

Selección Óptima de *p-Cycle* en Redes WDM Un Enfoque Basado en Algoritmo Genético

Carlos Colmán, Diego Pinto y Benjamín Barán

Universidad Nacional de Asunción
Casilla de Correos 2111 - Paraguay
ccolman@ing.una.py, {dpinto, bbaran}@pol.una.py

Resumen. Con la expansión de Internet y el crecimiento de la demanda de capacidad para transmisión de datos, incrementa el interés en la tecnología de Redes Ópticas WDM. Con el enorme ancho de banda potencialmente provisto por estas redes, gana importancia estratégica el estudio de prevención y protección contra fallas. La protección basada en *p-Cycle* es un novedoso enfoque, basado en ciclos de protección pre-configurados que proveen una rápida recuperación. En consecuencia, este trabajo propone la resolución del problema de selección óptima de *p-Cycles* utilizando Algoritmos Genéticos. Resultados experimentales han demostrado que esta propuesta es un enfoque prometedor, al lograr mejores resultados sobre distintas topologías de redes, al compararlo con un conocido algoritmo del estado del arte llamado *Capacitated Iterative Design Algorithm* (CIDA).

Palabras Claves: Redes Ópticas, WDM, Protección, *p-Cycle*, SLA, GROW, Algoritmo Genético y CIDA.

1 Introducción

La expansión de Internet, convierte a la tecnología de transmisión óptica *WDM* (*Wavelength Division Multiplexing*) en la potencial sucesora de las redes basadas en tecnología *SONET* y *SDH*, por ofrecer mayores capacidades de transmisión de datos y aprovechamiento de recursos [1].

Las redes de transmisión ópticas son también propensas a fallas causadas por diversos motivos, que incluyen cortes en conductos y/o cables, como también fallas en equipos de comunicación, teniendo muchas veces consecuencias catastróficas para el tráfico que en ella circula debido a su alta capacidad [2]. En consecuencia, resulta estratégico el estudio de la tolerancia a fallas contemplando todos los escenarios posibles de las Redes Ópticas *WDM*.

Entre las estrategias de protección más novedosas y prometedoras [3, 4, 5], se considera el empleo de ciclos de protección pre-configurados (*preconfigured protection cycles*, o simplemente *p-Cycle*), por lo que el diseño de algoritmos o heurísticas competitivas que optimicen un conjunto de ciclos de protección óptimo resulta crucial. Numerosos trabajos han tratado el problema de protección *p-Cycle* con

enfoques heurísticos basados en población unitaria, por lo que en el presente trabajo propone aplicar alternativamente, un *Algoritmo Genético (AG)*. Se realizaron distintas pruebas experimentales sobre diversos tipos y tamaños de topologías de redes. Básicamente, el enfoque propuesto mejora los resultados obtenidos por un conocido algoritmo del estado del arte propuesto por Doucette y colegas [6].

2 Ciclos de Protección Pre-configurados – p-Cycle

Considerando la importancia de la tolerancia a fallas en Redes *WDM*, Grover et al, proponen un nuevo método inspirado en estrategias de rápida recuperación de redes anilladas *SONET / SDH*, aplicadas a redes malladas, denominada *p-Cycle (Preconfigure Protection Cycle)* [5].

El *p-Cycle* es un método de protección basado en enlace, precalculado y dedicado, orientado a proteger la capacidad de trabajo de cada enlace, reservando hasta un 50 % de la capacidad total de cada enlace. Básicamente, el método proporciona protección de una *unidad de capacidad de trabajo* (o *u.c.t.*) a los enlaces que se encuentran sobre un ciclo (“*on-link*”), y de dos *unidades de capacidad de trabajo* cuando un enlace es rodeado por el ciclo (“*straddling-link*”) [7]. Un ejemplo puede observarse en la Figura 1, donde ante cualquier falla en los enlaces protegidos se puede tomar de forma inmediata el ciclo de protección ilustrado.

