



INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE FÍSICA Y MATEMÁTICAS

**“DISEÑO RADIAL DE COMBUSTIBLE USANDO PATH
RELINKING”**

TESIS

**QUE PARA OBTENER EL GRADO DE
INGENIERA MATEMÁTICA**

P R E S E N T A:

YERANIA CAMPOS SILVESTRE

BAJO LA DIRECCIÓN DE:

M. en C. JOSÉ ALEJANDRO CASTILLO MÉNDEZ

Dr. EDMUNDO DEL VALLE GALLEGOS

MARZO DEL 2007

Dedicado a mi familia a quienes amo y admiro. Gracias por enseñarme, cuidarme, amarme y prepararme para afrontar la vida. Los llevo en mi corazón en cada paso que doy.

Felicidades papá por ser éste tu mes...

Índice

	Página
I. Introducción	<i>iii</i>
Capítulo 1. Diseño Radial de Celdas de Combustible	1
1.1 Antecedentes	1
1.2 Planteamiento del problema	11
Capítulo 2. Técnica Path Relinking	19
2.1 Path Relinking	20
2.2.1 Algoritmo de Path Relinking	25
2.2.2 Generación de trayectorias	25
2.2 Búsqueda Dispersa	27
2.2.1 Elementos de la Búsqueda Dispersa	28
2.2.1.1 Generación de soluciones dispersas	29
2.2.1.2 Conjunto Referencia	32
2.2.1.3 Combinación convexa	34
2.2.1.4 Actualización	36
Capítulo 3. Implementación de la Técnica Path Relinking al problema	38

del diseño radial.	
3.1 Algoritmo de Path Relinking en el diseño radial	39
3.2 Implementación	40
3.2.1 Generación de la semilla	40
3.2.2 Conjunto Referencia	41
3.2.2.1 Generador de soluciones dispersas	42
3.2.2.2 Conjunto referencia	42
3.2.3 Mejora del conjunto Referencia	48
3.2.3.1 Método de combinación	49
3.2.3.2 Actualización	51
3.2.4 Re-encadenamiento de trayectorias	51
3.2.5 Actualización del conjunto referencia	53
Capítulo 4. Resultados	55
4.1 Características de las celdas analizadas	56
4.2 Obtención del conjunto Referencia	59
4.3 Actualización del conjunto referencia	62
4.4 Celdas Obtenidas	64
Capítulo 5. Conclusiones	73
Referencias	77
Agradecimientos	81

INTRODUCCIÓN

El presente trabajo muestra los resultados obtenidos al implementar la técnica de optimización combinatoria conocida como Path Relinking (Re-encadenamiento de Trayectorias), al problema del diseño radial de ensambles de combustible nuclear, para reactores de agua en ebullición (BWR Boiling Water Reactor por sus siglas en inglés), este tipo de reactores son los que se utilizan en la Central Nucleoeléctrica de Laguna Verde Veracruz. Como en cualquier otra planta generadora de electricidad que hace uso de algún combustible para producir calor y que necesita cada determinado tiempo (de 12 a 14 meses) hacer un abastecimiento del mismo, debido a que éste se desgasta o se quema, en las plantas nucleoeléctricas a esta actividad se le denomina recarga de combustible. En dicha recarga intervienen distintas actividades, entre las cuales destacan los diseños radial y axial de ensambles de combustible, los patrones de barras de control y el estudio de multiciclos, cada una de estas etapas con su propia complejidad. Este trabajo se limitará a estudiar en forma independiente el diseño radial, sin considerar las otras actividades. Estas fases son básicas para el diseño de recarga de combustible y de estrategias de operación del reactor.

El diseño radial es un trabajo difícil y complicado, por los parámetros involucrados, así como los recursos de cómputo necesarios para dicha tarea. En la literatura especializada se han reportado algunos estudios en los que se han dado soluciones óptimas al problema, con cierto enfoque, haciendo uso de métodos heurísticos, como por ejemplo, Búsqueda Tabú [15] y Algoritmos Genéticos [7],

por mencionar algunos. En estos trabajos se han reportado avances importantes para dar una aproximación aceptable a la solución.

La solución al problema planteado conlleva una serie de pasos, los cuales están plasmados en el presente trabajo de acuerdo a la estructura siguiente; la tesis consta de 5 capítulos en los cuales se desarrolla cada una de las etapas del trabajo en forma explícita.

En el Capítulo 1 se dan los antecedentes y conocimientos necesarios para poder entender la problemática a resolver. En este mismo capítulo se da el planteamiento del problema, así como la importancia y los criterios que se consideran para obtener una configuración definida del acomodo de los combustibles en un ensamble como parte integral de la recarga de combustible.

De acuerdo a la estructura, el Capítulo 2 está enfocado a la explicación de la técnica conocida como Path Relinking (PR), se exponen sus características, fundamentos y orígenes. Es importante mencionar que Path Relinking requiere de un conjunto de soluciones iniciales para su implementación, por lo cual es necesario obtener dichas soluciones con algún otro método; en este caso, la técnica empleada para ello fue la Búsqueda Dispersa (BD), en este capítulo se explican brevemente sus características.

Por otro lado, en el Capítulo 3 se presenta la implementación de la técnica PR al problema del diseño radial de combustibles. El capítulo incluye la función objetivo empleada, las distintas estrategias, así como las dificultades que se presentaron a lo largo de la implementación.

En el Capítulo 4 se discuten los resultados obtenidos con la implementación de la técnica PR de acuerdo a los criterios establecidos y el desempeño del sistema creado para tal fin. También se muestran los mejores diseños obtenidos con el sistema creado.

Finalmente, en el capítulo 5 se destaca la eficacia de la técnica para encontrar diseños de celdas que cumplan con los requerimientos establecidos en el Capítulo 1. Para concluir el capítulo se presentan las conclusiones del trabajo realizado.

Capítulo 1

DISEÑO RADIAL DE CELDAS DE COMBUSTIBLE

*Todo lo que una persona puede imaginar, otras personas
podrán hacerlo realidad*

Julio Verne

1.1 ANTECEDENTES

En los últimos tres decenios, la población mundial casi se ha duplicado y continuará en aumento. Según las estimaciones actuales, en el año 2020 habitarán en este planeta unas 8100 millones de personas, y casi el 90% del incremento de esa población se registrará en los países en desarrollo. El consumo actual de electricidad por habitante en estos países, puede utilizarse como indicador del nivel de vida, además de las numerosas necesidades urgentes, tales como los alimentos, la vestimenta, el alojamiento y el trabajo, por lo que la

disponibilidad de energía es un requisito previo importante para el desarrollo socioeconómico en cualquier parte del mundo (Figura 1.1).

La energía eléctrica se ha convertido en parte fundamental de nuestra vida diaria, la importancia de ésta radica en que es una de las principales formas de energía usadas en el mundo actual, de hecho puede decirse que se usa en todas partes. En múltiples maneras, afecta e influye nuestra vida diaria. Vemos el uso de la electricidad directamente en nuestros hogares para la iluminación, el funcionamiento de los aparatos electrodomésticos, el televisor, estufas, etc.

La electricidad se ha usado en la fabricación de la mayoría de las cosas que empleamos y usamos, difícilmente podríamos imaginarnos los niveles de progreso que el mundo ha alcanzado. Hoy en día representa un mayor reto generarla, debido a que el consumo aumenta considerablemente y para satisfacer esta demanda se requiere generar cada día mayor cantidad de energía.



Figura 1.1. Electricidad como indicador del desarrollo de un país.

Las plantas generadoras de energía eléctrica constan de una máquina motriz, que mueve un generador eléctrico. Las [fuentes](#) que se utilizan para generar electricidad son: el movimiento del agua que corre o cae, el calor para producir

vapor, la geotermia, las energías renovables (solar, eólica, el carbón y leña) y finalmente las no renovables (gas natural, petróleo y la energía nuclear).

Es importante saber que en México más del 50% de la electricidad se genera a base de combustibles fósiles utilizados en centrales termoeléctricas las cuales consumen [gas natural](#), combustóleo y carbón. Como se muestra en la Figura 1.2, en México la energía nuclear representa un 5.01% de la producción total [22].

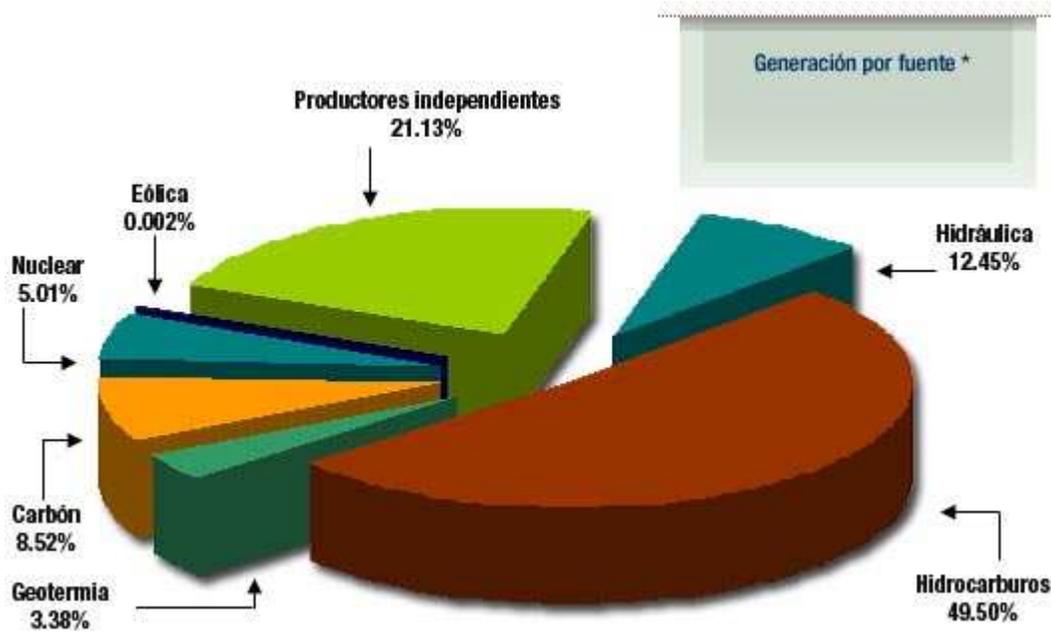


Figura 1.2. Generación de electricidad.

Una buena parte de las plantas generadoras queman algún combustible fósil para calentar calderas que contienen agua y de esta manera producir vapor. El vapor se lleva a una turbina, la cual está conectada a un generador y cuando éste gira, convierte ese movimiento giratorio en electricidad. Después de que el vapor pasa a través de la turbina, es llevado a una torre de enfriamiento, donde se condensa y

se convierte nuevamente en agua líquida para ser utilizada otra vez en la caldera y repetir el [proceso](#).

En el esquema de la Figura 1.3 se observa el recorrido de un fluido (agua o vapor de agua), el cual generará la fuerza necesaria para mover la turbina. El movimiento que se genera (energía mecánica) es transformada en el generador.

La preocupación por los efectos ambientales, resultado del quemado de combustibles fósiles ha fomentado el interés por las fuentes de energía poco contaminantes. Dependiendo de la ubicación geográfica y el nivel de industrialización, la energía final se consume de distintas maneras. Entre las distintas fuentes energéticas con las que cuentan los países, la energía nucleoelectrónica representa una de las pocas opciones que permite producir electricidad, en forma económica y en condiciones ambientalmente aceptables, por lo que podría representar para algunas naciones una opción atractiva para contribuir a un suministro energético seguro y confiable a largo plazo.

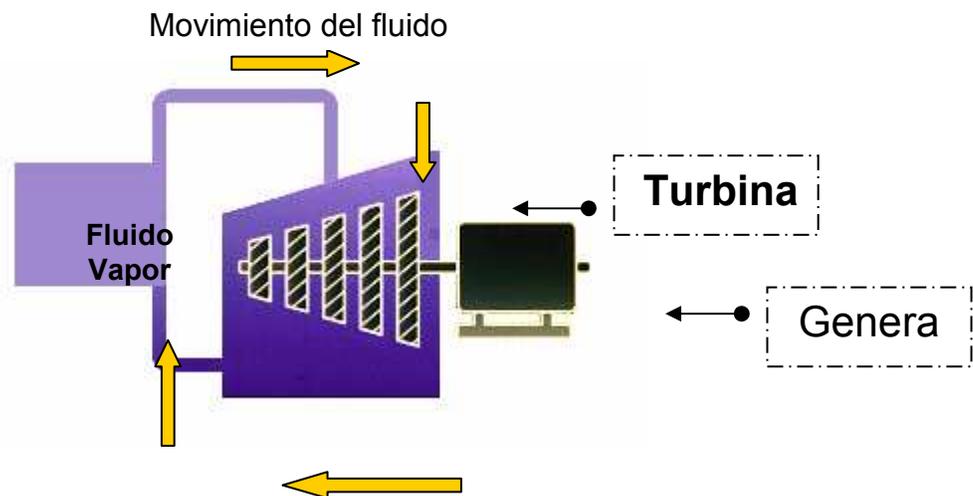


Figura 1.3. Generación de electricidad.

Las plantas nucleoelectricas utilizan la energía nuclear para producir el calor necesario que convierte el agua en vapor y de esta manera mover las turbinas y los generadores. Como en otras plantas, estas centrales también requieren de un combustible, que puede ser el Uranio, ya sea enriquecido o en forma natural, principalmente. El uranio natural se encuentra distribuido en la corteza de la tierra ya sea a nivel del suelo o bajo éste. La mayoría de las minas de uranio que actualmente están en trabajos de extracción tienen una concentración menor del 0.5% de uranio por tonelada de extracción. El uranio natural está formado por U^{238} , U^{235} y U^{234} el contenido de cada uno de estos elementos es el siguiente.

Uranio natural:

- 99.29% de U^{238}
- 0.71% de U^{235}
- 0.0057% de U^{234}

En el proceso de enriquecimiento el U^{235} se incrementa a un rango del 2 al 5%, de tal manera que el uranio continúa estando formado por un 95% de U^{238} aproximadamente.

En términos generales, la energía nuclear se basa en el concepto fundamental de la fisión nuclear y ésta ocurre cuando un átomo pesado, por ejemplo el Uranio, absorbe un neutrón, con este suceso el núcleo del átomo puede llegar a ser inestable y de esta manera dividirse, como se muestra en la Figura 1.4.

La fisión nuclear es una reacción que se produce mediante el bombardeo de determinados núcleos, denominados núcleos fisiónables utilizando neutrones como proyectiles.

En la fisión acontece que al romperse el núcleo se liberan varios neutrones con una energía igual o superior a la de los neutrones incidentes, lo que permite que los neutrones producidos den lugar a nuevas fisiones y los liberados en ellas a

otras nuevas y así sucesivamente, además de que surgen diversos productos de fisión. Con ello se puede conseguir que una vez iniciada la reacción no sea necesario continuar con el bombardeo mediante neutrones externos, puesto que la reacción se mantiene por sí misma. A esto se le denomina reacción en cadena (ver Figura 1.5).

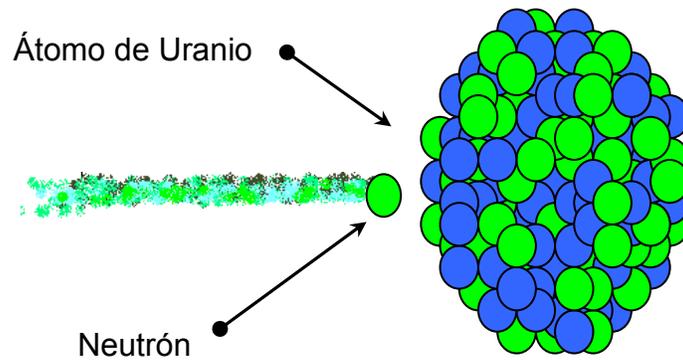


Figura 1.4. Fisión de un átomo de Uranio.

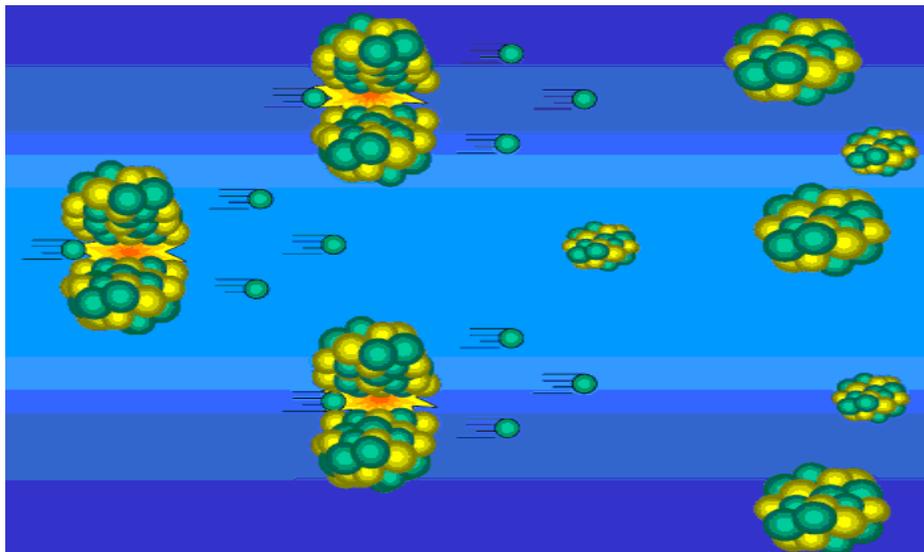


Figura 1.5. Reacción en cadena.

