892	0.1	5	15	6.26639785	2
4	1.9	6.4	2.50713268	0.1	5	15	8.02853073	2
8	1	7	1.5411035	0.1	5	15	10.328828	2
10	0.6	7.2	1.341076	0.1	5	15	11.41076	2
15	0.35	7.4	1.09459954	0.1	5	15	14.4189931	2
20	0.3	7.8	0.92838826	0.1	5	15	16.5677652	2

Tabla 2 (cont.). Valores iniciales de los experimentos y simulaciones.

N (1 a 20)	w supuesto	q supuesto	w_cal-sup	maxp	min	max	q calculado	B
1	6.4	5.2	9.16060406	0.2	5	15	7.16060406	2
2	3.2	5.6	3.78226857	0.2	5	15	5.56453715	2
4	1.85	5.65	2.59684226	0.2	5	15	8.38736906	2
8	0.9	6.2	1.42400062	0.2	5	15	9.39200499	2
10	0.6	6.6	1.15108644	0.2	5	15	9.51086443	2
15	0.3	6.65	0.97243143	0.2	5	15	12.5864715	2
20	0.2	6.8	0.8541836	0.2	5	15	15.0836721	2
N (1 a 20)	w supuesto	q supuesto	w_cal-sup	maxp	min	max	q calculado	B
1	6.2	5.15	7.49258893	0.4	5	15	5.49258893	2
2	3	5.25	4.13319892	0.4	5	15	6.26639785	2
4	1.6	5.65	1.88107989	0.4	5	15	5.52431955	2
8	0.8	5.8	1.26243812	0.4	5	15	8.09950494	2
10	0.45	5.85	1.12415546	0.4	5	15	9.24155456	2
15	0.25	6.6	0.78704793	0.4	5	15	9.80571895	2
20	0.15	6.7	0.71537732	0.4	5	15	12.3075463	2

El objetivo final fue determinar la estabilidad del sistema evaluando el polinomio característico de la matriz A. Esta es generada para los casos indicados en la tabla 2. Graficando la magnitud de los polos de los casos mencionados, se puede observar cómo varía la estabilidad relativa del sistema TCP-RED (figuras 3,4,5,6 y 7).

En la figura 3 con $\text{max}_p=0.01$, la magnitud de uno de los polos tiende a 1, pero se mantiene la estabilidad para todos los valores de N.

En la figura 4 con $\text{max}_p=0.05$, la magnitud de uno de los polos tiende a 1 más rápidamente que el caso anterior.

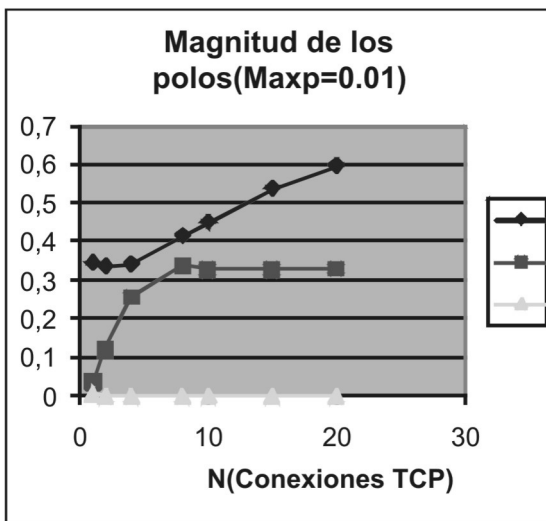


Figura 3. Magnitud de los polos 1,2 y 3 con $\text{max}_p = 0.01$.

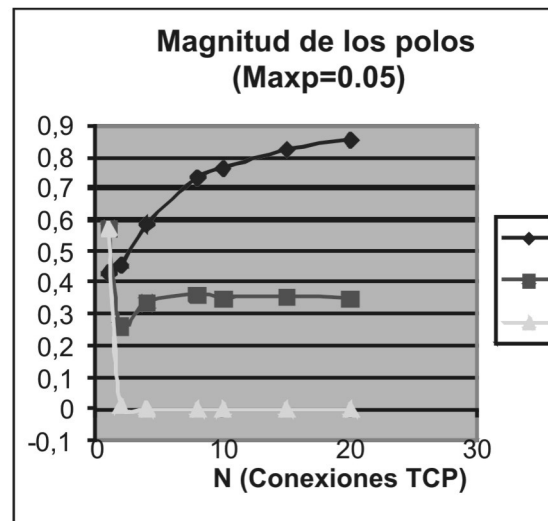


Figura 4. Magnitud de los polos 1,2 y 3 con $\text{max}_p = 0.05$.