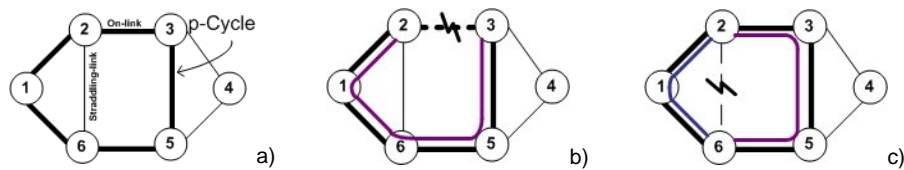


Fig. 1. Protección Cíclica Pre-configurada. a) Un ejemplo de *p-Cycle*. b) Protección de una unidad de trabajo ante la caída de un enlace “*on-link*”. c) Protección de dos unidades de trabajo ante la caída de un enlace “*straddling-link*”. Note que en b) la protección ofrece una ruta de recuperación mientras que en c) se permiten dos rutas alternativas.

El problema de selección de *p-Cycles* óptimos o ciclos óptimos, es considerado en la literatura como un problema NP-completo. Por lo tanto, trabajos anteriores plantearon el uso de una combinación de algoritmos heurísticos, dividiendo el problema en dos etapas: primero, generar un conjunto de ciclos candidatos [5 - 11] y luego seleccionar un subconjunto de ciclos óptimos o solución al problema [5, 7, 11]. Para la generación de ciclos candidatos, en [11, 12, 13] se proponen algunas heurísticas para una topología de N enlaces, obteniendo N^N ciclos candidatos. La complejidad del algoritmo de generación de ciclos propuesto en [12], fue estudiada en [8], proponiendo un algoritmo denominado *SLA (Straddling Link Algoritmo)*, que genera solo N ciclos candidatos. Además, se propusieron tres algoritmos complementarios *ADD*, *EXPAND* y *GROW*, logrando una mejora notable en la calidad de los ciclos seleccionados como candidatos.

Como elemento de evaluación y de preselección, se propusieron en [7], el grado topológico (*Topological Score - TS*), la eficiencia (η), y la Redundancia, con el fin de mejorar la selección de ciclos óptimos. Para el cálculo de una solución, se propone una formulación *ILP (Integer Linear Programming)* en [5], con una complejidad impracticable para redes de gran porte. En [6] se propone una heurística denominada *CIDA (Capacitated Iterative Design Algorithms)*, logrando un buen desempeño al combinarse con los generadores de ciclos *SLA* y *GROW*. Cabe resaltar que como solución alternativa, este trabajo propone utilizar como generadores de ciclos el *SLA* y el *GROW*, combinados con un *AG* para el cálculo de la protección cíclica.

3 Formulación del Problema

Sea $G = \{V, E\}$ un grafo que representa una topología de red, siendo V un conjunto de nodos y E un conjunto de enlaces. La siguiente nomenclatura es utilizada en el resto del trabajo:

- $v_i \in V$: Nodo i perteneciente a V con $i = 1, 2, \dots, |V|$, donde $|\cdot|$ indica cardinalidad.
- $e_j \in E$: Enlace j perteneciente a E , con $j = 1, 2, \dots, |E|$.
- $p_k \subset E$: Conjunto de enlaces que conforman un ciclo en G , i.e. $p_k = \{e_1, e_2, \dots, e_{|p_k|}\}$.
- P : Conjunto de ciclos pertenecientes a G , i.e. $P = \{p_1, p_2, \dots, p_{|P|}\}$.
- $X_k(e_j) \in \{0, 1, 2\}$: Variable que indica el tipo de protección proveída a e_j por el ciclo p_k . Si e_j es un *straddling-link* en p_k entonces $X_k(e_j)=2$, si e_j es un *on-link* en p_k entonces $X_k(e_j)=1$, en otro caso $X_k(e_j)=0$.
- $Z_k(e_j) \in \{0, 1\}$: Variable que indica si p_k provee protección a e_j . Si e_j es *on-link* o *straddling-link* en p_k entonces $Z_k(e_j)=1$, en otro caso $Z_k(e_j)=0$.