Los neutrones liberados del núcleo mediante una fisión surgen a altas velocidades por lo que éstos se tienen que frenar con un moderador para que pueda producirse la reacción en cadena. Para reactores de agua en ebullición (BWR por sus siglas en inglés) el moderador empleado es el agua (Figura 1.6), dicho elemento también sirve como refrigerante del reactor. El refrigerante es el fluido utilizado para extraer el calor producido en el núcleo de un reactor nuclear.

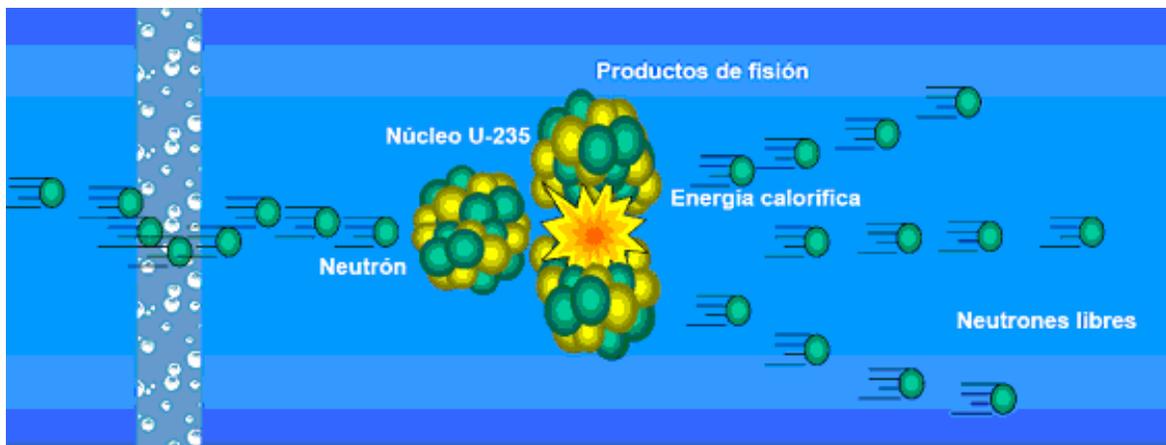


Figura 1.6. El agua como moderador de la velocidad de los neutrones.

A consecuencia de la fisión, la energía desprendida es suficiente para calentar el agua utilizada para estos fines, el vapor de agua generado se transporta por tuberías hasta una turbina, en la que esa energía calorífica es convertida en energía cinética (de movimiento o fuerza). Después, esa energía se aprovecha para mover un generador, y éste produce la electricidad que pasa a la red eléctrica nacional para su distribución, después el vapor pasa por un condensador, ya en su estado líquido el agua se vuelve a usar como moderador y refrigerante. Éste es un ciclo que permite la funcionalidad del sistema el cual se observa en la Figura 1.7.

Como se mencionó con anterioridad, la planta requiere de un combustible para su funcionamiento, a éste se le llama combustible nuclear y se denomina así a cualquier material que contiene núcleos fisionables y puede emplearse en un reactor nuclear para que en él se desarrolle una reacción en cadena. Dicho combustible se encuentra en el núcleo del reactor como se observa en la Figura 1.7. El núcleo es la parte más importante del reactor, es aquí donde se llevan a cabo las reacciones nucleares y se tienen los mecanismos necesarios para el control del sistema el cual consiste de un recipiente llamado vasija que contiene al núcleo y a los componentes internos que lo soportan, así como aquellos que proporcionan las trayectorias de flujo adecuadas para los conductos de agua y vapor.

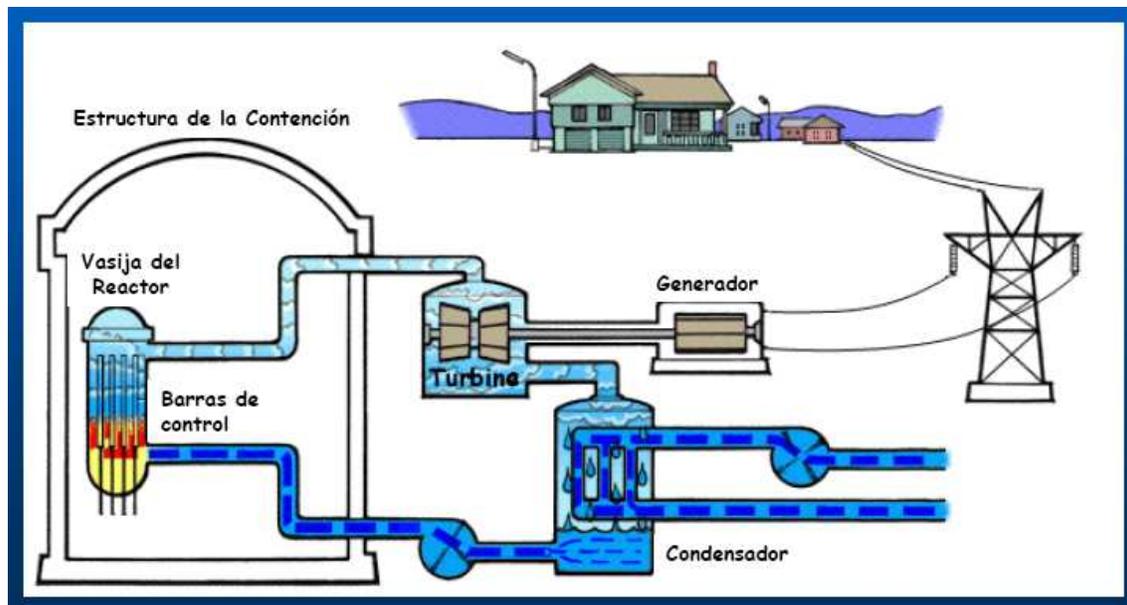


Figura 1.7. Funcionamiento de una planta nucleoelectrica.

Debido al proceso de fisión que se lleva a cabo en el núcleo del reactor los combustibles que se utilizan en un ensamble se desgastan, por esta razón los combustibles deben cambiarse periódicamente por nuevos, de forma tal que

cumplan con las metas de obtención de energía, a cada periodo se le llama un ciclo de combustible. No obstante, sólo una parte de los ensambles son retirados ya que el “quemado” de éstos no es uniforme. Además, no sería una buena política, por razones de eficiencia y economía, por lo que se tiene que hacer un estudio previo para poder determinar las ubicaciones óptimas de los combustibles nuevos y los usados. De este modo, los combustibles disponibles se colocan de tal manera que se maximice el rendimiento, siempre que se cumplan los límites de seguridad y las restricciones de funcionamiento.

En consecuencia, los operadores de reactores se enfrentan con el llamado **problema de recarga de combustible óptimo**, que consiste en acomodar los conjuntos de combustibles usados y nuevos, de modo que se optimice la reactividad en el núcleo del reactor.

La radioactividad es la propiedad de algunos elementos de emitir partículas u ondas electromagnéticas. Esta propiedad se debe a la existencia de una descompensación entre el número de neutrones y de protones del núcleo del átomo, que provoca una inestabilidad y una liberación de la energía acumulada en forma de partículas u ondas. La radiactividad natural se debe a elementos que emiten radiaciones espontáneamente, como es el caso del uranio, el torio, el radón, etc.

El generar un diseño óptimo para el acomodo de los combustibles produce una reducción de los costos del ciclo de combustible debido al mejor quemado del mismo. En la recarga de combustible intervienen distintas actividades, entre las que destacan las siguientes:

- Diseño radial de combustible
- Diseño axial de combustible
- Diseño de la recarga de combustible

- Diseño de patrones de barras de control
- Estudio de multiciclos

Cada una de las etapas anteriores tiene su propia complejidad y se encuentran estrechamente relacionadas, el presente trabajo tiene como finalidad estudiar el diseño radial en forma independiente.

Se han realizado estudios previos que han dado buenas aproximaciones a la solución del problema [6] y han aportado información importante acerca del mismo. Los detalles del planteamiento del problema así como la forma de resolverlo se analizarán con mayor profundidad en las siguientes secciones.



Figura 1.8. Central Nucleoeléctrica de Laguna Verde.

Existen distintos tipos de reactores nucleares, cada uno con sus propias características, como ya se ha mencionado, el presente estudio es para reactores BWR, debido a que son el tipo de reactores que se emplean en la Central

Nucleoeléctrica de Laguna Verde (CNLV) la cual se encuentra ubicada sobre la costa del Golfo de México en el Km. 42.5 de la carretera federal Cardel-Nautla, en la localidad denominada Punta Limón municipio de Alto Lucero, Estado de Veracruz, cuenta con un área de 370 Ha. Geográficamente situada a 60 km. al Noreste de la ciudad de Xalapa, 70 km. al Noroeste del Puerto de Veracruz y a 290 km. al Noreste de la Ciudad de México.

La CNLV está compuesta por dos unidades independientes destinadas a la producción de energía eléctrica, cada una con una capacidad de 682.44 MWe (Mega Watts eléctricos). Ver Figura 1.8.

1.2 PLANTEAMIENTO DEL PROBLEMA

El combustible utilizado en los reactores en la CNLV es el U^{235} , dicho material se encuentra en forma de pastillas cilíndricas (Figura 1.9) de aproximadamente 1 cm de diámetro y 1 cm de largo. Dichas pastillas se colocan en el interior de barras (*fuel rods*), cuya longitud es de 3.81 metros (Figura 1.10).

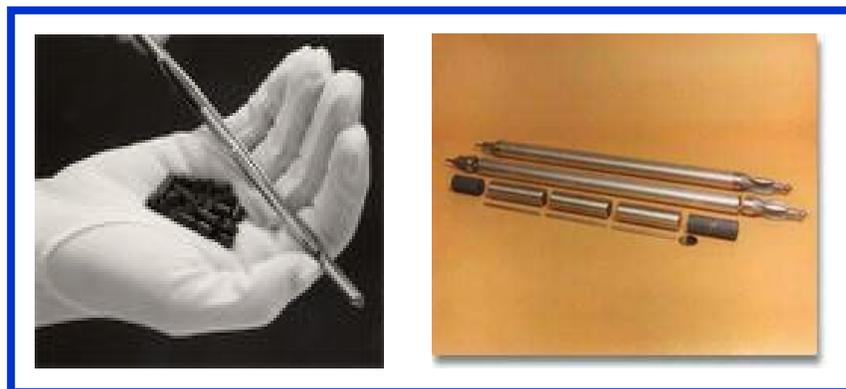


Figura 1.9. Pastillas de combustible.

Los ensambles combustibles están compuestos por un conjunto de estas barras, las cuales se colocan en arreglos de 8x8, 9x9 ó 10x10 que dan forma a un ensamble para reactores tipo BWR (Ver Figura 1.11).

En un BWR moderno cada ensamble consta de entre 74 y 100 barras de combustible. En concreto el número de ensambles combustibles en un reactor depende de la energía a generar, el tamaño del núcleo y otras características más.

En la CNLV, cada núcleo está compuesto por 444 ensambles combustibles, cada ensamble pesa alrededor de 178 Kg., sumando un total de aproximadamente 81 toneladas de uranio en el núcleo de cada reactor.

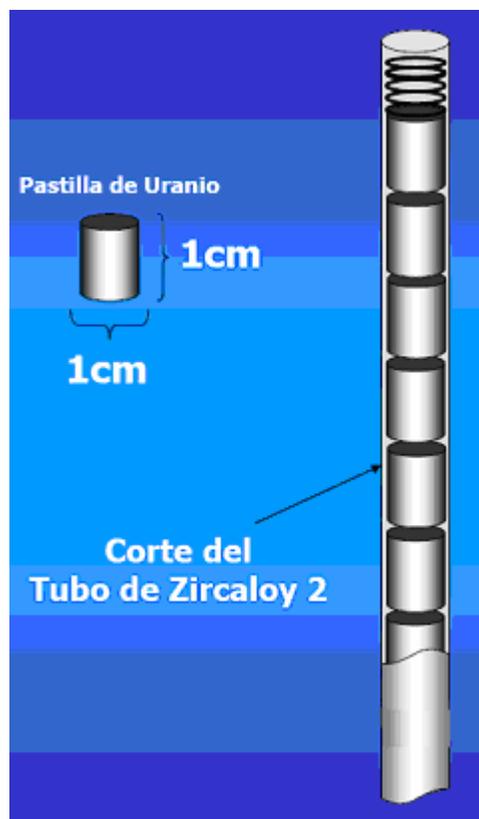


Figura 1.10. Varilla en la que se colocan combustible.

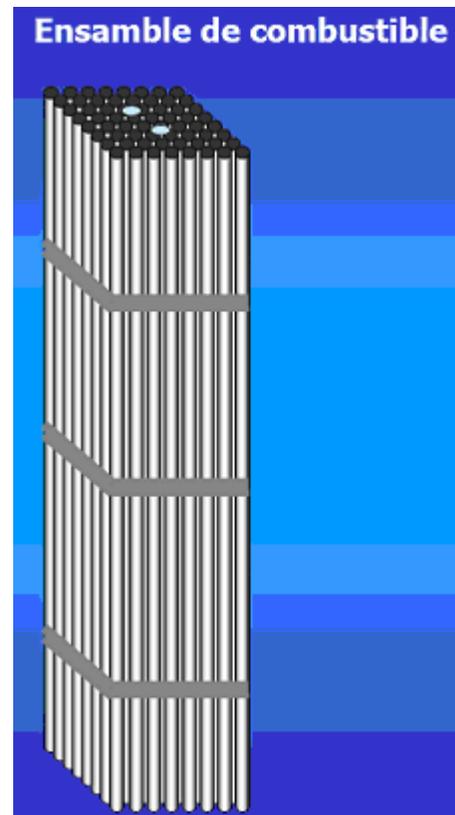


Figura 1.11. Ensamble de las pastillas de combustible.

Los ensambles combustibles que se encuentran en el núcleo del reactor pueden estar en distintas proporciones, así como algunos pueden estar mezclados con otro compuesto llamado gadolinia (Gd_2O_3), en la Figura 1.12 se observa otra imagen de la estructura de los ensambles combustibles. La gadolinia está compuesta de dos átomos de gadolinio y tres átomos de oxígeno. El gadolinio absorbe neutrones y compensa el exceso de reactividad durante un ciclo de operación.

El ciclo de operación es el intervalo de tiempo que puede fluctuar entre 12 y 24 meses en el cual el combustible permanece dentro del reactor con la finalidad de generar electricidad. Ese tiempo concluye cuando el combustible es incapaz de sostener la reacción en cadena.



Figura 1.12. Estructura de un ensamble de combustible.

Los porcentajes disponibles de Uranio y Gadolinia que se utilizaron en el presente trabajo para la composición de los ensambles combustibles se dan en la Tabla 1.1, donde la “ X ” indica que la composición puede ser utilizada en los ensambles combustibles, la ausencia de ésta indica que la composición no es permitida en el diseño.

Por otro lado, se hará una breve mención acerca del diseño axial, esto para ubicar la parte del ensamble con que se trabajó el diseño radial. Un ensamble de combustible consta de 25 secciones axiales llamadas nodos (el simulador que se utiliza para valorar el diseño considera este número de secciones axiales). Cada sección axial tiene una altura de 15.24 cm.

Tabla 1.1. Porcentajes de Uranio y Gadolinia disponibles.

% de Uranio	Gadolinia al 0%	Gadolinia al 2%	Gadolinia al 3%	Gadolinia al 4%	Gadolinia al 5%	Gadolinia al 6%
2.0	X					
2.4	X					X
2.8	X			X		
3.2	X	X	X	X		
3.6	X	X	X		X	X
3.95	X	X	X	X	X	X
4.4	X	X		X	X	X
4.9	X					X

Si se juntan algunos nodos con las mismas características, tanto en contenido de uranio y/o gadolinia, como en la estructura del ensamble, se forman secciones axiales, con las cuales opera mejor el reactor, el número de zonas puede variar. No obstante, en los últimos años y de acuerdo a diversos estudios sobre el tema, el número de zonas se encuentra entre 6 y 7 (ver Figura 1.13). Estas zonas difieren entre ellas por el número de barras con Uranio enriquecido, así como las distintas mezclas con Gadolinia y por el número de barras vacías o ausentes.

Es importante mencionar que, dependiendo de la zona axial con la que se trabaje, el problema puede variar, debido a que la estructura del ensamble cambia

conforme la altura aumenta. Si se realiza un corte transversal en algún punto del ensamble, a ésta sección transversal se le conoce como celda de combustible, por lo que una celda de combustible será un conjunto de barras que contienen distintas composiciones de uranio y gadolinia. Cabe aclarar que el corte que se realiza abarca un conjunto de nodos con las mismas características.

El presente estudio se enfoca únicamente a las características de la Zona B de la Figura 1.13. Si observamos el ensamble de combustible correspondiente a la zona B se ve como se muestra en la Figura 1.14.

