

**Universidad Técnica Federico Santa María.  
Departamento de Electrónica.**

IPD-438 Seminario de Redes de Computadores.

II Semestre 2002.

Profesor: Agustín González.

Segundo informe de avance.  
**Redes WDM de enrutamiento por  
longitud de onda.**

**Nombre alumno:** Felipe Carrillo Oliva.

**Rol:** 9621013-2

**Fecha:** 9 de octubre de 2002.

## **Introducción.**

En la primera entrega se abarcaron aspectos básicos de las redes WDM, en especial de las redes de enrutamiento WDM.

Además, se presentó el problema RWA (Routing and Wavelength Assignment). En esta entrega, se intenta abarcar este problema, primeramente en forma genérica, para continuar con ejemplos concretos de algoritmos de enrutamiento. En especial, se verá el problema de tráfico estático.

El objetivo es comprender la metodología usada para efectuar enrutamiento estático en redes WDM.

Además, en el apéndice a este informe, se incluye información que ha sido necesaria para complementar el desarrollo, y que en alguna medida, han ayudado a entender el problema aquí tratado.

## **Resumen.**

Primeramente se explica qué es el problema RWA en redes WDM. Después se hace una revisión funcional de algoritmos de enrutamiento y asignación de longitud de onda en redes WDM. Después, se presentan dos ejemplos, uno que efectúa una formulación de Programación Lineal Entera para enrutamiento y asignación de longitud de onda, y un algoritmo de Coloreo Secuencial de Grafo para asignación de longitud de onda. Finalmente conclusiones, referencia y anexo.

## Problema RWA.

La topología física de una red de enrutamiento por longitud de onda está constituida por los nodos terminales, nodos enrutadores y las fibras que los conectan. Los lightpaths establecidos entre nodos terminales a través de la topología física forman la topología lógica de la red. Como lightpath, se designa al camino entre dos nodos terminales de la red, y que puede pasar a través de varios nodos enrutadores. Encontrar rutas a través de la topología física y la asignación de longitudes de onda para los lightpaths es llamado el *problema de enrutamiento y asignación de longitud de onda* (Routing and Wavelength Assignment (RWA) ).

Es deseable establecer lightpaths entre cada par de nodos de una red. Sin embargo, en las redes de enrutamiento WDM el número de longitudes de onda disponibles en las fibras restringe el número de lightpaths.

Si un equipo de enrutamiento está provisto de conversión de longitud de onda, entonces la restricción de continuidad de longitud de onda desaparece, y el problema de enrutamiento es similar que una red de circuito conmutado, donde el único factor limitante es el número de canales disponibles por enlace. Pero si un lightpath opera en una misma longitud de onda en todos los enlaces de fibra que atraviesa, el Enrutamiento y Asignación de Longitud de onda (RWA) debe satisfacer la restricción de continuidad de longitud de onda. Esta restricción implica un uso ineficiente de los canales, y además una alta probabilidad de bloqueo.

En las redes WDM, es posible soportar conversión completa de longitud de onda. Esto se refiere a que, en cada nodo enrutador es posible convertir las señales de salida a cualquier longitud de onda, siempre y cuando esté disponible

El objetivo en RWA es maximizar el throughput por asignación óptima de rutas y longitudes de onda para un patrón de tráfico dado.

Los algoritmos RWA disponibles en la literatura difieren en sus modelos de tráfico y métricas de desempeño. Los modelos de tráfico generalmente están en las categorías: 1) *tráfico estático* y 2) *tráfico dinámico*. En el primer caso, un conjunto de pares de nodos fuente-destino están dados y son fijos, mientras que en el segundo los requerimientos de conexión arriban y terminan desde la red uno por uno aleatoriamente. Las métricas de desempeño generalmente caen en alguna de las categorías: 1) *número de longitudes de onda requeridas*, 2) *probabilidad de bloqueo de conexión (o throughput)* definida como la razón entre el número de conexiones bloqueadas y el número total de conexiones que arriban o dadas, y por último 3) *número de recursos de fibra manejados en los nodos enrutadores (o costo de fibra)*.

