



Monografía

Presentada el 24 de mayo de 2016 en

Universidad de La República, UdelaR

Facultad de Ciencias Económicas y Administración, Instituto de Estadística

para obtener el título de

LICENCIADO EN ESTADÍSTICA

para

María José ACUÑA SOSA

Título :

Estadística Aplicada al Cálculo de Confiabilidad de Redes

Orientadores: Dr. Franco Robledo, Dr. Pablo Romero

Comité de examinadores:

Dr. Gustavo Guerberoff

Dr. Juan Piccini

Dr. Pablo Sartor

Agradecimientos

Quiero agradecer a mis tutores por su apoyo y todo el tiempo que me han dedicado, compartiendo generosamente sus conocimientos, Dr. Franco Robledo y Dr. Pablo Romero. También a mi familia que me ha acompañado y comprendido desde los comienzos de mi carrera, así como a todas aquellas personas que de alguna forma me han ayudado a lo largo de este camino, como lo fueron mis amistades, mis compañeros de estudio y de trabajo.

Índice general

I	INTRODUCCIÓN	3
II	CONOCIMIENTOS PRELIMINARES	9
1.	Conocimientos Preliminares	11
1.1.	Teoría de Grafos	11
1.2.	Sistemas Binarios Estocásticos	13
1.3.	Confiabilidad Clásica	15
1.4.	Complejidad Computacional	16
III	ARISTA CONFIABILIDAD	21
2.	Métodos de Cálculo Exacto	23
3.	Métodos aproximados para el Problema Clásico	27
3.1.	Introducción	27
3.2.	Método Monte Carlo Crudo	28
3.3.	Muestreo por Importancia.	32
3.4.	Método secuencial IS.	34
3.5.	Muestreo Basado en Cotas.	39
3.6.	Muestreo Dagger	42
3.7.	Método de Reducción Recursiva de la Varianza (RVR)	46
3.8.	Método Entropía Cruzada.	52
3.9.	Entropía Cruzada y Monte Carlo	54
4.	Estudio experimental	55
4.1.	Resultados numéricos	58
5.	Conclusiones Generales	63
	Bibliografía	67

Resumen

La realización del cálculo exacto de la probabilidad de funcionamiento de una red sensible a probables fallas en alguno de sus componentes corresponde a un problema combinatorio perteneciente a la clase \mathcal{NP} -Difícil, por tal motivo se decide utilizar métodos de aproximación para su estimación, presentando soluciones eficientes.

Usualmente se opta por técnicas basadas en la simulación Monte Carlo. Cuando la red presenta una baja probabilidad de falla, el método Monte Carlo Crudo, abreviado CMC debido a sus siglas inglesas, se vuelve poco eficiente para estimar la confiabilidad. Existen diferentes métodos estadísticos que logran reducir la varianza obteniendo una mejor precisión de la estimación de la confiabilidad respecto al método CMC. Específicamente en este trabajo se describen los métodos de Monte Carlo Crudo, Muestreo por Importancia (IS), Muestreo basado en Cotas, Reducción Recursiva de la Varianza (RVR), Muestreo Dagger y Entropía Cruzada. Para cada uno de ellos se estudia el rendimiento en cuanto a tamaño muestral, error cuadrático medio y esfuerzo computacional.

Se realiza un estudio comparativo entre IS y RVR, donde los resultados indican que al estimar la confiabilidad de una red con posibles fallas en los enlaces, mediante el primer método se genera una varianza menor pero a un costo computacional mucho mayor que en RVR.

Palabras clave: Confiabilidad de Redes, Complejidad Computacional, Monte Carlo Crudo, Muestreo por Importancia, Muestreo basado en Cotas, Reducción Recursiva de la Varianza, Muestreo Dagger, Entropía Cruzada.

Parte I

INTRODUCCIÓN

Motivación

En los últimos años se ha dado un desarrollo tecnológico que originó un incremento en el número de usuarios de redes de comunicación, lo que trae como consecuencia un aumento en el interés por realizar diversas investigaciones en esta área con el fin de ofrecer un servicio que sea accesible y seguro. Este trabajo centra su objetivo en estudiar la confiabilidad de una red, la probabilidad de que la misma continúe funcionando en perfectas condiciones a pesar de que alguno de sus componentes falle, en otras palabras, la capacidad que tiene la misma de resistir alguna falla en alguno de sus componentes y continuar brindando el servicio que originalmente brindaba.

Es importante destacar que realizar investigaciones con modelos estáticos se refiere a estudiar la confiabilidad de que la red esté funcionando correctamente en un instante del tiempo y cuando se tienen modelos dinámicos se estudia la confiabilidad de una red en un intervalo de tiempo. Existen varias medidas de confiabilidad de una red, una de ellas es la clásica. En la confiabilidad clásica se cuenta con un grafo con diferentes nodos los cuales son llamados terminales, es decir, diferentes puntos a los cuales se desea hacer llegar una determinada información y los enlaces son los que comunican los mismos, que pueden fallar en forma independiente.

La línea de estudio se centrará en calcular la confiabilidad de una red cuando la posible falla se encuentra en los enlaces, se dirá entonces que los nodos son perfectos (no fallan).

El lector interesado puede hallar una valiosa discusión de confiabilidad de redes y sus aplicaciones en [9].

Objetivo de la Monografía

El objetivo principal de la presente monografía es el cálculo aproximado de la confiabilidad clásica de una red representada mediante un grafo. Dado un grafo simple donde sus enlaces fallan de modo estocástico e independiente, se define la confiabilidad clásica como la probabilidad de comunicación de sus terminales, es decir, un conjunto de nodos que se desea estén comunicados. Se presentarán diversos métodos aproximados al cálculo de la confiabilidad, con énfasis en métodos estadísticos, que serán presentados a lo largo del documento. Es importante mencionar que existen numerosos métodos, por lo cual se realizó una selección de aquellos que son robustos ante escenarios donde la falla es un evento raro. La confiabilidad de una red, se define como la probabilidad de que la misma continúe operativa aunque alguno de sus componentes falle. En este trabajo se supone que los únicos componentes que fallan son los enlaces, con idénticas e independientes probabilidades $q = (1 - p)$, siendo p la probabilidad de que el enlace se encuentre operativo, es decir, la confiabilidad del mismo.

Organización del documento

A continuación se realiza una breve descripción de la organización del documento, el cual se encuentra estructurado en tres partes.

Parte I - Introducción

Se presenta la Introducción, donde se establece el objetivo principal del trabajo, como también su motivación.

Parte II - Conocimientos Preliminares

En el Capítulo 1, se resume terminología relacionada a Grafos, Sistemas Binarios Estocásticos, Confiabilidad Clásica de una red y Complejidad Computacional, a efectos de una mejor comprensión de esta monografía por parte del lector.

Parte III- Arista Confiabilidad

El Capítulo 2, corresponde a la presentación de diversos métodos de cálculo exacto relacionados al Problema Clásico de confiabilidad en redes.

Luego en el Capítulo 3, se expone una selección de diversos métodos aproximados para el Problema Clásico.

En el capítulo 4 se presenta un estudio comparativo de los métodos Muestreo por Importancia y Reducción Recursiva de la Varianza, así como sus conclusiones. Finalmente en el capítulo 5 se presentan las conclusiones generales de esta monografía.

Parte II

**CONOCIMIENTOS
PRELIMINARES**

Capítulo 1

Conocimientos Preliminares

1.1. Teoría de Grafos

En esta sección se introducen conceptos correspondientes a la Teoría de Grafos, existe una amplia literatura en relación a este tema, el lector interesado puede encontrar valiosa información en los libros de Frank Harary [18] o de Ralph P. Grimaldi [17], entre otros.

Se define un *grafo* G dirigido (sobre V) o digrafo (sobre V), como un par (V, E) , donde V es un conjunto no vacío y $E \subseteq V \times V$, siendo V el conjunto de *vértices* o *nodos* y E el conjunto de *aristas* o *enlaces*, se denota usualmente $G = (V, E)$. Si $|V|$ es finito, entonces G es también finito. Cuando no interesa la dirección de las aristas es un grafo no dirigido, en ese caso E es una relación simétrica $((a; b) \in E \rightarrow (b; a) \in E)$.

Un par de nodos u, v son adyacentes si existe una arista $x \in E$ que los une (es incidente a ambos vértices), donde $x = \{u, v\} = uv = e_{uv}$. Un grafo G tiene aristas múltiples cuando contiene aristas con los mismos nodos iniciales y finales. Un grafo puede tener *Loops*, es decir, una arista que une un vértice consigo mismo. Cuando un grafo no dirigido no tiene *Loops* se denomina simple. Un grafo es completo si y solo si para todo par de vértices distintos existe una arista que los conecta.

Definición 1.1.1 *Un camino $u - v$ de un grafo no dirigido es una sucesión finita (sin lazos) de vértices adyacentes y aristas en G , con extremos en los vértices u y v . La cantidad de aristas que hay en el camino es la longitud del mismo.*

Definición 1.1.2 *Un camino es cerrado cuando $u = v$.*

Definición 1.1.3 *Cuando en el camino $u - v$ no se repite ninguna arista, lo llamamos un recorrido $u - v$.*

Definición 1.1.4 *Un circuito es un recorrido $u - u$ cerrado.*

Definición 1.1.5 *Un camino simple $u - v$ es un camino en donde ningún vértice es visitado más de una vez. Cuando el mismo es cerrado $u - v$ es un ciclo.*

Definición 1.1.6 *Una componente conexa de G es un subgrafo conexo maximal de G .*

Definición 1.1.7 Sea $G = (V, E)$ un grafo no dirigido, se dice que G es conexo si existe un camino simple entre dos vértices cualesquiera de G . Un grafo que no es conexo es desconexo, o sea, V (conjunto de nodos) puede separarse en al menos dos subconjuntos V_1 y V_2 de forma tal que no exista una arista de E de la forma $\{u, v\}$ en donde $u \in V_1$ y $v \in V_2$. Por lo tanto un grafo es conexo si solamente está compuesto por una componente, el número de componentes conexas se denota $K(G)$.

Definición 1.1.8 Un subgrafo de $G = (V, E)$, se denota $G' = (V', E')$, donde $V' \subseteq V$ y $E' \subseteq E$, que es igual a que $G' \subseteq G$. Se dice que G' es un grafo inducido de G .

Definición 1.1.9 Un conjunto de corte de un grafo $G(V, E)$ no dirigido, es un subconjunto de aristas $E' \in E$ tal que al eliminarlas se tiene que $K(G) < K(G')$, donde $G' = (V, E - E')$.

Definición 1.1.10 El grado de un vértice v de G se denota $gr(v)$, es equivalente al número de aristas de G que resultan incidentes en v .

Definición 1.1.11 Un corte minimal (mincut), es un corte que no contiene otro corte que el mismo.

Definición 1.1.12 Dado un grafo $G = (V, E)$ con $|V| \geq 3$, G tiene un ciclo Hamiltoniano si existe un ciclo en G que contiene cada vértice de G .

Definición 1.1.13 Un camino Hamiltoniano es un camino simple de G que posee todos los vértices del grafo.

Definición 1.1.14 Un grafo aleatorio es aquel donde sus nodos y aristas son etiquetados con cierta probabilidad. Entonces $G = (V, E, \vec{p})$, con $p_i \in [0, 1] \forall i \in V \cup E$ y $\vec{p} = (p_1, \dots, p_N)$.

En el presente documento se trabaja con el supuesto de que los únicos que pueden fallar son los enlaces, entonces en este caso \vec{p} se refiere a los mismos y no a los vértices.

1.2. Sistemas Binarios Estocásticos

Se realiza a continuación una introducción a Sistemas Binarios Estocásticos que se abreviará SBS debido sus siglas.

Definición 1.2.1 *Un Sistema Binario Estocástico [3, 30], es un sistema que falla o se encuentra operativo en forma aleatoria, en función de la probabilidad de falla u operación de sus componentes. Se define en base a la terna (T, ϕ, P) , donde:*

- $T = \{1, 2, \dots, m\}$ es el conjunto ordenado de los m componentes del sistema.
- Cada componente puede encontrarse únicamente en dos estados posibles, o se encuentra operativo o falla, siendo independiente el estado de un componente en relación al estado de otro.
- Una palabra binaria $x = (x_1, \dots, x_m) \in \{0, 1\}^m$ representa el estado binario de los componentes, donde $x_i = 1$ significa que la componente i -ésima opera y $x_i = 0$ que falla.
- Se define función de estructura del sistema a $\phi: \{0, 1\}^m \rightarrow \{0, 1\}$, siendo una función binaria cuyos valores posibles son : $\phi(x) = 1$ cuando el sistema se encuentra operativo aunque fallen sus componentes bajo x , el cual es un vector ordenado y $\phi(x) = 0$ en caso contrario.
- $P = (p_1, \dots, p_m)$ es el vector de probabilidades de operación de cada componente.

Definición 1.2.2 *La confiabilidad de un Sistema Binario Estocástico es la probabilidad de que funcione el sistema, y se define como:*

$$R = P(\phi(X) = 1) = E(\phi(X)), \quad (1.1)$$

donde $X = (X_1, \dots, X_m)$, son variables aleatorias independientes que siguen una ley Bernoulli con $P(X_i = 1) = p_i$, es decir, la probabilidad de que la componente i se encuentre funcionando, y $P(X_i = 0) = 1 - p_i$ en el caso contrario.

Definición 1.2.3 *Un SBS es monótono si $\phi(x)$ es monótona. Si se tiene un orden establecido en relación a las palabras binarias: $x = (x_1, \dots, x_m) \in \{0, 1\}^m$, $y = (y_1, \dots, y_m) \in \{0, 1\}^m$ entonces $x \leq y$ si y solo si $x_i \leq y_i \forall i \in \{1, \dots, m\}$. Entonces $\phi(x)$ es monótona si se cumple que $\phi(x) \leq \phi(y)$.*

El concepto de monotonía es de componente a componente con un orden de inclusión establecido y se mira en base a palabras binarias.

Definición 1.2.4 *Un Sistema Binario Estocástico (SBS) es coherente si cumple las siguientes características, además de cumplir con la propiedad de monotonía explicada antes:*

$$a) \phi(\vec{0}) = 0,$$

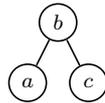
$$b) \phi(\vec{1}) = 1.$$

Dada una configuración operativa x , donde $\phi(x) = 1$, y si por algún motivo se decide cambiar x_i que vale 1 a que valga 0 y el sistema cae, entonces x es una configuración operativa minimal. Los estados de cortes minimales determinan completamente la estructura de un SBS monótono [3].

A continuación se presentan algunos ejemplos de Sistemas Binarios Estocásticos monótonos:

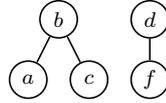
- All-Terminal Reliability [3]: dado un grafo aleatorio $G = (V, E)$ con nodos perfectos, el sistema está operativo si entre cualquier par de nodos pertenecientes a G existe al menos un camino que los comunica.
- Diameter Constrained Reliability (DCR) [29]: dado un grafo aleatorio $G = (V, E)$ donde $K \subseteq V$ y $d \in \mathbb{Z}^+$, el sistema está operativo si todo par de nodos objetivos en el conjunto K se encuentran comunicados entre sí por al menos un camino de largo menor o igual a d luego de que se han removido las aristas que fallaron.
- Social Reliability: dado un grafo aleatorio donde hay enlaces que fallan, el sistema está operativo si y solo si hay un K -clique en el subgrafo resultante, es decir, hay un grafo completo simple de K vértices.
- k - m -Survivability: un sistema con m componentes se encuentra en estado operativo si y solo si por lo menos k componentes de los m también lo están. Se conoce como k -out-of- m system.

Se ilustran a continuación dos ejemplos de caída de nodos [8, 11, 15], es decir, los nodos del grafo están sujetos a posibles fallas y la propiedad de monotonía no se cumple. Por ejemplo, dado un grafo simple $G(V, E)$ con tres nodos a, b, c .



Ejemplo 1

El nodo a se comunica con el nodo b mediante la arista (a, b) y el nodo b se comunica con el nodo c mediante la arista (b, c) . Si el nodo intermedio b cae, entonces el grafo resulta no conexo. La configuración del sistema es entonces $x = (1, 0, 1)$ donde $\phi(x) = 0$, dado que el grafo quedó no conexo. Si se cuenta con una segunda configuración de estado del sistema donde el nodo a y b han fallado, es decir, $y = (0, 0, 1)$, al quedarnos únicamente con el nodo c que es el que ha quedado activo, el grafo es conexo, entonces $\phi(y) = 1$. En este caso $y \leq x$, pero $\phi(y) > \phi(x)$, por lo tanto no se cumple la propiedad de monotonía.