Considerando las anteriores definiciones, el problema consiste en calcular un conjunto ciclos $P' \subset P$ que maximice la eficiencia de protección [7]:

$$\eta = \frac{\text{Protección - total}(P')}{\text{Costo - total}(P')} \quad (1)$$

donde:

$$\text{Protección - total}(P') = \sum_{p_k \in P'} TS(p_k) \quad (2)$$

$$\text{Costo - total}(P') = \sum_{p_k \in P'} C(p_k) \quad (3)$$

la Protección Cíclica o *Topological Score - TS* es definida en [7] como:

$$TS(p_k) = \sum_{e_j \in E} X_k(e_j) \quad (4)$$

Por su parte, el Costo Cíclico es definido en [7] como:

$$C(p_k) = |p_k| \quad (5)$$

Sujeto a la restricción de una protección completa de todo el grafo:

$$\sum_{e_j \in E} \sum_{p_k \in P'} Z_k(e_j) \geq |E| \tag{6}$$

Como ejemplo práctico, considere la topología de red presentada en la Figura 2. En la Figura 2.a se muestra el conjunto de ciclos candidatos $P = \{p_1, p_2, \dots, p_8\}$, mientras que en la Figura 2.b se observa un sub-conjunto de ciclos $P' = \{p_4, p_8\}$ que representa una solución posible para la protección de la red, expuesta en la Figura 2.c. donde todos los enlaces se encuentran protegidos. En particular, los enlaces (2,6), (1,3) y (3,5) son *straddling-links*. En la Figura 2.c son mostrados los cálculos relacionados a esta solución. Note que todos los enlaces del grafo se encuentran protegidos.

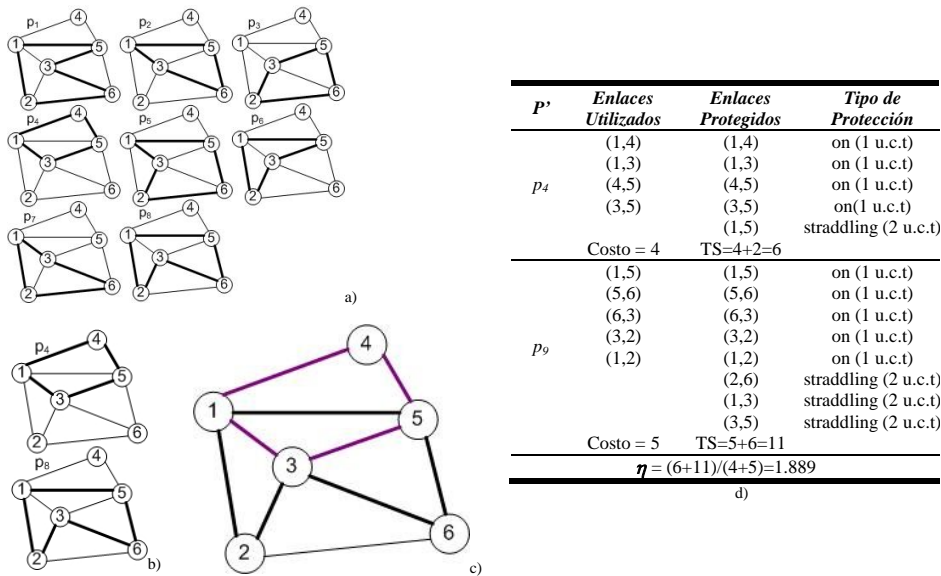


Fig. 2. Ejemplo de una protección basada en p-Cycle. a) Conjunto candidatos de ciclos P, b) Sub-conjunto P' de ciclos seleccionados, c) Topología protegida y d) Cálculos de la solución.