Para los algoritmos donde se asume un patrón de tráfico estático, el objetivo es minimizar el número de longitudes de onda para establecer un cierto número de conexiones, o maximizar el número de éstas para un número dado de longitudes de onda disponibles.

Por otra parte, en algoritmos donde se trabaja con tráfico dinámico, el objetivo es minimizar la probabilidad de bloqueo de conexión. Es imperativo que estos algoritmos sean simples y rápidos para que la atención de las peticiones de conexión sea tan rápida como sea posible.

## El problema RWA con patrón de tráfico estático

Básicamente, un problema RWA puede ser planteado como: dado un conjunto de lightpaths que necesitan establecerse en la red, y dada una restricción en el número de longitudes de onda, se necesita determinar las rutas y las longitudes de onda que deben asignarse a los lightpaths de modo de maximizar el número de lightpaths establecidos (o minimizar el número de longitudes de onda requeridas, o bien, alcanzar la mínima probabilidad de bloqueo de lightpath). El problema de enrutamiento se resuelve por técnicas basadas en algoritmos shortest path. La asignación de longitud de onda se determina por técnicas de coloreo de grafos para las rutas seleccionadas. Entonces el problema RWA puede definirse como un problema de optimización de modo que usa varias funciones de costo. Por ejemplo, 1) establecer todos los lightpaths usando un número mínimo de longitudes de onda, 2) establecer todos los lightpaths usando un número mínimo de

El problema RWA puede dividirse en dos funciones: enrutamiento y asignación de longitud de onda. Más aún, ambas funciones pueden dividirse en dos componentes: 1) búsqueda y 2) selección.

### Elementos funcionales de Algoritmos de Enrutamiento.

La función de búsqueda es realizada por técnicas de shortest path, como los algoritmos de Dijkstra y Bellman-Ford.

*Shortest path*: Los algoritmos shortest path buscan la ruta más corta entre un par de nodos en un grafo. La ruta más corta es la que tiene menor costo que cualquiera otra posible. La función de costo es frecuentemente la suma de los pesos de las etapas en la ruta. El algoritmo genera una ruta que es independiente de otras selecciones. Entonces no se requiere ningún orden ni regla de búsqueda.

*Weight Shortest path* : algoritmos iguales a los anteriores, pero el costo de enlace puede variar dinámicamente dependiendo del número de rutas establecidas. Por lo tanto se necesita cierto criterio de búsqueda. Por ejemplo:

- Mayor tráfico primero, ordena los lightpath a ser enrutados comenzando con aquél que tenga mayor tráfico con el objeto de encontrar una ruta.
- Esquema aleatorio, se ordenan los lightpaths a enrutar de forma aleatoria.

*k-Shortest Path*: algoritmos que buscan más de una ruta para cada par fuente-destino, proporcionando flexibilidad para la selección de ruta. Sin embargo, el problema de enrutamiento se transforma en un problema de selección, donde las rutas se eligen para obtener un costo mínimo (número de saltos o costo de enlace) para todos los pares fuente destino.

El proceso de selección es bien hecho por algoritmos secuenciales o de optimización combinatoria. El algoritmo secuencial (conocido como algoritmo Greedy) es simplemente aquél en que la selección para cada lightpath es hecha secuencialmente. Esta técnica no cambia los resultados de selecciones previas, pero los toma en cuenta. Se necesitan dos

subfunciones: orden de selección y norma de selección. La primera es la secuencia en que los lightpath son enrutados (o asignados). La norma de selección es un criterio de decisión para elegir una de las rutas candidatas.

- Orden de selección:
  - Esquemas aleatorios ordenan los lightpaths a ser enrutados aleatoriamente de forma de seleccionar rutas.
  - Esquemas fijos ordenan los lightpaths a ser enrutados en un orden dado (por ejemplo orden alfabético).
  - Esquemas “más largo primero”, ordena los lightpaths a enrutar colocando primero el que posea mayor costo o saltos.
  - Esquemas “más corto primero”, igual que anterior, pero comenzando con menor costo o saltos.
- Norma de selección:
  - Selección aleatoria de una ruta entre las candidatas.
  - Elección de la primera ruta acertada.
  - Selección de una ruta de acuerdo a regla de probabilidad.
  - Selección de ruta en un enlace que tenga el menor número de rutas establecidas.