Ejemplo 2

En el segundo ejemplo, el razonamiento es análogo al anterior dado que si los vértices d y f caen, el grafo que inicialmente era no conexo pasaría a ser conexo.

1.3. Confiabilidad Clásica

El problema de la confiabilidad clásica en redes se enmarca también dentro de los sistemas binarios coherentes estocásticos [3]. Se desea calcular la probabilidad de que un conjunto K de nodos objetivos denominados terminales se comuniquen entre sí, con posibles fallas en los enlaces.

Una red puede ser modelada mediante un grafo simple $G = (V, E)$, donde $V = (v_1, \dots, v_n)$ y $E = (e_1, \dots, e_m)$ son el conjunto de nodos y aristas respectivamente. A lo largo de este trabajo, se hace el supuesto de que los nodos son perfectos, o sea, únicamente pueden fallar las aristas (enlaces) y lo hacen en forma independiente, siendo $p = (p_1, \dots, p_m)$ el vector de operación de los mismos. Los componentes en este caso son las aristas, es decir, $T = E$, además se cuenta con un conjunto de terminales $K \subseteq V$. Se establece un orden para el conjunto de aristas E , obteniendo para cada $x \in \{0, 1\}^m$ el valor de $\phi(x)$, de modo tal que valdrá 1 si y solo si en el subgrafo en el cual sus aristas son $E' = \{e_i \in E : x_i = 1, i = 1, \dots, m\}$ todo par de terminales en K están conectados.

La confiabilidad clásica $R_{K,G}(\vec{p}) = P(\phi(X) = 1)$ denominada *K-terminal reliability*, surge de tomar $X = (X_1, \dots, X_m)$ que es la configuración del espacio de estados del sistema, cuya operación es independiente en base al vector p , de modo que $p_i = P(X_i = 1)$. Es importante mencionar cuando $K = V$, es *all-terminal reliability* y *source-terminal reliability* si $K = \{s, t\}$. La confiabilidad clásica es un caso particular de DCR con $d = |V| - 1$.

1.4. Complejidad Computacional

En esta sección se presentarán algunos conceptos fundamentales de la Teoría de la Complejidad Computacional.

Definición 1.4.1 Problema de decisión. *Un problema es una interrogante que necesita una respuesta y recibe como entrada ciertos parámetros. Cuando acepta en cada instancia, es decir, en cada especificación de sus parámetros una respuesta de sí o no, es un problema de decisión.*

Definición 1.4.2 *Dado un problema de decisión π , una instancia del mismo es un conjunto concreto de parámetros de modo que se pueden utilizar para alimentar de manera unívoca un algoritmo a fin de encontrar una respuesta (Sí o No).*

Definición 1.4.3 Algoritmo. *Se define como un conjunto finito de pasos que resuelve en su totalidad las instancias de un problema.*

Definición 1.4.4 Orden computacional de un algoritmo. *Se define como orden computacional de un algoritmo a $f(n)$, de forma tal que toda instancia del mismo se puede ejecutar en una cantidad de operaciones que por lo menos es $f(n)$. Se dice que la función $f(n)$ es de orden $O(g(n))$ si existe por lo menos alguna constante c de modo que $|f(n)| \leq cg(n)$ para todo n natural.*

Definición 1.4.5 Algoritmo polinomial. *Un algoritmo es de tiempo polinomial cuando su función de complejidad es $O(p(n))$, siendo p un polinomio, de no ser así es de tiempo exponencial.*

Definición 1.4.6 Complejidad de un problema. *Es equivalente a la complejidad del algoritmo más eficiente que permite resolverlo.*

Se brinda a continuación un resumen conceptual sobre complejidad computacional en base a la tesis del Dr. Claudio Risso [26].

Cuando nos referimos a la eficiencia de un algoritmo nos estamos refiriendo a la relación eficiencia-complejidad, es decir, la cantidad de tiempo empleada por el mismo en su ejecución como función del tamaño de la cadena de datos de entrada (*input string set*). Para poder comparar dos algoritmos se utiliza la siguiente definición:

Definición 1.4.7 *Sea $f(x)$ y $g(x)$ dos funciones reales definidas en alguna cadena de datos de entrada cuyas salidas son reales positivos, las mismas representan el tiempo gastado por dos algoritmos ejecutándose en un ordenador en encontrar una solución a una instancia asociada con cada cadena de datos de entrada. Se dice que g es de orden computacional mayor que f , lo cual se denota ($f(x) = O(g(x))$) si y solo si existe una constante positiva M , de modo que para toda cadena x , $f(x)$ es a lo sumo M multiplicada por $g(x)$. Lo anterior significa que existe x_0 y M tal que :*

$$f(x) \leq Mg(x), \text{ para todo } |x| > x_0.$$

La definición anterior se centra en la tendencia del algoritmo a medida que aumenta el tamaño de los datos de entrada.

Definición 1.4.8 *Un algoritmo se conoce como eficiente o tratable, si y solo si para cualquier instancia x de tamaño n , el algoritmo encuentra una solución en tiempo polinomial en n .*

Existen muchos problemas para los cuales un algoritmo de tiempo polinomial no ha sido encontrado. Alan Turing, Stephen Cook y Richard Karp han realizado excepcionales contribuciones a la Teoría de la Complejidad Computacional.

A continuación se presentará un conjunto de conceptos relacionados con problemas de decisión, es decir, cuando el problema tiene como respuesta solamente dos valores: Sí o No.

Definición 1.4.9 *Cualquier problema de decisión π tiene asociado un conjunto de dominio de instancias D_π en el cual el problema tiene sentido, siendo $Y_\pi \subseteq D_\pi$ el subconjunto de instancias donde la respuesta es Sí.*

Definición 1.4.10 *Se dice que un problema de decisión pertenece a la clase \mathcal{P} si y solo si existe un algoritmo que puede resolverlo en tiempo polinomial respecto al tamaño de la instancia. Decimos que un problema de decisión pertenece a la clase \mathcal{NP} si y solo si existe un algoritmo capaz de chequear soluciones en base a una máquina secuencial determinista en tiempo polinomial en función al tamaño de la instancia. Un problema que puede ser resuelto en tiempo polinomial, también puede ser chequeado en tiempo polinomial, por lo tanto $\mathcal{P} \subseteq \mathcal{NP}$. Es importante señalar que la inclusión opuesta parece poco probable, hasta hoy nadie ha encontrado una prueba formal de la misma. De hecho, la conjetura $\mathcal{P} \neq \mathcal{NP}$ es probablemente el problema abierto más importante en la ciencia de la computación.*

Definición 1.4.11 *Dado cualquier par de problemas de decisión π y π' , y D_π y $D_{\pi'}$ sus respectivos conjuntos de instancias, se llama reducción polinomial de π' a π ($\pi' \preceq \pi$) a alguna función $f : D_{\pi'} \rightarrow D_\pi$ de complejidad polinomial de modo que para todo $d \in D_{\pi'}$ se verifica que $d \in Y_{\pi'}$ si y solo si $f(d) \in Y_\pi$.*

Si existe reducción polinomial de π' a π ($\pi' \preceq \pi$), entonces un algoritmo eficiente para encontrar soluciones para alguna instancia de π mediante el proceso de reducción puede ser utilizado como núcleo para la construcción de un algoritmo eficiente para encontrar soluciones a cualquier instancia de π' .

Definición 1.4.12 *Dado un problema π , se dice que el mismo es \mathcal{NP} -Difícil si y solo si para todo $\pi' \in \mathcal{NP}$ se verifica ($\pi' \preceq \pi$).*

Cuando además se cumple que $\pi \in \mathcal{NP}$, se dice que es \mathcal{NP} -Completo (\mathcal{NP} -C).

En 1971 Stephen Cook [10] demostró que el problema de Satisfactibilidad booleana (SAT) es \mathcal{NP} -Completo. Este problema consiste en determinar si existe una asignación de verdad al conjunto de literales x_j de una proposición lógica de modo tal que vuelva verdadera la proposición. El problema SAT es por lo tanto tan difícil como cualquier otro problema en \mathcal{NP} .

En 1972 Richard Karp [20] comenzó la construcción de una lista de reducciones polinimiales de SAT a otros problemas \mathcal{NP} , demostrando de esa manera que todos ellos pertenecen a la

clase \mathcal{NP} -Completo (21 problemas \mathcal{NP} -Completo de Karp). Hoy en día, el procedimiento estándar para demostrar que un problema $\pi \in \mathcal{NP}$ -Completo, consiste en buscar un problema π' del que se tenga conocimiento que es \mathcal{NP} -Completo y realizar una reducción de π' a π , es decir, probar que $(\pi' \preceq \pi)$. La transitiva de “ \preceq ” garantiza que $SAT \preceq \pi$. Como SAT es el problema \mathcal{NP} más difícil (Teorema de Cook), las complejidades de ambos problemas son equivalentes y π es \mathcal{NP} -Completo también. Es importante destacar que el procedimiento anterior garantiza que π es \mathcal{NP} -Difícil aunque no se pueda probar que $\pi \in \mathcal{NP}$.

Las principales contribuciones que realizó Karp en relación a este tema fueron entonces definir un método de reducibilidad entre problemas combinatorios de decisión y la demostración de los 21 problemas de decisión \mathcal{NP} -Completo.

Uno de los 21 problemas de la lista de Karp es por ejemplo el problema 3-SAT, el cual se diferencia del problema SAT en que las instancias tienen exactamente tres literales por cláusula. Para demostrar que el mismo es \mathcal{NP} -Completo, se parte de que es \mathcal{NP} , dado que se puede verificar en tiempo polinomial si en base a una asignación para las variables se satisfacen todas las cláusulas dadas, es decir, si la expresión es verdadera. Luego se realiza la reducción del problema SAT al problema 3-SAT. Para realizar la reducción se debe reemplazar la cláusula $\sigma_1 \cup \sigma_2 \cup \dots \cup \sigma_m$, donde σ_i son los literales y $m > 3$, por:

$$(\sigma_1 \cup \sigma_2 \cup u_1)(\sigma_3 \cup \dots \cup \sigma_m \cup \bar{u}_1)(\bar{\sigma}_3 \cup \sigma_1) \dots (\bar{\sigma}_m \cup \sigma_1),$$

siendo u_1 una nueva variable, este procedimiento se repite hasta que ninguna cláusula tenga más de tres literales.

Existen importantes problemas que son \mathcal{NP} -Completo o incluso \mathcal{NP} -Difícil.

A continuación se exponen otros ejemplos:

- Determinar si hay un conjunto de corte de determinado tamaño en un grafo $G = (V, E)$ es un problema \mathcal{NP} -Completo.
- Contar el número de cortes minimales de cierto tamaño correspondiente a un grafo es un problema \mathcal{NP} -Difícil. Contar cuantos ciclos hay también es un problema \mathcal{NP} -Difícil.
- El problema de Partición consiste en: dado un conjunto de enteros no negativos, determinar si hay división del mismo en dos subconjuntos de manera tal que la suma de los números en cada uno coincida, es un problema \mathcal{NP} -Completo.
- El problema del Viajero (Travelling Salesman Problem) es en general \mathcal{NP} -Difícil. Dada una lista de ciudades y las distancias entre cada par de ellas, se basa en encontrar cual es el ciclo de longitud más baja que se extiende por todas las ciudades. Si las distancias son números enteros el problema TSP es \mathcal{NP} -Completo.
- El problema de Cubrimiento de Vértices de un grafo (Vertex Cover), se basa en que hay un subconjunto de vértices de modo que cada vértice del grafo es incidente en al menos un vértice del subconjunto. El problema de encontrar un cubrimiento mínimo, es decir, con un mínimo número de vértices es un problema \mathcal{NP} -Difícil.

- El problema del Circuito Hamiltoniano. El mismo consiste en determinar si dado un grafo G existe un ciclo que incluya todos los nodos exactamente una vez, se encuentra también dentro de los problemas \mathcal{NP} -Completo.

Parte III

ARISTA CONFIABILIDAD

Capítulo 2

Métodos de Cálculo Exacto

Cuando se desea calcular $R_{K,G}$ de manera exacta, nos enfrentamos al problema asociado a la complejidad y costo computacional que esto genera. En algunos casos es posible aplicar métodos de cálculo exacto sin enfrentarse al problema antes mencionado.

Se exponen a continuación diferentes métodos de cálculo exacto que permiten obtener la medida de confiabilidad $R_{K,G}$ basados en Trivedi et. al [33]. Se recomienda al lector realizar una revisión de la bibliografía citada para una mayor comprensión de cada método. Se pueden encontrar diferentes tipos de métodos exactos, pudiendo ser clasificados en:

- Factorización (basado en el estado de una arista).
- Inclusión - Exclusión.
- Suma de Productos Disjuntos, SDP.

Los dos últimos se basan en la enumeración de conjuntos, en el caso de redes en base a min-paths o mincuts.

Enumeración del espacio de configuraciones

El recuento del espacio de configuraciones de G , es uno de los métodos más sencillos cuando se desea calcular la medida de confiabilidad $R_K(G)$. Este método también es conocido bajo el nombre de método de Fuerza Bruta.

Se deben identificar las posibles configuraciones de la red en su totalidad, calculando para cada caso la probabilidad de ocurrencia de un estado operativo o pathset.

Esta técnica en su forma teórica es relativamente simple, pero tiene el inconveniente de tener un alto costo computacional al realizar su implementación. El costo computacional es grande debido a que el mismo se incrementa en forma exponencial con el tamaño de la red [2], donde el espacio de vectores posee $2^{|E|}$ elementos.

Principio de inclusión y exclusión

El siguiente método se basa en la aplicación del principio de inclusión y exclusión, siendo el teorema de Poincaré su generalización, el cual logra el cálculo exacto de la medida de confiabilidad de la red en tiempo exponencial en relación a su tamaño. No es posible usar este método en forma generalizada, pero se destaca su buen desempeño en redes pequeñas. Para implementar esta técnica es necesario calcular la probabilidad de que cada pathset esté operativo, llegando a ser más costoso que utilizar la técnica de enumeración completa de vectores de estado.

Llamamos $\{P_1, \dots, P_H\}$ al conjunto de minpaths de G y E_i al evento en el que todas las aristas de P_i se encuentran operativas siendo $P(E_i)$ la probabilidad del evento E_i . La confiabilidad $R_k(G)$ es la probabilidad de que uno o más eventos E_i ocurran, siendo $i = 1, \dots, H$. Como los eventos E_i no son disjuntos se llega a que el cálculo de la confiabilidad puede ser expresado como:

$$R(G) = \sum_{j=1}^H (-1)^{j+1} \sum_{I \subseteq \{1, \dots, H\}, |I|=j} P(E_I),$$

con E_I el evento en donde todos los caminos P_i se encuentran en estado operativo, $i \in I$

$$E_I = \bigcap_{i \in I} E_i$$

La expresión de $R(G)$ parte de la expresión del cálculo de la probabilidad de eventos disjuntos:

$$\begin{aligned} R_k(G) &= P\left(\bigcup_{i=1}^H E_i\right) = \sum_{i=1}^H P(E_i) - \sum_{i < j} P(E_i E_j) + \dots + (-1)^{H+1} P(E_1 E_2 \dots E_H) = \\ &= \sum_{j=1}^H (-1)^{j+1} \sum_{I \subseteq \{1, \dots, H\}, |I|=j} P(E_I). \end{aligned}$$

En este método se tienen $2^H - 1$ términos, siendo por lo tanto también exponencial.

Método de Suma de Productos Disjuntos

Este método [1] se fundamenta en la utilización de eventos mutuamente disjuntos, obteniendo la confiabilidad de la red como la suma de las probabilidades de dichos eventos. La confiabilidad de la red queda definida como:

$$\prod_{i \in I} r_i \prod_{h=1}^H (1 - \prod_{j \in J_h} r_j).$$

Siendo r_i la probabilidad de que la arista i se encuentre operativa e I, J_1, \dots , conjuntos disjuntos de aristas del grafo.

Método de Satyanarayana-Prabhakar

El método de Inclusión-Exclusión puede ser mejorado en base a la utilización de conceptos de dominancia, que es precisamente en lo que se fundamenta este método [34]. La técnica utilizada se basa en que una formación par contribuye positivamente a la medida de confiabilidad y una impar lo hace negativamente. Se define como i -formación de un grafo G a un conjunto de i minipaths donde su unión es el propio grafo. Una formación de un grafo es par si i es par e impar si i es impar. Una formación par va a contribuir en forma positiva a la confiabilidad, por lo tanto, una impar lo hace negativamente.

El número de formaciones pares de G menos el número de formaciones impares se define como *Signed Domination* de G , $sdom(G)$. La confiabilidad de G queda definida como:

$$R_k(G) = \sum_{H \subseteq G} sdom(H)P(H). \quad (2.1)$$

La probabilidad de que ocurra H es $P(H)$, con H variando en todos los vectores de estado de G . La ventaja de este método frente a Poincaré consiste en que en lugar de ser necesario utilizar la enumeración completa de configuraciones de la red alcanza con el cálculo de $sdom$ de cada estado.

