Elección de un método Monte Carlo para el cálculo de la medida R_V de confiabilidad en redes de comunicaciones y su implementación en la herramienta HEIDI

Héctor Cancela *María E. Urquhart*

cancela@fing.edu.uy urquhart@fing.edu.uy

Investigación Operativa

Instituto de Computación, PeDeCiBa Informática

Facultad Ingeniería, Montevideo, Uruguay

Resumen — En un trabajo anterior [3] hemos presentado un estado del arte de los distintos métodos existentes para la evaluación por Monte Carlo de la confiabilidad de una red de comunicaciones. El objetivo de este documento es el justificar la elección de uno de estos métodos para el caso específico de la evaluación de la medida R_V (confiabilidad entre todas las terminales), y presentar su implementación en la herramienta HEIDI junto con algunos resultados numéricos.

Palabras clave: Confiabilidad de redes, métodos Monte Carlo, reducción de varianza.

1 Introducción

Este reporte tiene como objetivo presentar el método de cálculo de R_V por simulación elegido para su implementación en la herramienta HEIDI. El contexto general de esta presentación es el mismo del trabajo [3] (estado del arte), que no será reiterado.

La organización de este informe es la siguiente. En la sección 2 se discuten los criterios y motivaciones utilizados para discriminar entre los distintos métodos existentes, de cuya consideración hemos elegido el método de muestreo antitético generalizado [4]. La sección 3 realiza una descripción detallada de dicho método. Por último, en la sección 4 se dan algunos resultados numéricos de la implementación realizada.

2 Elección del algoritmo a implementar

Para realizar la elección de un algoritmo entre todos los relevados en el trabajo [3], hemos considerado los siguientes criterios, de acuerdo a las necesidades de la herramienta en desarrollo:

- fácil adaptación al cálculo de R_V: ya que ciertos algoritmos han sido diseñados teniendo únicamente en cuenta el cálculo de R_{st};
- eficiencia computacional: el algoritmo elegido debe ser lo más eficiente posible frente al método Monte Carlo crudo, lo que puede alcanzarse gracias a la reducción de la varianza propiamente dicha o gracias a un menor tiempo de cálculo por muestra; un problema encontrado es la casi inexistencia de resultados numéricos en la literatura sobre la evaluación de la medida R_V, lo que dificulta la comparación de los distintos métodos;
- tiempo de pre-procesamiento poco importante o nulo: algunos algoritmos requieren un pre-procesamiento que implica la generación y almacenamiento de todos los cortes o todos los caminos del grafo previo a su simulación, tarea de complejidad exponencial en el tamaño del grafo, y casi equivalente al cálculo exacto de la confiabilidad;
- especial adaptación al caso en el que las confiabilidades elementales son muy cercanas a 1: nos interesa un método eficiente en los grafos altamente confiables, que son aquellos que aparecen en las aplicaciones en telecomunicaciones;
- facilidad de extensión al caso en el que existen fallas tanto en las aristas como en los nodos: estos modelos resultan también de gran interés, y no todos los métodos pueden adaptarse fácilmente al cálculo de la confiabilidad en este contexto.

Pasemos entonces revista a los distintos métodos relevados en [3]:

1. **Muestreo basado en cotas [5].**
   Los métodos de esta familia han sido especialmente estudiados en el contexto R_{st}, no habiendo muchas referencias en la literatura al caso R_V. Sería entonces necesario buscar un par de funciones cota adaptadas, computacionalmente eficientes.. Otro problema es el de la facilidad de extensión al caso de fallas en los nodos, que no es inmediata.

2. **Muestreo "dagger" [8] y derivados [4].**
   El método "dagger" es especialmente eficiente en el caso en que las confiabilidades son muy cercanas a la unidad. Sin embargo, en su versión original puede presentar serios problemas cuando las confiabilidades elementales son distintas, con requerimientos de memoria muy importantes. En cambio, la modificación presentada en [4] bajo el nombre de "algoritmo antitético generalizado" no tiene este problema, y resulta muy fácil adaptarlo al caso de fallas en los nodos, o incluso a modelos con dependencias

entre los elementos de la red. Los resultados numéricos existentes en la literatura se refieren especialmente al caso R_{st}, pero el principio del algoritmo no depende de la elección del conjunto de terminales, por lo que puede suponerse que resultará igualmente eficiente para el cálculo de R_V.