Por otra parte, las técnicas de selección combinatoria considera la interdependencia del enrutamiento de lightpath. Estos métodos se dividen en mecanismos óptimos y heurísticos. Los óptimos ocupan todas las combinaciones posibles de la interdependencia, obteniendo el mejor resultado, pero el costo por complejidad computacional es crítico. Los métodos heurísticos reducen el espacio de combinaciones.

- Para solución óptima, se usa un programa entero mixto. Esto es en extremo dificultoso en términos de complejidad computacional.
- Para una solución heurística, se propone un algoritmo de redondeo aleatorio. Aquí, el algoritmo de enrutamiento se ejecuta repetidamente para distintos conjuntos de rutas mientras el máximo número de enlaces en todas las rutas se decremente a través de selección alternativa de rutas. Se repite el proceso hasta que no hayan más mejoras posibles.

### **Elementos funcionales de algoritmos de Asignación de Longitud de onda**

El problema de asignación de longitud de onda también puede dividirse funcionalmente en búsqueda y selección. La búsqueda es simplemente saber que cualquier longitud de onda disponible en la ruta elegida puede ser asignada al lightpath. El paso siguiente es la selección de la longitud de onda entre las disponibles de modo de minimizar la utilización de éstas. Similarmente al caso de enrutamiento, la selección puede ser llevada a cabo en forma secuencial o combinatoria. El modo secuencial consiste en ordenar las rutas para asignarles una longitud de onda.

- Orden de selección.
  - Mayor número de vecinos primero, se ordenan las rutas de acuerdo al número de vecinos de modo de asignarles longitud de onda disponible.
  - Más longitudes de onda disponibles, ordena las rutas en orden al número de longitudes de onda disponibles.
  - Mayor tráfico primero, se ordenan las rutas en orden de requerimientos de tráfico.
  - Mayor número de pasos, se ordenan las rutas de acuerdo al mayor número de saltos contados para cada una.
  - Más cortas primero, de acuerdo al menor número de saltos.
  - Orden aleatorio.
- Norma de selección.
  - Asignación de la primera longitud de onda disponible.
  - Asignar la longitud de onda más utilizada disponible.
  - Asignar la longitud de onda menos utilizada.
  - Asignación aleatoria.

La selección combinatoria considera la interdependencia de cada selección. También se usan aproximaciones optimales o heurísticas. Las aproximaciones optimales se aplican a un problema NP, difícil de aplicar a redes muy complejas. Las aproximaciones heurísticas son robustas para reducir el espacio de búsqueda a un pequeño conjunto de lightpaths, sin embargo ello puede incrementar el número de longitudes de onda. Los algoritmos heurísticos propuestos están basados en métodos de colooreo de grafos.

- Selecciones óptimas se resuelven por búsquedas exhaustivas. Estos algoritmos generan el mejor resultado de colooreo para un grafo dado, pero no pueden asegurar un manejo de grafos complejos.

Los algoritmos de selección heurística trabajan bien con problemas de colooreo de grafos. Pueden ser clasificados en:

- *Algoritmos genéticos*: son técnicas estándar para problemas de optimización combinatoria complejos. La idea básica es estimular la evolución de genotipos y selección natural, que han sido aplicados a optimización global, y especialmente a problemas de optimización combinatoria. La idea está basada en la especificación de tres operaciones (cada una es probabilística) sobre objetos denominados “cadenas”; la reproducción combinada de cadenas en la población para crear una nueva cadena (descendiente), alteración por mutación espontánea de caracteres en un cadena, y cruce de cadenas para intercambiar valores, y crear nuevos cadenas en su lugar. La reproducción y operaciones de intercambio puede incluir competencia dentro de la población. Todas estas operaciones ocurren hasta encontrar hasta encontrar una cadena óptima.
- *Algoritmos de recocido estimulado*: son otra técnica estándar para problemas complejos de optimización combinatoria. Ver apéndice A.2.
- *Algoritmos TABÚ*, son métodos heurísticos relativamente nuevos. La idea básica es una búsqueda aleatoria local, pero algunos movimientos son prohibidos.