Método de Factorización.

Moskowitz [27], definió en 1985 el método exacto de factorización, que utiliza técnicas como: reducciones en serie, reducciones en paralelo, contracción de aristas y eliminación de aristas irrelevantes. Se reduce el grafo original $G = (V, E)$ en subgrafos de menor tamaño y se propone una fórmula recursiva que es aplicada a los mismos, donde su topología es obtenida condicionando G al estado de un enlace e . En base a este método el cálculo exacto de la medida de confiabilidad $R_K(G)$ es:

$$R_k(G) = r_e R_{k'}(G * e) + (1 - r_e) R_k(G - e).$$

La confiabilidad elemental de e es r_e , siendo e la arista pivot y $(G * e)$, $(G - e)$ los grafos resultantes de la contracción y eliminación de la arista e . La fórmula recursiva es repetida en cada una de las subredes que son obtenidas, siendo el número de repeticiones como mínimo equivalente a la enumeración de los componentes de la red, dando como resultado un tiempo de cómputo exponencial. En relación al volumen de memoria de almacenamiento que se necesita, el mismo es por lo menos cuadrático en relación al tamaño de la red. También es importante señalar que la elección de la arista pivot va a influir en el rendimiento del algoritmo. A continuación se presentan algunas definiciones importantes:

Definición 2.0.13 (Contracción de una arista) Dado un grafo $G = (V, E)$ y $e = (x, y) \in E$, la contracción de e se define como la fusión entre sus nodos extremos. $G * e$ es el grafo resultante que queda definido como $G_e = (V', E')$, siendo $V' = V - \{x\}$ y $E' = E - \{e\}$. Cada arista de la forma (w, x) es sustituida por (w, y) . La confiabilidad de G luego de la contracción de e queda expresada mediante la relación:

$$R(G) = r_e * R(G_e).$$

Definición 2.0.14 (Eliminación de aristas irrelevantes) Eliminar aristas cuando se está trabajando en el cálculo de la medida de confiabilidad de una red, consiste en eliminar las aristas irrelevantes, es decir, aquellas que no afectan el valor de la confiabilidad. Sea e una arista irrelevante del grafo $G = (V, E)$, al eliminar e del grafo G , se obtiene un nuevo grafo $(G - e) = (V, E')$, siendo $E' = E - \{e\}$. Como la arista es irrelevante, se puede afirmar que $R_K(G) = R_K(G - e)$, lo cual es válido para 2 y k terminal.

Definición 2.0.15 (Reducción en serie) Dado el grafo $G = (V, E)$ y las aristas $\{e_1, e_2\} \in E$, si estas últimas son incidentes a un mismo vértice de grado 2, entonces las mismas están en serie. Si $e_1 = (x, y)$ y $e_2 = (y, z)$, la reducción en serie implica reemplazarlas por una única arista $e = (x, z)$. La confiabilidad elemental del enlace que ha sido generado es: $r_e = r_1 r_2$, con r_1 y r_2 las confiabilidades correspondientes a e_1 y e_2 .

Definición 2.0.16 (Reducción en paralelo) Si se tiene un grafo $G = (V, E)$, dos aristas que se encuentran en paralelo, $e_1 = (x, y)$, $e_2 = (x, y)$, es decir comparten los mismos extremos, pueden ser reducidas a una única arista e de modo tal que se conserva la confiabilidad de la red. La confiabilidad elemental de e es igual a : $r_e = r_1 + r_2 - r_1 r_2$.

Capítulo 3

Métodos aproximados para el Problema Clásico

3.1. Introducción

El problema clásico de confiabilidad pertenece a la clase \mathcal{NP} -Difícil [7]. Cuando se desea resolver analíticamente la confiabilidad de una red, el esfuerzo computacional requerido se vuelve elevado, creciendo en forma exponencial a medida que aumenta el tamaño del grafo. Debido a la creciente importancia que ha tomado el cálculo de la medida de confiabilidad ($R_k(G)$) de redes actuales de gran tamaño, se ha optado por la utilización de métodos aproximados para su cálculo. Como fue previamente mencionado se presentarán diversos métodos de aproximación a la confiabilidad clásica, con énfasis en métodos estadísticos. Uno de los métodos utilizados en estos casos es el Monte Carlo Crudo [13, 30, 31], abreviado CMC debido a sus siglas inglesas. Es importante señalar que dicho método, aunque es de fácil aplicación, tiene el inconveniente de no obtener un buen desempeño en redes altamente confiables, es decir, cuando la falla de la misma es un evento raro. Si bien este método es de fácil aplicación, es necesario utilizar técnicas de reducción de varianza, que también serán presentadas en este capítulo.

Antes de exponer los diferentes métodos es importante recordar que dada una sucesión de variables aleatorias independientes (X_i) con idéntica distribución, decimos que un estimador $\hat{\theta}_n(X_1, \dots, X_n)$ es insesgado para θ si se verifica que $E(\hat{\theta}_n - \theta) = 0$ para todo entero positivo n , coincidiendo el error cuadrático medio de $\hat{\theta}_n$ con su varianza. A continuación se presentan algunas definiciones que serán vistas a lo largo del presente documento, donde γ es la anticonfiabilidad de la red.

Definición 3.1.1 Error Relativo. *Se define como el cociente entre el radio de confianza y el valor correcto del parámetro γ .*

Definición 3.1.2 Error Relativo Acotado o BRE (por sus siglas inglesas) [24], establece que cuando $\gamma \rightarrow 0$ el error relativo (ER) tiene cota superior.

$$\lim_{\gamma \rightarrow 0} ER(\gamma) = L$$

Definición 3.1.3 Error Relativo Desvaneciente VRE (Vanishing Relative Error) [24], indica que $ER \rightarrow 0$ cuando $\gamma \rightarrow 0$.

$$\lim_{\gamma \rightarrow 0} ER(\gamma) = 0$$

Es evidente que un estimador cumpla con la propiedad VRE asegura que cumpla con la propiedad BRE, pero el recíproco no se cumple.

A continuación se presenta el método CMC enfocado a la estimación de la medida de confiabilidad clásica de una red.

3.2. Método Monte Carlo Crudo

Dado un grafo $G = (V, E)$ con $V = (v_1, \dots, v_n)$ y $E = (e_1, \dots, e_m)$ que modela por ejemplo una red de telecomunicaciones, se puede estar interesado en calcular la probabilidad de que un subconjunto K de terminales de dicha red estén comunicados entre sí. A esta probabilidad se la denomina K -confiabilidad. Es relevante recordar que en este documento se hace el supuesto de que los enlaces son los únicos que pueden experimentar fallas, es decir, los nodos son perfectos, siempre están operativos.

El método Monte Carlo Crudo tiene un buen desempeño en este tipo de cálculos, aunque tiene el inconveniente de que el Error Relativo asociado al mismo tiende a infinito cuando la probabilidad de un evento raro tiende a cero. La propiedad de error relativo acotado o BRE no se cumple en CMC cuando la red es altamente confiable, ya que el error relativo tiende a infinito. El vector aleatorio de estado de la red se denota $X = (X_1, \dots, X_m)$, donde $X_i = 1$ si e_i se encuentra operativo, y su valor es 0 en caso contrario.

La técnica empleada por este método consiste en generar N réplicas aleatorias e independientes de vectores de estado $(X^{(i)})$ de la red, donde $i = 1, \dots, N$. Para generar cada componente del vector de estado $(X^{(i)})$ se realiza un sorteo utilizando una variable Uniforme $(0,1)$, se analiza el valor de la variable Uniforme (U) , se establece que si el valor de la misma es menor o igual a la probabilidad de que el enlace j esté operativo el mismo es etiquetado operativo, y en caso contrario es definido en estado de falla, con valores 1 y 0 respectivamente.

Mediante CMC se construye un estimador de la medida de confiabilidad en base a la proporción de estados operativos que fueron sorteados de la cantidad de sorteos efectuados. De modo que el estimador insesgado de la confiabilidad K -terminal del sistema puede ser planteado de la siguiente forma:

$$\hat{R}_k(G) = \frac{1}{N} \sum_{i=1}^N \phi(X^{(i)}) \quad (3.1)$$

Donde $\phi: \{0,1\}^m \rightarrow \{0,1\}$ es la función estructura, tal que vale 1 si el sistema se encuentra operativo y cero en caso contrario, $R_k(G) = P(\phi(X) = 1) = E\{\phi(X)\}$ y las replicaciones son independientes. Si se definen las variables Y^1, \dots, Y^N de modo que $Y^i = \phi(X^{(i)})$, como la función ϕ es medible y además la muestra X^1, \dots, X^N es i.i.d, entonces Y^1, \dots, Y^N es también una muestra i.i.d. Se puede afirmar que \bar{Y} converge casi seguramente a $E(Y^i)$ en base a la Ley Fuerte de los Grandes Números. La Ley establece que si se tiene una sucesión de

variables aleatorias i.i.d, donde $E(|X_i|) < \infty$ y tienen valor esperado μ , entonces el promedio de las variables aleatorias converge casi seguramente a μ . El estimador por este método es un estimador insesgado dado que es un promedio muestral:

$$E(\hat{R}_k(G)) = E\left(\frac{1}{N} \sum_{i=1}^N \phi(X^{(i)})\right) = \frac{\sum_{i=1}^N E[\phi(X^{(i)})]}{N} = \frac{NE[\phi(X)]}{N} = \frac{NR_k(G)}{N} = R_k(G).$$

La varianza del estimador que coincide con el error cuadrático medio y la estimación de la varianza son:

$$\text{Var}(\hat{R}_k(G)) = \frac{\text{Var}(\phi(X))}{N} = \frac{R_k(G)Q_k(G)}{N}. \quad (3.2)$$

$$\hat{V} = \frac{\hat{R}_k(G)(1 - \hat{R}_k(G))}{(N - 1)} = \frac{1}{N(N - 1)} \sum_{i=1}^N (\phi(X^{(i)}) - \hat{R}_k(G))^2. \quad (3.3)$$

Siendo $Q_k(G)$ la medida clásica de anticonfiabilidad:

$$Q_k(G) = P(\phi(X) = 0) = E\{1 - \phi(X)\} = 1 - R_k(G).$$

Es fundamental recordar que un estimador sea Consistente implica que a medida que crece el tamaño de la muestra las estimaciones que nos proporciona el mismo se aproximan cada vez más al verdadero valor del parámetro que se desea estimar, o sea, que converge casi seguramente al valor que se desea estimar cuando el tamaño de muestra tiende a infinito. Como el estimador es insesgado, que es una condición más fuerte que consistencia, el estimador resultante de CMC es un estimador consistente.

A continuación se presenta una descripción del algoritmo utilizado por esta técnica. Como entradas se tienen: el grafo $G(V, E)$, los nodos objetivos K y las probabilidades de que los enlaces estén operativos, la cantidad de iteraciones N , basadas en la precisión y tiempo de ejecución establecidos. Como salida se obtiene un estimador de la medida de confiabilidad $R_k(G)$ así como su varianza. El algoritmo aplicado en este método es:

Algorithm 1 Monte Carlo

 INPUT: $N, G = (V, E), P = (p_1, \dots, p_m)$

```

1: C = 0
2: for  $i = 1:N$  do
3:   for  $l = 1:m$  do
4:     Sortear una muestra  $u$  Uniforme  $(0,1)$ ;
5:     if  $(u \in [0, p_l])$  then
6:        $x_l^{(i)} = 1$ 
7:     else
8:        $x_l^{(i)} = 0$ 
9:     end if
10:  end for
11:  Evaluar  $\phi(x^{(i)})$ 
12:   $C = C + \phi(x^{(i)})$ 
13: end for
14: Estimar  $R_k(G) : \hat{R}_k(G) = \frac{C}{N}$ 
15: Estimar  $Var(\hat{R}_k(G)) : \hat{V} = \frac{\hat{R}_k(G)(1-\hat{R}_k(G))}{(N-1)}$ 
16: return
17:  $\hat{R}_k(G), \hat{V}(\hat{R}_k(G))$ 

```

En una primera etapa se genera una variable de conteo C , que va acumulando la cantidad de estados operativos de G que fueron sorteados, el número de iteraciones va a coincidir con el tamaño de la muestra N . Luego se realiza el sorteo del vector de estados de G entre los pasos 3 y 10. En la línea 4 del algoritmo se genera una variable Uniforme $U(0, 1)$ para cada enlace del grafo donde si el valor de la Uniforme es menor o igual a la probabilidad de que el enlace se encuentre operativo el mismo se etiqueta como tal, y en caso contrario se etiqueta como en estado de falla, $x_l^{(i)} = 1$ y $x_l^{(i)} = 0$ respectivamente. En el paso 12, si el estado que es sorteado fue clasificado como operativo entonces la variable de conteo aumenta en una unidad, y en caso contrario permanece igual que antes. Finalmente, se calcula el estimador de la confiabilidad así como también la estimación de la varianza. Como resultado $\hat{R}_k(G)$ es equivalente a la proporción de estados que resultaron operativos en el sorteo (acumulados en C) en relación al número total de iteraciones realizadas, es decir, es la frecuencia de subredes en donde se verifica que los K nodos están conectados. En cada muestra la complejidad de cálculo es: $O(\max(|E|, |V|))$ que es lo mismo que $O(|E|)$.

Como fue antes mencionado, el método Monte Carlo Crudo tiene como principal desventaja frente a otras técnicas que el error relativo crece en forma no acotada cuando R_k es cercana a 1, creando la necesidad de realizar un alto número de experimentos para poder reducirlo.

Error Relativo en CMC

Si γ es la probabilidad asociada a un evento R que se desea estimar mediante CMC, entonces se realizan N muestras aleatorias e independientes del modelo almacenando cuando R

ocurre o no. Como $X_i = 1$ si R ocurre en la muestra i , y $X_i = 0$ en caso contrario, la estimación de γ es obtenida mediante el promediado de Bernoullis, siendo $\hat{\gamma} = (X_1 + \dots + X_N)/N$.

La esperanza y varianza de X_i son: $E(X_i) = \gamma$ y $Var(X_i) = \gamma(1 - \gamma) = \sigma^2$. En base al Teorema Central del Límite, se tiene que $X_1 + \dots + X_N$ es aproximadamente normal cuando N es lo suficientemente grande, entonces $E(X_1 + \dots + X_N) = N\gamma$ y $Var(X_1 + \dots + X_N) = N\sigma^2$ por lo tanto la variable $Z = [N\sigma^2]^{-\frac{1}{2}}(X_1 + \dots + X_N - N\gamma)$ tiene media 0 y varianza 1, cuando $N \rightarrow \infty$ se llega a que $Z \rightarrow N(0, 1)$.

En el caso de que R sea un evento raro, $\gamma \rightarrow 0$, el Error Relativo (ER) [31], tiende a infinito:

$$ER = z \frac{\sqrt{\gamma(1-\gamma)}}{\sqrt{N}\gamma} \approx \frac{z}{\sqrt{N}\sqrt{\gamma}} \rightarrow \infty,$$

por lo tanto CMC no verifica la propiedad de BRE [24], lo cual es el denominado *problema central de simulación de eventos raros*. El pobre desempeño de CMC en este contexto ha motivado a la investigación y desarrollo en relación a la simulación de eventos raros.

A continuación se presenta un ejemplo numérico del cálculo del Error Relativo.

Si $\hat{\gamma}$ es un estimador cuya varianza es σ^2 y z_α el valor asociado a la distribución normal estándar, donde $z_\alpha = \phi^{-1}(1 - \alpha)$, se define el error relativo asociado a $\hat{\gamma}$ como:

$$ER = z_\alpha \frac{\sigma_n}{\gamma}.$$

En CMC se llega a que:

$$ER = z_\alpha \frac{\sigma_n}{\hat{Q}} \approx \frac{1}{(N\hat{Q})^{\frac{1}{2}}}.$$

Siendo $\sigma_n = \sqrt{\frac{\hat{Q}(1-\hat{Q})}{N}}$, decreciendo a medida que \sqrt{N} aumenta.

Si se desea calcular el menor tamaño de muestra N para lograr un $RE < 10^{-2}$ cuando $Q = 10^{-10}$, se despeja N y se obtiene :

$$ER = \frac{1}{(N10^{-10})^{\frac{1}{2}}} < 10^{-2} \Rightarrow N > 10^{14}$$

Entonces el menor tamaño de muestra N debería ser $10^{14} + 1$. Un resultado con estas características era de esperar cuando se utiliza el método Monte Carlo Crudo, ya que el gran problema que presenta este método es no cumplir con la propiedad BRE cuando la red es altamente confiable, necesitando un número elevado de muestras.