4 Generadores de p-Cycles Candidatos

Para topologías de redes de gran porte, el cálculo de un conjunto de ciclos candidatos P_{full} que contenga todos los ciclos posibles conlleva a un excesivo costo computacional [14]. Varios trabajos proponen la construcción cuidadosa de un conjunto de ciclos $P \subset P_{full}$ candidatos [9, 11]. Entre los algoritmos de generación de ciclos candidatos más referenciados y estudiados se encuentran el *Straddling Link Algorithm - SLA* y el *GROW* [14]. Ambos enfoques calculan un conjunto de ciclos candidatos eficientes y adaptables a topologías de diferentes tamaños, densidades de nodos y enlaces [6, 9, 14]. Básicamente, el *SLA* realiza dos llamadas al algoritmo *Dijkstra Shortest Path*, para construir un ciclo por cada enlace, generando un número

máximo de ciclos, equivalente a la cantidad de enlaces de la topología. De esta forma se construye un conjunto de ciclos candidatos reducido, escalable y eficiente [8]. El algoritmo *GROW* considerado para este trabajo corresponde a la *versión exhaustiva* propuesta en [14]. El *GROW* construye los nuevos ciclos a partir de un conjunto de ciclos calculados *a priori* por otro algoritmo. En consecuencia, el *SLA* se utiliza para generar un conjunto de ciclos P_1 mientras que el *GROW* genera otro conjunto de ciclos P_2 con nuevos ciclos de alta calidad, donde generalmente $P_1 \subset P_2$ [6].

5 CIDA (Capacitated Iterative Design Algorithm)

El *CIDA* es una herramienta heurística propuesta en [6], que recibe un conjunto de ciclos candidatos P , calculados previamente por otros algoritmos como: el *SLA* y el *GROW*. Se calcula la η de todos los ciclos y se seleccionan los mejores ciclos hasta que la capacidad de trabajo sea totalmente protegida. La excelente eficiencia de este algoritmo, combinado con el *SLA* y el *GROW* se estudia en [6, 14] y se compara con otras técnicas como el *ILP* en [9, 11] demostrando todo su potencial.

Algoritmo 1: CIDA

Entrada: P (conjunto de ciclos candidatos) y $G=\{V, E\}$
Salida: P' (conjunto de ciclos de protección)

```

1: Inicializa Conjunto-Ciclos,  $P' = \emptyset$ ;
2: Conjunto-Ciclos =  $P$ ;
3: Mientras Protección < 100 % hacer
4:   Para cada ciclo  $p_k \in$  Conjunto-Ciclos hacer
5:     Calcular y evaluar  $\eta(p_k)$ ;
6:     Si  $\eta(p_k) > \eta(\text{Mejor-Ciclo})$  entonces
7:       Mejor-Ciclo =  $p_k$ ;
8:     Fin si
9:   Fin Para
10:   $P' = P' \cup$  Mejor-Ciclo;
11:  Conjunto-Ciclos = Conjunto-Ciclos - Mejor-Ciclo;
12: Fin Mientras
13: Retornar  $P'$ ;
14: Fin Algoritmo 1

```

6 Algoritmo Genético

Los Algoritmos Genéticos (o *AGs*) son heurísticas que se inspiran en la teoría de *Darwing* de la evolución de las especies, aplicado a la resolución de problemas complejos [15]. En este trabajo se considera un *AG* elitista, que toma como dato de entrada una topología de red y un conjunto de ciclos P candidatos, calculando un subconjunto óptimo $P' \subset P$ de ciclos de protección. En un *AG* elitista, los l mejores individuos de la población actual sobreviven directamente al siguiente ciclo evolutivo sin modificaciones. En este trabajo se ha considerado una baja presión elitista tomando $l=1$. En el Algoritmo 2 se presenta el enfoque propuesto, cuyos detalles son explicados brevemente a continuación.

Construir Tabla de Protección. Sea $P = \{p_1, p_2, \dots, p_{|P|}\}$ un conjunto de ciclos candidatos. La Tabla de Protección es un almacenamiento ordenado de los $p_k \in P$, de forma a simplificar la representación del cromosoma. Cada cromosoma representa una solución $P' \subset P$, el cual consta de dos campos: a) *CantCycle*: la cantidad de ciclos de la solución P' ; y b) *IDs p-Cycles*: índices a la Tabla de Protección donde se encuentran los ciclos seleccionados. Note que la representación del cromosoma es dinámica dependiendo de *CantCycle*. En la Figura 3 se expone la relación entre Tabla de Protección, Cromosoma y una Solución P' considerando el ejemplo presentado en la sección 3. La estructura del cromosoma fue inspirada en [22].