## Un ejemplo de formulación de Programa Lineal Entero.

Dependiendo si los nodos enrutadores poseen o no conversión, podemos clasificar los lightpath como *wavelength path* (*WP*) para el caso sin conversión, y *virtual wavelength path* (*VWP*) para el caso con conversión. En ambos tipos de enrutamiento, se asigna a los lightpath una ruta y longitud(es) de onda. Además deben satisfacerse ciertas restricciones. En el caso de *VWP* el número de lightpaths pasando a través de un enlace de fibra debe ser menor que la capacidad del enlace. Para *WP* se agrega la restricción de continuidad de longitud de onda, que limita los lightpaths que son asignados a una misma longitud de onda al número de fibras del enlace. En una topología física fija entonces, puede que no sea posible enlutar todos los lightpaths. Por lo tanto, el problema se formula como una programación lineal entera que maximice el número de lightpaths acomodados en la red. Se consideran dos formulaciones para ambos casos, *WP* y *VWP*. Una *formulación de ruta* enumera todas las rutas entre pares fuente-destino, y se determina cuántas veces se usa cada ruta. Para *formulaciones de flujo*, la variable de decisión básica son los flujos en los enlaces generados a través de cada par de nodos terminales.

La topología física de la red se modela como un grafo unidireccional  $G = (V, E)$ , donde  $V = \{v_1, v_2, \dots, v_n\}$  es el conjunto de nodos enrutadores y terminales y  $E = \{e_1, e_2, \dots, e_m\}$  es el conjunto de enlaces de fibra. En este grafo cada enlace puede contener una o más fibras, por tanto, un vector de capacidad  $K = [k_j]$  se define para representar la capacidad de los enlaces.  $k_j$  es el número de fibras en el enlace  $j$  multiplicado por el número de canales de longitud de onda en cada fibra. Para facilitar la formulación, el conjunto de pares fuente-destino  $S = \{v_i v_j \mid 1 \leq i < j \leq n\}$  con tamaño  $s$  se usa para representar los pares de nodos terminales. Luego, las peticiones de lightpath pueden ser representadas por un vector de demanda  $D = [d_i]_{1 \times s}$  donde  $d_i$  es el número de lightpaths que se establecen entre el par fuente-destino  $i$ .

En formulaciones de ruta, todas las posibles rutas cíclicas libres entre los pares fuente-destino se determinan y el conjunto  $R = \{r_1, r_2, \dots, r_r\}$  se forma por esas rutas. En correspondencia con el conjunto  $R$ , la matriz  $Q$  de incidencia ruta con par fuente-destino y la matriz de incidencia ruta con enlace de fibra, se formulan como:

$$Q = [q_{ij}]_{r \times s} \quad q_{ij} = \begin{cases} 1 & \text{si la ruta } r \text{ incide con el par fuente-destino } i \\ 0 & \text{otro caso} \end{cases}$$

$$B = [b_{ij}]_{r \times m} \quad b_{ij} = \begin{cases} 1 & \text{si la etapa } e_i \text{ se usa en la ruta } r \\ 0 & \text{otro caso} \end{cases}$$

En enrutamiento *VWP* usando formulación de ruta, las variables ILP (integer linear programming) se pueden representar por el vector de enrutamiento  $F = [f_i]_{1 \times r}$ , donde  $f_i$  es el número de veces que se usa una ruta. Por lo tanto, la formulación para *VWP* que maximiza el tráfico transportado se da como:

$$\max \sum_{i=1}^r f_i$$

sujeto a:

(la demanda es parcialmente o completamente satisfecha)

$$F \cdot Q \leq D$$

(número de lightpath que comparten un enlace está limitado a la capacidad de éste)

$$F \cdot B \leq K^T$$

En enrutamiento *WP* usando formulación de ruta, las variables ILP se pueden representar por una matriz de ruta-longitud de onda  $W = [w_{ij}]_{r \times ?}$ , donde  $w_{ij}$  es el número de pasos usados en la ruta  $r_i$  y longitud de onda  $j$ , y  $?$  es el número de canales de longitud de onda en cada fibra. Por tanto, la formulación de ruta para *WP* que maximiza el tráfico transportado puede ser:

$$\max \sum_{i=1}^r \sum_{j=1}^l w_{ij}$$

sujeto a:

(la demanda es parcialmente o completamente satisfecha)

$$1_l \cdot W^T \cdot Q \leq D$$

(el número de lightpaths que usan la misma longitud de onda y comparten el mismo enlace se limita al número de fibras del enlace)

$$W \cdot B \leq \frac{1}{I} 1_l^T K$$

donde  $1_?$  es un vector  $1 \times ?$  en que todos los elementos son 1.

En formulaciones de flujo, las variables de decisión básicas son los flujos en los enlaces de fibra generados por cada par fuente-destino. Las variables en *VWP* se representan por la matriz de enrutamiento  $X = [x_{ij}]_{m \times s}$ , con  $x_{ij}$  igual a la cantidad de tráfico fluyendo del par fuente-destino  $k$  a través del enlace  $j$ . La formulación de flujo para enrutamiento *VWP* que maximiza el tráfico transportado puede ser como:

$$\max \sum_{j=1}^s p_j$$

sujeto a:

(la demanda es parcialmente o completamente satisfecha)

$$P \leq D$$

(para cada par fuente-destino el número de lighpath enrutados y la conservación del flujo deben ser válidos en cada vértice)

$$\sum_{a_j \in a(u_i)} x_{jk} - \sum_{a_j \in b(u_i)} x_{jk} = \begin{cases} p_k & \text{si } ?_i \text{ es fuente del par fuente-destino k} \\ -p_k & \text{si } ?_i \text{ es destino del par fuente-destino k} \\ 0 & \text{otro caso} \end{cases}$$

(el flujo total en cada etapa no excede la capacidad física)

$$\sum_{k=1}^S x_{jk} \leq k_j$$

donde  $P = [p_j]_{1 \times s}$  representa el tráfico enrutado donde  $p_{ij}$  es el número de lightpaths enrutados entre el par fuente-destino  $j$ ;  $a(?_i)$  y  $b(?_i)$  representan al conjunto de enlaces que tienen a  $?_i$  como origen y destino respectivamente.

Las variables en  $WP$  usando formulación de flujo son representadas por la matriz ruta-longitud de onda,  $Y = [y_{ijk}]_{s \times m \times s}$ , siendo  $y_{ijk}$  igual a la cantidad de tráfico que fluye entre el par fuente-destino  $k$  a través del enlace  $j$  y en la longitud de onda  $i$ . Luego, la formulación para enrutamiento  $WP$  que maximiza el tráfico puede darse:

$$\max \sum_{z=1}^Z \sum_{j=1}^S t_{zk}$$

sujeto a:

(la demanda es parcialmente o completamente satisfecha)

$$\sum_{z=1}^I t_{zk} \leq d_j$$

(para cada par fuente-destino el número de lighpath enrutados y la conservación del flujo deben ser válidos en cada vértice)

$$\sum_{a_j \in a(u_i)} y_{zjk} - \sum_{a_j \in b(u_i)} y_{zjk} = \begin{cases} t_{zk} & \text{si } ?_i \text{ es fuente del par fuente-destino k} \\ -t_{zk} & \text{si } ?_i \text{ es destino del par fuente-destino k} \\ 0 & \text{otro caso} \end{cases}$$

(restricciones de capacidad)

$$\sum_{k=1}^S y_{zjk} \leq \frac{k_j}{T}$$

donde  $T = [t_{zk}]$  representa el tráfico enrutado y  $t_{zk}$  es el número de lightpaths enrutados entre el par fuente-destino  $j$  y en la longitud de onda  $z$ .

El objetivo de todas las formulaciones anteriores es la maximización del número de lightpath que se requiere acomodar en una topología física dada. Otras formulaciones pueden apuntar a minimizar el número de canales de longitud de onda requeridos en cada fibra (en efecto, el número total de longitudes de onda).