3. **Construcción/Destrucción Secuenciales [9].**
   Los métodos de esta familia han sido especialmente estudiados para el caso en el que todas las aristas poseen la misma la confiabilidad elemental. Son también aplicables cuando hay confiabilidades diferentes, pero en ese caso es necesario recurrir a rutinas más generales y de menor eficiencia. Los (pocos) resultados de la literatura para el caso R_V son buenos, aunque tienen el inconveniente de haber sido obtenidos con un muy bajo número de replicaciones (10000 replicaciones). Otro característica es que no resulta muy fácil realizar la extensión al caso en que hay fallas también en los nodos.

4. **Método de cubrimiento (Karp&Luby) [7].**
   El gran inconveniente de este método es el de exigir el cálculo y almacenamiento previo de la lista de todos los cortes mínimos del grafo considerado. Dado que el tamaño de esta lista crece en forma exponencial con el tamaño del grafo, tenemos que el método resulta inaplicable para grafos de mediano tamaño (más de 100 elementos). Por lo tanto, podemos considerar que no será adecuado a nuestros fines.

5. **Método del riesgo total [10].**
   Este método, si bien tiene buenos resultados experimentales para el caso R_{st}, no garantiza una reducción de varianza respecto al método crudo en todos los casos. Los resultados experimentales de los que se dispone para el caso R_V (ver [3]) no son demasiado prometedores. En un trabajo reciente [1] se demuestra que de todos modos la reducción de varianza obtenida por este método es siempre inferior a la obtenida por el de reducción recursiva de varianza, que veremos a continuación.

6. **Método de reducción recursiva de varianza [2].**
   Si bien este método obtiene muy buenos resultados experimentales para el caso R_{st}, existe fuerte presunción de que su eficacia decrece al aumentar el número de terminales consideradas. Esta conjetura está respaldada por los resultados experimentales presentados en [3]. Un punto a favor del método es su relativa facilidad de adaptación al caso de fallas en los nodos, que no presenta mayor complejidad.

En base a las consideraciones precedentes, el método que hemos escogido para su implementación en la herramienta HEIDI es el método antitético generalizado, que presenta el mayor número de características favorables.

3 Descripción del algoritmo antitético generalizado

El método antitético generalizado fue propuesto en [4] a partir del muestreo dagger propuesto por Kumamoto et al. [8]. Puede ser visto como una extensión de la técnica de variables antitéticas. La idea base es la de generar bloques de un número L de muestras, dentro de los cuales las variables aleatorias son elegidas de manera de introducir correlación negativa entre las muestras individuales. Sin embargo, la principal ventaja no consiste en la reducción de varianza obtenida, sino en el empleo de mecanismos muy similares a aquellos empleados en la simulación a eventos discretos, que permiten una gran reducción en el tiempo de cálculo por muestra. En este método se lleva una lista con los próximos fallos correspondientes a cada arista, y se busca la primer replicación con un fallo, lo que permite no sólo ganar en tiempo de ejecución, sino también reducir la complejidad en memoria al mínimo, al mismo tiempo que la hace independiente del tamaño de bloque L.

El tamaño L de cada bloque es un parámetro del algoritmo. Para cada arista l la secuencia de L replicaciones de un bloque es partida en exactamente $N_l - 1$ sub-bloques de tamaño $S_l = min(L, \lfloor 1/q_l \rfloor)$ y un sub-bloque de tamaño $T_l = (L-1)\ mod\ S_l + 1$, siendo $N_l = \lceil L/S_l \rceil$. Para cada uno de esos subbloques se elige aleatoriamente (utilizando una variable uniforme) una única posición en la que la arista l va a fallar (esto genera un patron de fallos para las L replicaciones en el que la frecuencia de fallas de cada arista está en proporción a la tasa media de fallo de la misma). Luego se recorre todas las replicaciones del bloque, controlando en cuales de ellas el grafo es $\mathcal{K}$-conexo y en cuales no lo es, para de esta forma obtener una estimación del valor de $R_{\mathcal{K}}$.