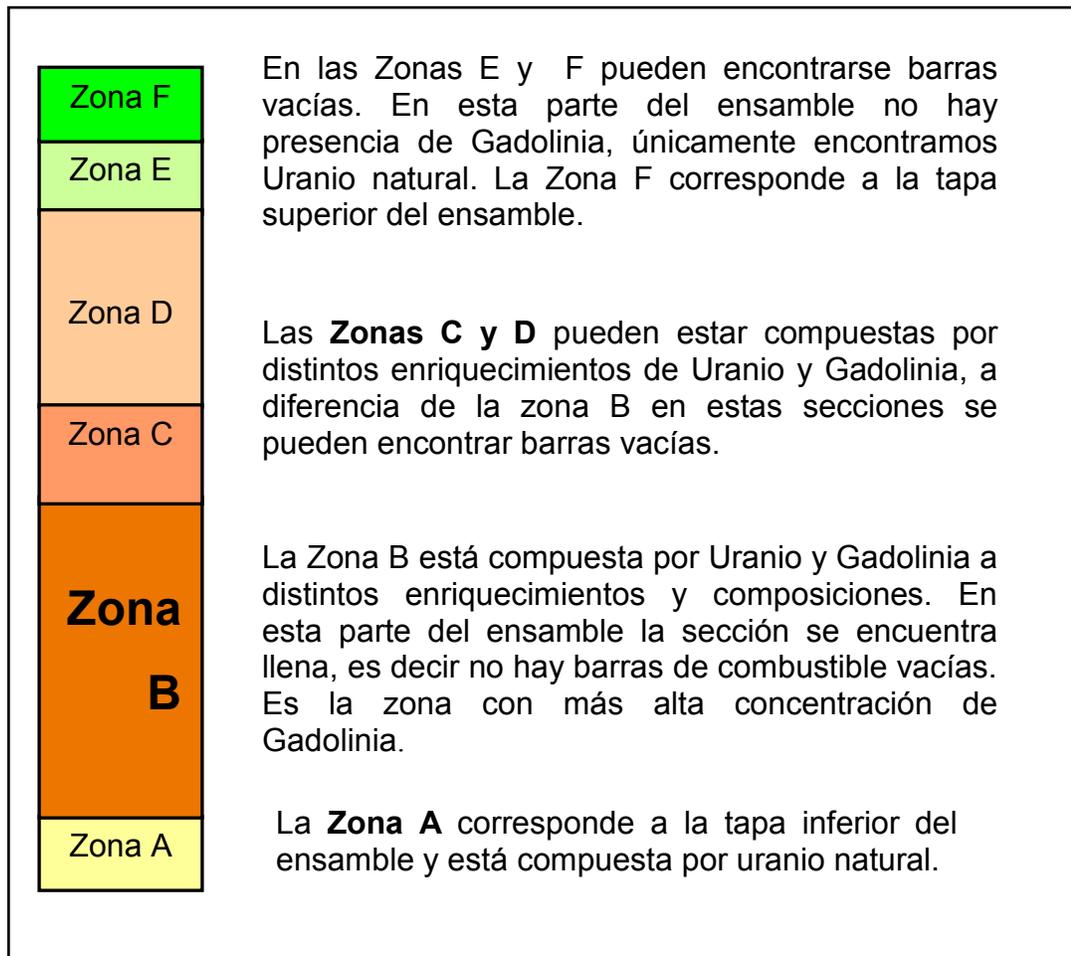


Figura 1.13. Distribución axial del ensamble.

El diseño con el que se trabajó fue para ensambles combustibles del tipo GE12 el cual es un arreglo de 10x10, 8 posiciones están reservadas para 2 canales de agua, estos canales de agua se encuentran en el centro de la celda (ver Figura 1.14). En algunas celdas se pueden tener menos de 92 barras, es decir algunas posiciones no tienen combustible y estos espacios son ocupados por el refrigerante.

Se considera la simetría de $\frac{1}{2}$ o de tipo espejo con respecto a las barras de control, esto significa que el diseño de la parte inferior de la diagonal es el mismo para la parte superior y viceversa (Figura 1.14), por tal motivo se trabajarán solamente con 51 posiciones de un total de 100 que existen en la celda.

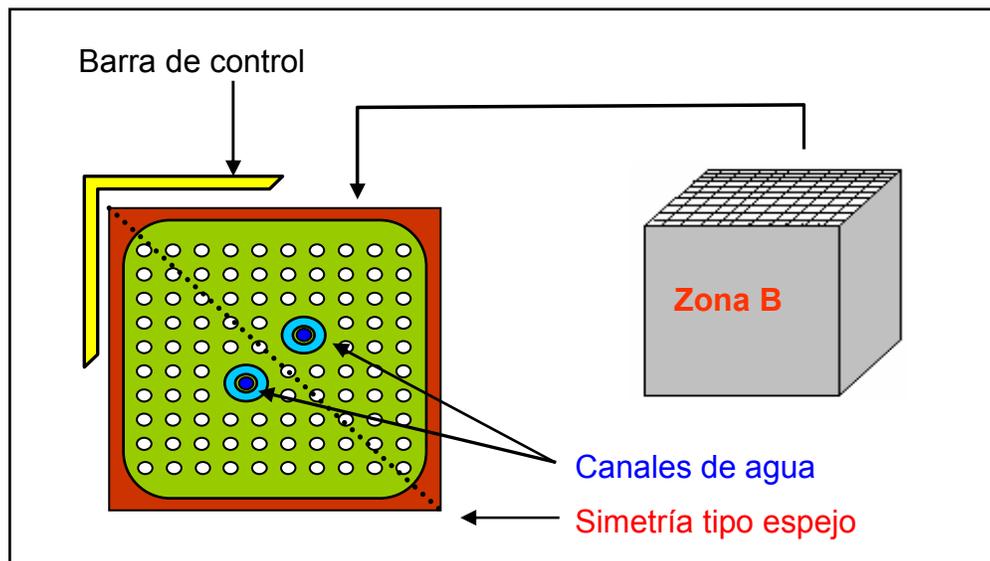


Figura 1.14. Vista superior del ensamblaje de combustible.

Las barras de control, son los elementos que se encargan de mantener la intensidad de la reacción en cadena, dentro de los límites deseados y de

conformidad con la cantidad de energía que se quiera producir. Las barras de control contienen carburo de boro, mismo que tiene la propiedad de capturar neutrones y debido a esto la función de control se establece. Si se desea disminuir la intensidad de la reacción nuclear que ocurre dentro del reactor, basta con insertar las barras de control entre los ensambles combustibles del núcleo, en la medida de la disminución deseada. Las barras se encargan de capturar gran parte de los neutrones libres, reduciéndose la cantidad de fisiones y por lo tanto la energía térmica producida por el reactor. En caso de querer subir la potencia del reactor (aumentar la intensidad de la reacción nuclear) sólo hay que extraer las barras de control, hasta lograr la potencia deseada.

Con lo descrito anteriormente a continuación se exponen las restricciones que se tienen para el diseño de las celdas de combustible.

Restricciones:

1. Los valores de más bajo enriquecimiento deben estar colocados en las esquinas.
2. No debe haber presencia de Gadolinia en las partes laterales del diseño.

Considerando las restricciones anteriores, el problema se reduce a determinar la posición radial de los 51 enriquecimientos tomados de la tabla 1.1 para encontrar su mejor distribución, de modo que se obtenga una celda que cumpla con las restricciones de seguridad y operación del reactor.

Cabe aclarar que las restricciones se han modificado con el transcurso del tiempo, gracias a la experiencia adquirida por los investigadores que han tratado el problema. Una vez ubicadas las posiciones de cada uno de los combustibles se tiene que verificar que el diseño cumpla con los requerimientos establecidos, además se tiene que evaluar su desempeño.

Es importante mencionar que las características del combustible para una recarga, esto es, el promedio de enriquecimiento de Uranio y la concentración promedio de Gadolinia que se coloca dentro de los elementos combustibles para controlar el exceso de reactividad durante el ciclo, deben permanecer constantes.

La calidad de la celda de combustible se mide en base a ciertos parámetros, obtenidos a partir de su desempeño como parte integral de un ensamble de combustible, y éste a su vez como parte de una recarga. Cabe mencionar que no todos los diseños tienen un desempeño adecuado, de hecho, muchos no cumplen con las restricciones impuestas por los parámetros mencionados, por lo que se vuelve un trabajo difícil de realizar.

Con todo lo anterior, se observa que el diseño radial se puede ver como un problema de optimización combinatoria por manejar intercambios entre las posiciones de las pastillas en la celda.

Si en principio no se toma en cuenta ninguna restricción, esto es, las 92 barras pueden tener concentraciones de Uranio y Gadolinia diferentes, entonces se tienen en total 1.2438×10^{142} posibles soluciones, lo que es difícil de solventar con los métodos ordinarios, debido a la complejidad que cada cálculo representa. Se han propuesto muchos métodos numéricos para resolver el problema radial, y se han escrito muchas aplicaciones de software para ayudar en la administración del combustible. Éste es un tema en investigación sin que se haya conseguido todavía una solución óptima, por lo que los operadores utilizan una combinación de técnicas de cálculo y empíricas para tratar el problema.

Capítulo 2

TÉCNICA PATH RELINKING

Una máquina puede hacer el trabajo de cincuenta hombres mediocres; ninguna máquina puede hacer el trabajo de un hombre extraordinario.

E. Hubbard

La existencia de problemas difíciles de resolver debido a su complejidad, han impulsado el desarrollo de técnicas y métodos que los resuelven en forma eficiente encontrando soluciones que bajo ciertas condiciones dan una aproximación a la solución real del problema, estas soluciones no necesariamente son óptimas. A estos métodos se les llamó heurísticos [13]. La implementación de dichos algoritmos y técnicas depende de las características del problema a resolver. Los métodos de tipo combinatorio, en los que se requiere maximizar o minimizar una función sobre un conjunto discreto de soluciones, se consideran dentro de este campo. Entre los más conocidos podemos mencionar los Algoritmos Genéticos [8], GRASP [16], Búsqueda Tabú [14], Búsqueda Dispersa [7], entre otros. Estos métodos se consideran entre los llamados métodos evolutivos los cuales generan, seleccionan, combinan y reemplazan una solución o un conjunto de éstas, los heurísticos que trabajan con un conjunto de soluciones

mantiene constante el tamaño de dicho conjunto a lo largo del proceso de búsqueda. Es importante destacar que una solución se considera buena si ésta se aproxima en gran medida a la solución real del problema, opuesto a lo descrito anteriormente se considera que una solución es mala. En este caso el significado de bueno no se restringe únicamente a la calidad, si no que también se considera la diversidad que la solución aporta al problema.

Para el presente estudio nuestro interés se enfoca en la técnica conocida como Re-encadenamiento de Trayectorias o Path Relinking (PR por sus siglas en inglés), la cuál forma parte de las técnicas heurísticas.

2.1 PATH RELINKING

Path Relinking [10] es un método evolutivo que en los últimos años ha sido aplicado a distintos problemas difíciles de resolver obteniendo buenos resultados. Esta técnica surgió a finales de los ochentas y es en este transcurso de tiempo cuando Laguna y Martí comienzan a publicar textos en los que hablan de la técnica, PR es un método de búsqueda global y como tal su objetivo principal es crear un buen equilibrio entre lo que se denomina intensificación y diversificación. La finalidad de estas dos características son las siguientes:

Intensificación:

- Enfocar la búsqueda de la mejor solución encontrada hasta el momento con el fin de encontrar aún ligeras mejoras.
- Se favorece la aparición de atributos asociados a las mejores soluciones encontradas.

Diversificación:

- Visitar nuevas regiones no exploradas.
- Se modifican las reglas de elección para incorporar atributos que no han sido usados frecuentemente.

PR usa la estructura del problema en forma inteligente para obtener una buena solución. La característica principal de PR es básicamente la generación de trayectorias [8] entre dos soluciones, de tal manera que las soluciones que se encuentran en la trayectoria trazada, contengan atributos de las soluciones que las originaron. Las soluciones iniciales pueden ser generadas con algún método ya sea de búsqueda local o global.

El proceso de generar combinaciones lineales entre dos soluciones puede ser visto como generar caminos dentro y fuera de éstas, donde las soluciones en tales caminos también pueden tomarse como base para generar caminos adicionales. Esta idea es reforzada por el hecho de que los caminos entre soluciones en un espacio vecino generarán nuevas soluciones que comparten un conjunto significativo de atributos. Las características de tales caminos son especificadas por los atributos que le son añadidos o eliminados. Para generar los caminos se requiere tener dos soluciones, una llamada solución inicial y la otra conocida como solución guía. Estas dos soluciones deben ser obtenidas previamente con otro método. La característica de la solución guía es que su evaluación en la función objetivo sea de buena calidad, a diferencia de la solución inicial que puede ser de mala calidad o puede verse como la más alejada de las soluciones en el espacio de búsqueda. Esto con el fin de mezclar la calidad con la diversidad y así explorar en mayor proporción el espacio de búsqueda y evitar caer en máximos o mínimos locales.

Para la generación de trayectorias es necesario seleccionar movimientos que comiencen en la solución inicial, tales cambios pueden progresivamente introducir atributos de ésta y la solución guía (o reducir la distancia entre los atributos de ambas soluciones).

Para la incorporación de atributos a la nueva solución, se tiene que hacer una selección de tal manera que los atributos que se escojan sean los mejores de cada solución. La generación de múltiples caminos surge al considerar la combinación de atributos provistos por una solución guía, donde a las variables que determinan en mayor grado el problema se les da mayor peso y de esta manera se determinan los movimientos de mayor prioridad. Las vecindades para este proceso pueden diferir, de acuerdo a las fases de búsqueda. La importancia de tener variables que determinen fuertemente al problema, ayudan a definir de mejor manera las vecindades sobre las que se trabajará y así reducir el espacio de búsqueda, lo cual es una gran ventaja en tiempo de cálculo computacional. En forma general, para PR se debe considerar que se tiene una colección de soluciones previamente encontradas, ya sea con un método de búsqueda local o global, al cual llamaremos Conjunto Base¹, a dicho conjunto se le aplica la estrategia de generación de caminos.

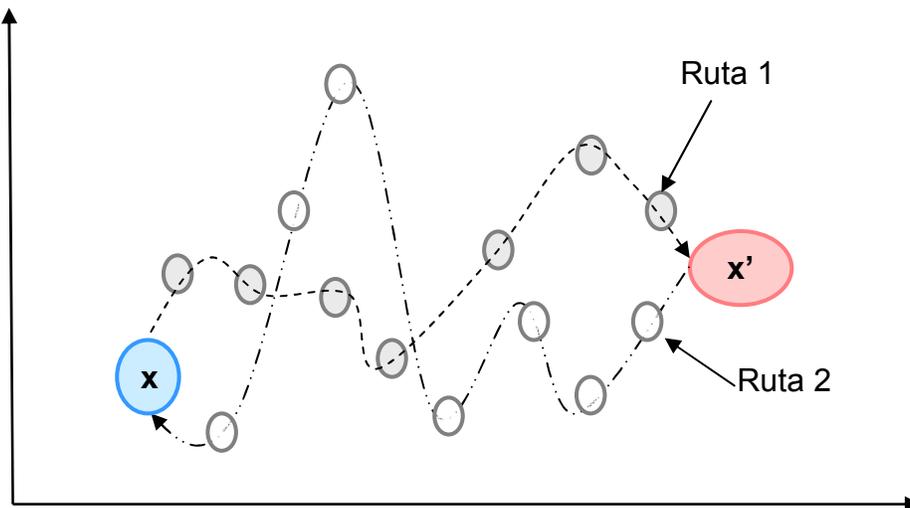


Figura 2.1. Trayectorias generadas de x a x' .

¹ Desde este momento al Conjunto Base se le llama Conjunto Referencia (CR) por haber usado la Búsqueda Dispersa para la generación de las soluciones previas.

Todos los pares de soluciones que se encuentran en el CR son llamados a participar en la fase de re-encadenamiento. Para cada par de soluciones (x, x') donde x representa la solución inicial y x' la solución guía, se pueden establecer dos caminos que las unen, uno que parte de x a x' (ruta 1) y otro que parte de x' a x (ruta 2) como se puede observar en la Figura 2.1. Las dos soluciones obtenidas en dicho proceso son bastante similares y difieren solamente en los atributos que han sido introducidos por un solo movimiento. Por esta razón no es conveniente realizar todas las trayectorias en cada iteración puesto que el tiempo consumido en cada ejecución sería bastante amplio.

Una posibilidad es trazar las trayectorias cada cierto número de iteraciones, o cada que las soluciones cumplan un determinado criterio de calidad. El proceso de re-encadenamiento comienza con dos puntos x y x' simultáneamente, produciendo dos secuencias, la primera de ellas $x = x(1), x(2), \dots, x(n)$ y la segunda $x' = x'(1), x'(2), \dots, x'(m)$. Las elecciones en este caso son dirigidas con el fin de obtener $x(n) = x'(m)$, para los valores finales de n y m como se muestra en la Figura 2.2 para una ruta. Esto se ve como un proceso en donde dos soluciones se intercambian dinámicamente hasta su convergencia a un sólo punto. El diseño se puede generalizar, lo que permite generar caminos fuera de los extremos.

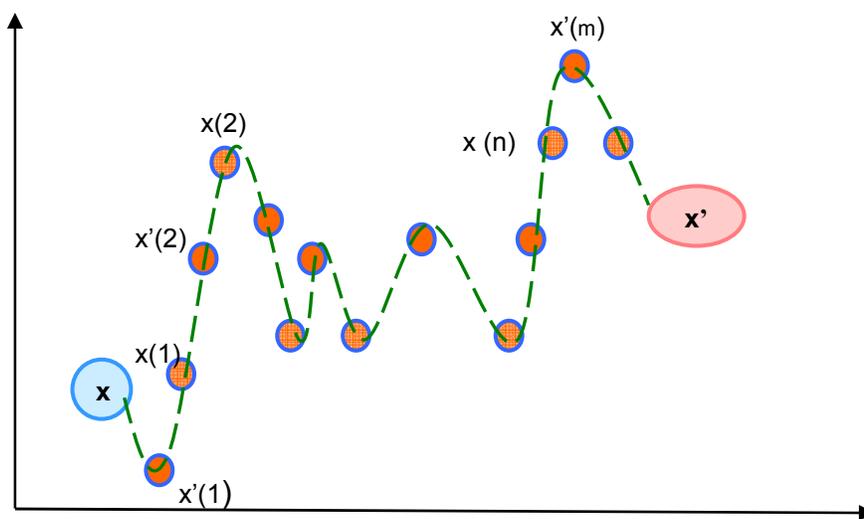


Figura 2.2. Intercambio de atributos entre x y x' .