La tabla siguiente compara los tamaños de los problemas formulados. Puede verse que el tamaño del problema *VWP* es independiente del número de longitudes de onda, mientras que el caso de *WP* crece cuadráticamente con el número de longitudes de onda. Por otra parte el número de variables es proporcional al número de rutas en formulaciones de ruta. Entonces, el número de variables crece exponencialmente con el tamaño de la red para redes con muchas conexiones. Aunque el número de variables es mucho menor en formulaciones de flujo, el número de restricciones sube exponencialmente con el tamaño de la red.

	Formulaciones de ruta.		Formulaciones de flujo.	
	VWP-routing	WP-routing	VWP-routing	WP-routing
Número de Variables	$r$	$rx\lambda$	$mxs$	$mxsx\lambda$
Número de Restricciones	$m+s$	$mx\lambda+s$	$m+nxs+s$	$mx\lambda+nxsx\lambda+2s$

### Un ejemplo de enrutamiento y asignación de longitud de onda: Algoritmo de Coloreo Secuencial de Grafo.

El problema de asignación estática de longitud de onda puede ser reducido a un problema de colooreo secuencial de grafo, el cuál se dice que es NP-completo<sup>1</sup>. Dado un conjunto de lightpaths y sus rutas, se construye un grafo  $G(V,E)$  donde cada lightpath se representa por un nodo en el grafo  $G$ . Si dos lightpaths atraviesan un enlace de fibra común, entonces se dibuja una línea no dirigida entre ambos nodos. Resolviendo el problema de dar color a los nodos de manera que dos nodos adyacentes no tengan el mismo color, se puede reducir el número de longitudes de onda a utilizar.

El número de colores necesitados para colorear el grafo  $G$ , conocido como el Número Cromático  $X(G)$ , es difícil de determinar. Sin embargo, este algoritmo entrega un número óptimo de colores. En el método secuencial de colooreo, se asignan secuencialmente colores a los nodos, agregando nuevos colores para acomodar nuevos vértices, resguardando que el número total de colores sea mínimo. Dos métodos principales usados son los algoritmos Largest-first y Shortest-first. Se dará a continuación una breve explicación del primero.

---

<sup>1</sup> Las clases de complejidad en ciencias computacionales, define a los problemas NP como aquellos cuya solución hasta la fecha no han podido ser resueltos de manera exacta por medio de algoritmos deterministas eficientes, pero que pueden ser resueltos por algoritmos no-deterministas.

- **Algoritmo Largest-first**

En esta aproximación, los vértices  $V(G) = v_1, v_2, \dots, v_n$  se ordenan de acuerdo al número de líneas que los conecta a otros (grado del nodo) en forma decreciente. En cada paso, al nodo con mayor grado se le asigna un color, y las líneas correspondientes se borran del grafo, reduciendo el grado de los nodos adyacentes. Esto asegura un mínimo de colores usados para colorear el grafo.

## Conclusiones.

- El enrutamiento estático en redes WDM abarca varios aspectos de la optimización y la teoría de grafos, poniendo en evidencia la base teórica matemática que se usa en enrutamiento en redes.
- La metodología general en la primera parte del desarrollo, puede considerarse como una base de métodos concretos.
- Esto se corrobora a través de los ejemplos, de Programación Lineal Entera y Coloreo Secuencial de Grafo, que representan a los métodos óptimos combinatorios.
- Sin embargo, queda pendiente algún ejemplo que represente a los métodos heurísticos.
- Los algoritmos que buscan un óptimo no pueden aplicarse a redes con muchos nodos, debido a su complejidad computacional. Los métodos heurísticos permiten obtener una solución de mucho menor complejidad, pero no garantizan la solución óptima.
- Finalmente, la elaboración de segunda entrega me permitió refrescar los conocimientos en optimización y algoritmos, y algunos conceptos hasta ahora desconocidos, como los métodos heurísticos.