El algoritmo antitético genérico tiene una complejidad por muestra de $O(\alpha|\mathcal{E}|)$, donde $\alpha = max(1/L, 1-p)$, cuando las confiabilidades de todas las aristas son iguales a p. Esto marca una mejora frente al método crudo que es creciente con el tamaño L elegido para los bloques, pero acotada por el inverso de las confiabilidades elementales (es decir que más allá de un cierto nivel, el aumentar L no incrementa la eficiencia del método).

La figura 1 muestra en pseudocódigo la rutina principal teniendo como entrada al grafo $\mathcal{G}$, el conjunto $\mathcal{K}$, y el tamaño del bloque L, y retornando como salida un valor de R.

4 Resultados numéricos

Esta sección presenta algunos resultados obtenidos con la implementación del algoritmo antitético generalizado en la herramienta HEIDI.

Para evaluar un esquema de simulación es importante tener en cuenta tanto su precisión (varianza o intervalo de confianza) como el esfuerzo computacional requerido para obtener dicha precisión. Fijado un tamaño de muestra, notaremos T y V respectivamente el tiempo de ejecución y la varianza. Para tener en cuenta los aspectos nombrados anteriormente utilizaremos la eficiencia relativa (a veces llamada aceleración) del método antitético generalizado ($ANTI$) respecto a Monte Carlo crudo (MCC). Este parámetro denotado como W_{MCC}/W_{ANTI} [6] es el el producto de dos términos, el cociente de varian-

Entradas: grafo $\mathcal{G}$, conjunto $\mathcal{K}$, tamaño del bloque L
Salida: un valor de R

Inicialización del bloque:
Para $e = 1, \ldots, |\mathcal{E}|$
    Sortear $proxFalla[e]$;
finPara
Adelantar el tiempo: $t := min(proxFalla[e])$;
Inicializar variables: $tprevio := 0$; $R := 0$;
Mientras $t \leq L$
    $deltaT := t - tprevio - 1$;
    $R := R + deltaT \times 1$;
    Determinar estado en el instante t**:**
        $x_e := 0 \iff proxFalla[e] = t, x_e := 1$ de otro modo;
    Si $\Phi(X) = 1$ **entonces** $R := R + 1$;
    Para $e/proxFalla[e] = t$
        Sortear $proxFalla[e]$;
    finPara
Adelantar el tiempo: $t := min(proxFalla[e])$;
finMientras
$deltaT := t - tprevio - 1$;
$R := (R + deltaT \times 1)/\mathsf{L}$;
Retornar R;

Figura 1: Pseudo-código del algoritmo antitético generalizado

zas V_{MCC}/V_{ANTI} y el cociente de tiempos T_{MCC}/T_{ANTI}, para un tamaño fijo de muestra.

Los resultados están organizados en dos grupos. Primero utilizamos la red de 20 nodos y 30 aristas que se ve en la Figura 2, tomando todas las aristas la misma confiabilidad elemental $q_l = 0.10$, 0.01 y 0.001. Utilizamos $N = B \times L$ replicaciones para cada valor de la confiabilidad elemental, repartidas en $B = 10^4$ bloques de L iteraciones cada uno. La determinación de L se realizó para cada confiabilidad de acuerdo a lo discutido en [4], donde se demuestra que cuando las aristas son equiprobables con confiabilidad q_l, el valor de L que proporciona el mejor rendimiento del algoritmo es $L = \lceil 1/q_l \rceil$. Esto corresponde a elegir $L = 10, 100, 1000$ para $q_l = 0.100, 0.010, 0.001$. Por lo tanto el tamaño total de la muestra para los distintos valores de q_l varía, siendo $N = 10^5, 10^6, 10^7$ respectivamente.