Es importante aclarar que el proceso no asegura que se encuentre una solución, de hecho ésta es una desventaja de los métodos y técnicas heurísticas, debido a que éstas no tienen un fundamento analítico, por lo que no se puede afirmar que se encuentre una solución al problema.

Cabe aclarar que se pueden trazar más de dos rutas para el mismo par de soluciones como se observa en la Figura 2.3. Debido a que se pueden trazar distintas rutas, se deben establecer criterios bien definidos basados en la calidad de la solución para evitar trazar caminos innecesarios, lo cual ahorrará tiempo de cálculo.

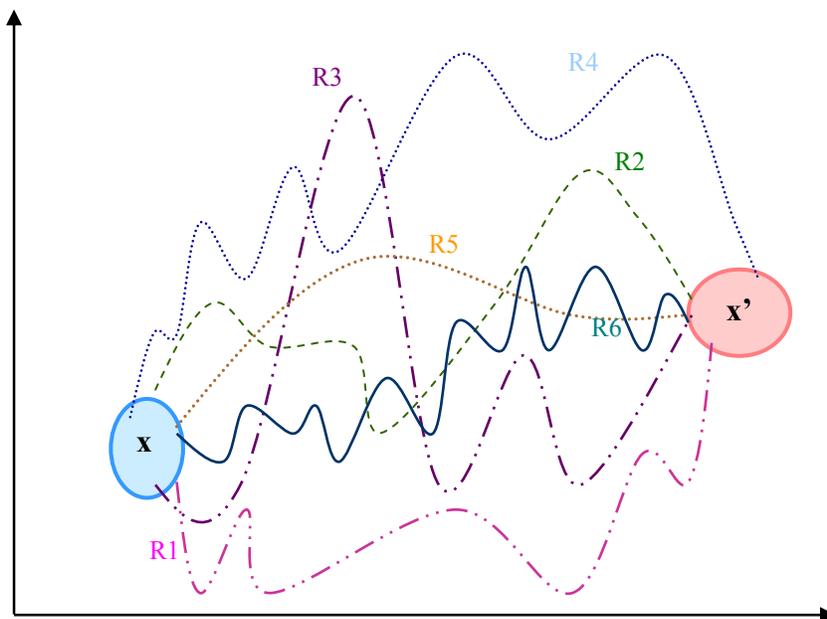


Figura 2.3. Múltiples caminos entre dos soluciones.

El siguiente algoritmo muestra cómo actúan los elementos descritos anteriormente, el cual nos permitirá realizar la implementación de Path Relinking.

2.2.1 Algoritmo de Path Relinking

Paso 1. Determinar los elementos que forman el CR con alguna técnica o método previo.

Paso 2. Aplicar método de mejora al CR.

Paso 3. Trazar trayectorias entre la mejor y la peor solución que se encuentren en el CR.

Paso 4. Actualización del CR con las trayectorias trazadas en el paso 3.

Paso 5. Si después de n iteraciones el CR no presenta mejora entonces se detiene el proceso.

La actualización se basa en mejorar la peor solución que se encuentra en el CR, es decir si una ruta resulta ser mejor que alguna solución de las que se encuentran en el conjunto referencia, ésta se introduce desplazando a la peor solución. Los pasos 2, 3 y 4 se repiten hasta alcanzar el número de iteraciones deseado, o cuando los valores ya no presentan mejora alguna.

2.1.2 Generación de trayectorias

Ya se han mencionado las características de las trayectorias generadas entre la solución inicial y la solución guía [12], ahora se muestra un ejemplo ilustrativo de cómo generarlas. Considérese un problema de permutación de dos soluciones A y B. El objetivo es llegar a B partiendo de A. Los valores para cada solución son los siguientes.

Sea la solución inicial el vector siguiente

Solución A	4	3	10	1	5	2	7	6	9	8
------------	---	---	----	---	---	---	---	---	---	---

La solución guía es el vector mostrado a continuación

Solución B	4	2	7	1	5	10	6	9	3	8
------------	---	---	---	---	---	----	---	---	---	---

Trazar las trayectorias de A con destino en la solución B, es decir, realizar los cambios necesarios, de tal manera que la solución A se vaya pareciendo a la solución B en cada trayectoria trazada, hasta que A sea igual a B.

Solución A	4	3	10	1	5	2	7	6	9	8
Trayectorias	4	2	10	1	5	3	7	6	9	8
trazadas	4	2	7	1	5	3	10	6	9	8
	4	2	7	1	5	10	3	6	9	8
	4	2	7	1	5	10	6	3	9	8
	4	2	7	1	5	10	6	3	9	8
Solución B	4	2	7	1	5	10	6	9	3	8

Como se puede observar las trayectorias consecutivas son bastante similares y sólo difieren en el intercambio de posiciones de dos elementos en cada movimiento, tales movimientos se interpretan como agregar atributos de la solución guía a la solución inicial. El pseudo código que muestra cómo se generan las trayectorias se observa en la Figura 2.4.

Las soluciones iniciales que se usan para aplicar Path Relinking son previamente encontradas con otra metodología, como por ejemplo, Búsqueda Tabú (BT), Búsqueda Dispersa (BD), Algoritmos Genéticos (AG), GRASP entre otros. En este

sentido, cabe señalar que diversos autores ven a Path Relinking como una extensión de la búsqueda dispersa [11], debido a que surgió como parte de las estrategias avanzadas de esta metodología.

BD y PR son métodos basados en poblaciones que operan sobre un conjunto de soluciones y básicamente difieren en la forma en como construyen, manipulan y actualizan el conjunto de soluciones. En particular, para el presente trabajo se empleó la BD para generar el conjunto de soluciones previas, por este motivo se dará a continuación la descripción de este método.

```
L = 0
DO I = 1, N      N es la longitud de los vectores a combinar.
  IF (SOLINICIAL (I) .EQ. SOLGUIA (I)) THEN
    Pasar al siguiente elemento
  ELSE
    SOLGUIA
    Buscar en el vector SOLINICIAL el elemento de
    Cambiar de posición los elementos
    DO J =1, N
      TRAYECTORIAS (L, N) = SOLINICIAL (I)
    END DO
    L = L + 1
  END IF
END DO
NCAMINOS = L - 1      Número de caminos generados
```

Figura 2.4. Seudo código para generar trayectorias.

2.2 BUSQUEDA DISPERSA

La Búsqueda Dispersa (BD) es una técnica eficiente para encontrar buenas soluciones, éstas no necesariamente son óptimas pero sí se aproximan en gran medida a la solución real. La BD no está fundamentada en la aleatorización sobre un conjunto relativamente grande de soluciones, como en algoritmos genéticos, si no en la elección sistemática y estratégica sobre un conjunto pequeño de éstas. Por ejemplo AG trabaja con una población de 100 soluciones mientras que la BD es común que trabaje con únicamente 10 de éstas, a este conjunto de 10 soluciones le llamaremos conjunto referencia (RefSet). Al igual que los AG, la BD se basa en mantener un conjunto de soluciones y realizar combinaciones con éstas.

La BD se basa en el principio de que la información sobre la calidad o el atractivo de un conjunto de reglas, restricciones o soluciones pueden ser utilizados mediante la combinación de éstas. En concreto, dadas dos soluciones, se puede obtener una nueva mediante su combinación de modo que mejore a las que la originaron. Una descripción del método fue publicada por Fred Glover [8], en este libro se determina que la BD realiza una exploración sistemática sobre una serie de buenas soluciones llamadas conjunto referencia, cabe destacar que el significado de buena no se restringe a la calidad de la solución, sino que también se considera la diversidad que ésta aporta al conjunto. La Búsqueda Dispersa consta básicamente de los siguientes elementos.

2.2.1 Elementos de la Búsqueda Dispersa

- I. Un *generador de soluciones dispersas*: El método se basa en generar un conjunto P de soluciones dispersas (alrededor de 100), del que extraeremos un subconjunto pequeño (alrededor de $b = 10$) con el que realizaremos combinaciones e identificaremos como Conjunto Referencia.
- II. *Conjunto referencia* (RefSet). Extraído del conjunto de soluciones dispersas según el criterio de contener soluciones de calidad y diferentes entre sí. El tamaño del RefSet se denota con b , el cual ésta compuesto por las $b/2$ mejores soluciones y las otras $b/2$ se extraen por el criterio de máxima distancia, para lo cual se tiene que definir una métrica.
- III. Un *método de combinación*. Se encarga de combinar todas las soluciones del conjunto. Para esto, se consideran subconjuntos de 2 o más elementos del RefSet y se combinan mediante una rutina diseñada a tal efecto.
- IV. *Actualización*. La solución o soluciones que se obtienen de las combinaciones pueden incorporarse al RefSet si éstas mejoran alguna de las soluciones que se encuentran en este conjunto. La actualización puede ser de manera dinámica o estática, el usuario debe seleccionar cual usará de acuerdo a las características del problema.

En adelante se describirá con más detalle estos elementos para especificar con mayor claridad las características más importantes.

2.2.1.1 Generador de soluciones dispersas

Generar soluciones dispersas no es lo mismo que generar soluciones aleatorias. La meta de la diversificación es producir soluciones en la cual una difiere de la otra

en caminos significantes, puede verse como lo opuesto a la conglomeración. En contraste, producir soluciones en forma aleatoria puede que difieran una de la otra en algún punto, tales caminos pueden no tener relación alguna. En la aleatoriedad la presencia de cualquier solución no es predecible.

En la Búsqueda Dispersa se consideran dos tipos de generadores diversificadores, uno para problemas que pueden ser formulados como la optimización de una función en la que las variables pueden tomar valores de ceros y unos, y otra en la que pueden ser formulados como una permutación de elementos.

En el presente trabajo se hará uso del segundo tipo de generador, debido a que en el problema a tratar se manejan permutaciones, por este motivo se dará una explicación amplia de éste.

Generador diversificador para problemas de permutaciones.

Dada una permutación P compuesta por i elementos, se usa a P como semilla para generar más permutaciones a partir de ésta. Sin pérdida de generalidad, para este ejemplo los elementos de P aparecen en orden consecutivo, es decir $P = (1, 2, 3, \dots, n)$. Se define la subsecuencia $P(h:s)$, donde h es un valor que se encuentra preferiblemente en el rango de 1 a $n/2$, la cual se puede considerar como una distancia, y s es un entero positivo entre 1 y h .

De esta manera se define $P(h:s)$ como:

$$P(h:s) = (s, s+h, s+2h, \dots, s+rh) \quad (2.1)$$

Donde r es el entero no negativo más grande tal que $s + rh \leq n$. Entonces se define la permutación $P(h)$, para $h \leq n$, para ser $P(h) = (P(h:h), P(h:h-1), \dots, P(h:1))$.

Para mayor claridad se muestra un ejemplo. Sea P la permutación de la cual se desea generar un Conjunto Disperso (CD).

$$P = (1,2,3,4,5,6,7,8,9,10)$$

Se tiene que el valor de $n = 10$ (n es el número de elementos que forman a P) por lo que se toma una h que se encuentre en el intervalo de 1 a 5. Si se escoge $h=5$. Se genera el CD de permutaciones.

$$P(5) = \{P(5:5), P(5:4), P(5:3), P(5:2), P(5:1)\}$$

$$P(5:5) = (5,10), \quad P(5:4) = (4,9), \quad P(5:3) = (3,8), \quad P(5:2) = (2,7), \quad P(5:1) = (1,6)$$

$$P(5) = \{5,10,4,9,3,8,2,7,1,6\}$$

De la misma manera si se escoge $h = 4, 3, 2, 1$ se obtiene lo siguiente.

$$P(4) = \{P(4:4), P(4:3), P(4:2), P(4:1)\} = \{4,8,3,7,2,6,10,1,5,9\}$$

$$P(3) = \{P(3:3), P(3:2), P(3:1)\} = \{3,6,9,2,5,8,1,4,7,10\}$$

$$P(2) = \{P(2:2), P(2:1)\} = \{2,4,6,8,10,1,3,5,7,9\}$$

$$P(1) = \{P(1:1)\} = \{1,2,3,4,5,6,7,8,9,10\}$$

Finalmente P queda de la siguiente manera.

	P									
$P(5)$	5	10	4	9	3	8	2	7	1	6
$P(4)$	4	8	3	7	2	6	10	1	5	9
$P(3)$	3	6	9	2	5	8	1	4	7	10
$P(2)$	2	4	6	8	10	1	3	5	7	9
$P(1)$	1	2	3	4	5	6	7	8	9	10

Es importante hacer notar que para valores grandes de n el proceso comienza a ser extenso.

Para complementar el CD, se genera el inverso el cual se denota como $P^*(h)$. En el presente trabajo, el inverso se refiere al cambio de posición de los elementos de una permutación, es decir el último elemento se pasa a la primera posición, el penúltimo a la segunda y así sucesivamente, como se ve en el siguiente ejemplo. Donde A^* denota la inversa de A .

A	5	10	4	9	3	8	2		A^*	2	8	3	9	4	10	5
Posición	1	2	3	4	5	6	7		posición	1	2	3	4	5	6	7

Con lo anterior P^* queda de la siguiente forma:

	P^*									
$P^*(5)$	6	1	7	2	8	3	9	4	10	5
$P^*(4)$	9	5	1	10	6	2	7	3	8	4
$P^*(3)$	10	7	4	1	8	5	2	9	6	3
$P^*(2)$	9	7	5	3	1	10	8	6	4	2
$P^*(1)$	10	9	8	7	6	5	4	3	2	1

De esta manera se obtiene el CD, el cual está compuesto por los elementos de P y P^* .

Conjunto Disperso de P

5	10	4	9	3	8	2	7	1	6
4	8	3	7	2	6	10	1	5	9
3	6	9	2	5	8	1	4	7	10
2	4	6	8	10	1	3	5	7	9
1	2	3	4	5	6	7	8	9	10
6	1	7	2	8	3	9	4	10	5
9	5	1	10	6	2	7	3	8	4
10	7	4	1	8	5	2	9	6	3
9	7	5	3	1	10	8	6	4	2
10	9	8	7	6	5	4	3	2	1

Cabe aclarar que cuando los valores de h son pequeños hay preferencia de $P^*(h)$ sobre $P(h)$. Por ejemplo cuando $h = 1$, $P(h)=P$ y $P^*(h)$ es el inverso de P , o visto de otra manera $P(n) = P^*(1)$. El código para generar un conjunto disperso se anexará en el disco compacto adjunto al presente trabajo.

2.2.1.2 Conjunto Referencia

Una vez obtenido el conjunto de soluciones dispersas, se extrae un subconjunto de 10 elementos (tamaño del conjunto referencia) el cual debe cumplir con los criterios de calidad y diversidad, estos elementos forman el conjunto referencia (*RefSet*). Las soluciones en el *RefSet* están ordenadas de mejor a peor respecto de su calidad.

El *RefSet* se divide en 2 subconjuntos [13] a los cuales se identificará como *RefSet1* y *RefSet2*. En el primero se encuentran los elementos de calidad, mientras que en el segundo están los elementos más alejados al *RefSet1* de acuerdo a la métrica empleada en cada problema. La integración de éstos se hace de la siguiente forma:

RefSet1. Soluciones de calidad

Las soluciones que se encuentran en el CD son evaluadas con la función objetivo. Para la construcción del *RefSet1* se seleccionan las 5 mejores soluciones que se encuentran en el CD, estas soluciones integran el *RefSet1*.

RefSet2. Soluciones más distantes al *RefSet1*

Se hace uso del criterio de máxima distancia para obtener este subconjunto. Para esto se define una métrica de la siguiente manera, se calcula la distancia mínima de cada elemento \hat{x} que se encuentra en el CD a cada uno de los elementos que se encuentran en el *RefSet1*. Esto es

$$d(\hat{x}, \text{RefSet1}) = \min \{ d(\hat{x}, y_i) \mid y_i \in \text{RefSet1}, \hat{x} \in \text{CD} \} \quad (2.2)$$

Donde $d(\hat{x}, y_i)$ es la distancia Euclídeana entre \hat{x} y y_i . Una vez calculadas las distancias, se selecciona de este conjunto los elementos que tengan las máximas distancias, estos elementos forman el *RefSet2*. El esquema se puede ver en Figura 2.5.

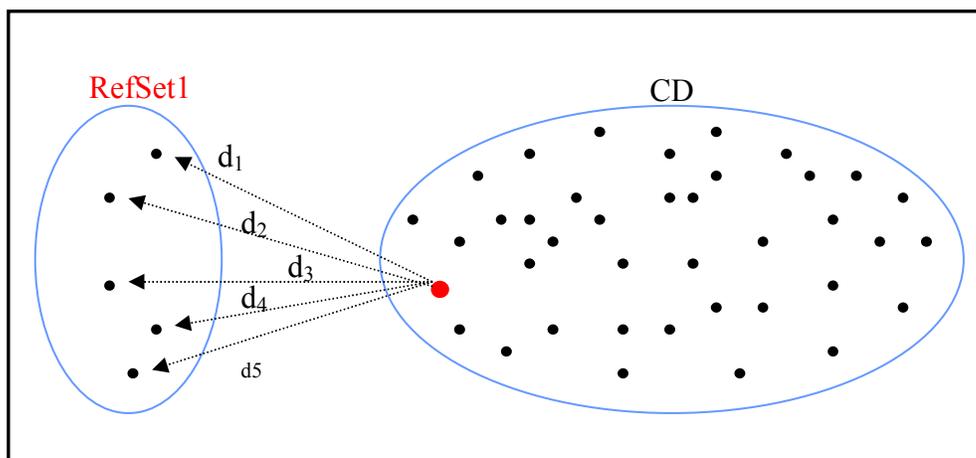


Figura 2.5. Distancias del *RefSet1* al CD.