Algoritmo 2: Algoritmo Genético (AG)

Entrada: P (conjunto de ciclos candidatos) y $G = \{V, E\}$

Salida: P' (conjunto de ciclos de protección)

```

1: Inicializar parámetros genéticos;
2: Construir Tabla de Protección;
3: Generar Población Evolutiva;
4: Asegurar Protección;
5: Calcular Fitness;
6:  $P' =$  Guardar mejor-solución;
7: Mientras criterio de parada no se cumpla hacer
8:   Seleccionar Padres;
9:   Cruzar padres y aplicar Mutación;
10:  Asegurar Protección;
11:  Calcular Fitness;
12:   $P' =$  Guardar mejor-solución;
13: Fin Mientras
14: Retornar  $P'$ ;
15: Fin Algoritmo 2
    
```

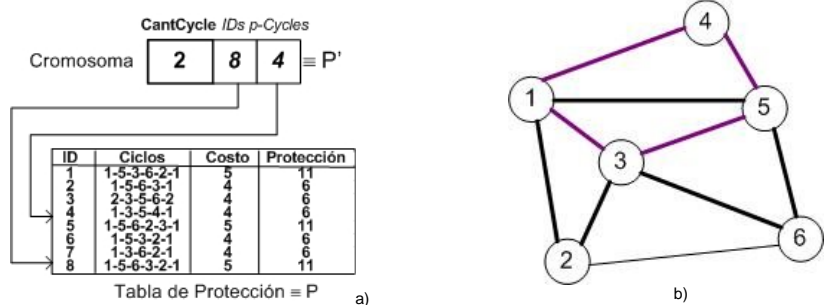


Fig. 3. Relación entre Cromosoma, Tabla de Protección y Solución. Note que en a) *CantCycle* = 2 indica el número de *ciclos* de P' . Los siguientes números en un cromosoma indican la posición en la Tabla de los ciclos $p_k \in P'$. En b) la solución dada por el cromosoma es graficada.

Generar Población Evolutiva. Cada individuo se construye de forma aleatoria. Inicialmente a *CantCycle* es asignado aleatoriamente un numero entero positivo, $CantCycle \in \{1, 2, \dots, |P|\}$. *CantCycle* define el número de genes a ser utilizados. A cada gen_i , donde $i=1,2,\dots, CantCycle$, es asignado aleatoriamente un *ID* que apunta a una posición en la Tabla de Protección. Para mantener la consistencia, es impuesta la siguiente restricción sobre los genes, $gen_i \neq gen_j$ cuando $i \neq j$.

Asegurar Protección. La solución propuesta por un cromosoma debe verificar la

restricción de protección total de los enlaces, conforme a la ecuación (5). Por lo tanto, ésta sub-rutina agrega nuevos ciclos al cromosoma en caso de no cumplir con dicha restricción.

Calcular Fitness. Se define como *fitness* del cromosoma la eficiencia (η) del conjunto de ciclos candidatos P' propuesta por el cromosoma, ver ecuación (1).

Operadores genéticos. En este trabajo se ha implementado como método de selección de padres el torneo binario [15]. Por su parte, el operador de cruce toma dos padres y genera un nuevo hijo. Dicho operador consta de dos etapas: primero cruce de *CantCycle* y luego cruce de genes *ID*. Para el cruce de *CantCycle* es utilizado el conocido operador *BLX* [15]. Una vez definida *CantCycle* para el hijo, los genes *ID* idénticos en los padres son copiados directamente al hijo. Los siguientes genes *ID* vacíos son completados de forma aleatoria con los *ID* no comunes de los padres. Finalmente, la mutación consiste en el reemplazo total de individuos de la generación por nuevos individuos, considerando una probabilidad de mutación $p_m=0.2$ adoptado experimentalmente.