Es importante resaltar que el interés por utilizar diferentes tipos de métodos de estimación radica en que al ser los mismos insesgados, su varianza coincide con el error cuadrático medio, lo cual hace necesario comparar las varianzas de los distintos métodos estadísticos. A la hora de optar por un método de estimación insesgado, es ventajoso hacerlo por el que tenga menor varianza asociada. Se buscará entonces obtener resultados precisos y con un menor costo computacional asociado.

3.3. Muestreo por Importancia.

El pobre desempeño de CMC en el contexto de la simulación de eventos raros, que es debido a que necesita un elevado número de replicaciones para obtener una precisión aceptable, ha impulsado el estudio y desarrollo de métodos más eficientes en el área de la simulación de este tipo de eventos. En el presente cada vez son más las redes que poseen una baja probabilidad de fallar, pero cuando ocurre los resultados pueden ser muy negativos; por lo cual fue necesario encontrar otros métodos con un mejor desempeño al estimar la confiabilidad de una red cuando la falla de la misma es un evento raro.

Existen técnicas que logran mejorar la eficiencia del estimador a partir de una reducción en la varianza las cuales se ubican dentro de las técnicas de reducción de varianza. Se logra una precisión dada utilizando una cantidad menor de replicaciones,. Es importante destacar que al momento de evaluar tanto la reducción en la varianza así como la eficiencia, se toma como referencia al método CMC.

La concepción de Muestreo por Importancia, abreviado IS por sus siglas inglesas [24, 31], apunta a lidiar con el problema central de eventos raros. Cuando se aplica el método IS al cálculo de la medida de confiabilidad en redes, el mismo permite obtener una considerable reducción de la varianza a partir del muestreo de estados de la red en base a un cambio de probabilidad del evento raro (falla de la red). La idea detrás de este método es asignar mayor probabilidad de ocurrencia al evento raro, de modo que el mismo va a ocurrir ahora con más frecuencia en base a la nueva Ley de probabilidades \tilde{P} .

Una vez realizado el cambio de probabilidad al fallo de la red, es necesario corregir los resultados que se obtienen mediante el cociente de verosimilitud entre P y \tilde{P} , logrando de ese modo un estimador insesgado al realizar la estimación de la medida de confiabilidad de una red, donde $R_k(G) = E(\phi(X))$. En este caso se trabajará con la anticonfiabilidad, dado que se define la función estructura de la siguiente manera: $\phi(X) = 1$ si la red falla y $\phi(X) = 0$ si está operativa. La expresión para la anticonfiabilidad sea entonces $Q_k(G) = E(\phi(X))$, donde X es una variable aleatoria con función de cuantía P . Si $X \in R^n$ y las $X^{(i)}$ son i.i.d con distribución P , entonces el valor que se desea calcular es:

$$E(\phi(X)) = \sum_{x \in \Omega_G} \phi(X)P(X = x),$$

siendo Ω_G es el espacio de los vectores de estado $\in G$.

Cabe recordar que el estimador obtenido por el método Monte Carlo era: $\frac{1}{N} \sum_{i=1}^N \phi(X^{(i)})$, en IS como fue mencionado previamente se utiliza una nueva distribución \tilde{P} , obteniendo:

$$E(\phi(X)) = \sum_{x \in \Omega_G} \phi(X)P(X = x) = \sum_{x \in \Omega_G} \phi(X)P(X = x) \frac{\tilde{P}(X = x)}{\tilde{P}(X = x)} =$$

$$\sum_{x \in \Omega_G} \phi(X)L(X)\tilde{P}(X = x) = \tilde{E}(\phi(X)L(x)).$$

Donde \tilde{E} es el valor esperado en base a \tilde{P} , $\phi(X)$ la función de estructura y $L(X)$ es el cociente de verosimilitud de P y \tilde{P} :

$$L(X) = \frac{P(X = x)}{\tilde{P}(X = x)} \quad (3.4)$$

El nuevo estimador insesgado para $E(\phi(X))$ es:

$$\frac{1}{N} \sum_{i=1}^N \phi(X^{(i)}) L(X^{(i)}) \quad (3.5)$$

La simulación es realizada en base a la ley de probabilidad original mediante $L(X)$. Las varianzas correspondientes son :

$$\tilde{\sigma}^2(\phi(X)L(X)) = \tilde{E}[(\phi(X)L(x))^2] - (E[(\phi(X))])^2.$$

$$\sigma^2(\phi(X)) = E[\phi(X)^2] - (E(\phi(X)))^2.$$

En este método se busca encontrar un cambio de medida de P , es decir, encontrar \tilde{P} que de como resultado un estimador de menor varianza que el método Monte Carlo.

El cambio de medida óptimo mediante el cual se llega a un estimador de **varianza cero** es el siguiente:

$$L(X) = \frac{\partial P}{\partial \tilde{P}} = \frac{E[|\phi(X)|]}{|\phi(X)|}, \quad (3.6)$$

que es igual a decir

$$\tilde{P}(X) = \frac{P(X) |\phi(X)|}{E[|\phi(X)|]}. \quad (3.7)$$

Cabe destacar que cuando $\phi(X) \geq 0$, al realizar el cambio de medida óptimo se llega a:

$$\tilde{E}[(\phi(X)L(x))^2] = \tilde{E}[(\phi(X) \frac{E[|\phi(X)|]}{\phi(X)})^2] = (E[(\phi(X))])^2. \quad (3.8)$$

Por lo tanto, $\tilde{\sigma}^2(\phi(X)L(X)) = 0$, dando lugar a un estimador de varianza cero y dado que es un estimador insesgado es un estimador consistente. La formulación "*ideal*" del método verificaría las propiedades BRE [24] y VRE [24], aunque sería trivial dado que en una sola muestra se obtendría el valor que se desea estimar.

En conclusión, mediante este método se puede encontrar un cambio de medida óptimo que de como resultado un estimador de varianza cero, pero se tiene que conocer $E(\phi(X))$, que es justamente lo que queremos estimar siendo la **principal dificultad** que enfrenta el método, por lo tanto no se puede obtener con exactitud.

3.4. Método secuencial IS.

En esta sección se realiza una descripción en base al artículo de L'Ecuyer et al [24], el lector puede encontrar también valiosa información en el libro de Geral Folland [14]. El cambio de medida basa en un muestreo secuencial del estado de las aristas reformulando el muestreo del vector de estados como una cadena de Markov. Se genera un orden para los estados de los links X_1, \dots, X_m , siendo X_i la coordenada generada en el paso i de la cadena. La probabilidad de falla del link es definida como $P(X_i = 0) = 1 - P(X_i = 1) = q_i$, donde las X_i son independientes $\forall i$. Se cambia la probabilidad q_i en cada paso haciendo que dependan de los valores anteriormente generados X_1, \dots, X_{i-1} . Mediante esta técnica se puede llegar a un estimador de varianza cero.

Se define en base al esquema teórico de varianza cero:

$$u_i(x_1, \dots, x_{i-1}) = E[\phi(X)/X_1 = x_1, \dots, X_{i-1} = x_{i-1}].$$

La expresión anterior representa la anticonfiabilidad del grafo G' , definido como el grafo condicionado al estado de las aristas $\{e_1, \dots, e_{i-1}\}$ y u es la medida de anticonfiabilidad del mismo, siendo:

$$u_i(x_1, \dots, x_{i-1}) = q_i u_{i+1}(x_1, \dots, x_{i-1}, 0) + (1 - q_i) u_{i+1}(x_1, \dots, x_{i-1}, 1).$$

La cuantía utilizada para realizar el sorteo de la arista en el paso i es:

$$\tilde{q}_i = P[X_i = 0/X_1 = x_1, \dots, X_{i-1} = x_{i-1}] = \frac{q_i u_{i+1}(x_1, \dots, x_{i-1}, 0)}{u_i(x_1, \dots, x_{i-1})},$$

$$\tilde{p}_i = 1 - \tilde{q}_i = P[X_i = 1/X_1 = x_1, \dots, X_{i-1} = x_{i-1}] = \frac{(1 - q_i) u_{i+1}(x_1, \dots, x_{i-1}, 1)}{u_i(x_1, \dots, x_{i-1})}.$$

El cociente de verosimilitud en el paso i entre P y \tilde{P} es :

$$L_i(x_i) = x_i \frac{1 - q_i}{1 - \tilde{q}_i} + (1 - x_i) \frac{q_i}{\tilde{q}_i} = \frac{u_i(x_1, \dots, x_{i-1})}{u_{i+1}(x_1, \dots, x_{i-1}, x_i)},$$

$$L(x) = \prod_{i=1}^m L_i(x_i),$$

siendo $\phi(X)L(X)$ el estimador insesgado final.

En base a este esquema de muestreo secuencial, donde las probabilidades q_i son reemplazadas por \tilde{q}_i , se tiene que $\phi(X) = 1$, con probabilidad 1, lo que es igual a generar sólo estados de falla de la red y $L(x) = u = Q_k(G)$ con probabilidad 1, ver [24]. Lo anterior se traduce a que $\phi(X)L(X) = u$. Como resultado se obtiene un estimador insesgado de varianza cero,

por lo tanto es consistente. Como la resolución exacta no puede ser aplicada dado que la nueva cuantía \tilde{P} depende del parámetro a estimar, una solución es aproximar el valor de dicho parámetro utilizando la probabilidad del conjunto de corte de máxima probabilidad condicionado al estado de las aristas que fueron previamente sorteadas.

Se llega a que :

$$u_i(x_1, \dots, x_{i-1}) = P[C(G/x_1, \dots, x_{i-1})],$$

donde $C(G/x_1, \dots, x_{i-1})$ es un mincut del grafo G , condicionado al estado de las aristas e_1, \dots, e_{i-1} . Dada \tilde{q}_i la nueva probabilidad de falla en base a la distribución \tilde{P} , junto con el reemplazo de $E(\phi(X))$ por la probabilidad máxima de un conjunto de corte en \tilde{q}_i , se llega a una versión aproximada del método llamada **AZVIS** (Approximate Zero-Variance Importance Sampling).

La aproximación a varianza cero consiste en reemplazar la función u_i por una aproximación \hat{u}_i que sea más fácil de calcular.

$$\tilde{q}_i = \tilde{P}[X_i = 0] = \frac{q_i \hat{u}_{i+1}(x_1, \dots, x_{i-1}, 0)}{q_i \hat{u}_{i+1}(x_1, \dots, x_{i-1}, 0) + (1 - q_i) \hat{u}_{i+1}(x_1, \dots, x_{i-1}, 1)}. \quad (3.9)$$

De esta manera, el muestreo es realizado con probabilidades que no están muy alejadas de las óptimas, dando como resultado una significativa reducción de la varianza.

A continuación se presentan algunas definiciones que serán utilizadas posteriormente:

- $q_i = a_i \epsilon^{b_i}$, siendo ϵ el parámetro de rareza.
- $u = u(\epsilon) = \Theta(\epsilon^c)$
- si se tiene dos funciones $f, g : (0, \infty) \rightarrow \mathfrak{R}$, entonces $f(\epsilon) = o(g(\epsilon))$ si $\lim_{\epsilon \rightarrow 0^+} \frac{f(\epsilon)}{g(\epsilon)} = 0$.
- $f(\epsilon) = \underline{O}(g(\epsilon))$ si $|f(\epsilon)| \geq bg(\epsilon)$, para alguna constante $b > 0 \forall \epsilon$ suficientemente pequeño.
- $f(\epsilon) = \underline{O}(g(\epsilon))$ si $|f(\epsilon)| \leq cg(\epsilon)$ para alguna constante $c > 0 \forall \epsilon$ suficientemente pequeño.
- $f(\epsilon) = \Theta(g(\epsilon))$ si $f(\epsilon) = \underline{O}(\epsilon^d)$ y $f(\epsilon) = \underline{O}(\epsilon^d)$
- $S_1 = \{x \in \{0, 1\}^m : \phi(x) = 1 \text{ y } \tilde{P}[X = x] = \Theta(1)\}$

En las líneas siguientes se exponen dos teoremas relacionados con las condiciones suficientes que permiten que esta técnica cumpla con las propiedades BRE y VRE.

Teorema 3.4.1 Para cada i y $(x_1, \dots, x_i) \in \{0, 1\}^i$, $1 \leq i \leq m$, existe una constante $a_{i+1}(x_1, \dots, x_i)$ que es independiente del parámetro de rareza ϵ de modo tal que:

$$\hat{u}_{i+1}(x_1, \dots, x_i) = a_{i+1}(x_1, \dots, x_i)u_{i+1}(x_1, \dots, x_i) + o(u_{i+1}(x_1, \dots, x_i)),$$

entonces el estimador resultante del método IS luego de realizarse el reemplazo de u_i por una aproximación como se presenta en la ecuación (3.9), cumple con la propiedad BRE. La prueba puede ser estudiada en [24].

Teorema 3.4.2 Bajo el supuesto de que se cumple el Teorema (3.4.1) y que además para todo $x = (x_1, \dots, x_m) \in S_1$ y para cada i una de las siguientes condiciones se cumple:

$$\frac{\hat{u}_{i+1}(x_1, \dots, x_{i-1}, 1)}{u_{i+1}(x_1, \dots, x_{i-1}, 1)} = \frac{\hat{u}_{i+1}(x_1, \dots, x_{i-1}, 0)}{u_{i+1}(x_1, \dots, x_{i-1}, 0)} + o(1),$$

$$x_i = 0, a_{i+1}(x_1, \dots, x_i) = 1 \text{ y } (1 - q_i)\hat{u}_{i+1}(x_1, \dots, x_{i-1}, 1) = o(q_i\hat{u}_{i+1}(x_1, \dots, x_{i-1}, 0)),$$

$$x_i = 1, a_{i+1}(x_1, \dots, x_i) = 1 \text{ y } q_i\hat{u}_{i+1}(x_1, \dots, x_{i-1}, 0) = o((1 - q_i)\hat{u}_{i+1}(x_1, \dots, x_{i-1}, 1)),$$

el estimador bajo IS en base a (3.9) cumple con la propiedad VRE, la demostración también puede ser vista en [24].

Aproximación basada en cortes minimales.

El método para aproximar \hat{u}_{i+1} basado en cortes minimales “mincut-maxprob approximation” [24], es una herramienta de fácil aplicación, donde se requiere la identificación de conjuntos de corte de máxima probabilidad. Un corte o k -corte en un grafo G es un conjunto de aristas tal que al eliminarlas del grafo este queda desconectado. Un corte minimal (mincut), es un corte que no contiene otro corte que el mismo, es decir, si se mira un subconjunto estricto de esas aristas deja de ser un corte. Para cualquier corte γ , $E(\gamma)$ es el evento donde todos los enlaces de γ fallan. La red se encontrará en estado de falla si por lo menos uno de los cortes minimales tiene todos sus enlaces en estado de falla, por lo tanto, la anticonfiabilidad puede ser planteada como:

$$u = P\left[\bigcup_{\gamma} E(\gamma)\right]. \quad (3.10)$$

El sistema deja de estar operativo si por lo menos uno de los mincuts (cortes minimales) tiene todos sus enlaces en estado de falla.

$$P(E(\gamma)) = \prod_{i \in \gamma} q_i \quad (3.11)$$

Un corte minimal de máxima probabilidad, es un corte minimal γ de manera que:

$$\gamma = \operatorname{argmax}_{\gamma'} \{P[E(\gamma')]\} : \gamma' \text{ es un mincut de } G\}.$$

Si se llama H a la cantidad de cortes minimales de máxima probabilidad $p = P(E(\gamma))$, se

tiene que $p = \Theta(\epsilon^c)$ y $u = Hp + o(\epsilon^c)$.

Sea $G_i = G_i(x_1, \dots, x_i)$ el grafo obtenido luego de remover todos los links $j \leq i$ para cada $x_j = 0$ y exigiendo que los enlaces donde $x_i = 1$ estén operativos, C_i es el conjunto de mincuts de G_i que no contiene ningún enlace $j \leq i$ para $x_j = 1$. Se utiliza como una aproximación de $u_{i+1}(x_1, \dots, x_i)$ a:

$$P[E(\gamma_i)] = \max\{P[E(\gamma)] : \gamma \in C_i\}, \quad (3.12)$$

donde γ_i es un corte minimal de máxima probabilidad en C_i . La aproximación anterior es factible debido a que al ocurrir $E(\gamma_i)$ todos los links en γ_i fallan y el sistema se encuentra en estado de falla. Se deben rechazar los γ_i que contengan enlaces $j \leq i$, con $x_j = 1$ debido a que esos enlaces no están en estado de falla.