De esta manera se forma el Conjunto Referencia con los elemento del *RefSet1* mas los elementos del *RefSet2*.

2.2.1.3 Combinación Convexa

Ésta es la parte esencial del método, Glover introduce la combinación ponderada como un nuevo mecanismo para generar nuevas soluciones [12]. El objetivo de realizar combinaciones lineales lo podemos definir en los siguientes puntos:

- Combinar dos o más soluciones del conjunto referencia.
- Generar soluciones en la línea que une las dos soluciones.
- Seleccionar pesos apropiados al realizar las combinaciones.
- Realizar combinaciones convexas y no convexas de las soluciones.
- La distribución de las soluciones debe ser dispersa.

La combinación de dos soluciones en la que una se caracteriza por su calidad y la otra por su diversidad nos permite mejorar las soluciones en cada etapa del proceso y de esta manera acercarnos a una solución óptima.

Como se puede observar en la Figura 2.6, se realizan combinaciones convexas y no convexas entre los puntos A, B y C. Con los nodos resultantes se pueden realizar también combinaciones lineales. De esta manera los nodos que tienen el número 1 son resultado de combinaciones convexas, mientras que los nodos con el número 2 y 3 son resultado de la no convexidad.

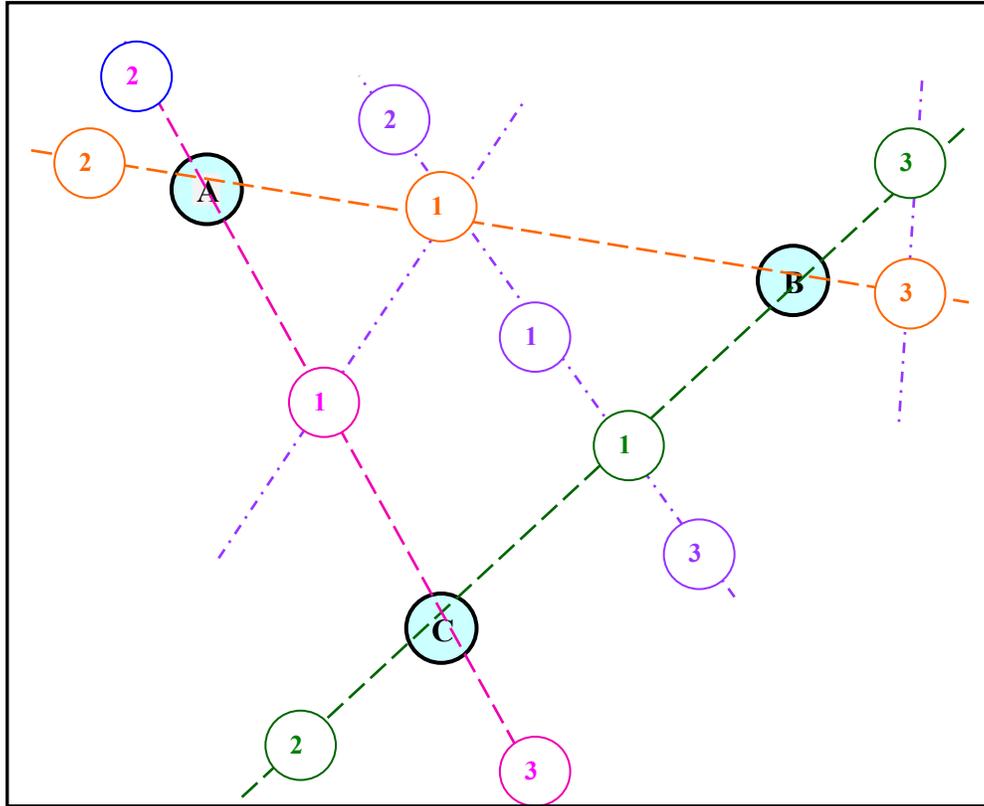


Figura 2.6. Combinación Convexa.

Hablar de combinaciones lineales cuando tenemos variables enteras o binarias no representan ningún problema, pero hablar de combinaciones lineales entre dos permutaciones no tiene sentido, por lo que para cada problema de este tipo se deben diseñar métodos específicos de combinación.

En el presente trabajo se manejan permutaciones, por lo que se tuvo que desarrollar un algoritmo particular para este problema considerando las características del mismo para poder realizar las combinaciones entre dos permutaciones.

Con todo lo anterior decimos que las combinaciones lineales entre una solución de buena calidad y una que se encuentra dispersa en el universo de estudio, da como resultado una solución de buena calidad, además de asegurar que no se cae en

algún mínimo o máximo local. Un esquema de esto se puede observar en la Figura 2.7.

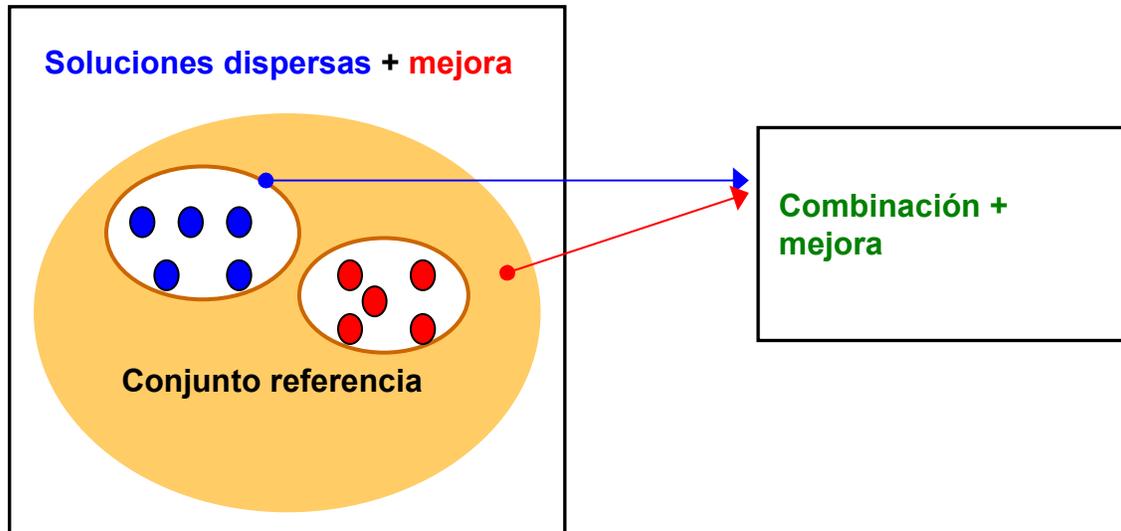


Figura 2.7. Esquema de Búsqueda Dispersa.

2.2.1.4 Actualización

Cabe aclarar que en el presente trabajo la actualización del *RefSet* se realizó en forma dinámica, lo que significa que se podrán sustituir las soluciones que forman éste conjunto si al realizar una combinación se encuentra una solución que mejore alguna de las que están el *RefSet*. El tamaño del conjunto referencia es constante, pero el valor de las soluciones va mejorando a lo largo del proceso de búsqueda. A esto se le denomina actualización del conjunto referencia.

En resumen, se considera a la BD como un método evolutivo que todavía se encuentra en desarrollo. En la actualidad no existe un esquema único para aplicar esta técnica, sin embargo se han obtenido buenos resultados al aplicar la técnica

en la solución de problemas. Como se mencionó con anterioridad, en el presente estudio se trabajó con Path Relinking, considerando a Búsqueda Dispersa como técnica previa para encontrar el conjunto de soluciones previas. En el siguiente Capítulo se realiza la integración de ambas técnicas para la implementación al diseño radial de celdas de combustible nuclear.

Capítulo 3

IMPLEMENTACIÓN DE LA TÉCNICA PATH RELINKING AL PROBLEMA DEL DISEÑO RADIAL

Lo opuesto a una formulación correcta es una formulación falsa. Pero lo opuesto de una verdad profunda puede ser muy bien otra verdad profunda.

N. H. Bohr

El objetivo del trabajo es encontrar un diseño óptimo de una celda de combustible respetando ciertas restricciones. Se tienen 51 posiciones en las cuales se tiene que determinar el porcentaje de enriquecimiento de U^{235} que se colocará en cada barra, así como a cuáles de éstas se adicionará gadolinia. La distribución obtenida en la celda debe satisfacer ciertas restricciones.

En capítulos previos se analizó la problemática del diseño radial y la técnica Path Relinking en forma independiente. En el presente capítulo se desarrollará la implementación de la técnica al problema en cuestión, para esto se desarrolló el

sistema ohtli-RT (ohtli es una palabra en náhuatl que significa camino y RT es Re-encadenamiento de Trayectorias). De esta manera se describen las dificultades que se presentaron a lo largo del proceso. Primero se muestra el algoritmo considerado y después se desarrolla la implementación de cada una de las etapas.

3.1 ALGORITMO DE PATH RELINKING EN EL DISEÑO RADIAL

A continuación se presenta el algoritmo empleado en la implementación de la técnica Path Relinking en el diseño de celdas de combustible. En dicho algoritmo se emplea la Búsqueda Dispersa para generar el conjunto referencia, a partir del cual se trazarán las trayectorias y también se emplea como método de mejora del mismo conjunto.

Algoritmo:

- Paso 1.* Generar una solución semilla en forma aleatoria que cumpla con los requisitos del diseño.
- Paso 2.* Determinar los elementos que forman el Conjunto Referencia (CR) con la Búsqueda Dispersa, tomando como base la semilla generada en el paso 1.
- Paso 3.* Aplicar la Búsqueda Dispersa para mejorar las soluciones que se encuentran en el CR.
- Paso 4.* Trazar trayectorias entre la mejor y la peor solución que se encuentren en el CR.
- Paso 5.* Actualización del CR cuando se encuentre alguna trayectoria que mejore alguna de las soluciones que se encuentren en este conjunto.

3.2 IMPLEMENTACIÓN

En adelante se presentará cada una de las etapas de la implementación del algoritmo, la metodología se aplicó a una celda de 10x10 barras de combustible con distintos enriquecimientos, diferentes concentraciones de Gadolinia y dos canales de agua en simetría diagonal. Para cada etapa se presentan sus respectivas características, las cuales en conjunto son parte integral del sistema ohtli-RT, el cual se codificó en el lenguaje de programación Fortran, y es un programa en el que se automatizó desde la entrada de los datos, hasta la obtención final del diseño radial.

3.2.1 Generación de la semilla

Se tomó como semilla un vector de 51 posiciones, en éste se encuentran los enriquecimientos propuestos para el diseño. Para la generación de la semilla se utilizaron 8 enriquecimientos distintos (ver Tabla 1.1 del Capítulo 1).

El algoritmo empleado para la generación de la semilla es el siguiente:

Paso 1. Solicitar datos de entrada: el enriquecimiento promedio de la celda y el número de barras de Gadolinia con sus respectivos porcentajes.

Paso 2. Tomar en forma aleatoria 51 elementos de los enriquecimientos disponibles, los cuales en conjunto deben cumplir con el promedio requerido y el número de barras de Gadolinia, con un margen de error de ± 0.005 en el enriquecimiento. En caso contrario regresar al paso 1.

Paso 3. Verificar que se cumplan las restricciones del diseño. En caso de no cumplirse realizar una permutación de los elementos o regresar al paso 1.

En la Figura 3.1 se muestra un vector de 51 posiciones en el cual se almacenarán los enriquecimientos del diseño, así como el porcentaje de gadolinia; esta estructura es la que se empleará a lo largo de la implementación del problema.

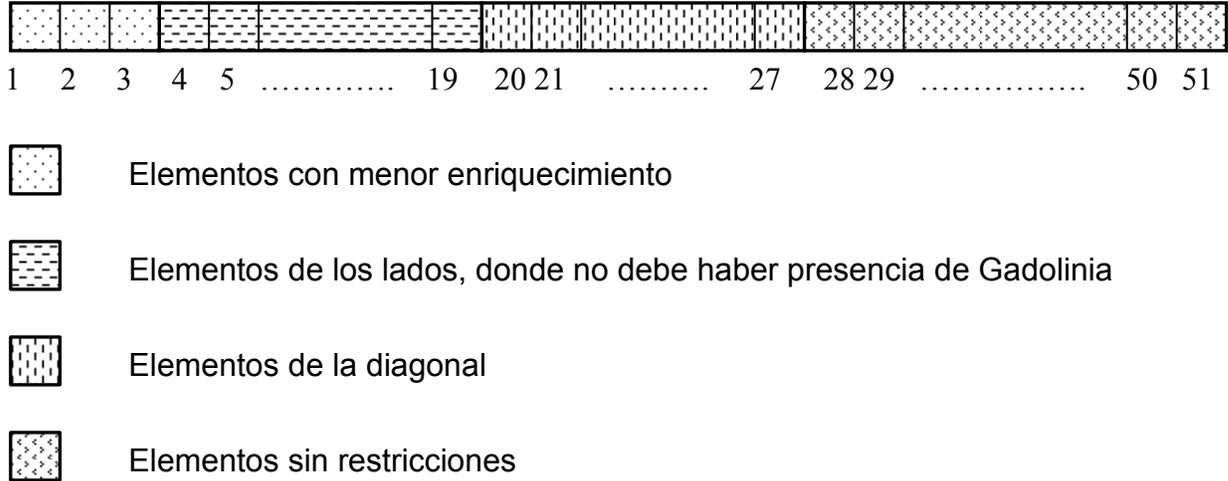


Figura 3.1. Distribución de los elementos.

3.2.2 Conjunto Referencia

Para generar el conjunto referencia se hace uso de la Búsqueda Dispersa. Se empleará el mismo mecanismo descrito en el capítulo anterior para generar el conjunto referencia. A continuación se describirá la manera en como se implemento al problema que nos ocupa.

3.2.2.1 Generador de soluciones dispersas

Una vez obtenida la solución semilla se procede a generar el conjunto disperso. En este caso se tienen que realizar permutaciones con 48 posiciones, puesto que las primeras 3 corresponden a los elementos de las esquinas y éstas permanecerán fijas a lo largo del proceso. El método indica que se harán tantas permutaciones como elementos se tengan en el vector a combinar. Como resultado se obtienen 48 permutaciones obtenidas con ésta metodología, a este conjunto de elementos se le denotará como Q .

El conjunto de soluciones dispersas estará compuesto por los elementos de Q mas los elementos de Q^* . Donde Q^* representa el inverso de Q (se entiende como inverso al cambio de posición de los elementos, como se describió en el Capítulo 2), por lo que Q^* está compuesta por 48 elementos.

En total obtendremos 96 permutaciones, hay que considerar que hay dos soluciones que se repiten por la forma en como se está generando el conjunto. El primer renglón de Q^* corresponde al ultimo renglón de Q , por este motivo se descarta el primer renglón de Q^* , por lo que nos quedan únicamente 95 permutaciones en el conjunto.

Con este proceso se obtiene finalmente el conjunto de soluciones dispersas derivadas de una solución semilla.

3.2.2.2 Conjunto referencia

El *RefSet* está integrado por las 5 permutaciones que representan las soluciones de mejor calidad y las 5 más alejadas de éstas que se encuentran en el conjunto disperso.

Antes de continuar es conveniente describir y definir algunos términos, debido a que son fundamentales para continuar con el proceso.

La distribución de potencia en el núcleo del reactor no es uniforme lo cual trae como consecuencia una distribución de potencia característica que requiere ser verificada en cada momento de la operación de la central, a esta distribución se le conoce como Factor de Pico de Potencia (FPP). El FPP es un indicador de la distribución de potencia en el núcleo del reactor, existen varias formas de medir el FPP, para el presente trabajo se considera el FPP de la barra que genera mayor potencia en una celda.

Otro parámetro importante es el factor de multiplicación infinita (k_{∞}) que se define como:

$$k_{\infty} = \frac{\# \text{ de neutrones en una generación dada}}{\# \text{ de neutrones en una generación previa}}$$

Si $k_{\infty} > 1$: La población de neutrones aumenta; en este caso, el sistema (reactor) se conoce como supercrítico.

Si $k_{\infty} = 1$: La población de neutrones permanece constante, en este caso el sistema (reactor) se conoce como crítico.

Si $k_{\infty} < 1$: La población de neutrones disminuye, en este caso, el sistema (reactor) se conoce como subcrítico.

También se puede definir de la siguiente manera:

$$k_{\infty} = \frac{\text{rapidez de producción de neutrones}}{\text{rapidez de pérdida de neutrones}}$$

En la valoración de los diseños que se generan, estos dos términos son fundamentales junto con otras características no menos importantes pero que no se consideraron en el presente estudio, en conjunto se utilizan para verificar el funcionamiento del reactor y son utilizadas por el sistema de información para elaborar cálculos más complejos.