7 Pruebas Experimentales

Los experimentos se realizaron considerando el siguiente entorno. Los algoritmos se ejecutaron en *Mathworks Matlab 6.5*, para plataforma *Microsoft Windows XP*. Se ha empleado una Computadora *IBM ThinkPad* con procesador Intel Pentium 4 de 2.8 MHz, 768 Mbytes de memoria RAM, con sistema operativo *Microsoft Windows XP profesional*. Para este trabajo, se consideraron 11 Topologías de Red de diferentes tamaños y densidades, las cuales son presentadas seguidamente en la Tabla 1.

Tabla 1. Topologías de Redes consideradas en las pruebas experimentales. Detalle de cantidad de nodos, enlaces y cantidad de ciclos candidatos generados por el *SLA* y el *GROW*.

Símbolos	Topologías de Red	Nodos	Enlaces	SLA P ₁	GROW P ₂	GROW P ₃
NSF	USA National Science Foundation Backbone Network [1]	14	25	25	144	756
BCBY	Bell Core Backbone Yerse LATA [17]	15	27	27	122	780
ULHN	USA Long Haul Network [6]	28	45	45	181	780
ARPA	USA ARPA Backbone Network II [18]	21	25	25	30	30
BCN	Bell Core Backbone Network [16, 18]	25	28	28	120	404
PAN	PAN European COST 239 [1, 16]	11	26	25	106	366
ELHN	European Long Haul Network - France Telecom [6,11]	43	71	70	321	1683
GOBN	German Optical Backbone Network [19]	17	27	26	79	212
ECN	ECNet European Backbone Network [20]	18	39	25	144	420
CNBN	China National Backbone Network [21]	66	120	120	662	4257
NTT	Nippon Telephone Telegraph Backbone Network [22]	49	66	66	213	545

Los siguientes pasos definen el proceso experimental:

- Para cada topología se generaron 3 conjuntos de ciclos candidatos, el primero P_1 , generado por el *SLA*. El segundo conjunto P_2 generado por el *GROW* en una primera ejecución sobre P_1 , y el tercer conjunto P_3 , nuevamente utilizando el *GROW* en una segunda ejecución sobre P_2 ; totalizando 33 conjuntos de ciclos

candidatos, conforme se muestra en la Tabla 1.

- Se realizaron 50 ejecuciones independientes con el *AG* y el *CIDA* sobre cada conjunto de ciclos candidatos, totalizando 3.300 *ejecuciones de optimización*. Fue considerado como criterio de parada para cada ejecución un tiempo máximo de ejecución de 20 minutos, llevando un total de 28 *días* de pruebas experimentales.

En la Tabla 2 se muestran las Eficiencias de Protección obtenidas sobre las diferentes topologías de red, mientras que en la Tabla 3 se presenta un resumen de los promedios globales de todas las pruebas experimentales, definiendo un ranking.

Puede observarse que para todas las topologías de red y para los distintos conjuntos de ciclos candidatos, el *AG* presenta una superioridad notable. Esto se ve reforzado con los resultados de la Tabla 3, donde se enfatiza resultados con celdas en gris. Para la topología ARPA, considerada de tamaño y densidad media, los dos algoritmos (*CIDA* y *AG*) lograron hallar el mismo conjunto óptimo de protección.

Por otra parte, corresponde destacar que las eficiencias de las soluciones se incrementan conforme aumenta el tamaño del conjunto de ciclos candidatos. Esto indica una dependencia de la calidad de las soluciones óptimas con relación al tamaño y la calidad del conjunto de ciclos candidatos. Considerando el conjunto de ciclos P_1 y P_3 y los resultados globales de la Tabla 3, se puede observar que el *CIDA* ha obtenido un porcentaje de mejoramiento del 18% ($= 100 \cdot (1.83 - 1.55) / 1.55$). Por su parte el *AG* obtuvo un porcentaje de mejoramiento del 45% ($= 100 \cdot (2.44 - 1.68) / 1.68$) superando ampliamente al obtenido por el algoritmo *CIDA*. Estas tendencias indican que la eficiencia del *AG* aumenta más rápidamente que el *CIDA*, demostrando intuitivamente su alta capacidad exploratoria del espacio de soluciones.