## Referencias.

- **A Functional Classification of Routing and Wavelength Assignment Schemes in DWDM networks:**
  - Static Case. Jin Seek Choi, Nada Golmic, Francois Lapayrere, Frederic Monvcaux and David Sue.
  - Advanced Network Technologies Division, Information Technology Laboratory, Nacional Institute of Standards and Technology, Gaithersburg, MD, USA.
- **All Optical Networking.**
  - Semih Bilgen, Altan Koçyigit, September 2001.
- **Wavelegth Assignment In Optical Networks.**
  - Ashish Natani, Rajiv Jayant, Saurabhi Vadalkar, Vijay Sharma and Vinod Vokkarane.
  - Department of Computer Science, The University of Texas at Dallas, Texas.
- **Introducción a la programación heurística(presentación PPT).**
  - Ignacio Olmeda
  - Profesor Titular de Universidad, Departamento de Ciencias de la Computación y Escuela Universitaria de Turismo, Universidad de Alcalá.
  - Febrero 1999.

## Apéndice.

### A. Algoritmos heurísticos.

Un *algoritmo heurístico* es un procedimiento de búsqueda de soluciones casi optimales a un coste computacional razonable, sin ser capaz de garantizar la optimalidad o factibilidad de las soluciones empleadas ni determinar a qué distancia de la solución óptima nos encontramos (Reeves, 1995)

#### A.1 Algoritmos genéticos.

Los *Algoritmos Genéticos (AG)* son procedimientos de búsqueda de soluciones basados en el principio de evolución natural (Darwin y Wallace).

*Principio de evolución natural:* los individuos mejor adaptados al medio tienen mayores probabilidades de sobrevivir y reproducirse, transmitiendo su información genética a sus sucesores.

Al cabo de muchas generaciones, la población estará integrada por individuos bien adaptados al medio y con características genéticas similares.

Tres observaciones:

- La evolución se manifiesta sobre una población, no sobre un individuo.
- La selección actúa sobre cada uno de los individuos
- La información se transmite genéticamente codificada.

La idea de los AG consiste en aplicar estas ideas sobre conjuntos de soluciones de un problema matemático, logrando nuevas y mejores soluciones.

Dado el problema combinatorio:

$$\begin{aligned} \max \quad & f(x) \\ \text{s.a.} \quad & x \in S \subset Z^n \end{aligned}$$

Podemos hacer la correspondencia:

$$\begin{array}{lll} \text{Población} & \Leftrightarrow & \text{Conjunto de soluciones} \\ \text{Adaptación} & \Leftrightarrow & \text{Valor de la función objetivo} \end{array}$$

Obsérvese que, a diferencia de los métodos de descenso por gradiente, búsqueda aleatoria y recocido simulado, los AG actúan sobre *conjuntos de soluciones* que van mejorando iteración a iteración y no sobre una sola solución.

En lo que sigue, sin pérdida de generalidad, supondremos que queremos resolver el problema de una sola variable:

$$\begin{aligned} \max \quad & f(x) \\ \text{s.a.} \quad & x \in S \subset Z \end{aligned}$$

Los AG son procedimientos *iterativos* que emplean cuatro ideas fundamentales:

- a) Codificación binaria
- b) Selección
- c) Cruce
- d) Mutación

La selección es el mecanismo por el cual soluciones más próximas al óptimo (*individuos mejor adaptados*) tienen mayor probabilidad de sobrevivir y ser elegidos (*seleccionados*) para reproducirse.

Supongamos que, en una determinada iteración del algoritmo, tenemos un conjunto de soluciones, que denominaremos *población*:

$$P = \{x1, x2, \dots, xr\}, xi \in S \subset Z$$

y que cada una de ellas (que denominamos *individuos*) tiene un valor de la función objetivo

$$f(x1), f(x2), \dots, f(xr)$$

La selección supone que cuanto mayor sea  $f(xi)$ , mayor sea la probabilidad  $pi$  de que  $xi$  sea elegido y se reproduzca.

Una manera sencilla de hacer esto es tomar:

$$p_i = \frac{f(x_i)}{\sum_{j=1}^r f(x_j)}$$

Después asignamos realizamos extracciones aleatorias de la distribución de probabilidad obtenida.