Regresando, el *RefSet* se generará como sigue. Para determinar las primeras se deben evaluar los elementos del conjunto disperso con la siguiente función objetivo.

$$\text{Minimizar} \quad f(FPP, k_{\infty}) = w_1 * FPP + G(k_{\infty}) \quad (3.1)$$

con

$$G(k_{\infty}) = \begin{cases} w_2 |k_{\infty}| & \text{si } |k_{\infty} - 1.1| > 0.1 \\ 0 & \text{en otro caso} \end{cases}$$

donde

FPP : Factor de Pico de Potencia

k_{∞} : Factor de multiplicación infinita

w_1, w_2 : Factores de peso

Los factores de peso ponderan la importancia entre los términos de la función. Es importante mencionar que para que la celda se considere de buena calidad los parámetros FPP y la k_{∞} deben cumplir con las restricciones siguientes:

- El valor del FPP debe ser menor a 1.45
- La k_{∞} debe estar en el siguiente intervalo $1.1 < k_{\infty} < 1.3$

Cabe señalar que los parámetros anteriores son una parte de los requerimientos solicitados para una celda, en una segunda etapa se verifican otros como la energía, los límites térmicos, el margen de apagado en frío, así como el exceso de reactividad en caliente, sin embargo, éstos no están considerados en el presente trabajo, por lo que no se definirán.

Para obtener los parámetros involucrados en la función objetivo se utilizó el simulador Helios[18], el cual es un código de transporte de simulación neutrónica de celdas de combustible, que ha sido validado para el cálculo de bancos nucleares para BWRs. Helios utiliza como preprocesador de datos de entrada al código Aurora[17] y como posprocesador el código Zenith[19].

Para evaluar neutrónicamente diferentes distribuciones radiales de enriquecimiento y gadolinia en la celda de combustible, los datos requeridos por Helios, son el enriquecimiento de uranio y el contenido de gadolinia en las distintas barras de combustible, así como sus posiciones en el diseño radial, entre otras características más. El vector que contiene las barras de la celda se distribuyen como se muestra en la Figura 3.2, esto ayuda debido a que el código Helios lee la información de esta forma.

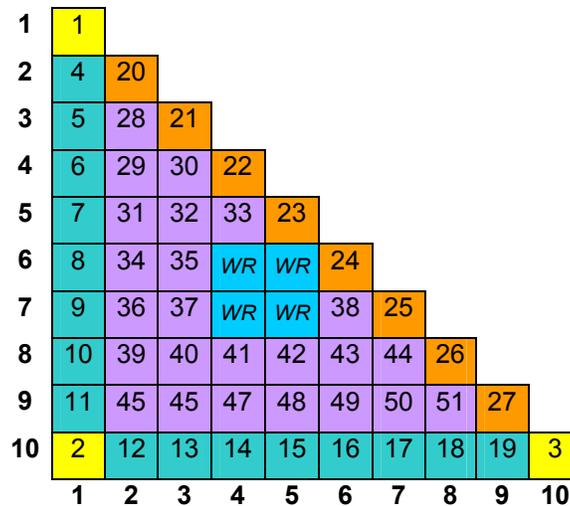


Figura 3.2. Elementos de una celda en forma triangular.

Para evaluar un elemento del conjunto disperso, primero se tiene que generar un archivo de cierto nombre con extensión “*.inp” con los datos del vector, después este archivo es evaluado con el código Aurora, el cual se encarga, entre otras cosas, de verificar que las definiciones correspondan a los enriquecimientos que se encuentran en el diseño así como a sus respectivas posiciones. Cuando Aurora termina, arroja dos archivos de salida uno con extensión “*.out” en el que se puede verificar si la celda tiene errores y de qué tipo son éstos, y el archivo con extensión “*.hrf”, el cual es utilizado en la siguiente fase.

Una vez que el código Aurora termina en forma exitosa se procede a evaluar la celda en el código Helios. Al terminar Helios regresa nuevamente el archivo *.hrf con los cálculos del desempeño de la celda, también arroja un nuevo archivo con extensión “*.out” en el que se puede verificar si Helios encontró problemas en el diseño.

Finalmente se ejecuta el código Zenith, el cual lee los datos que necesita para su ejecución de un archivo de entrada, además utiliza el archivo *.hrf. Zenith lo que hace básicamente es traducir la información de los datos que arroja Helios (en

formato binario) para que el usuario pueda entenderlos. Zenith produce un archivo con extensión “*.out”, en el cual se encuentra entre otros datos, los resultados de la valoración de la celda entre ellos el valor del FPP y la k_{∞} que son los que interesan para poder evaluar la función objetivo.

La forma en la que interactúan los códigos mencionados se muestra en la Figura 3.3. Al final la información requerida se obtiene del último archivo de salida generado en este proceso.

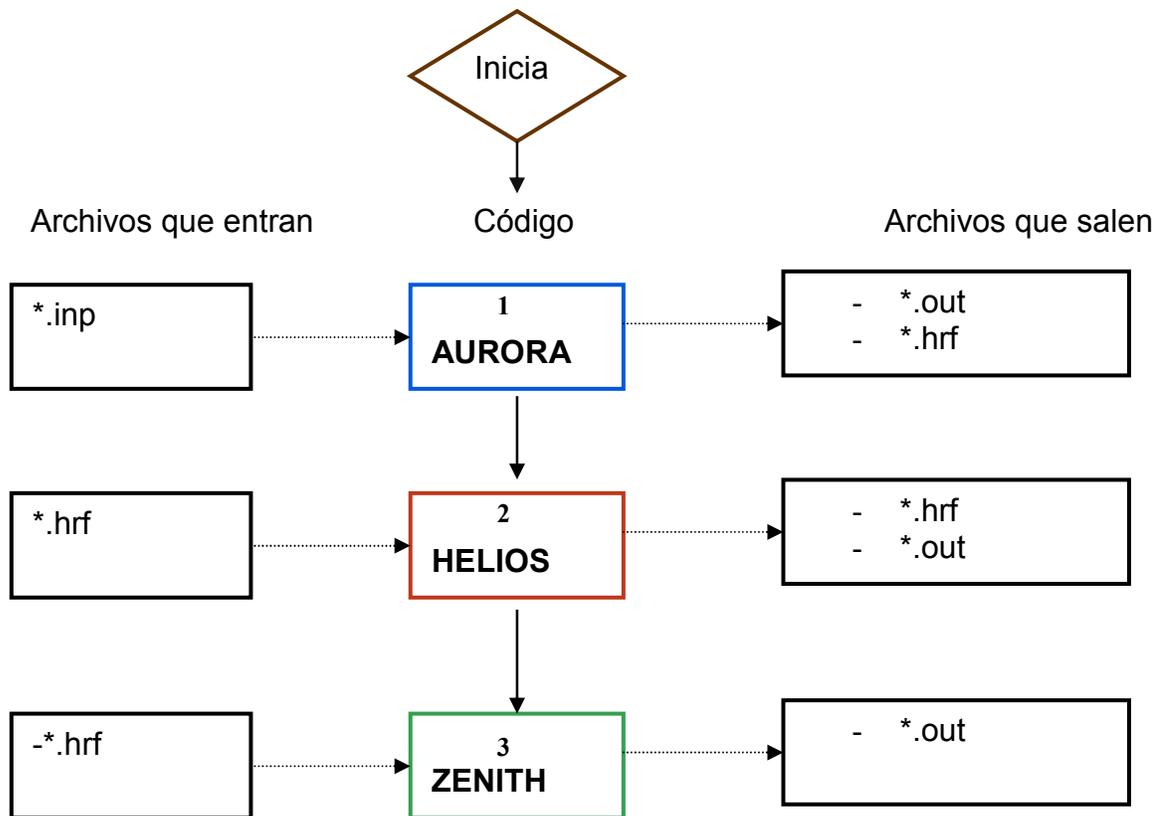


Figura 3.3. Proceso de valoración de una celda.

Es importante mencionar que la secuencia anteriormente descrita se realiza de manera continua, esto significa que si alguno de los códigos encuentra alguna falla

en el diseño propuesto, el proceso se detiene y se debe corregir el error antes de continuar con la siguiente fase de valoración de las celdas.

Una vez que se obtienen los parámetros FPP y k_{∞} de los elementos del conjunto disperso se procede a evaluarlos con la función objetivo. El $RefSet$ estará compuesto por 10 soluciones, a estos elementos los dividiremos en dos subconjuntos, los cuales se identificarán como $RefSet1$ y $RefSet2$. El $RefSet1$ se forma con las 5 mejores soluciones que se encuentran en el conjunto disperso, es decir, se eligen las 5 celdas en las que sea mínimo el valor de la función objetivo. Los elementos del $RefSet2$ se extraen por el criterio de máxima distancia, para esto se utiliza la ecuación (2.2) definida en el Capítulo 2, donde la distancia $d(\hat{x}, y_i)$ para este problema en particular se calcula de la siguiente manera:

$$d(\hat{x}, y_i) = |k_{\infty, \hat{x}} - k_{\infty, y_i}| + |FPP_{\hat{x}} - FPP_{y_i}| \quad (3.2)$$

donde

$k_{\infty, \hat{x}}$: Valor del parámetro k_{∞} del elemento \hat{x} que pertenece al CD.

k_{∞, y_k} : Valor del parámetro k_{∞} del elemento y_i que pertenece al subconjunto $RefSet1$.

$FPP_{\hat{x}}$: Valor del parámetro FPP del elemento \hat{x} que pertenece al CD.

FPP_{y_k} : Valor del parámetro FPP del elemento y_i que pertenece al $RefSet1$.

Lo que significa que se calculará la distancia de cada uno de los elementos del conjunto disperso a los 5 elementos que se encuentran en el $RefSet1$, como resultado de esto se obtienen 5 distancias para cada elemento del conjunto disperso. De estas distancias se elige la que sea menor y es la que se toma para

cada elemento. Una vez obtenidas estas 95 distancias se eligen los 5 elementos que tengan el máximo valor de estas distancias y son los que formarán el *RefSet2* que representarán los 5 elementos más alejados de los elementos del *RefSet1* que se encuentran en el conjunto disperso.

3.2.3 Mejora del Conjunto Referencia

Una vez que se obtiene el conjunto referencia, se realizarán las combinaciones entre los elementos de éste, con el fin de mejorar la calidad de los elementos que pertenecen al mismo conjunto. En esta etapa se desarrollan los pasos 3 y 4 del algoritmo de la Búsqueda Dispersa.

3.2.3.1 Método de combinación

Como se mencionó en el Capítulo anterior, el hablar de una combinación lineal entre 2 permutaciones no tiene sentido, por lo que se tuvo que diseñar un algoritmo específico para generar dichas combinaciones, de tal manera que se respeten las restricciones impuestas para el diseño de las celdas. Las combinaciones se harán por parejas, es decir se combinará un elemento de *RefSet1* con un elemento de *RefSet2*.

Para entender el mecanismo usado se da un ejemplo. Se combinarán los siguientes vectores A y B aplicando el algoritmo diseñado para dicha combinación. Por comodidad los vectores A y B son de longitud 10, la generalización para los 51 elementos no tiene mayor problema.

A	3.95	3.95	3.2	4.9	4.9	3.95	3.95	3.95	3.60	4.4
	1	2	3	4	5	6	7	8	9	10

B	4.9	3.95	3.95	4.9	3.60	3.95g6	3.2	3.95	3.2	3.6g6
	1	2	3	4	5	6	7	8	9	10

Cada uno de los elementos de los vectores A y B representan porcentajes de enriquecimientos de Uranio, los elementos que tienen una “g” seguida de un número significa que son enriquecimientos mezclados con Gadolinia, por ejemplo el 4.4g4 significa que la pastilla está compuesta por una mezcla de 4.4% de Uranio enriquecido y 4% de Gadolinia. Los elementos que hay en A son los mismos que hay en B, la diferencia entre ambos son las posiciones de cada uno de los elementos. En el ejemplo anterior, de la posición 1 a la 4 están restringidas a elementos que contengan gadolinia, esto es indispensable para no violar las restricciones impuestas. De esta manera se inicia la descripción de la combinación de los dos elementos.

Los elementos usados en A y B son: 3.2, 3.6, 3.95, 4.4 y 4.9. Se hará la combinación por parejas, es decir un elemento de A con un elemento de B. $A(1) = 3.95$ y $B(1) = 4.9$, por lo que se debe elegir un valor distinto a éstos, puede ser el 3.2, 3.6 y el 4.9, escogemos uno de ellos en forma aleatoria y es el que se coloca en el vector resultante C, por lo que $C(1) = 3.2$. Es importante mencionar que una vez que se selecciona un elemento, éste ya no debe elegirse nuevamente para no alterar el enriquecimiento promedio de la celda. De esta manera se realiza el proceso hasta completar la combinación de las permutaciones. Cabe aclarar que se debe verificar que los enriquecimientos que se vayan a colocar en las posiciones del 1 al 4 deben ser revisadas, puesto que no debe haber presencia de Gadolinia en estos lugares. Con lo anterior el vector resultante C es:

C	3.2	4.4	3.95	3.95	3.2	3.95	3.6g6	3.95	4.4g4	4.9
	1	2	3	4	5	6	7	8	9	10

Las combinaciones entre los elementos de *RefSet1* con un elemento de *RefSet2* se efectúan de manera similar. En este caso las combinaciones se llevan a cabo con 51 elementos. Se debe considerar que las 3 primeras posiciones permanecen fijas, y que de la posición 4 a la 19 se debe verificar que no se coloque Gadolinia para no violar las restricciones. En total se tienen como resultado, 5 combinaciones del Conjunto Referencia. Se pueden realizar un número mayor de combinaciones, lo cual dependerá, en buena medida, de los recursos disponibles.

3.2.3.2 Actualización

Las combinaciones resultado de la fase anterior, son evaluadas de acuerdo a la función objetivo. Si alguna de estas combinaciones mejora algún elemento que se encuentre en el *RefSet1* entonces se introduce el nuevo elemento a este conjunto, respetando el orden y desplazando de éste a la solución cuya evaluación en la función objetivo es más alta, esto sucede porque el tamaño del conjunto referencia permanece constante a lo largo del proceso. En la implementación del problema se realizó la actualización en forma dinámica, es decir cada que se encontraba una solución buena se introducía de inmediato en el *RefSet1*.

3.2.4 Re-encadenamiento de trayectorias

Después de aplicar Búsqueda Dispersa al Conjunto Referencia como método de mejora, se procede a realizar las trayectorias entre los elementos de este conjunto. En esta parte, el interés está únicamente en el conjunto *RefSet1*, el cual contiene las soluciones de mejor calidad. Se trazarán las trayectorias tomando como solución inicial el último elemento de este conjunto y como solución guía el primer elemento del mismo, esto porque las soluciones se encuentran ordenadas de mejor a peor.

En forma análoga a las combinaciones, se trabaja únicamente con 48 posiciones debido a que las 3 primeras permanecen fijas. El número máximo de caminos que se tracen para llegar a la solución guía partiendo de la solución inicial, son 48. Si se evaluaran todas las trayectorias se consumirían muchos recursos en el tiempo de cálculo. Por este motivo se eligen en forma aleatoria 5 caminos. Es importante mencionar que el número de trayectorias elegidas puede variar dependiendo, entre otras cosas, de los recursos disponibles.

 Las celdas con este contorno indican qué elemento se mueve para ir alcanzando a la solución guía.

 A diferencia de las anteriores, estas celdas indican las posiciones que no se necesitan cambiar, debido a que se encuentran en su posición correspondiente con la solución guía.

Tabla 3.1. Generación de trayectorias.

A	3.95	3.95	3.2	4.9	3.95g2	3.6	4.4g4	4.4	3.95	3.95g6	3.6	4.4
	1	2	3	4	5	6	7	8	9	10	11	12

Trayectorias

1	4.4	3.95	3.2	4.9	3.95g2	3.6	4.4g4	3.95	3.95	3.95g6	3.2	4.4
2	4.4	3.95	3.6	4.9	3.95g2	3.2	4.4g4	3.95	3.95	3.95g6	3.2	4.4
3	4.4	3.95	3.6	3.95	3.95g2	3.2	4.4g4	4.9	3.95	3.95g6	3.2	4.4
4	4.4	3.95	3.6	3.95	3.95g6	3.2	4.4g4	4.9	3.95	3.95g2	3.2	4.4
5	4.4	3.95	3.6	3.95	3.95g6	3.2	4.4	4.9	3.95	3.95g2	3.2	4.4g4
6	4.4	3.95	3.6	3.95	3.95g6	3.2	4.4	4.9	3.95g2	3.95	3.2	4.4g4
7	4.4	3.95	3.6	3.95	3.95g6	3.2	4.4	4.9	3.95g2	3.95	4.4g4	3.2

B	4.4	3.95	3.6	3.95	3.95g6	3.2	4.4	4.9	3.95g2	3.95	4.4g4	3.2
	1	2	3	4	5	6	7	8	9	10	11	12

Se toman 3 trayectorias de las 7 generadas en forma aleatoria. Las seleccionadas son las siguientes: 2,6 y 7. Los vectores correspondientes se muestran en la Tabla 3.2.

Tabla 3.2. Trayectorias seleccionadas aleatoriamente.

	1	2	3	4	5	6	7	8	9	10	11	12
2	4.4	3.95	3.6	4.9	3.95g2	3.2	4.4g4	3.95	3.95	3.95g6	3.2	4.4
5	4.4	3.95	3.6	3.95	3.95g6	3.2	4.4	4.9	3.95	3.95g2	3.2	4.4g4
7	4.4	3.95	3.6	3.95	3.95g6	3.2	4.4	4.9	3.95g2	3.95	4.4g4	3.2

En el Capítulo 2 se describió el algoritmo empleado para obtener las trayectorias, el cual es el mismo que se empleó en la implementación del problema. Se dará un ejemplo esta vez empleando 12 elementos, las primeras cuatro posiciones están restringidas a pastillas que se encuentren mezcladas con gadolinia. De esta manera veremos la diferencia entre las trayectorias y las combinaciones. En la

Tabla 3.1 se muestran las trayectorias generadas. En cada una de las trayectorias generadas, se señala con negritas los elementos que cambiaron de posición.