Tabla 2. Resultados experimentales clasificados por Topología de Red.

Topologías	Algoritmos	Promedio de Eficiencias de Protecciones		
		<i>SLA-P₁</i>	<i>GROW-P₂</i>	<i>GROW-P₃</i>
NSF	CIDA	1.48	1.73	2.57
	AG	1.70	2.73	2.83
ULHN	CIDA	1.5	1.8	1.8
	AG	1.6	2.28	2.4
BCN	CIDA	1.8	1.86	1.86
	AG	1.8	2.52	2.87
ELHN	CIDA	1.43	1.53	1.61
	AG	1.60	1.97	2.05
ECN	CIDA	1.65	1.90	1.95
	AG	1.80	2.60	2.72
NTT	CIDA	1.35	1.40	1.50
	AG	1.41	1.60	1.73
BCBY	CIDA	1.78	1.9	1.95
	AG	1.95	2.36	2.56
ARTA	CIDA	1.10	1.20	1.20
	AG	1.10	1.20	1.20
PAN	CIDA	1.81	1.82	2.03
	AG	2.00	2.81	3.32
GOBN	CIDA	1.53	1.93	1.96
	AG	1.70	2.70	2.89
CNBN	CIDA	1.60	1.62	1.70
	AG	1.81	2.06	2.23

Tabla 3. Promedios Globales de Eficiencias.

Algoritmos	SLA-P ₁		GROW-P ₂		GROW-P ₃	
	Promedios	Desviación Estándar	Promedios	Desviación Estándar	Promedios	Desviación Estándar
CIDA	1.55	0	1.70	0	1.83	0
AG	1.68	0	2.26	0.03	2.44	0.03

8 Conclusiones y Trabajos Futuros

Este trabajo propone la utilización de un *AG* elitista para el tratamiento del problema de protección de redes basada en ciclos pre-configurados *p-Cycle*. Siguiendo el esquema del estado del arte, la resolución del problema se trata en dos pasos, empezando por la construcción de un conjunto de ciclos candidatos utilizando dos conocidos algoritmos, el *SLA* y el *GROW*. Como segundo paso se propone aplicar un *AG* para el cálculo de los ciclos óptimos de protección. Los resultados indican que el enfoque propuesto es promisorio, dado que obtiene mejores conjuntos de ciclos de protección en comparación con uno de los mejores algoritmos del estado del arte.

Otra cuestión interesante de mencionar, es el mejoramiento de las eficiencias de protección, conforme aumenta la calidad y cantidad ciclos candidatos, especialmente en sucesivas ejecuciones del *GROW*, demostrando que puede seguir mejorando, dependiendo del tamaño y la densidad de la topología. Esto es lógico, puesto que al considerar un conjunto de ciclos candidatos reducido y de baja eficiencia, se restringe el espacio de búsqueda.

Considerando lo anteriormente expuesto, los autores proponen como trabajo futuro el diseño de algoritmos de optimización independientes de un conjunto de ciclos candidatos, de forma a lograr una exploración efectiva de todo el espacio de búsqueda. Además, se propone considerar la protección de grupos de riesgos compartidos, como también estudiar el caso en que se conoce un tráfico *a priori*.

Referencias

1. Somani, A. K.: *Survivality and Traffic Grooming in WDM Optical Networks*. Cambridge University Press. (2005).
2. Falcao, P. F.: Pan-European multi-wavelength transport networks. Network: design, architecture, survivability and SDH networking. In: *Proceeding of the First International Workshop on Reliable Communication Networks (DRCN)*. Bélgica (1998).
3. Mohan, G., Siva Ram Murthy, C., Somani, A. K.: Efficient algorithms for routing dependable connections in WDM optical networks. *IEEE/ACM Transactions on Networking* 9 (5), pp. 553 – 566 (2001).
4. Frederick, M. T., Somani, A. K.: A single-fault recovery strategy for optical networks using sub-graph routing. In: *Proceeding in the Seventh IFIP Working Conference on Optical Networks Design and Modeling (ONDM)*. Hungría (2003).
5. Grover, W. D., Stamatelakis, D.: Cycle-oriented distributed preconfiguration: Ring-like speed with mesh-like capacity for self-planning network restoration. In: *Proceeding in the IEEE International Conference on Communications (ICC)*, pp. 537—543. EUA (1998).