Para poder obtener \tilde{q}_i se necesita $\hat{u}_{i+1}(x_1, \dots, x_{i-1}, 0)$ y $\hat{u}_{i+1}(x_1, \dots, x_{i-1}, 1)$, por tal motivo es necesario construir un corte minimal $\gamma_i^- = \gamma_i$ para x_1, \dots, x_{i-1} con $x_i = 0$, siendo $\hat{u}_{i+1}(x_1, \dots, x_{i-1}, 0) = P[E(\gamma_i^-)]$. Luego se debe hacer lo mismo pero con $x_i = 1$ para encontrar un corte minimal $\gamma_i^+ = \gamma_i$ de modo que $\hat{u}_{i+1}(x_1, \dots, x_{i-1}, 1) = P[E(\gamma_i^+)]$. Es importante destacar que los cortes minimales anteriores pueden ser calculados en tiempo polinomial [35]. Una vez que se tiene \tilde{q}_i se genera X_i . Mediante la utilización de la aproximación basada en cortes minimales se llega a que se cumplen las propiedades BRE [24] y VRE [24]. A continuación se presenta el algoritmo perteneciente a AZVIS [16].

Algorithm 2 AZVIS

INPUT: $N, G = (V, E), (s, t), P = (p_1, \dots, p_m)$ y $\tilde{P} = (\tilde{p}_1, \dots, \tilde{p}_m)$

- 1: $S_1 = 0$
 - 2: $S_2 = 0$ {Iniciar S_1 y S_2 }
 - 3: **for** $j = 1:N$ **do**
 - 4: **for** $i = 1:m$ **do**
 - 5: Identificar mincut $(G/x_1, \dots, x_{i-1}, x_i = 1) = C_i$
 - 6: Calcular $P(C_i)$
 - 7: Identificar mincut $(G/x_1, \dots, x_{i-1}) = C_{i-1}$
 - 8: Calcular $P(C_{i-1})$
 - 9: Calcular $\tilde{P}_i = 1 - \frac{p_i \cdot P(C_i=1)}{P(C_i)}$
 - 10: Sortear $X_i^j \sim \text{Bernoulli}(\tilde{p}_i)$
 - 11: **end for**
 - 12: $L(X^{(j)}) = \frac{\tilde{P}(X^{(j)})}{P(X^{(j)})}$
 - 13: $S_1 = S_1 + \Phi(X^{(j)})L(X^{(j)})$
 - 14: $S_2 = S_2 + (\Phi(X^{(j)})L(X^{(j)}))^2$
 - 15: **end for**
 - 16: $\hat{Q}_k(G) = \frac{S_1}{N}$ {Estimador puntual de $Q_k(G)$ }
 - 17: $\hat{V}(\hat{Q}_k(G)) = \frac{S_2 - \frac{S_1^2}{N}}{N(N-1)}$ {Estimador puntual de la varianza de $\hat{Q}_k(G)$ }
 - 18: **return** $\hat{Q}_k(G) \hat{V}(\hat{Q}_k(G))$
-

En el algoritmo anterior la idea general es la realización del sorteo del estado global de la red N veces, donde en cada réplica se realiza una muestra secuencial del estado de las aristas en base a la probabilidad \tilde{q} , la cual está sujeta al estado de las aristas que fueron sorteadas previamente. Es necesario corregir la proporción de estados de falla de la red por el cociente de verosimilitud para poder realizar la estimación de $Q_k(G)$. Al principio del algoritmo se inicializan las variables S_1 y S_2 en las cuales se acumulan los resultados obtenidos correspondientes a la estimación de $Q_k(G)$ y la varianza $\hat{V}(\hat{Q}_k(G))$. Se sortea el estado de cada arista en cada replicación. Bajo el supuesto de que la arista e_i se encuentra operativa, se identifica un mincut de G en el paso 5 restringido al estado de las aristas que fueron ya sorteadas. Se calcula la probabilidad del mincut anterior en el paso 6. Luego, en el paso 7 se identifica un nuevo mincut, esta vez condicionado al estado de las aristas que fueron previamente sorteadas, computándose su probabilidad en el paso 8. Se calcula \tilde{p} en el paso 9, que es la probabilidad de que la arista e_i se encuentre operativa bajo la nueva ley de probabilidad \tilde{P} tomando en cuenta las probabilidades de los mincuts que fueron identificados previamente. Utilizando una distribución Bernoulli con parámetro \tilde{p}_i se sortea el estado de la arista e_i en el paso 10 del algoritmo. Se calcula el cociente de verosimilitud $L(X^{(j)})$ para la j -ésima iteración en el paso 12, luego se acumulan los resultados de S_1 y S_2 . Este método aplica una búsqueda a lo ancho (BFS) [17]. Finalmente se calcula $\tilde{Q}_k(G)$ y $\hat{V}(\hat{Q}_k(G))$. Es importante destacar que el método iterativo AZVIS posee un tiempo de ejecución de gran tamaño, siendo superior al asociado al CMC para el mismo tamaño de muestra.

3.5. Muestreo Basado en Cotas.

[28]

El Muestreo Basado en Cotas [31], utiliza en su procedimiento dos funciones Φ^L y Φ^U acotando la función de estructura Φ inferior y superiormente, cuando las mismas son conocidas este método puede ser utilizado en cualquier problema donde se desee evaluar la confiabilidad. Dichas funciones deben cumplir:

$$\Phi^L(x) \leq \Phi(x) \leq \Phi^U(x), \forall x \in \Omega_G.$$

Se define a Ω_G como el espacio de vectores de estado del grafo $G(V, E)$ de configuración X . El método se basa en trabajar en un espacio menor mediante la utilización de las cotas antes mencionadas, de modo de poder utilizar un subespacio de Ω_G , realizando un muestreo en el mismo.

Es importante destacar que Van Slyke y H. Frank [36] en 1971, fueron quienes realizaron la primer aplicación de este método en el contexto de confiabilidad de redes, seguidos por Kummamoto [22] y Fishman [13].

La probabilidad para cada cota que su valor sea 1, condicionada al estado de k componentes es la siguiente:

$$R_k^L(\tilde{X}^{(k)}) \equiv P(\Phi^L(X) = 1 \mid X_1 = \tilde{x}_1, \dots, X_k = \tilde{x}_k)$$

$$R_k^U(\tilde{X}^{(k)}) \equiv P(\Phi^U(X) = 1 \mid X_1 = \tilde{x}_1, \dots, X_k = \tilde{x}_k)$$

Donde $k = 0, \dots, m$ y $\tilde{X}^{(k)} = (\tilde{X}_1, \dots, \tilde{X}_k)$ es una asignación de valores para los primeros k componentes, pudiendo ser las probabilidades anteriores calculadas en tiempo polinomial. Si se trabaja con $k = 0$ entonces R_0^L y $1 - R_0^U$ son los valores correspondientes a las probabilidades de eventos conocidos, donde la función de estructura tiene valores 1 y 0 respectivamente, siendo: $R_0^L = P(\Phi^L(X) = 1)$ y $R_0^U = P(\Phi^U(X) = 1)$.

Para aplicar este método se debe definir un espacio sobrante, en el mismo las muestras son extraídas de manera proporcional a su probabilidad en el espacio original, logrando obtener un estimador del sistema a partir del obtenido en el subespacio. Se define dicho espacio como:

$$W = \{X : \Phi^L(X) = 0, \Phi^U(X) = 1\}.$$

En este espacio la cota inferior vale 0 y la superior 1. Para obtener cada componente \tilde{X}_i del vector de estados \tilde{X} se realiza un sorteo con la siguiente probabilidad:

$$\tilde{P}_i = P(X_i = 1 \mid X_1 = \tilde{x}_1, \dots, X_{i-1} = \tilde{x}_{i-1}, \Phi^U(\tilde{X}) = 1, \Phi^L(\tilde{X}) = 0) =$$

$$\left[\frac{R_i^U(\tilde{X}^{(i-1)}) - R_i^L(\tilde{X}^{(i-1)})}{R_{i-1}^U(\tilde{X}^{(i-1)}) - R_{i-1}^L(\tilde{X}^{(i-1)})} \right] * P_i.$$

El estimador que se construye a partir de la muestra para estimar la confiabilidad del sistema es el siguiente:

$$\hat{R}_k(G) = R_0^L + \phi(\tilde{X})(R_0^U - R_0^L).$$

Como:

$R_0^L = P(\Phi^L(X) = 1) = E(\Phi^L(X))$ y $R_0^U = P(\Phi^U(X) = 1) = E(\Phi^U(X))$, además $\Phi^L(X) = 0$ y $\Phi^U(X) = 1$, entonces $E(\Phi^L(X)) = 0 = R_0^L$ y $E(\Phi^U(X)) = 1 = R_0^U$, al calcular la esperanza del estimador se llega a :

$$E(\hat{R}_k(G)) = E(R_0^L + \phi(\tilde{X})(R_0^U - R_0^L)) =$$

$$E(R_0^L) + E(\phi(\tilde{X})(R_0^U - R_0^L)) = E(R_0^L) + E(\phi(\tilde{X})(R_0^U)) - E(\phi(\tilde{X})(R_0^L)) =$$

$$E(\phi(\tilde{X})) = R_k(G).$$

La varianza correspondiente a cada muestra es la siguiente:

$$Var = R_k(R_0^U - R_k) - R_0^L(R_0^U - R_k) = R_k(1 - R_k) - (1 - R_0^U)R_k - R_0^L(R_0^U - R_k).$$

Mediante este método se obtiene un estimador insesgado cuya varianza es menor que la obtenida por el método Monte Carlo Crudo que era $R_k(1 - R_k)$, como el estimador es insesgado es por lo tanto consistente. La diferencia entre estas dos varianzas va a depender del grado de precisión de las cotas y el tiempo de ejecución dependerá de la dificultad a la hora de calcular las funciones de las cotas. Se obtiene una reducción en la varianza directamente proporcional a la porción de probabilidad total que queda contenida en W .

El algoritmo empleado por este método es:

Algorithm 3 *Muestreo basado en cotas*

INPUT:

grafo G ,conjunto objetivo de nodos K , R_0^L y R_0^U .

RETURN:

 $\hat{R}_K(G)$ 1: **for** $i = 1 : m$ **do**2: $\tilde{X}_i = \text{Bernoulli}(\tilde{p}_i)$ 3: **end for**4: $\hat{R}_k(G) = R_0^L + \phi(\tilde{X})(R_0^U - R_0^L)$.

El algoritmo anterior se basa en el sorteo de una variable aleatoria Bernoulli para cada enlace, donde la probabilidad de éxito del i -ésimo enlace es \tilde{p}_i , generando así el vector \tilde{X} . Finalmente la medida de confiabilidad es estimada en base a la muestra mediante la ecuación que fue previamente presentada.

3.6. Muestreo Dagger

Este método ha sido formulado por Kumamoto et al [23]. El lector puede acceder a información acerca de este tema en [4, 28, 31], en la cual también se basa este documento. El método utiliza variables antitéticas logrando una reducción del tiempo de cómputo en relación a CMC. Se combina éxito y fracaso de los estados, generando de esa forma vectores de estado correlacionados negativamente. Es importante mencionar antes de continuar la definición de dos variables antitéticas, que se basa en tomar un vector aleatorio $(X^{(1)}, X^{(2)})$, donde $X^{(1)}$ y $X^{(2)}$ se distribuyen Bernoulli(p), entonces:

$$X^{(i)} = (1 - X)1_{U \in I^{(i)}} + X1_{U \notin I^{(i)}},$$

con $X = 1_{\{p \geq 0,5\}}$.

La idea detrás del Muestreo Dagger es generar bloques de muestras de tamaño L , de modo que se busca que exista correlación negativa entre las muestras individuales. El tamaño L se fija para cada bloque de manera para cada arista e se pueda particionar la secuencia de las L muestras en L_e sub-bloques de tamaño $M_e = \frac{L}{N_e}, \forall e \in E$, donde:

$$N_e = \lfloor \frac{1}{q_e} \rfloor \text{ y } L = mcm\{N_e : e \in E\},$$

siendo q_e la anticonfiabilidad elemental correspondiente al enlace e .

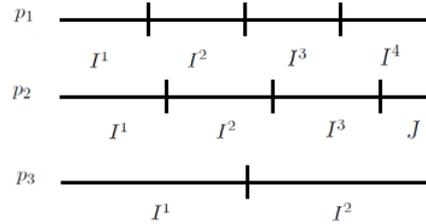
A cada una de las aristas se le asignan L_e intervalos de amplitud q denominados $I^{(i)}$ con $i = 1, 2, \dots, L_e$, de forma tal de cubrir todo el intervalo $[0, 1]$ con la unión de los mismos, puede suceder que quede un intervalo que tenga amplitud menor a q , que se denota

$$J = [0, 1] - \bigcup_{i=1}^{L_e} I^{(i)}.$$

La técnica utilizada en este método se basa en sortear una variable aleatoria Uniforme $(0,1)$ para cada arista perteneciente a cada sub-bloque, definiendo un parámetro $f = \lceil \frac{U}{q_e} \rceil$ de modo que la arista e únicamente falla en el elemento f de cada sub-bloque. En base a este procedimiento se llega a que la frecuencia de fallas de cada arista es proporcional a q_e . Para obtener una estimación de R_k se deben recorrer todas las muestras del bloque, registrando en cuales el grafo es K -conexo y cuando no.

Si se cuenta por ejemplo con un grafo $G = (V, E)$ con tres enlaces, $E = \{e_1, e_2, e_3\}$, donde $p_1 = 0,75$, $p_2 = 0,70$, y $p_3 = 0,5$, se obtienen los siguientes valores:

- $q_1 = 0,25$, $q_2 = 0,30$ y $q_3 = 0,50$
- $N_1 = 4$, $N_2 = 3$ y $N_3 = 2$
- $M_1 = 3$, $M_2 = 4$ y $M_3 = 6$
- $L = mcm\{N_e : e \in E\} = 12$



En el segundo enlace existe un intervalo remanente J , mientras que para el resto de los enlaces

$$\bigcup_{i=1}^{L_e} I^{(i)} = [0, 1].$$

A continuación es presentado el algoritmo correspondiente al Muestreo Dagger.

Algorithm 4 *Muestreo Dagger*

INPUT: grafo G ,

conjunto K (nodos objetivo),

confiabilidad de los enlaces.

RETURN: $\hat{R}_K(G)$

- 1: Se calcula el vector de enteros N de modo que $N_e = \lfloor \frac{1}{q_e} \rfloor$, ($N_e : e \in E$).
 - 2: Seleccionar el tamaño de muestra $L = \text{mcm}\{N_e : e \in E\}$
 - 3: $T=0$
 - 4: **for** $e = 1:|E|$ **do**
 - 5: $|L_e| = \frac{L}{N_e}$
 - 6: **for** $i = 1:L_e$ **do**
 - 7: Sortear U Uniforme
 - 8: $f = \lceil \frac{U}{q_e} \rceil$
 - 9: **if** $f \leq N_e$ **then**
 - 10: $X_e^{(i)} = 0$
 - 11: **end if**
 - 12: **end for**
 - 13: **end for**
 - 14: **for** $1 : L$ **do**
 - 15: $T = T + \Phi(x^{(i)})$
 - 16: **end for**
 - 17: $R_K(G) = \frac{T}{L}$
-

En el algoritmo correspondiente al Muestreo Dagger se utilizan como parámetros de entrada: el grafo G , la confiabilidad de las aristas y el conjunto K , devolviendo una estimación de $R_K(G)$. En el primer paso del algoritmo es calculado el vector de enteros N en base al parámetro N_e . Posteriormente se elige el tamaño de muestra L , de modo tal que sea el mínimo común múltiplo entre los diferentes N_e . En el paso 3, se inicializa T que es una variable de conteo, que será posteriormente utilizada para ir contando el número de estados operativos de

la red. En los pasos 4 y 5 se presentan las iteraciones correspondientes a cada enlace e y también para cada sub-bloque. En los pasos siguientes se generan los valores L_e para los enlaces de G luego dentro del bloque de tamaño $\frac{L}{N_e}$ el enlace e se fija en estado operativo para todos los vectores pertenecientes al mismo. En la línea posterior se cambian de estado los enlaces en base a una variable aleatoria Uniforme (0,1) y se presenta f . Luego se calcula f , de forma tal que si $f \leq \frac{L}{N_e}$, entonces e falla en el elemento f correspondiente i -ésima submuestra de las L_e . Este procedimiento es reiterado en todos los enlaces de G . Finalizando el algoritmo se evalúa cada uno de los L estados generados y se cuenta el número de estados que resultaron operativos en T , obteniéndose el estimador de $R_K(G)$, siendo el porcentaje de estados operativos de G en relación al tamaño de muestra L .

Estimador basado en Muestreo Dagger

A continuación en base al texto de Leslie Murray [12] se presenta como llegar al estimador basado en este método. La expresión utilizada para determinar $\hat{R}_k(G)$ correspondiente a CMC toma en cuenta muestras o replicaciones que son elegidas en forma independiente. En Muestreo Dagger se levanta el supuesto anterior, introduciendo correlación negativa entre $(X^{(i)}, X^{(j)})$, donde también es posible obtener un estimador insesgado por lo tanto consistente $\hat{R}'_K(G)$, siendo este último más preciso que CMC.