Una vez seleccionadas las trayectorias, se pasa a la siguiente fase de la técnica, en la cual se determina si alguna de éstas puede actualizar el conjunto referencia.

3.2.5 Actualización del Conjunto Referencia

Una vez que se obtienen las trayectorias, se lleva a cabo nuevamente el proceso de valoración de las celdas, se evalúan en el código Helios, de esta manera se obtienen los parámetros k_{∞} y FPP que evaluados en la función objetivo determinan la calidad del diseño obtenido resultado de las trayectorias, y se verifica si alguna de éstas es mejor a las que se encuentran en el subconjunto *RefSet1*, si ocurre esto se introduce este diseño al *RefSet1*, el tamaño de este subconjunto es constante, lo que implica que la solución de menor calidad que se encuentre en *RefSet1* se elimine de este subconjunto. De esta manera se actualiza el *RefSet* mediante las trayectorias trazadas desde la solución inicial a la solución guía. Cabe aclarar que a diferencia de la actualización en las combinaciones, en esta parte la actualización se realizó en forma estática, es decir, primero se evalúan las 5 trayectorias seleccionadas y después se determina si alguna de ellas entra al *RefSet1*.

Finalmente, los criterios de paro empleados en el sistema ohtli-RT son los siguientes:

- Cumplir con un número máximo de iteraciones.
- Cuando el conjunto referencia no se actualice después de 15 iteraciones.

El sistema ohtli-RT se programó en lenguaje FORTRAN 77, en una estación de trabajo alpha con plataforma UNIX.

Capítulo 4

RESULTADOS

Lo que sabemos es una gota de agua; lo que ignoramos es el océano.

Isaac Newton

El sistema ohtli-RT se ha utilizado para la valoración de diversos ejemplos, éstos es, celdas con diferentes porcentajes de uranio enriquecido y gadolinia. Por cuestiones de espacio, en este capítulo sólo se presentarán 5 ejemplos, para los cuales se hace la recopilación de los resultados obtenidos a lo largo del proceso de búsqueda.

Estos ejemplos comparten las mismas características en los diseños, es decir el enriquecimiento promedio de las celdas es el mismo, así como el número de barras de gadolinia que se encuentran en dichas celdas. Es importante mencionar que el resto de los resultados obtenidos son similares a los mostrados aquí.

4.1 CARACTERÍSTICAS DE LAS CELDAS ANALIZADAS

Para los ejemplos exhibidos, las características de las celdas generadas son las siguientes. El enriquecimiento promedio de U^{235} de las celdas es de 3.9%, con 12 barras de Gadolinia las cuales se encuentran distribuidas como se muestra en la Tabla 4.1.

Tabla 4.1. Porcentajes de Gadolinia en la celda.

Número de barras	Porcentaje de Gadolinia
2	2%
4	6%
2	4%
2	3%
2	5%

Es importante mencionar que por la simetría de espejo que existe en la celda, en el presente estudio se trabaja con la mitad de las barras de Gadolinia, es decir 6 barras. Los criterios de parada del sistema ohtli-RT son los siguientes, el número máximo de iteraciones es 300 y el sistema se detiene después de 15 iteraciones si el conjunto referencia no se actualiza. Como se indica en el algoritmo, el primer paso consiste en encontrar una solución semilla obtenida aleatoriamente, con base en los requerimientos del usuario, esto es, Uranio y Gadolinia. En esta primera etapa se muestran las soluciones semilla de los ejemplos, la Figura 4.1 corresponde al primer ejemplo, la Figura 4.2 al segundo y así sucesivamente. La nomenclatura presentada en las Figuras es la requerida para hacer uso del código HELIOS. Como ya se mencionó aquellas pastillas que incluyen una “g”, indican la cantidad de gadolinia en las mismas. Adicionalmente, las barras que contienen agua están señaladas con las letras WR.

2.0									
3.95	4.4g2								
2.8	2.8	3.6g6							
4.4	4.4g4	3.95g3	4.4g6						
3.6	2.4	3.95	4.9	4.9					
4.9	3.6g5	4.9	WR	WR	3.9				
3.6	3.95	4.4	WR	WR	3.6	4.9			
4.4	3.2	3.6	4.9	3.2	2.4	4.4	4.9		
4.9	3.6	2.8	3.95	4.9	4.9	4.9	4.9	4.9	
2.0	3.2	4.9	3.6	3.6	2.4	3.2	3.6	4.9	2.4

Figura 4.1. Semilla ejemplo 1

2.0									
2.4	4.4g2								
3.6	4.9	3.6g6							
3.95	3.6	4.4g6	3.95						
4.9	4.9	3.95	4.4	2.4					
3.6	3.95g4	4.9	WR	WR	3.2				
3.95	3.6	4.9	WR	WR	3.6	3.95			
4.9	4.9	3.95	3.6	3.95g3	4.9	3.95g5	3.2		
4.9	4.4	3.6	3.95	4.9	2.4	3.95	4.9	3.2	
2.4	2.8	3.95	4.4	4.9	4.4	4.9	3.6	2.8	2.4

Figura 4.2. Semilla ejemplo 2.

2.0										
2.8	3.95g2									
4.9	2.4	3.6g6								
2.4	4.4	4.9	3.6g6							
4.9	3.95g3	3.6	3.95	3.6						
4.4	3.95	3.6	WR	WR	3.6					
2.4	4.9	2.4	WR	WR	4.9	4.9				
4.9	3.6g5	3.6	4.9	3.6	3.95	3.6	4.4g4			
4.9	3.95	4.9	4.4	4.9	3.6	3.6	4.4	4.4		
2.0	3.95	4.9	4.9	3.95	4.9	4.4	2.4	3.95	2.0	

Figura 4.3. Semilla ejemplo 3.

2.0										
3.95	4.4g2									
3.95	2.4	2.4								
3.6	4.4	2.4	3.95							
3.6	4.4	3.6	4.9	4.9						
4.9	4.9	3.95	WR	WR	3.6					
4.4	3.95	3.6	WR	WR	4.9	3.95g6				
3.6	4.4	3.6	2.4	3.2	4.4	2.8	3.2			
4.9	4.4g6	4.4g4	3.95g3	4.9	4.4	3.6	3.95g5	4.9		
2.0	3.6	4.4	4.4	4.4	3.95	4.4	4.9	4.9	2.0	

Figura 4.4. Semilla ejemplo 4.

2.0										
4.4	3.6									
3.95	4.9	3.95g2								
4.4	4.9	4.4	3.6							
4.4	3.6	2.0	3.95	4.4g6						
4.9	4.9	3.6	WR	WR	4.9					
3.95	3.95g4	3.95g3	WR	WR	4.4	3.95g6				
3.6	3.2	4.9	4.4	3.6g5	4.9	2.0	3.2			
2.4	3.6	4.4	4.9	3.6	3.2	2.4	3.2	4.9		
2.0	3.95	4.9	4.4	3.6	4.9	4.9	4.4	4.4	2.0	

Figura 4.5. Semilla ejemplo 5.

4.2 OBTENCIÓN DEL CONJUNTO REFERENCIA

Con base en la semilla de cada ejemplo, se generaron 96 soluciones con el algoritmo de soluciones dispersas para cada uno de éstos. De las cuales se tomaron 95², de éstas se eligieron las 5 mejores, de acuerdo a la evaluación de la función objetivo, esto es, las soluciones con un menor valor, y son los elementos que componen el subconjunto *RefSet1* inicial. Las Tablas 4.2, 4.3 y 4.4 muestran los valores de la Función objetivo, de la k_{∞} y del FPP del subconjunto *RefSet1* inicial para cada uno de los ejemplos.

Una vez obtenido el *RefSet1* inicial, el siguiente paso es encontrar el conjunto más alejado de éste, con base en la métrica definida en el Capítulo anterior para tal efecto. La función usada es la ecuación (3.1) del Capítulo 3. Es importante señalar

² De acuerdo al algoritmo presentado en el capítulo 3, se debe recordar que una de las soluciones se repite, por tal motivo únicamente se toman 95, en lugar de las 96 soluciones.

que el número de elementos del conjunto en cuestión es el mismo que el del *RefSet1*.

Tabla 4.2. Valor de la función objetivo en el *RefSet1* inicial.

Diseño	Ejemplo 1	Ejemplo 2	Ejemplo 3	Ejemplo 4	Ejemplo 5
1	6.6044	1.3842	1.9421	7.6595	7.8725
2	6.6248	1.4165	1.9526	7.9708	7.9015
3	6.7218	1.4277	1.9781	8.0810	8.0215
4	6.7269	1.4334	1.9824	8.1166	8.0885
5	6.7284	1.4413	2.0361	8.1195	8.2065

Tabla 4.3. Valor del factor de multiplicación infinita en el *RefSet1* inicial.

Diseño	Ejemplo 1	Ejemplo 2	Ejemplo 3	Ejemplo 4	Ejemplo 5
1	1.17333	1.18169	1.22526	1.18208	1.5745
2	1.14272	1.15949	1.24530	1.23528	1.5803
3	1.20768	1.16123	1.24343	1.15991	1.6043
4	1.24164	1.16476	1.24800	1.24020	1.6177
5	1.19016	1.18650	1.24033	1.18218	1.6104

Tabla 4.4. Valor del factor de pico de potencia en el *RefSet1* inicial.

Diseño	Ejemplo 1	Ejemplo 2	Ejemplo 3	Ejemplo 4	Ejemplo 5
1	1.6511	1.5574	1.7845	1.5319	1.5745
2	1.6562	1.5937	1.4995	1.5430	1.5803
3	1.6372	1.6063	1.5658	1.6162	1.6043
4	1.6130	1.6127	1.5038	1.5692	1.6177
5	1.6821	1.6216	1.6993	1.6239	1.6104

En la Tabla 4.5 se muestran los valores obtenidos para las mayores distancias entre el *RefSet1* y el conjunto disperso de cada uno de los ejemplos. De la Tabla 4.6 a la 4.8 se muestran los elementos que forman el subconjunto *RefSet2*. En dichas Tablas se incluye el valor de los parámetros obtenidos con dicho conjunto, esto es, el *FPP* y la k_{∞} , así como el respectivo valor de la función objetivo de cada uno de los ejemplos. De esta manera se completa el *RefSet*.

Tabla 4.5. Distancias más grandes entre el conjunto disperso y el *RefSet1*.

Diseño	Ejemplo 1	Ejemplo 2	Ejemplo 3	Ejemplo 4	Ejemplo 5
1	0.60855	0.41608	0.51601	0.45094	0.42505
2	0.42065	0.41155	0.42185	0.34173	0.41084
3	0.41194	0.39489	0.35924	0.34071	0.38377
4	0.35106	0.34755	0.34071	0.32691	0.36945
5	0.34031	0.34122	0.31102	0.32514	0.36021

Tabla 4.6. Valor de la función objetivo en el *RefSet2* inicial.

Diseño	Ejemplo 1	Ejemplo 2	Ejemplo 3	Ejemplo 4	Ejemplo 5
1	8.9908	1.7685	2.8132	10.4257	10.1865
2	8.2660	1.7980	2.5616	9.8622	10.0715
3	8.2712	1.7854	2.5151	9.8744	9.9205
4	8.0008	1.7153	2.5414	9.7578	9.8635
5	7.8968	1.7354	2.4610	9.7365	9.7765

Tabla 4.7. Valor del factor de multiplicación infinita en el RefSet2 inicial.

Diseño	Ejemplo 1	Ejemplo 2	Ejemplo 3	Ejemplo 4	Ejemplo 5
1	1.12567	1.12578	1.32358	1.20472	1.15437
2	1.15391	1.16351	1.28289	1.21341	1.13468
3	1.16392	1.17881	1.29237	1.20479	1.13155
4	1.13566	1.13441	1.29600	1.22859	1.13447
5	1.12041	1.17618	1.28685	1.18044	1.17524

Tabla 4.8. Valor del factor de pico de potencia en el RefSet1 inicial.

Diseño	Ejemplo 1	Ejemplo 2	Ejemplo 3	Ejemplo 4	Ejemplo 5
1	2.2477	1.9898	1.6163	2.0523	2.0373
2	2.0665	2.0230	1.8492	1.9344	2.0143
3	2.0678	2.0088	1.6372	1.9420	1.9841
4	2.0002	1.9299	1.6222	1.9044	1.9727
5	1.9742	1.9525	1.6389	1.9473	1.9553

4.3. ACTUALIZACIÓN DEL CONUNTO REFERENCIA

Con la construcción de los conjuntos anteriores se inicia el proceso iterativo donde en cada paso se llevan a cabo dos tareas perfectamente definidas, en ambas se actualiza el conjunto referencia (cuando se encuentra una mejor solución). En el primer paso se realizan combinaciones entre los subconjuntos *RefSet1* y *RefSet2* (aplicando la BD). En el presente trabajo se tomaron parejas de elementos, una por cada conjunto. En total se realizan 5 combinaciones por iteración y se lleva a cabo una actualización dinámica en el *RefSet1*.

Al concluir el proceso anterior, se procede al re-encadenamiento de trayectorias, tomando como solución inicial el peor elemento del *RefSet1* (de acuerdo a la función objetivo) y como solución guía el mejor valor de dicho conjunto. El número máximo de trayectorias que se pueden trazar entre ambas soluciones es 48, tomando en cuenta que son permutaciones las que se realizan para conseguir esto. De las posibles trayectorias se eligen al azar 5 de ellas. Es importante señalar que las trayectorias se construyen en cada iteración, lo cual puede ser modificable, dependiendo de los resultados obtenidos.

En la Tabla 4.9 se muestran, el número de combinaciones y trayectorias que ayudaron a actualizar el *RefSet* durante el proceso de búsqueda en cada uno de los ejemplos mostrados. Se puede observar que en la mayoría de los ejemplos la actualización del conjunto referencia se llevó a cabo con Path Relinking, lo que muestra que la solución obtenida toma atributos de las trayectorias generadas. De la misma Tabla se observa el número de actualizaciones totales que se hicieron al conjunto referencia en cada ejemplo.

Tabla 4.9. Actualizaciones del RefSet.

Ejemplo	Número de combinaciones	Número de trayectorias	Actualizaciones totales del <i>RefSet</i>
1	11	19	30
2	5	6	11
3	14	24	38
4	13	23	36
5	11	12	23

Por otro lado, en las pruebas realizadas, la actualización del conjunto referencia con PR se realizó en forma estática, es decir, primero se construyen las trayectorias elegidas y al final se actualiza el subconjunto *RefSet1*. En la Tabla 4.10 se muestra la iteración en la que cada ejemplo se detuvo, de acuerdo a uno de los

criterios de paro establecidos anteriormente, así como el número de evaluaciones de la función objetivo realizados para cada ejemplo.

Tabla 4.10. Iteración de parada.

Ejemplo	Iteración	Evaluaciones de la Función
1	57	665
2	26	355
3	43	525
4	55	645
5	34	435

4.4 CELDAS OBTENIDAS

En el mismo orden de ideas, de la Tabla 4.11 a la 4.13 se presentan los resultados finales del *RefSet1* de cada uno de los ejemplos donde se muestra el valor de la función objetivo y los parámetros obtenidos respectivamente.

Tabla 4.11. Valor de la función objetivo en el *RefSet1* final.

Diseño	Ejemplo 1	Ejemplo 2	Ejemplo 3	Ejemplo 4	Ejemplo 5
1	5.5912	1.3212	0.9698	7.1860	7.0420
2	5.5912	1.3256	0.9750	7.1865	7.0970
3	5.6076	1.3259	0.9766	7.1865	7.1265
4	5.6076	1.3449	0.9774	7.1865	7.1265
5	5.6136	1.3538	0.9822	7.2615	7.1265

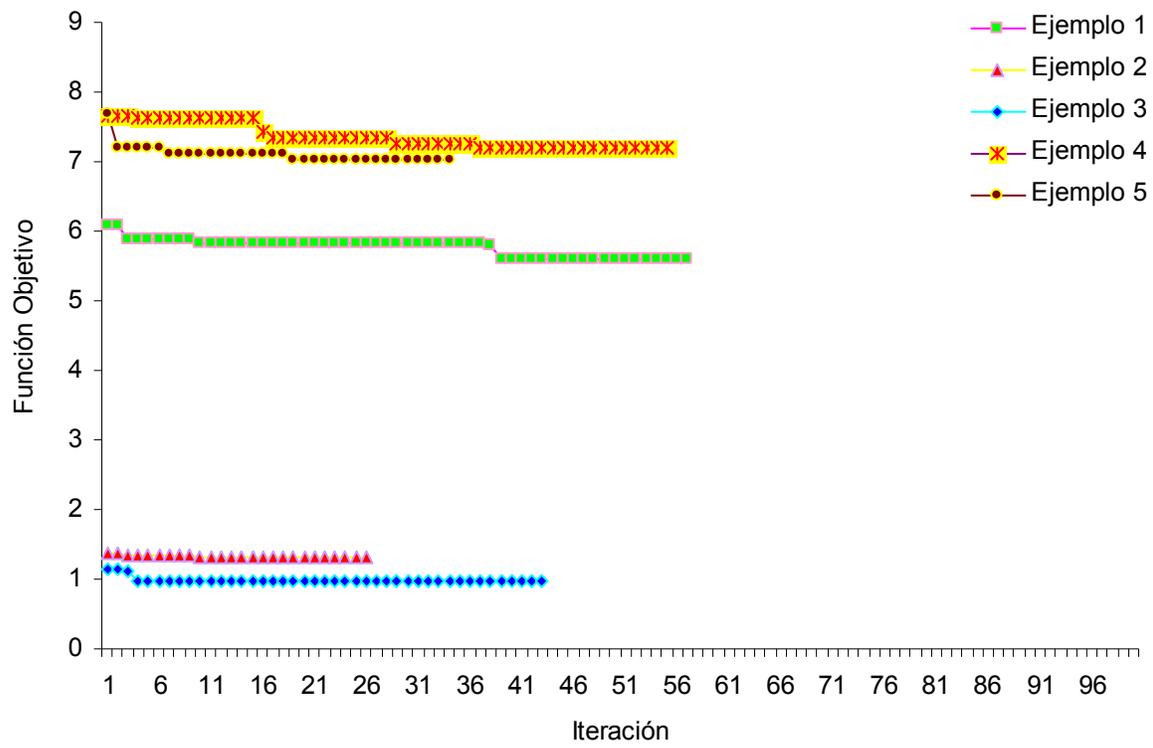
Tabla 4.12. Valor del factor de multiplicación infinita en el *RefSet2* final.