El cruce consiste en el intercambio de una sección cualquiera de las cadenas binarias de dos individuos al azar de la población seleccionada.

El cruce sigue los siguientes pasos:

primero: elegir un punto de corte, al azar, de ambas cadenas:

- ejemplo:

$$\begin{array}{l} 110 | 000 \\ 011 | 011 \end{array}$$

segundo: intercambiar las secciones entre ambas ambas cadenas, generando una nueva solución (*individuo hijo*) con la parte inicial del primer individuo (*madre*) y la parte final del segundo (*padre*):

*ejemplo:*

$$\begin{array}{l} 110 | 000 \text{ (madre)} \\ \qquad\qquad\qquad\Rightarrow\qquad\qquad\qquad 110 | 011 \\ 011 | 011 \text{ (padre)} \end{array}$$

La *mutación* consiste en cambiar, con una probabilidad muy pequeña, cualquier elemento de una cadena, pasando de cero a uno o viceversa.

*ejemplo:*

$$\begin{array}{l} 110000 \text{ (antes de la mutación)} \\ 111000 \text{ (después de la mutación)} \end{array}$$

La mutación permite generar nuevas soluciones en poblaciones estancadas que generarían individuos idénticos a los padres.

Por ejemplo, la población:

$$x1=110001$$

$$x2=110001$$

$$x3=110001$$

generaría siempre los mismos hijos.

Esquema de un Algoritmo Genético

- paso 1:  $t=0$ ; generar una población aleatoria  $P=\{x1,...,xn\}$
- paso 2: calcular  $f(x1),...,f(xn)$
- paso 2:  $t=t+1$ ; aplicar los procedimientos de
  - selección
  - cruce
  - mutación
 generando una nueva población  $P'=\{x'1,...,x'n\}$
- paso 3: reemplazar  $P$  por  $P'$
- paso 4: si no se cumple un criterio de optimalidad ir al paso 2

## A.2 Algoritmo de recocido estimulado.

En física de la materia condensada, el *recocido* es el proceso físico consistente en calentar un sólido hasta que se funde, seguido de un enfriamiento lento hasta que el sólido cristaliza en un estado con una estructura molecular perfecta. Durante este proceso, se minimiza la energía libre del sistema.

El *recocido simulado* es un procedimiento heurístico que toma estas ideas y las aplica en la solución de problemas de optimización.

La idea básica consiste en establecer un paralelismo entre la *función objetivo* a minimizar y la *energía libre* de un sistema, así como entre *soluciones* del problema y los *estados* de dicho sistema.

Los *algoritmos de recocido simulado* emplean la formalización propuesta por Metropolis *et al.* (1953) para describir la evolución de un sólido sometido a un baño de calor hasta alcanzar el equilibrio termal.

El *criterio de Metropolis* determina la probabilidad de que un sólido sometido a un proceso de recocido que se encuentre en un determinado estado pase a un nuevo estado con distinta energía.

Supongamos que un sistema se encuentra en un estado  $i$ , con una energía libre  $Ei$ , y supongamos que generamos un nuevo estado  $j$  con energía  $Ej$ .

Si la diferencia de energía  $Ej - Ei$  es menor o igual que cero, el estado  $j$  se acepta como nuevo estado con probabilidad uno. Si la diferencia de energía es mayor que cero, dicho estado se acepta probabilidad:

$$\exp\left(\frac{E_i - E_j}{k_B T}\right)$$

donde  $T$  representa la temperatura del baño de calor y  $kB$  es una constante física (la *constante de Boltzmann*).

El algoritmo de recocido simulado no es más que un procedimiento de generación de soluciones seguido de la aplicación del criterio de Metropolis de manera repetitiva:

- paso 1:  $t=0$ ; generar un estado aleatorio  $i$
- paso 2:  $t=t+1$ ; generar un estado  $j$  en una vecindad de  $i$
- paso 3:
  - si  $Ej < Ei \rightarrow i=j$
  - si  $Ej \geq Ei \rightarrow i=j$  con probabilidad  $\exp(-(Ej-Ei)/c(t))$
- paso 4: si no se cumple un criterio de optimalidad ir al paso 2