Es importante destacar que este método se basa en la definición de la varianza de una suma de variables aleatorias:

$$V(X + Y) = V(X) + V(Y) + 2Cov(X, Y).$$

Al generar bloques con cierta cantidad de muestras L , de modo que exista correlación negativa entre $(X^{(i)}, X^{(j)})$ la covarianza es negativa, por tal motivo es que la varianza es menor que la obtenida por CMC. El estimador parcial es el siguiente:

$$\hat{R}_L = \frac{1}{L} \sum_{i=1}^L \phi(X^{(i)}).$$

La varianza del estimador es:

$$V(\hat{R}_L) = \frac{1}{L^2} \left\{ \sum_{1 \leq i \leq L} Var(\phi(X^{(i)})) + 2 \sum_{1 \leq i < j \leq L} Cov(\phi(X^{(i)}), \phi(X^{(j)})) \right\}.$$

Es significativo señalar que la correlación negativa debe darse entre $(\phi(X^{(i)}), \phi(X^{(j)}))$, pero dado que la función de estructura $\phi(X)$ es monótona, cualquier par $((X^{(i)}, X^{(j)}))$ que sea negativamente correlacionado implicará que en $(\phi(X^{(i)}), \phi(X^{(j)}))$ también exista correlación negativa. Cuando se tienen N muestras, es decir, se toman B conjuntos de L muestras,

$N = BL$, es posible expresar el estimador $\hat{R}'_k(G)$ en base al estimador parcial que toma valores diferentes para cada uno de los B conjuntos de L muestras, entonces:

$$\hat{R}_L^b = \frac{1}{L} \sum_{1 \leq i \leq L} \phi(X^{(i,b)}),$$

donde $b = 1, \dots, B$.

Se llega a que el estimador insesgado final es:

$$\hat{R}' = \frac{1}{B} \sum_{b=1}^B \hat{R}_L^b.$$

La varianza del estimador final puede expresarse como:

$$V(\hat{R}') = \frac{1}{N^2} \left\{ \sum_{i=1}^N \text{Var}(\phi(X^{(i)})) + \sum_{i \neq j} \text{Cov}(\phi(X^{(i)}), \phi(X^{(j)})) \right\}.$$

El primer sumando corresponde a la varianza generada por el método Monte Carlo. Las muestras utilizadas en CMC son independientes, entonces son no correlacionadas. Si se trabaja con muestras correlacionadas adecuadamente, es posible generar una disminución de la varianza y es lo que ocurre mediante la utilización de Muestreo Dagger. Observando la expresión de la varianza, como existe una dependencia entre $\phi(X^{(i)})$ y $\phi(X^{(j)})$ siendo negativa la suma de las covarianzas, se obtiene un estimador más preciso. Se puede afirmar entonces que la varianza correspondiente a este método es menor que la varianza generada por Monte Carlo debido a la correlación negativa que fue inducida entre muestras que pertenecen al mismo sub-bloque.

El Muestreo Dagger cuenta con un menor tiempo de ejecución si se lo compara con CMC. La reducción se debe a que el tiempo de cálculo en cada muestra es igual para cada método, $O(|E|)$, pero cuando se utiliza este método se necesita generar una menor cantidad de variables aleatorias en su ejecución. Se obtiene un estimador con la misma calidad que cuando las muestras eran independientes pero necesitando menos replicaciones.

Para cada bloque la complejidad computacional es $O(E \times L)$. La frecuencia de falla de cada enlace en este método es directamente proporcional a su tasa media de falla. Este método tiene un importante requerimiento de memoria, siendo desventajoso cuando se trabaja con ciertos grafos [12], debido a que se deben generar y almacenar L estados de la red.

3.7. Método de Reducción Recursiva de la Varianza (RVR)

Mediante la utilización de RVR es posible obtener una reducción en varianza en relación al método Monte Carlo Crudo.

Sus orígenes son atribuidos a Mohammed el Khadiri y Héctor Cancela [5], que luego realizan una nueva formulación reduciendo el tiempo de ejecución utilizando un muestreo multinomial [6]. La descripción del método en este trabajo es efectuada en base a [5]. La idea es realizar una descomposición en forma recursiva del sistema en base a particiones del espacio de estados. Se parte de un grafo $G(V, E)$, donde $\Omega_G = \{0, 1\}^m$ es el espacio de vectores de estado, siendo m el número de enlaces. Se busca estimar la confiabilidad k -terminal $R_{(K,G)}$ de un grafo $G(V, E)$ en base al cálculo de estimación de la anti-confiabilidad $Q_{(K,G)}$, siendo $R_{(K,G)} = 1 - Q_{(K,G)}$. En otras palabras, se desea estimar la confiabilidad de que un conjunto de K terminales pertenecientes a V se comuniquen entre sí, asumiendo que los únicos que pueden fallar son los enlaces. Dado un corte de la red $C = \{e_1, \dots, e_l\}$, donde las aristas siguen un orden $e_1 < e_2 < \dots < e_l$, existen dos eventos, por un lado que todas las aristas pertenecientes a C fallen, o que exista una primera arista e_l que funcione, de manera tal que todas las aristas $\{e_1, \dots, e_{l-1}\}$ fallan. La probabilidad de que un enlace, arista, falle se define como $q_l = 1 - p_l$, siendo p_l la probabilidad de que el mismo se encuentre operativo. La técnica consiste en crear una variable que es una combinación lineal de variables Bernoulli, utilizando eventos disjuntos para generar un operador recursivo que proporcione un estimador insesgado de la anticonfiabilidad.

Se define:

- A_C es el evento en el cual todos los componentes de C fallan.
- B_i es el evento en el cual e_i está operativa pero las aristas anteriores se encuentran en estado de falla, donde $P(B_i) = p_i \prod_{j < i} q_j$.
- $Q_C = \prod_{l \in C} q_l$, probabilidad de ocurrencia del corte C .
- $\phi: \{0,1\}^m \rightarrow \{0,1\}$ es la función de estructura que devuelve 1 si luego de eliminar las aristas que han fallado de la red la misma es k -conexa, y 0 si no lo es.
- $X = (X_1, \dots, X_m)$ con $m = |E|$ es el vector de estados de G , X_i es el i -ésimo elemento de X , de modo que si X_i es 1, entonces el i -ésimo enlace opera y es igual a 0 en caso contrario.
- $Y = 1 - \phi(X)$.

Es importante destacar que los eventos A_C y B_i son exhaustivos y mutuamente excluyentes, de modo tal que al condicionar el grafo G a esos eventos se pueda encontrar subgrafos en forma recursiva.

El grafo condicionado al evento B_i es $G_i = G - \{e_1, \dots, e_{i-1}\} * e_i$, se eliminan las aristas que han fallado y se contrae la arista e_i , es decir, se elimina la arista e_i y se fusionan los dos nodos extremos en uno solo, si alguno de los dos nodos era terminal, entonces luego de la fusión se mantiene el mismo terminal. Es importante tener presente que al trabajar con grafos

que presentan aristas múltiples la probabilidad de operación es la probabilidad de la unión, es decir, $1 - (1 - p)^2$, siendo p la probabilidad de operación de cada arista. Se define X_i como una variable aleatoria binaria, donde $P(X_i = x) = P(X = x|B_i)\forall x \in \{0, 1\}^m$, entonces $Y_i = 1 - \phi(X_i)$ es la indicatriz de la anticonfiabilidad condicionada a B_i .

$P(Y = 1)$ es la anticonfiabilidad, donde $\{Y = 1\} = A_C \cup \bigcup_{i=1}^l A_C^C \cap B_i \cap \{Y_i = 1\}$.

Se realizará una estimación de $P(Y = 1)$ en forma recursiva en base a los grafos G_i que son inducidos por el evento B_i . Se diferencian los casos en los que ocurre A_C y en los que no. Mediante la partición del intervalo $[0, 1]$ en l intervalos, donde $(J_i)_{1 \leq i \leq l}$ es la secuencia de intervalos ordenados de longitud $P(B_i)/(1 - Q_C)$ y la utilización de una variable Uniforme $(0,1)$, se podrá realizar una simulación de las probabilidades de B_i cuando no ocurre A_C .

Se define:

$$Z = Q_C + (1 - Q_C) \cdot \sum_{i=1}^l 1_{\{U \in J_i\}} \cdot Y_i, \quad (3.13)$$

Se probará que esta variable tiene igual esperanza que Y pero su varianza es menor. El conjunto de nodos que se desea que estén comunicados en el grafo G es K , pero en G_i es K_i . Si al contraer la arista e_i los nodos extremos son terminales, entonces K_i es diferente a G_i , en el caso en que uno solo de los dos nodos extremos es terminal o ninguno lo sea, $K_i = K$. El grafo G_i es K_i -conexo cuando en G_i existe una comunicación entre todo par de nodos pertenecientes a K_i .

Para poder estimar la anticonfiabilidad se define el siguiente operador recursivo $F_{(G,K)}$, que toma los siguientes valores:

- 1, si G no es K -conexo
- 0, si K es un conjunto unitario;
- $Q_{C(G,K)} + (1 - Q_{C(G,K)}) \cdot \sum_{i=1}^l 1_{\{U \in J_i\}} F_{(G_i, K_i)}$ en otro caso.

En RVR se llama N veces el operador recursivo $F_{(G,K)}$ y se almacenan los valores finales. En cada paso el conjunto de nodos se reduce, así como también varía la cantidad de terminales. Lo anterior se debe a que en cada paso $F_{(G,K)}$ es calculado tomando $F_{(G_i, K_i)}$, donde el conjunto de nodos es menor.

El estimador insesgado de la medida de anticonfiabilidad es el promedio de las muestras de $F_{(G,K)}$:

- $\hat{F}_{(G,K)} = \sum_{j=1}^N \frac{F^{(j)}}{N}$, el estimador insesgado de $Q_{(G,K)}$.
- $F_{(G,K)}^{(j)}$ una muestra aleatoria de $F_{(G,K)}$.
- $V\{\hat{F}_{(G,K)}\} = V\{F_{(G,K)}\}/N$
- $\hat{V}_F = \sum_{j=1}^N \frac{[F^{(j)}(G,K) - \hat{F}_{(G,K)}]^2}{[N \cdot (N-1)]}$.

A continuación se presentan los algoritmos correspondientes al procedimiento recursivo y principal empleados en RVR:

Algorithm 5 Procedimiento Recursivo $F_{(G,K)}$

INPUT:

grafo G ,

conjunto de nodos objetivo K ,

probabilidad de operación de los enlaces $P = (p_1, \dots, p_m)$.

OUTPUT:

una muestra aleatoria de $F_{(G,K)}$

- 1: **if** $|K| = 1$, G es K -conexo **then**
 - 2: **return** 0
 - 3: **end if**
 - 4: **if** G no es K -conexo **then**
 - 5: **return** 1
 - 6: **else**
 - 7: Encontrar $C = \{e_1, \dots, e_l\}$
 - 8: Calcular $Q_C = \prod_{l \in C} q_l$
 - 9: Sortear U Uniforme $(0,1)$
 - 10: Seleccionar J_i de modo que $U \in J_i$
 - 11: Construir $G_i = G - \{e_1, \dots, e_{i-1}\} * e_i$
 - 12: Determinar K_i
 - 13: **return** $Q_C + (1 - Q_C) \cdot F_{(G_i, K_i)}$
 - 14: **end if**
-

En el algoritmo recursivo, se obtiene una muestra aleatoria de $F_{(G,K)}$. En el paso 1, si el conjunto k es unitario, el grafo es conexo, por lo tanto la anticonfiabilidad tiene valor 0. En el paso siguiente si G no es K -conexo, el algoritmo devuelve 1, debido a que es el valor de la anticonfiabilidad para ese caso. En la línea 7 se obtiene un corte C , luego se calcula la probabilidad de que todos los componentes en C fallen. En la línea siguiente se sortea una variable aleatoria Uniforme $(0,1)$ para determinar J_i en la línea 10. Luego se calcula G_i y K_i en las líneas 11 y 12 respectivamente. En la Línea 12 se determina K_i . Finalizando se realiza la llamada recursiva. Se deberá continuar con el proceso hasta que G_i sea determinista.

Algorithm 6 RVR

INPUT:

grafo G ,conjunto de nodos objetivo K ,probabilidad de operación de los enlaces $P = (p_1, \dots, p_m)$,cantidad de iteraciones N .

OUTPUT:

 $\hat{F}_{(G,K)}$ estimación de $Q_{(G,K)}$, \hat{V} estimación de $V\{\hat{F}_{(G,K)}\}$ 1: **for** $j = 1 : N$ **do**2: $F^{(j)} = F(G, K)$ 3: **end for**4: $\hat{F}_{(G,K)} = \sum_{j=1}^N \frac{F^{(j)}}{N}$ 5: $\hat{V}_F = \sum_{j=1}^N \frac{[F^{(j)} - \hat{F}_{(G,K)}]^2}{[N \cdot (N-1)]}$.

En el algoritmo principal de RVR, los parámetros de entrada utilizados son el grafo G , el conjunto de terminales K , la probabilidad de operación de los enlaces y al tamaño de muestra N . Como salida se obtiene una estimación de la anticonfiabilidad así como también su varianza. En las líneas 1 y 2 el algoritmo llama al operador recursivo $F_{(G,K)}$, luego de que se han realizado las N iteraciones se obtienen las estimaciones de los parámetros.

Como el estimador de $Q_{(G,K)}$ es insesgado, se puede concluir que el estimador obtenido mediante el método RVR es consistente.

A continuación se prueba que el estimador por el método RVR es insesgado para $Q_{(G,K)}$ y tiene menor varianza que la generada por la técnica Monte Carlo.

Es importante recordar que el teorema de esperanza total establece que:

$$E(Y) = \sum_X E(Y|X = x)P_X(x).$$

En base a que los eventos A_C y B_i con $i = 1, \dots, l$, son exhaustivos y mutuamente excluyentes, utilizando el teorema anterior se llega a:

$$Q_{(G,K)} = E(Y) = Q_C \cdot E(Y|A_C) + \sum_{i=1}^l P(B_i) \cdot E(Y|B_i).$$

Como $P(Y = 0|A_C) = 0$ y Y únicamente toma valores 0 y 1, se tiene que $E(Y|A_C) = 1$ y

$$E(Y) = Q_C + (1 - Q_C) \sum_{i=1}^l [P(B_i)/(1 - Q_C)] \cdot E(Y|B_i).$$

Dado que $[P(B_i)/(1 - Q_C)]$ es la longitud de cada J_i , se puede plantear que:

$$E(Y) = Q_C + (1 - Q_C) \sum_{i=1}^l P(U \in J_i) E(Y|B_i)$$

$$= Q_C + (1 - Q_C) \sum_{i=1}^l E(1_{\{U \in J_i\}}) \cdot E(Y|B_i).$$

Utilizando que: $Y = 1 - \phi(X)$ y $P(X_i = x) = P(X = x \setminus B_i)$, se llega a que:

$$E(Y \setminus B_i) = E(1 - \phi(X) \setminus B_i) = E(1 - \phi(X_i)) = E(Y_i).$$

Sustituyendo $E(Y \setminus B_i)$ se obtiene:

$$Q_{(G,K)} = E(Y) = Q_C + (1 - Q_C) \sum_{i=1}^l E(1_{\{U \in J_i\}})E(Y_i),$$

dado que U es independiente de cada Y_i :

$$\begin{aligned} E(Y) &= Q_C + (1 - Q_C)E(\sum_{i=1}^l 1_{\{U \in J_i\}}Y_i) \\ &= E(Q_C + (1 - Q_C) \sum_{i=1}^l 1_{\{U \in J_i\}}Y_i) \\ &= E(Z). \end{aligned}$$

Mediante el resultado anterior se llega a que el estimador obtenido mediante el método RVR es insesgado.