En la figura 5 con $\text{max}_p=0.1$, la magnitud de uno de los polos tiende a 1 más rápidamente que los dos casos anteriores.

En la figura 6 con $\text{max}_p=0.2$, uno de los polos está muy cercano a 1 en todos los casos de N.

En la figura 7 con $\text{max}_p=0.4$ el sistema es totalmente inestable.

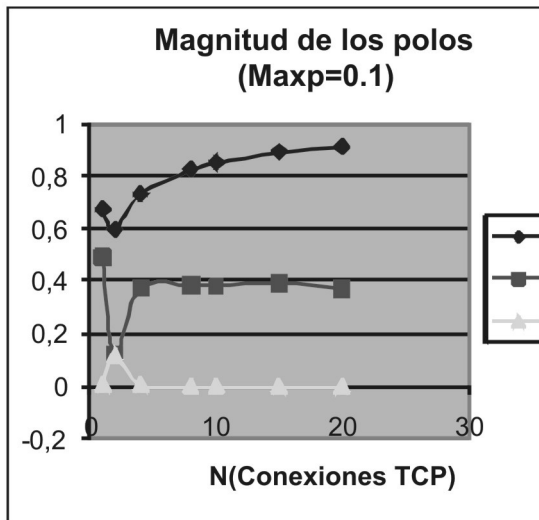


Figura 5. Magnitud de los polos 1,2 y 3 con $\max_p = 0.1$.

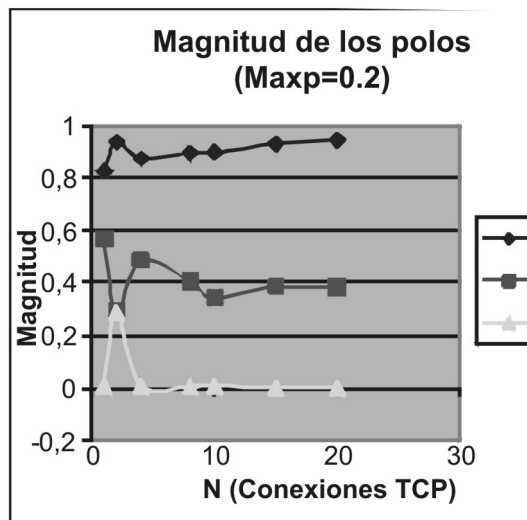


Figura 6. Magnitud de los polos 1,2 y 3 con $\max_p = 0.2$.

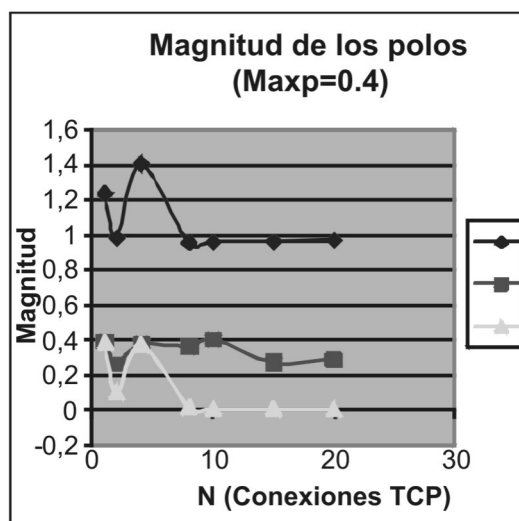


Figura 7. Magnitud de los polos 1,2 y 3 con $\max_p = 0.4$.

En general, se observó en las figuras anteriores que aumentar la probabilidad de marcado de los paquetes tiene un efecto de hacer inestable el sistema, lo cual confirma los resultados de Floyd *et al.* (1993). Floyd (2003) recomienda usar una probabilidad baja para que los paquetes no sean descartados con alta probabilidad, ya que con alta probabilidad se crea una necesidad de retransmisión de los paquetes que conduce a la congestión de sistema. También se observó que a medida que crece N (Numero de Conexiones TCP) el módulo de un polo (dominante) tiende a 1, lo que significa que la estabilidad relativa disminuye.