En las primeras columna de la Tabla 1 aparecen los valores exactos de la anti-fiabilidad Q_V, obtenidos por la herramienta HEIDI utilizando una implementación del algoritmo de Ahmad, así como las estimaciones $\widehat{Q}_V$ del método antitético generalizado, en función de la

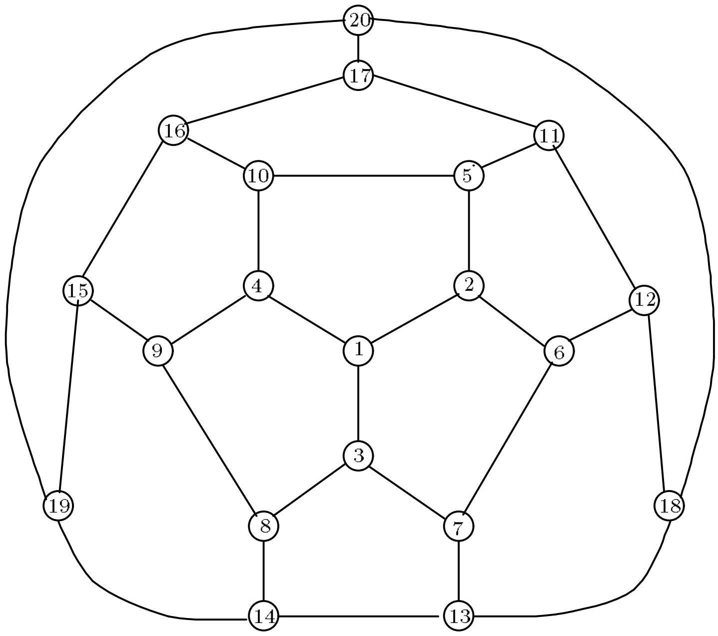

Figura 2: Topología de la red Dodecaedro

confiabilidad elemental de las aristas. Las columnas siguientes presentan la aceleración, así como los cocientes de varianzas y de tiempos de cálculo relativos al método Monte Carlo directo, correspondientes al método antitético. Todos los resultados fueron obtenidos en una estación Sun SPARCstation 5.

	Q_V	$\widehat{Q}_V$	V_{CMC}/V_{ANTI}	T_{CMC}/T_{ANTI}	W_{CMC}/W_{ANTI}
$q_l = .100$	2.287×10^{-2}	2.273×10^{-2}	6.083×10^{-1}	1.068×10^{0}	6.498×10^{-1}
$q_l = .010$	2.030×10^{-5}	1.901×10^{-5}	1.156×10^{0}	4.700×10^{0}	5.432×10^{0}
$q_l = .001$	2.003×10^{-8}	0	†	4.312×10^{1}	†
† Para $q_l = 0.001$, el método antitético obtuvo un intervalo de confianza nulo, por lo que no es posible calcular estos cocientes.					

Tabla 1: Resultados de simulación para la red en la Figura 1.

El segundo grupo de resultados corresponde a la topología que aparecen la figura 3, y que corresponde a una versión del núcleo de fibra óptica de la red telefónica de Montevideo. Esta topología posee 14 nodos y 21 aristas. También en este caso se consideraron las confiabilidades elementales $q_l = 0.10$, 0.01 y 0.001, y concomitentemente tamaños de bloque $L = 10, 100, 1000$ siendo constante el número de bloques $B = 10^4$; el número total de replicaciones es por lo tanto 10^5, 10^6 y 10^7 respectivamante.