6. Doucette, J., He, D., Grover, W. D., O.,Yang.: Algorithmic approaches for efficient enumeration of candidate p-cycles and capacitated p-cycle network design. In: Proceeding of the Fourth International Workshop on the Design of Reliable Communication Networks (DRCN), pp. 212—220. Canada (2003).
7. Schupke, D. A., Gruber, C. G., Autenrieth, A.: Optimal configuration of p-cycles in WDM networks. In: Proceeding of the IEEE International Conference on Communications (ICC), pp. 2761—2765. EUA (2002).
8. Zhang, H., Yang, O.: Finding protection cycles in DWDM networks. In: Proceeding of the IEEE International Conference on Communications (ICC), pp. 2756—2760. EUA (2002).
9. Zhao, T., Yu, H., Li, L.: A novel algorithm for node-encircling and link candidate p-Cycles design in WDM mesh network. *Journal of Chinese Institute of Engineers*, Vol. 29, No. 7, pp. 1227—1233 (2006).
10. Grover, W. D., Doucette, J.: Advances in optical network design with p-cycles: Joint optimization and pre-selection of candidate p-cycles. In: Proceeding of the IEEE LEOS Summer Topicals, pp. 49—50. Canada (2002).
11. Liu, C., Ruan, L.: Finding good candidate cycles for efficient p-cycle network design. In: Proceeding of the Thirteenth IEEE International Conference on Computer Communications and Networks (ICCCN), pp. 321—326. EUA (2004).
12. Johnson, D. B.: Finding all the elementary circuits of a directed graph. *SIAM Journal on Computing* 4, pp. 77 – 84 (1975).
13. Stamatelakis, D., Grover, W. D.: Network restorability design using pre-configured trees, cycles, and mixtures of pattern types. Technical Report TR-1999-05, TRILabs, Canada (2000).
14. Farkas, A., Szigeti, J., Cinkler, T.: p-Cycle Based Protection Scheme for Multi-Domain Networks. In: Proceeding of the Fifth International Workshop on Design of Reliable Communication Networks (DRCN). Italia (2005).
15. Goldberg, D. E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Publishing Company. EUA (1989).
16. Liu, C., Ruan, L.: p-Cycle Design in Survivable WDM Networks with Shared Risk Link Groups (SRLGs) (extended version). *Photonic Network Communications*, Volume 11, Issue 3, pp. 301—311, Springer, Heidelberg (2006).
17. Grover, W. D., y Stamatelakis, D.: Bridging the ring-mesh dichotomy with p-cycles. In: Proceeding of the Second International Workshop on the Design of Reliable Communication Networks (DRCN), pp. 92—104. Alemania (2000).
18. G., Shen, W. D, Grover. : Extending the p-cycle concept to path segment protection for span and node failure recovery. *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 8, pp. 1306—1319 (2003),.
19. Schupke, D. A.: Automatic Protection Switching for p-Cycles in WDM Networks. *Optical Switching and Networking (OSN)*, Elsevier, vol. 2, no. 1, pp. 35—48 (2005).
20. Huang, H., Copeland, J. A.: Hamiltonian cycle protection: A novel approach to mesh WDM optical network protection. In: Proceeding of the IEEE Workshop on High Performance Switching and Routing (HPSR), pp. 31—35. EUA (2001).
21. Zhang, Z., Zhang, Y., Zeng, Q., Zhao, Z., Luo, J., Cao, J.: P-cycle-based protection design for IP over WDM networks. In: Proceeding of the Asia-Pacific Optical and Wireless Communications Conference (APOC), vol. 5282, pp. 598—608. China (2003).
22. Crichigno, J., Barán, B.: Multiobjective Multicast Routing Algorithm for Traffic Engineering. In: Proceeding of the IEEE International Conference on Computers and Communication Networks (ICCCN). EUA (2004).