CONCLUSIONES

Después del estudio teórico y su verificación con casos prácticos, podemos concluir lo siguiente:

Se observó que a medida que crece N, la magnitud de los polos tiende a 1, por lo que se concluye que el sistema pierde estabilidad relativa. Este resultado es de esperarse, ya que a medida que N aumenta, crece el número de conexiones TCP y aumenta el tráfico que maneja la red, lo que conduce a congestión usando TCP y RED, es decir el sistema se hace ineficiente para manejar el tráfico que circula por la red. Este es el principal problema de las redes de datos que usan el algoritmo RED y cursan tráfico TCP.

Los resultados son cónsonos con la propuesta teórica, permitiendo así un análisis amplio y válido del sistema TCP-RED. Queda abierta la posibilidad de extender el estudio mediante la variación de los parámetros del algoritmo RED y el uso de otras ecuaciones de TCP. Como alternativas podemos recomendar el trabajo con nuevas versiones de TCP denominadas Illinois y África, así como nuevas versiones de RED, encontradas ambas en Internet.

No es posible usar una probabilidad alta para el descarte de paquetes, ya que en la simulación resultó que el sistema se comporta de forma inestable (módulo de los polos mayor que 1). Este resultado se refuerza con la variación de N.

El estudio abre el camino para otros trabajos que permitan ver la estabilidad del sistema con otros escenarios.

REFERENCIAS

CHRISTIANSEN, M., JEFFAY, K., OTT, D., DONELSON, S. (2001). Tuning RED for Web Traffic. Recuperado el 20 de Septiembre de 2009, de www.icir.org/floyd/papers/TCPPreport.pdf.

COMER, D. (1997). Redes de Computadores Internet e Intranets. Prentice Hall. p. 189.

- DE CNOODER, S., ELLOUMI, O., PAUWELS, K. Effect of different packet sizes on RED performance Traffic and Routing Technologies project. Recuperado el 20 de Septiembre de 2009, de <http://arxiv.org/pdf/cs/0001005/>.
- DORF, R. & BISHOP, R. (2005). *Sistemas de Control Moderno*. Pearson Prentice Hall. p. 386.
- FLOYD, S. & FALL, K. (1999). IEEE Promoting the Use of End-to-End Congestion Control in the Internet. *ACM Transactions on Networking*. Recuperado el 20 de Septiembre de 2009, de www.aciri.org/floyd/papers/collapse.may99.pdf/.
- FLOYD, S. (2000). A Report on Some Recent Developments in TCP Congestion Control. Recuperado el 20 de Septiembre de 2009, de www.aciri.org/floyd/papers/report_Jan01.pdf/.
- FLOYD, S., GUMMADI, R., SHENKER, S., (2001). Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management. Recuperado el 20 de Septiembre de 2009, de www.aciri.org/floyd/papers/adaptiveRed.pdf/.
- FLOYD, S., & JACOBSON, V. (1993). Random Early Detection gateways for Congestion Avoidance. Recuperado el 20 de Septiembre de 2009, de <http://www.icir.org/floyd/red.html/>.
- LA, R.J., RANJAN, P., ABED, E. Feedback control of random early detection. Recuperado el 20 de Septiembre de 2009, de <https://drum.umd.edu/dspace/bitstream/1903/200/1/umi-umd-1313.pdf/>.
- LE, L., AIKAT, J., JEFFAY, K., DONELSON, S. The Effects of Active Queue Management and Explicit Congestion Notification on Web Performance. Recuperado el 20 de Septiembre de 2009, de www.cs.unc.edu/~jeffay/papers/IEEE-ToN-05.pdf/.
- MAY, M., BOLOT, J., DIOT, C., LYLES, B. Reason to not deploy RED. Recuperado el 20 de Septiembre de 2009, de ieeexplore.ieee.org/iel5/6221/16606/00766502.pdf/.
- MCCABE, J. (2003). *Network analysis, architecture and design*. Elsevier Science. USA. p. 501.
- OHSAKI, H. & MURATA, M. (2002). Steady State Analysis of the RED gateway: stability, Transient Behavior, and Parameter Setting. *IEICE Trans. Commun.*, Vol. E85-B, No.1. Recuperado el 20 de Septiembre de 2009, de citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.63.4304&rep.
- WELZL, M. (2005). *Network Congestion Control*. Wiley. England. p. 263.
- ZHANG, H., LIU, M., VUKADINOVIC, V., TRAJKOVIC, L., Modelling TCP/RED: a dynamical approach. Recuperado el 20 de Septiembre de 2009, de www.ensc.sfu.ca/~ljlja/publications_date.html/.