En las siguientes líneas se demostrará que Z tiene menor varianza que Y , pudiendo de este modo concluir que el estimador correspondiente a RVR tiene menor varianza que el proporcionado por CMC, en el cual el estimador era \bar{Y}_N .

$$\begin{aligned} V(Z) &= V(Q_C + (1 - Q_C) \sum_{i=1}^l 1_{\{U \in J_i\}} \cdot Y_i) \\ &= 0 + (1 - Q_C)^2 V(\sum_{i=1}^l 1_{\{U \in J_i\}} Y_i). \end{aligned}$$

Como $\sum_{i=1}^l 1_{\{U \in J_i\}} \cdot Y_i = W$ es una variable aleatoria Bernoulli donde:

$$V(W) = E(W)(1 - E(W)), \text{ entonces:}$$

$$\begin{aligned} V(Z) &= (1 - Q_C)^2 V(\sum_{i=1}^l 1_{\{U \in J_i\}} \cdot Y_i) \\ &= (1 - Q_C)^2 \cdot E(\sum_{i=1}^l 1_{\{U \in J_i\}} \cdot Y_i) \cdot [1 - E\{\sum_{i=1}^l 1_{\{U \in J_i\}} \cdot Y_i\}]. \end{aligned}$$

Sea $E(Y) = Q_C + (1 - Q_C)E\{\sum_{i=1}^l 1_{\{U \in J_i\}} \cdot Y_i\} = E\{Z\}$, se tiene que:

$$\begin{aligned} E(\sum_{i=1}^l 1_{\{U \in J_i\}} Y_i) &= [E(Y) - Q_C]/(1 - Q_C) \\ &= (Q_{(K,G)} - Q_C)/(1 - Q_C). \end{aligned}$$

Finalmente, utilizando los resultados alcanzados en las líneas anteriores se llega a que:

$$V(Z) = (1 - Q_C)^2 \cdot (Q_{(K,G)} - Q_C)/(1 - Q_C) \cdot [1 - (Q_{(K,G)} - Q_C)/(1 - Q_C)]$$

$$= [1 - (Q_{(K,G)} - Q_C)](Q_{(K,G)} - Q_C) \leq (1 - Q_{(K,G)})Q_{(K,G)} = V(Y).$$

3.8. Método Entropía Cruzada.

El Método Entropía Cruzada [19, 21, 28], fue originalmente introducido por Rubinstein en 1997 y es utilizado para la estimación de eventos raros en el contexto de redes estocásticas, así como también en problemas complejos de optimización. Proporciona un procedimiento iterativo para estimar los parámetros de interés correspondientes al Muestreo por Importancia. Estima la probabilidad de interés realizando un cambio de la distribución original por una en la cual el evento raro tiene más probabilidad de ocurrir, una vez más, el evento raro para el caso en el que se estudia la confiabilidad de una red, es la falla de la misma. Se utiliza una técnica en donde los eventos raros no son tan raros y comienzan a serlo cada vez más en cada iteración. Además emplea la divergencia Kullback-Leibler, este método simplifica el cálculo numérico en cada paso, proporcionando algoritmos rápidos y eficientes. En general, consiste en la estimación de la siguiente esperanza l :

$$l = E_u = [H(X)] = \int H(x)f(x; u)dx. \quad (3.14)$$

$H(X)$ es una función positiva y $X = (X_1, \dots, X_n)$ es un vector aleatorio con cierta función de densidad conjunta correspondiente a una familia de densidades $f(\cdot; u)$ (depende de cierto parámetro u). Si se aplica el método Muestreo por Importancia para estimar l , se utiliza una función de densidad nueva, donde la probabilidad de ocurrencia del evento raro es mayor.

$$\hat{l} = \frac{1}{N_1} \sum_{i=1}^{N_1} H(X_{(i)})W(X_{(i)}; u, v). \quad (3.15)$$

Siendo el ratio de verosimilitud entre las dos densidades igual a :

$$W(X_{(i)}; u, v) = \frac{f(X; u)}{f(X; v)}. \quad (3.16)$$

La función de densidad asociada a $X_{(1)}, \dots, X_{(N_1)}$ es cierta $f(\cdot; v)$ que utiliza un parámetro de referencia diferente v . Mediante el método de Entropía Cruzada se busca encontrar un vector de referencia óptimo v^* de modo que la divergencia entre la función de densidad $f(\cdot; v^*)$ y la densidad óptima correspondiente a Muestreo por importancia que genera una varianza cero sea mínima, es decir, encontrar una densidad dentro de una clase específica de densidades donde la divergencia Kullback-Leibler sea mínima. La divergencia Kullback-Leibler entre dos funciones de densidad g y h , se define como:

$$D(g, h) = E_g[\ln \frac{g(X)}{h(X)}] = \int g(x) \ln \frac{g(x)}{h(x)} dx = \int g(x) \ln g(x) dx - \int g(x) \ln h(x) dx \quad (3.17)$$

Es importante destacar que la expresión anterior no es en realidad una distancia, debido a que no es simétrica, pero no causa inconvenientes. El problema radica en determinar v^* , es decir, cuál de todas las distribuciones de la familia de candidatas $f(X; v)$ es la que más se asemeja a $g^*(x)$, donde:

$$g^*(x) = \frac{H(x)f(x; u)}{l}$$

$$v^* = \arg \min_v D(g^*, f(\cdot; v)) = \arg \max_v \int H(x) f(x; v) \ln f(x; v) dx =$$

$$\arg \max_v E_u[H(X)] \log f(X; v) = \arg \max_v E_w[H(X)] W(X; u, w) \log f(X; v).$$

El vector óptimo de referencia correspondiente al método Entropía Cruzada puede ser estimado mediante el siguiente procedimiento iterativo:

$$v_t = \arg \max_v \frac{1}{N} \sum_{i=1}^N H(X_{(i)}) W(x_{(i)}; u, v_{t-1}) \log f(X_{(i)}; v). \quad (3.18)$$

En cada iteración t , se realiza una muestra aleatoria proveniente de $f(\cdot; v_{t-1})$.

Para el caso en el que se estudian eventos raros, $H(X) = I_{\{S(X) \geq \gamma\}}$, siendo γ una constante e I la función indicatriz que indica la ocurrencia o no del evento raro.

$$I = P[S(X) \geq \gamma] \quad (3.19)$$

La probabilidad anterior es muy pequeña, S es denominada Función de Rendimiento (Performance Function), siendo $S(X) \geq \gamma$ un evento raro. Existe un procedimiento en dos fases en Entropía Cruzada, que es utilizado precisamente en el caso de eventos raros ya que muchos de los valores correspondientes a $H(X_{(i)})$ serán cero. Se crea una secuencia de pares $\{(u_t, \gamma_t)\}$ donde v y γ se van actualizando con el fin de estimar v^* . Se comienza con $v_0 = u$ (parámetro original). Luego se toma una muestra aleatoria $X_{(1)}, \dots, X_{(N)}$ de $f(\cdot; v_{t-1})$, se toma en cuenta una proporción p de muestras que tienen un mejor rendimiento, donde γ_t es el $(1 - p)$ cuantil de $S(X_{(i)})$, siendo $i = 1, \dots, N$, siempre que el cuantil de la muestra sea menor que γ_t , de lo contrario se fija γ_t a γ . Un estimador de γ_t es un estadístico ordenado, por lo tanto se evalúa $S(X_i)$ para cada i y se ordenan los resultados, $S_{(1)} \leq S_{(2)} \leq \dots \leq S_{(N)}$.

$$\gamma_t = \min\{\gamma, S_{([\lceil(1-p)N\rceil])}\}. \quad (3.20)$$

Siendo j el orden del estadístico $S_{(j)}$. Se utiliza la misma muestra $X_{(1)}, \dots, X_{(N)}$ de $f(\cdot; v_{t-1})$ obteniendo:

$$v_t = \arg \max_v \frac{1}{N} \sum_{i=1}^N I\{S(X_{(i)}) \geq \gamma_t\} W(X_{(i)}; u, v_{t-1}) \log f(X_{(i)}; v). \quad (3.21)$$

Es importante resaltar que en muchos casos, por ejemplo cuando f pertenece a la familia de distribuciones exponenciales, encontrar v^* puede ser realizado directamente en forma analítica. Para el caso en que p es pequeño, como sucede cuando corresponde a la probabilidad de que un evento raro ocurra y N es lo suficientemente grande, v_t converge a v^* óptimo [32].

3.9. Entropía Cruzada y Monte Carlo

Es posible adaptar el método entropía cruzada al método Monte Carlo Crudo. En este caso en vez de realizar una muestra de estados de los enlaces, se realiza una muestra del tiempo en que se encuentra operativo cada enlace, luego se comprueba si la red está operativa en $t = 1$, siendo su probabilidad la confiabilidad de la red. Realizando este procedimiento se estaría cambiando el problema original, en el que se utilizan variables aleatorias Bernoulli, a otro en el que se utilizan variables aleatorias exponenciales Y_1, \dots, Y_m . El sistema en este caso estaría dependiendo del tiempo, por ejemplo en $t = 0$ todos los enlaces se encuentran en estado de falla y en reparación. Se tiene Y_1, \dots, Y_m con $Y_i \sim \text{Exp}(u_i^{-1})$ donde $u_i = \frac{1}{\lambda(i)} = \frac{-1}{\log q_i}$ son los tiempos independientes de reparación de los enlaces.

$$P(Y_i \geq 1) = e^{-\frac{1}{u_i}} = q_i \quad \text{con } i = 1, \dots, m$$

Sea $S(Y)$ el tiempo aleatorio en que el sistema se vuelve operativo, se llega a que:

$$Q = P(S(Y) \geq 1),$$

siendo posible aplicar directamente el método de entropía cruzada.

La aplicación del método consistirá en que en vez de realizar un muestreo independiente para cada i de $\text{Exp}(u_i^{-1})$ se hace de $\text{Exp}(v_i^{-1})$. Se tendrá entonces un vector de referencia por un lado $v = (v_1, \dots, v_m)$ y además se construye una secuencia de (v_t, γ_t) , de modo que v_t converge al vector de referencia óptimo y γ_t llega a uno. Al comienzo $v_0 = u = (u_1, \dots, u_m)$, en cada iteración se realiza una muestra $(Y_{(1)}, \dots, Y_{(N)})$ con $f(\cdot; v_{t-1})$ de Y y se actualiza el parámetro en base a 3.20 y el parámetro de referencia en base a 3.18, teniendo una solución analítica en este caso.

Se llega a que :

$$v_{t,j} = \frac{\sum_{i=1}^N I\{S(Y_{(i)}) \geq \gamma_t\} W(Y_{(i)}; u, v_{(t-1)}) Y_{(i)j}}{\sum_{i=1}^N I\{S(Y_{(i)}) \geq \gamma_t\} W(Y_{(i)}; u, v_{(t-1)})}$$

$$W(y; u, v) = \frac{f(y; u)}{f(y; v)} = \exp\left(-\sum_{j=1}^m y_j \left(\frac{1}{u_j} - \frac{1}{v_j}\right)\right) \prod_{j=1}^m \frac{v_j}{u_j}$$

Entonces luego de la iteración donde γ_T llega a uno, se puede estimar Q utilizando Muestreo por importancia de la siguiente manera:

$$Q = \frac{1}{N_1} \sum_{i=1}^{N_1} I\{S(Y_{(i)}) \geq 1\} W(Y_{(i)}; u, v_T)$$

Capítulo 4

Estudio experimental

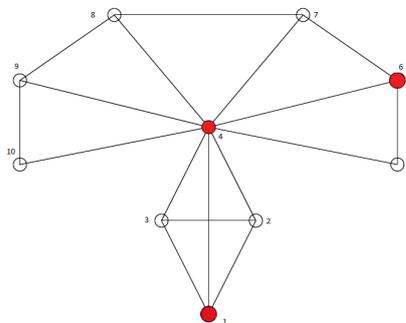
En este capítulo se exponen los resultados obtenidos al estimar la anticonfiabilidad $Q_k(G)$ de diversas redes mediante la implementación del método de Reducción Recursiva de la Varianza (RVR) y de Muestreo por Importancia en base a la versión aproximada de varianza cero (AZVIS).

Es importante destacar que se ha optado por utilizar tipologías que son usualmente utilizadas en este tipo de análisis, así como también redes reales actuales como lo son las de ANTEL y Atlanta que han sido estudiadas en otras oportunidades en diferentes trabajos monográficos, como los realizados por Elisa Bertinat [4], Fernanda González [16] y Daniel Lena [25] entre otros.

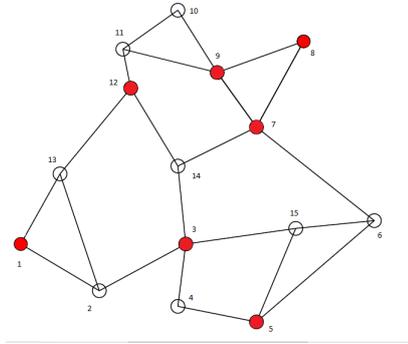
Casos de estudio

A continuación se presentan las diferentes tipologías utilizadas, ordenadas en forma ascendente en relación a la cantidad de vértices que poseen:

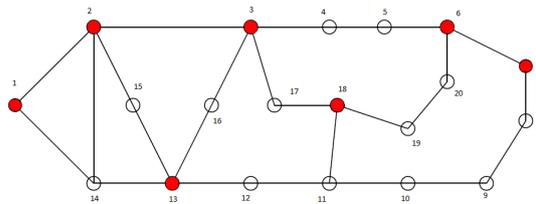
- RAU, con: $|V| = 10$, $|E| = 17$, $|K| = 3$.



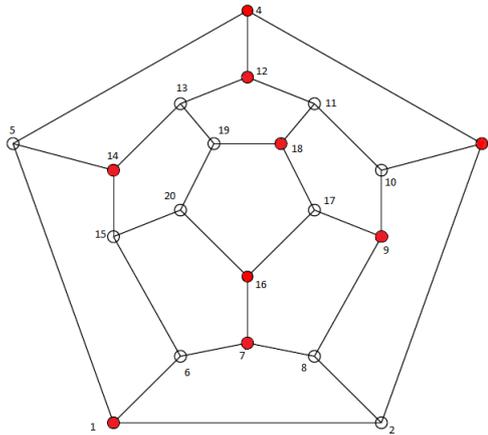
- Grafo Atlanta, con: $|V| = 15$, $|E| = 22$, $|K| = 7$.



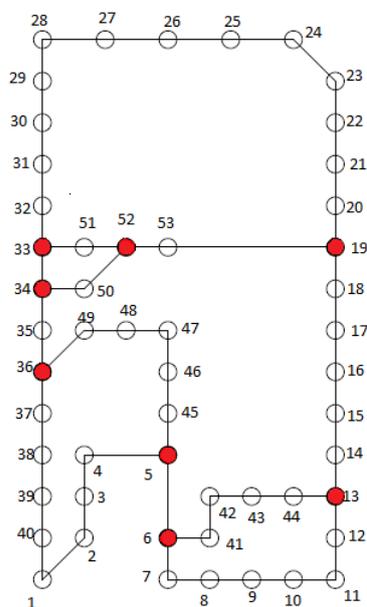
- Grafo Arpa Net, con: $|V| = 20$, $|E| = 26$, $|K| = 7$.



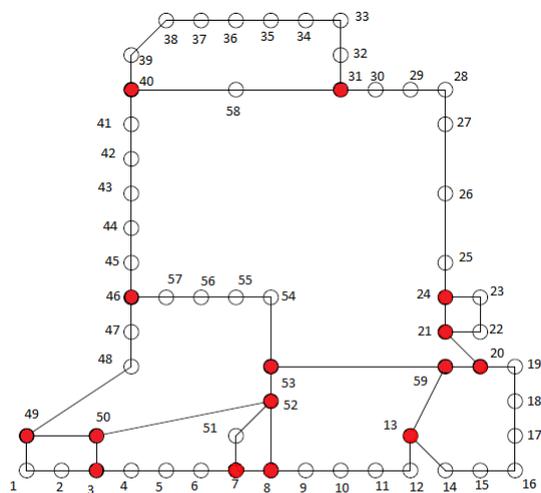
- Grafo Dodecaedro, con: $|V| = 20$, $|E| = 30$, $|K| = 9$.



- Grafo Antel Oeste, con: $|V| = 53$, $|E| = 57$, $|K| = 8$.



- Grafo Antel Este, con: $|V| = 59$, $|E| = 65$, $|K| = 15$.



4.1. Resultados numéricos

Se presentan en esta sección los resultados numéricos obtenidos para cada uno de los casos en base a diferentes métodos y probabilidades de operación de los enlaces, realizados en su Proyecto de Grado por Daniel Lena [25].

La implementación de los diferentes algoritmos fue realizada en MATLAB R2015b y el número de iteraciones es el mismo en todos los casos, $N = 10^4$.