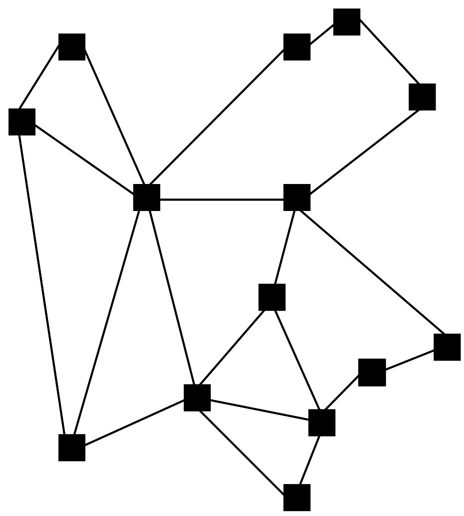

Figura 3: Versión parcial de la red telefónica de Montevideo

En las primeras columna de la Tabla 2 aparecen los valores exactos de la anti-fiabilidad Q_V, obtenidos por la herramienta HEIDI utilizando una implementación del algoritmo de Ahmad, así como las estimaciones $\widehat{Q}_V$ del método antitético generalizado, en función de la confiabilidad elemental de las aristas. Las columnas siguientes presentan la aceleración, así como los cocientes de varianzas y de tiempos de cálculo relativos al método Monte Carlo directo, correspondientes al método antitético. Todos los resultados fueron obtenidos en una estación Sun SPARCstation 5.

	Q_V	$\widehat{Q}_V$	V_{CMC}/V_{ANTI}	T_{CMC}/T_{ANTI}	W_{CMC}/W_{ANTI}
$q_l = .100$	1.025×10^{-1}	1.046×10^{-1}	1.765×10^{0}	1.280×10^{0}	2.259×10^{0}
$q_l = .010$	1.094×10^{-3}	1.150×10^{-3}	1.031×10^{0}	5.818×10^{0}	6.000×10^{0}
$q_l = .001$	1.099×10^{-5}	1.198×10^{-5}	8.607×10^{-1}	5.517×10^{1}	4.749×10^{1}

Tabla 2: Resultados de simulación para la red en la Figura 3.

5 Conclusiones

En este informe hemos presentado una discusión de los métodos de reducción de varianza para el cálculo Monte Carlo del parámetro Q_V de anticonfiabilidad de una red de comunicaciones. En base a esta discusión, se ha realizado la elección de uno de estos métodos para su implementación en la herramienta HEIDI. Se ha presentado el seudocódigo correspondiente al método elegido (método antitético generalizado), así como una serie de resultados numéricos basados en topologías de interés.

Referencias

[1] H. Cancela and M. El Khadiri. *Recursive Path Conditioning Monte Carlo Simulation of Communication Network Reliability.* Research Report PI915, IRISA, Rennes, France, Campus de Beaulieu 35042 France, March 1995.

[2] H. Cancela and M. El Khadiri. A recursive variance reduction method for estimating communication network reliability. *IEEE Transactions on Reliability*, (to appear), 1995.

[3] H. Cancela and M.E. Urquhart. Métodos Monte Carlo en la estimación de la confiabilidad de redes de comunicaciones – Estado del arte –. In *24 JAIIO*, SADIO, Buenos Aires, Argentina, Agosto 1995.

[4] M. El Khadiri and G. Rubino. *A Monte-Carlo Method Based on Antithetic Variates for Network Reliability Computations.* Technical Report 626, IRISA, Campus de Beaulieu, 35042 Rennes, France, 1992.

[5] G.S. Fishman. A Monte Carlo sampling plan for estimating network reliability. *Operational Research*, 34, 1986.

[6] J.M. Hammersley and D.C. Handscomb. *Monte Carlo Methods.* Halsted Press, Wiley & Sons. Inc., New York, 1979.

[7] R. Karp and M.G. Luby. A new Monte Carlo method for estimating the failure probability of an n-component system. *Computer Science Division*, University of California(Berkeley), 1983.

[8] H. Kumamoto, K. Tanaka, K. Inoue, and E.J. Henley. Dagger-sampling Monte Carlo for system unavailability evaluation. *IEEE Trans. Reliab.*, R-29(2), June 1980.

[9] C.K. Wong M.C. Easton. Sequential destruction method for Monte Carlo evaluation of system reliability. *IEEE Trans. Reliab.*, R-29(1), April 1980.

[10] S.M. Ross. Variance reduction in simulation via random hazard. *Probability in the Engineering and Informational Sciences*, 4, 1990.