En las siguientes tablas la notación utilizada es la siguiente:

- p : probabilidad de operación de los enlaces.
- \hat{Q}_k : estimación de la anticonfiabilidad.
- \hat{V} : estimación de la varianza.
- T : tiempo (segundos) de cómputo ejecutado en realizar la estimación de la anticonfiabilidad.
- Grafo RAU

Método	p	\hat{Q}_k	\hat{V}	T
RVR	0,95	5,28E-04	1,48E-10	16,8
IS	0,95	5,10E-04	1,69E-11	438,7
RVR	0,97	1,13E-04	1,46E-11	17,5
IS	0,97	1,10E-04	6,11E-13	424,5
RVR	0,99	3,99E-06	2,03E-14	17,0
IS	0,99	3,99E-06	6,14E-16	421,9

- Grafo Atlanta

Método	p	\hat{Q}_k	\hat{V}	T
RVR	0,95	5,63E-02	1,37E-07	114,8
IS	0,95	5,51E-02	2,35E-08	3.908,5
RVR	0,97	3,21E-02	2,71E-08	114,1
IS	0,97	3,31E-02	1,98E-08	3.985,9
RVR	0,99	1,02E-02	8,99E-09	112,5
IS	0,99	1,03E-02	2,04E-10	4.055,6

- Grafo Arpa Net

Método	p	\hat{Q}_k	\hat{V}	T
RVR	0,95	2,03E-02	1,87E-07	216,4
IS	0,95	1,95E-02	2,65E-07	5.294,3
RVR	0,97	6,27E-03	3,53E-08	216,7
IS	0,97	6,35E-03	2,58E-08	5.274,9
RVR	0,99	4,86E-04	8,67E-10	217,0
IS	0,99	4,48E-04	2,20E-10	5.247,6

- Grafo Dodecaedro

Método	p	\hat{Q}_k	\hat{V}	T
RVR	0,95	1,30E-03	1,55E-09	277,9
IS	0,95	1,26E-03	1,51E-09	30.646,3
RVR	0,97	2,61E-04	3,91E-10	251,0
IS	0,97	2,83E-04	2,42E-10	31.953,6
RVR	0,99	9,12E-06	1,06E-12	254,1
IS	0,99	9,23E-06	6,90E-14	36.094,4

- Grafo Antel Oeste

Método	p	\hat{Q}_k	\hat{V}	T
RVR	0,95	1,84E-01	3,39E-06	1.151,3
IS	0,95	1,89E-01	6,49E-06	33.315,4
RVR	0,97	7,10E-02	1,07E-06	1.216,6
IS	0,97	8,00E-02	1,09E-07	33.162,9
RVR	0,99	7,80E-03	5,13E-07	1.249,8
IS	0,99	7,75E-03	1,38E-08	31.671

- Grafo Antel Este

Método	p	\hat{Q}_k	\hat{V}	T
RVR	0,95	1,92E-01	3,35E-06	2.672,6
IS	0,95	1,91E-01	3,50E-07	193.108,9
RVR	0,97	7,37E-02	1,11E-06	3.056
IS	0,97	7,37E-02	3,97E-08	215.070,5
RVR	0,99	8,20E-03	5,21E-07	3.669,5
IS	0,99	8,10E-03	1,13E-09	294.439,1

Cuando se tienen diferentes métodos para resolver un problema determinado, se pueden comparar los resultados obtenidos en relación a la varianza de cada estimador, así como también al tiempo ejecutado por cada método en obtener la estimación en base a un número fijo de iteraciones.

La siguiente relación puede ser utilizada al momento de comparar dos métodos, A y B :

$$W_{RVR/IS} = \frac{Var_{RVR} T_{RVR}}{Var_{IS} T_{IS}}$$

Entonces, si el cociente es mayor que 1, significa que el método B tiene mejor desempeño que A , lo cual tiene tres lecturas posibles: B tiene la misma precisión que A pero consume menos tiempo en la ejecución o es más preciso que A pero a un costo computacional mayor, o bien, obtiene una mayor precisión en menor tiempo. Por lo tanto también es útil analizar que sucede con los cocientes de varianzas y de tiempos de cómputo:

$$V_{A/B} = \frac{Var_A}{Var_B}$$

$$T_{A/B} = \frac{T_A}{T_B}.$$

Es necesario que los resultados sean obtenidos utilizando el mismo procesador, debido a que en caso contrario el tiempo de cálculo puede modificarse de una versión a otra. Por tal motivo, los resultados expuestos han sido calculados utilizando el mismo procesador, Intel i7-6700HQ CPU 2.6 GHz, así como también se utilizó la misma cantidad de iteraciones.

A continuación se presentan los resultados obtenidos relacionando los dos métodos estudiados en los diferentes grafos, teniendo en cuenta distintas probabilidades de operación de los enlaces:

GRAFO	p	$\hat{V}_{RVR/IS}$	$T_{RVR/IS}$	$W_{RVR/IS}$
RAU	0,95	8,8	3,83E-02	0,3
	0,97	23,9	4,12E-02	1,0
	0,99	33,1	4,03E-02	1,3
Atlanta	0,95	5,8	2,94E-02	0,2
	0,97	1,4	2,86E-02	3,92E-02
	0,99	44,1	2,77E-02	1,2
Arpa Net	0,95	0,7	4,09E-02	2,88E-02
	0,97	1,4	4,11E-02	5,62E-02
	0,99	3,9	4,14E-02	0,2
Dodecaedro	0,95	1,03	9,07E-03	9,30E-03
	0,97	1,6	7,86E-03	1,27E-02
	0,99	15,3	7,04E-03	0,1
Antel Oeste	0,95	0,5	3,46E-02	1,80E-02
	0,97	9,9	3,67E-02	0,4
	0,99	37,0	3,95E-02	1,5
Antel Este	0,95	9,6	1,38E-02	0,1
	0,97	27,9	1,42E-02	0,4
	0,99	461,2	1,25E-02	5,7

Conclusiones

Al realizar un análisis de los resultados presentados para las diferentes tipologías en base a los dos métodos (IS y RVR), se puede apreciar que mediante la utilización del método IS bajo el esquema AZVIS se logran en general estimaciones más precisas si se lo compara con RVR, pero al aumentar el tamaño del grafo y K , el tiempo ejecutado para realizar la estimación es mayor. El tiempo de cómputo por parte del método IS es mayor al ejecutado por RVR debido a que se necesitan seleccionar por cada iteración $2 | E |$ conjuntos de corte de máxima probabilidad. Para poder encontrar el corte de máxima probabilidad se deben calcular los cortes máximos entre todas las combinaciones de K , eligiendo el máximo, entonces es razonable que a medida que aumenta K y el tamaño del grafo, el tiempo de cómputo también lo haga.

En relación a la varianza, como se dijo es evidente que el método IS es más preciso, es decir, el que presenta menores valores para la misma en casi todos los casos, disminuyendo significativamente a medida que aumenta la probabilidad de operación de los enlaces. La disminución de la varianza es coherente ya que el método cumple con la propiedad **VRE**, es decir $RE \rightarrow 0$ cuando $q \rightarrow 0$.

Si se observa el valor que toma el indicador $W_{RVR/IS}$, como fue previamente mencionado puede tener tres lecturas posibles, pero en este caso la lectura que se puede hacer es la misma en casi todos los casos, el método IS es más preciso pero insume un mayor tiempo de cómputo que RVR. Hay casos donde el valor de $W_{RVR/IS}$ es menor a 1, lo cual significa que el método RVR tiene en base a este indicador un mejor desempeño que IS, por el ejemplo en el grafo RAU el valor del mismo es 0,3 para $p = 0,5$ lo cual se debe a que si bien el valor de la varianza obtenida en IS es menor a la generada por RVR, no se logra compensar la diferencia de tiempo de cómputo siendo mucho mayor para el caso de IS. Hay dos casos en que el valor del indicador es menor que 1 y la lectura es diferente, los mismos se dan en los grafos Arpa Net y Antel Oeste con $p = 0,5$, donde RVR es más preciso que IS e insume menor tiempo de cómputo. En los casos en que el valor del indicador es mayor a 1 IS tiene un mejor desempeño que RVR, su mayor precisión compensa el mayor tiempo de cómputo en todos los casos. Si se observan por ejemplo los resultados obtenidos para el grafo Antel Este, siendo el de mayor tamaño dentro de las tipologías utilizadas en este caso, se puede apreciar que si bien se alcanza una mayor precisión mediante IS, su costo computacional es mucho mayor que el necesario por RVR, lo cual llevará al investigador a tener que optar por uno o por otro en base a los recursos que posea.

Capítulo 5

Conclusiones Generales

El presente trabajo se ha centrado en el estudio del cálculo de la Confiabilidad Clásica de una red, el cual pertenece a la clase \mathcal{NP} -Difícil. Como cuando se desea resolver mediante una forma analítica el esfuerzo computacional resulta cada vez más elevado a medida que aumenta el tamaño de la red, es necesario buscar otros métodos que involucren un menor esfuerzo aunque su resultado sea aproximado. Existen varios métodos, de los cuales algunos fueron presentados en este documento. Cuando el fallo de una red es un evento raro, el método Monte Carlo Crudo no obtiene un buen desempeño, siendo necesario realizar un gran número de experimentos con el fin de reducir el error relativo. Entre otros, se presentó el método RVR, el cual se basa en la descomposición en forma recursiva de la red en base a particiones del espacio de estados. Se demostró en forma teórica que mediante la utilización del mismo se logra reducir la varianza en relación a Monte Carlo Crudo. Los estimadores de los métodos presentados son insesgados, por lo tanto la varianza coincide con el error cuadrático medio, por tal motivo cuando se presentó cada uno de los métodos se analizó el comportamiento de la varianza en relación a CMC y se pudo concluir teóricamente que efectivamente la varianza es mayor en este último. Otro de los métodos presentados fue Muestreo por Importancia en base al esquema AZVIS. Este método cumple con la propiedad VRE, que establece que el Error Relativo tiende a cero cuando la probabilidad de falla se hace cada vez más pequeña. Según los resultados obtenidos se pudo concluir que a medida que aumenta la probabilidad de operación de los enlaces se obtiene una menor varianza si se compara IS con RVR, sin embargo, el costo computacional es mayor cuando el tamaño del grafo y K aumentan, debido al costo asociado al hallar cortes de máxima probabilidad. A la hora de optar por un método se tendrá que tener en cuenta tanto su precisión como también el costo computacional asociado al mismo, lo cual dependerá de cada caso.

Es importante destacar que si bien en esta monografía se ha presentado el método Monte Carlo en el entorno de la estimación de la confiabilidad de una red, el mismo tiene múltiples aplicaciones. En general el método Monte Carlo permite obtener soluciones a problemas matemáticos de elevada complejidad en base a pruebas aleatorias repetidas llevadas a la práctica mediante cálculos realizados con números aleatorios.

Este método puede ser utilizado por ejemplo en la estimación de parámetros poblacionales, en el análisis de riesgo de proyectos, entre tantas otras aplicaciones.

Bibliografía

- [1] S.H. Ahmad. A simple technique for computing network reliability. *Reliability, IEEE Transactions on*, R-31(1):41–44, 1982.
- [2] M. Ball, C. Colbourn, and J. Provan. Network Reliability. Technical report, University of Maryland TR 92-74, June 1992.
- [3] Michael O. Ball. Computational complexity of network reliability analysis: An overview. *IEEE Transactions on Reliability*, 35(3):230–239, aug. 1986.
- [4] Lic. María Elisa Bertinat. Estudio de métodos de reducción de varianza para modelos de confiabilidad con restricciones de diámetro. 2011. URL http://premat.fing.edu.uy/ingenieriamatematica/archivos/tesis_mariaelisa_bertinat.pdf.
- [5] H. Cancela and M. El Khadiri. A recursive variance-reduction algorithm for estimating communication-network reliability. *Reliability, IEEE Transactions on*, 44(4):595–602, Dec 1995.
- [6] H. Cancela and M. El Khadiri. The recursive variance-reduction simulation algorithm for network reliability evaluation. *IEEE Transactions on Reliability*, 52(2):207–212, 2003.
- [7] Héctor Cancela and Louis Petingi. Reliability of communication networks with delay constraints: computational complexity and complete topologies. *International Journal of Mathematics and Mathematical Sciences*, 2004:1551–1562, 2004.
- [8] Jacques Carlier and Corinne Lucet. A decomposition algorithm for network reliability evaluation. *Discrete Applied Mathematics*, 65(1?3):141 – 156, 1996. ISSN 0166-218X. First International Colloquium on Graphs and Optimization.
- [9] Charles J. Colbourn. *Telecommunications Network Planning*, chapter Reliability Issues In Telecommunications Network Planning, pages 135–146. Springer US, Boston, MA, 1999.
- [10] Stephen A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the third annual ACM symposium on Theory of computing*, STOC '71, pages 151–158, New York, NY, USA, 1971. ACM.

- [11] R.K. Dash, N.K. Barpanda, P.K. Tripathy, and C.R. Tripathy. Network reliability optimization problem of interconnection network under node-edge failure model. *Applied Soft Computing*, 12(8):2322 – 2328, 2012. ISSN 1568-4946.
- [12] Institut National de Recherche en Informatique et en Automatique, M.E. Khadiri, G. Rubino, and IRISA (117). *A Monte Carlo Method Based on Antithetic Variates for Network Reliability Computations*. Rapports de recherche. Institut National de Recherche en Informatique et en Automatique, 1992.
- [13] George S. Fishman. A Monte Carlo Sampling plan for estimating network reliability. *Operations Research*, 34(4):581–594, 1986.
- [14] Gert Folland. *Real Analysis: Modern Techniques and Their Applications*. Wiley Series of Pure and Applied Mathematics, 1999. ISBN 978-0471317166.
- [15] Ilya B. Gertsbakh and Yoseph Shpungin. *Models of Network Reliability: Analysis, Combinatorics, and Monte Carlo*. CRC Press, Inc., Boca Raton, FL, USA, 1st edition, 2009. ISBN 1439817413, 9781439817414.
- [16] Lic. Fernanda González. Muestreo por importancia y reducción recursiva de la varianza para la evaluación de la confiabilidad diámetro acotada. 2013. URL http://premat.fing.edu.uy/ingenieriamatematica/archivos/tesis_fernanda_gonzalez.pdf.
- [17] Ralph P. Grimaldi. *Matemáticas Discreta y Combinatoria*. 1994.
- [18] Frank Harary. *Graph Theory*. 1969.
- [19] Kin-Ping Hui, Nigel Bean, Miro Kraetzl, and Dirk P. Kroese. The cross-entropy method for network reliability estimation. *Annals of Operations Research*, 134(1):101–118, 2005.
- [20] R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.
- [21] Dirk P. Kroese, Kin-Ping Hui, and Sho Nariai. Network reliability optimization via the cross-entropy method. *IEEE Transactions on Reliability*, 56(2):275–287, 2007.
- [22] H. Kumamoto, Kazuo Tanaka, and Koichi Inoue. Efficient evaluation of system reliability by monte carlo method. *Reliability, IEEE Transactions on*, R-26(5):311–315, Dec 1977.
- [23] H. Kumamoto, K. Tanaka, K. Inoue, and E. J. Henley. Dagger sampling Monte Carlo for system unavailability evaluation. *IEEE Transactions on Reliability*, 29(2):376–380, 1980.
- [24] P. L’Ecuyer, G. Rubino, S. Saggadi, and B. Tuffin. Approximate Zero-Variance Importance Sampling for Static Network Reliability Estimation. *Reliability, IEEE Transactions on*, 60(3):590–604, Sept 2011.

- [25] Daniel Lena Messere. Confiabilidad de redes con fallas hostiles. 2016.
- [26] Claudio Riso Montaldo. Using grasp and ga to design resilient and cost-effective ip/mps networks. 2015.
- [27] Fred Moskowitz. The analysis of redundancy networks. *American Institute of Electrical Engineers, Part I: Communication and Electronics, Transactions of the*, 77(5):627–632, Nov 1958.
- [28] L. Murray. Comparación entre Cinco Métodos de Monte Carlo para Estimar la Confiabilidad Diámetro Acotada de Redes de Comunicaciones. Technical report, Facultad de Ciencias Exactas, Ingeniería y Agrimensura, Universidad Nacional de Rosario, Tech. Rep., 2007.
- [29] L. Petingi and J Rodríguez. Reliability of networks with delay constraints. *Congressus Numerantium*, 152, pages 117–123, 2001.
- [30] Dr. Pablo Gabriel Romero. Diámetro confiabilidad de una red. 2014.
- [31] G. Rubino and B. Tuffin. *Rare event simulation using Monte Carlo methods*. Wiley, 2009.
- [32] Y. Rubinstein and D. P. Kroese. *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning*. 2004.
- [33] M. Veeraraghavan S .Rai and K.S. Trivedi. A survey of efficient reliability computation using disjoint products approach. *NETWORKS*, 25:147–163, 1995.
- [34] A. Satyanarayana and A. Prabhakar. New topological formula and rapid algorithm for reliability analysis of complex networks. *Reliability, IEEE Transactions on*, R-27(2): 82–100, June 1978.
- [35] R. Sedgewick and M. Schidlowsky. Algorithms in Java, Part 5: Graph Algorithms. *Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.*, 2003.
- [36] R. Van Slyke and H. Frank. Network reliability analysis: Part 1. *Networks*, 1(3):279–290, 1971.