Diseño	Ejemplo 1	Ejemplo 2	Ejemplo 3	Ejemplo 4	Ejemplo 5
1	1.17947	1.17540	1.19403	1.15784	1.19569
2	1.17947	1.17414	1.19154	1.15780	1.18848
3	1.17888	1.17392	1.18912	1.15780	1.19491
4	1.17888	1.18694	1.19410	1.15780	1.19491
5	1.17961	1.17727	1.18906	1.15792	1.19491

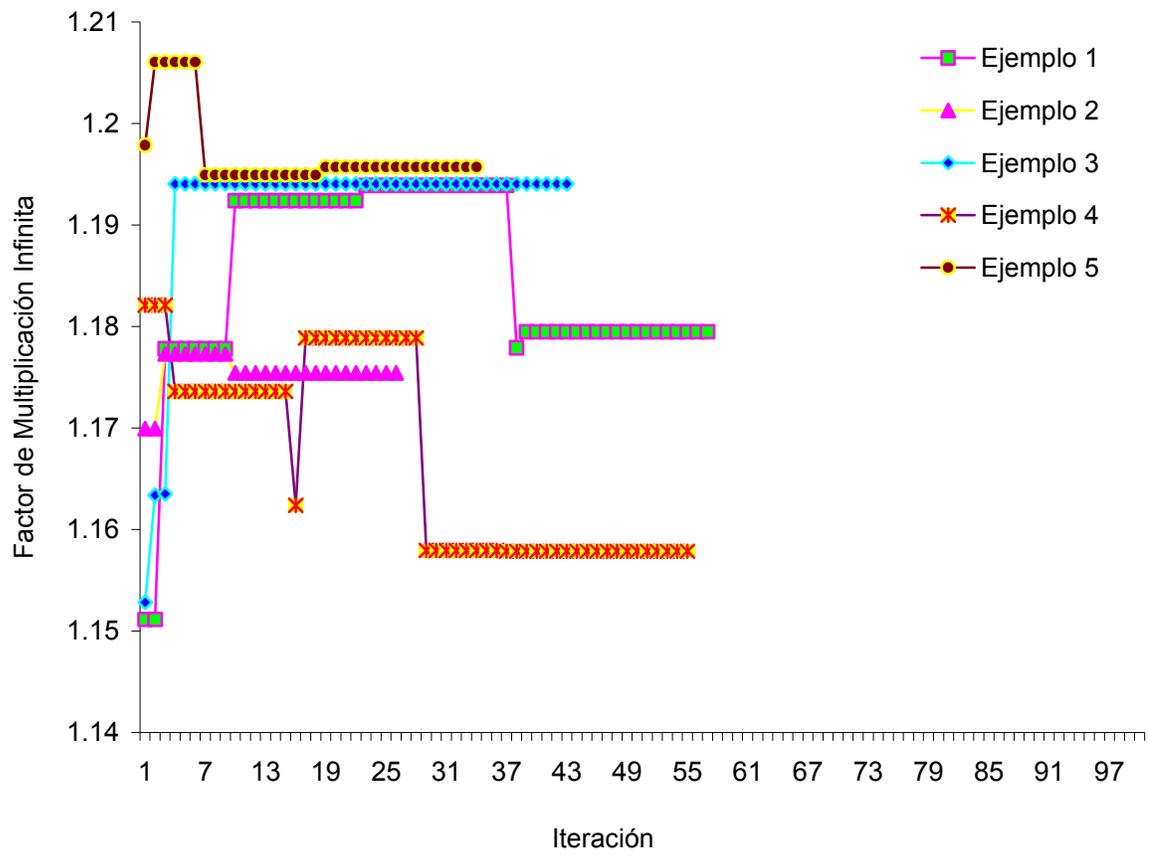
Tabla 4.13. Valor del factor de pico de potencia en el *RefSet1* final.

Diseño	Ejemplo 1	Ejemplo 2	Ejemplo 3	Ejemplo 4	Ejemplo 5
1	1.3978	1.4865	1.4548	1.4372	1.4084
2	1.3978	1.4915	1.4627	1.4373	1.4194
3	1.4019	1.4918	1.4651	1.4373	1.4253
4	1.4019	1.5132	1.4663	1.4373	1.4253
5	1.4019	1.5232	1.4734	1.4523	1.4253

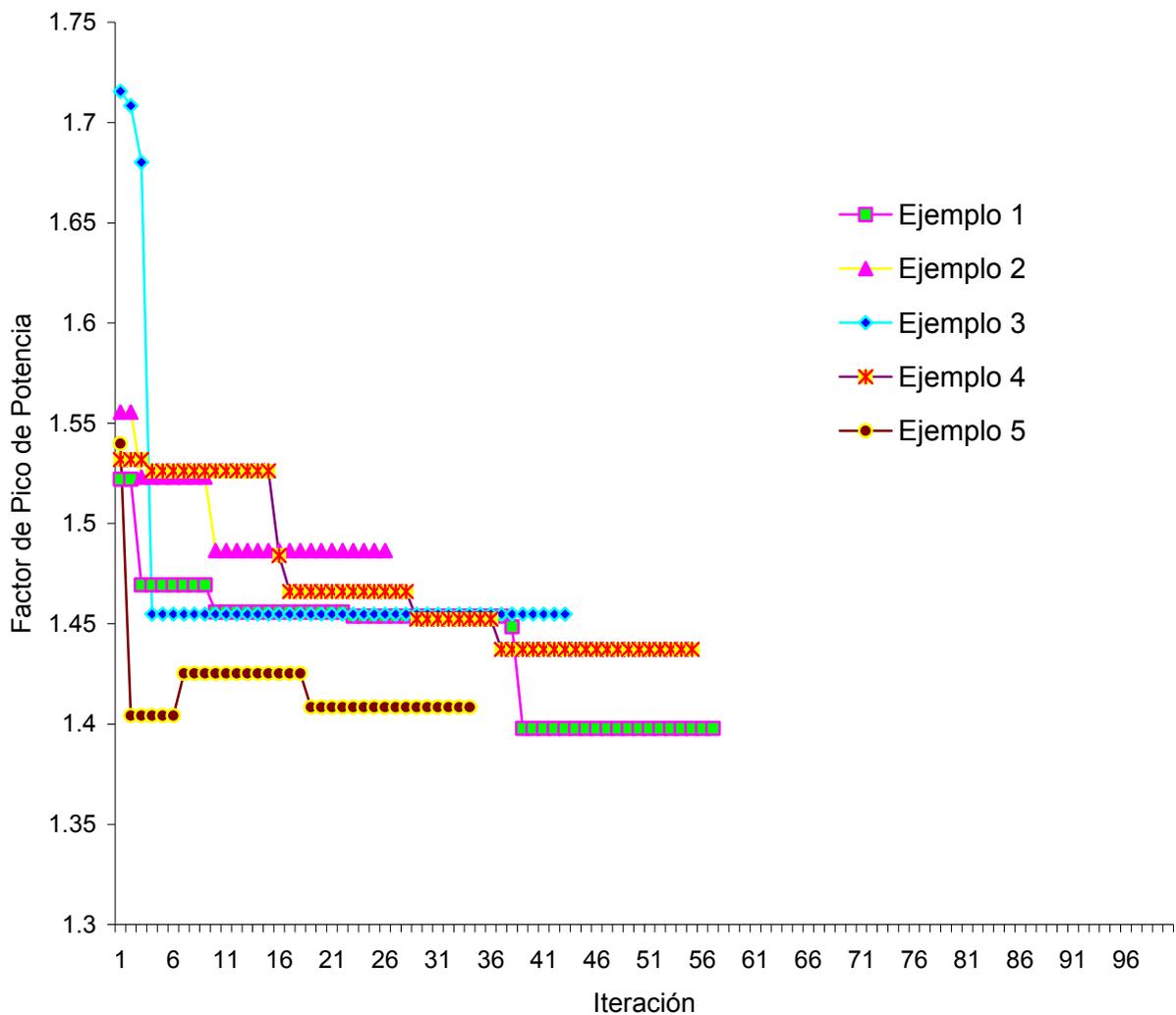
Asimismo, en las Gráficas 4.1, 4.2 y 4.3 se muestra el comportamiento de la función objetivo, el valor del factor de multiplicación infinita y del Factor de Pico de Potencia respectivamente, a lo largo del proceso de optimización.



Grafica 4.1. Comportamiento de la Función objetivo.



Grafica 2. Comportamiento del factor de multiplicación infinita.



Grafica 3. Comportamiento del factor de pico de potencia.

Es importante mencionar que sólo se presenta el subconjunto *RefSet1* debido a que es el único que se actualiza al lo largo del proceso de búsqueda, el *RefSet2* permanece fijo, es decir es el mismo al inicio y al final del proceso.

Con estos resultados, en la Tabla 4.14 se muestra la comparación entre el mejor valor del *RefSet1* al inicio del proceso y el mejor valor de este mismo conjunto al final del proceso iterativo de cada uno de los ejemplos mostrados. Las Tablas

anteriores nos permitirán analizar la manera en como se comporta el conjunto referencia a lo largo del proceso de búsqueda, en el Capítulo 5 se muestran las conclusiones a la que se llegó con estos resultados.

Tabla 4.14. Comparación de resultados del RefSet.

Ejemplo	Diseño	k_{∞}	FPP	Función Objetivo
1	Inicial	1.17333	1.6511	6.6044
	Final	1.17947	1.3978	5.5912
2	Inicial	1.18169	1.5574	1.3842
	Final	1.17540	1.4865	1.3212
3	Inicial	1.22526	1.7845	1.9421
	Final	1.19403	1.4548	0.9698
4	Inicial	1.18208	1.5319	7.6595
	Final	1.15784	1.4372	7.1860
5	Inicial	1.17635	1.5745	7.8725
	Final	1.19569	1.4084	7.0420

Finalmente se muestran los diseños obtenidos al concluir el proceso de búsqueda de cada ejemplo. La Figura 4.6 corresponde al ejemplo 1, la 4.7 al ejemplo 2 y así sucesivamente. Es necesario resaltar que en los cinco ejemplos anteriores se eligió el primer elemento del conjunto referencia como mejor diseño, tomando como base las restricciones para los parámetros.

2.0										
2.4	3.6									
2.4	3.95	3.6								
2.4	3.95	3.95g3	3.6							
3.6	4.4	4.4	4.4	3.6						
2.8	4.4	4.4g2	WR	WR	3.6g5					
4.9	4.4g4	4.4g6	WR	WR	4.9	3.6g6				
2.8	4.9	4.9	4.9	4.9	4.9	4.9	3.95			
2.8	4.9	4.9	4.9	4.9	4.9	4.9	4.9	3.95		
2.0	3.2	3.6	3.2	3.95	3.2	3.2	3.6	3.6	2.4	

Figura 4.6. Distribución final Ejemplo 1.

2.0										
2.4	3.95									
3.2	3.95	3.6								
4.9	3.95	3.95g4	3.6							
3.95	3.95	3.95	3.95	3.95g5						
2.4	3.95g3	4.4	WR	WR	3.6					
4.9	4.4	4.4	WR	WR	4.4	3.6				
2.4	4.4g2	4.4g6	4.9	4.9	4.9	4.9	3.6			
2.8	4.9	4.9	4.9	4.9	4.9	4.9	4.9	3.6g6		
2.4	2.8	3.2	3.6	3.2	3.6	3.6	3.95	3.95	2.4	

Figura 4.7. Distribución final Ejemplo 2.

2.0										
3.6	3.6									
4.9	3.95g3	3.6								
3.95	4.4	3.95	3.6							
2.4	3.95	3.95	3.95	3.6						
2.4	3.95g2	3.95	WR	WR	3.6g5					
2.4	4.4	4.4	WR	WR	4.4	3.6g6				
3.95	4.4g4	4.4	4.9	4.9	4.9	4.9	3.6g6			
2.4	4.9	4.9	4.9	4.9	4.9	4.9	4.9	4.4		
2.0	2.4	2.8	4.9	4.9	3.6	3.6	3.6	3.6	2.0	

Figura 4.8. Distribución final Ejemplo 3.

2.0										
3.6	3.95									
4.9	4.4	3.95								
2.4	4.4	4.4	3.95							
3.95	4.4	4.4	4.4	3.95g3						
2.4	4.4g2	4.4g4	WR	WR	3.95g5					
2.4	4.9	4.9	WR	WR	4.9	3.95g6				
2.8	4.9	4.9	4.9	4.9	4.9	2.0	4.4			
3.2	2.0	4.9	4.9	4.4	3.6	4.4	2.4	4.4		
2.0	3.6	3.6	3.6	3.6	3.6	3.6	3.6	3.95	2.0	

Figura 4.9. Distribución final Ejemplo 4.

2.0										
3.6	3.95g3									
4.9	4.4	3.95								
2.0	4.4	4.4	3.95							
2.4	4.4	4.4	4.4	3.95g2						
2.0	4.4g6	4.9	WR	WR	3.95g4					
2.4	4.9	4.9	WR	WR	4.9	3.95g6				
3.2	4.9	4.9	4.9	4.9	4.9	2.0	4.4			
3.6	2.0	4.9	3.6	4.9	2.0	3.6	3.6g5	4.4		
2.0	3.6	3.2	3.6	3.6	3.6	3.6	3.95	4.4	2.0	

Figura 4.10. Distribución final Ejemplo 5.

De esta manera se mostraron los resultados obtenidos al implementar la técnica Path Relinking en el diseño radial de celdas de combustible. En el Capitulo siguiente se incluyen las conclusiones a las que se llegaron con el presente trabajo.

Capítulo 5

CONCLUSIONES

Un libro abierto es un cerebro que habla; cerrado, un amigo que espera; olvidado, un alma que perdona; destruido, un corazón que llora...

Proverbio hindú

Los resultados obtenidos muestran que las celdas diseñadas en el presente trabajo, tienen un desempeño adecuado, de acuerdo al criterio establecido. Esto es, tomando en cuenta el Factor de Pico de Potencia (FPP) y el Factor de Multiplicación Infinita (k_{∞}). Resaltando uno de los ejemplos, en el cuál los parámetros antes mencionados son mejores que los encontrados en los trabajos revisados.

Por otro lado, se observa en la Tabla 4.14 del Capítulo 4, que se tienen resultados similares a los reportados por Martín del Campo *et al* (ver referencia [2]), en lo referente al Factor de Pico de Potencia, cuyo resultado es de 1.43. Es necesario resaltar que no es posible hacer una comparación completa con dichos resultados, debido a que los criterios manejados entre ambos trabajos son diferentes.

Durante las pruebas realizadas hasta el momento, el valor para el Factor de Multiplicación Infinita casi siempre se encuentra dentro de los límites propuestos, desde el inicio del proceso, lo cual no sucede con el Factor de Pico de Potencia. Lo que indica que es necesario darle un mayor peso a dichos parámetros para que se cumpla.

En la generación de la solución semilla de cada ejemplo, como se observa en las Figuras 4.1, 4.2 y 4.3 del Capítulo 4, las barras de Gadolinia se encuentran distribuidas en distintas formas, en algunos diseños este veneno quemable se encuentra en los primeros renglones, como se puede observar en la Figura 4.1, a diferencia de la Figura 4.4, donde los elementos se encuentran en los renglones inferiores. En el resto de los diseños la gadolinia se distribuye en todo el diseño sin mostrar alguna tendencia.

Analizando las soluciones que se encuentran en el subconjunto *RefSet1* inicial, se observa que para estos ejemplos en particular los valores de la k_{∞} se cumplen para los ejemplos 1, 2, 3 y 4 ya que se encuentran dentro del intervalo requerido, es decir, estos diseños tienen un valor de k_{∞} por debajo del valor establecido el cual es 1.3, en cuanto al ejemplo 5 todos los elementos que se encuentran en el *RefSet1* no cumplen con este criterio. Es importante aclarar que en esta etapa, lo único que se le pide a la solución semilla es que cumpla con el promedio de uranio enriquecido de la celda y con el número de barras de gadolinia requeridas.

En cuanto a los valores del *FPP* y siguiendo con el *RefSet1* inicial, en ninguno de los ejemplos se cumple con la restricción impuesta para este parámetro, siempre hablando del *RefSet1*, ya que todos los elementos se encuentran por arriba de 1.45 el cual es el valor máximo permitido.

Los elementos del *RefSet2* inicial, como era de esperarse, muy pocos de éstos cumplen con la restricción de la k_{∞} , y de los valores del *FPP* ninguno cumplen la condición, ya que presenta valores muy altos para este parámetro.

Con el *RefSet* inicial, se comenzó el proceso de optimización para hallar una solución de calidad. En los ejemplos mostrados el proceso se detuvo cuando el conjunto referencia no presentó mejora alguna después de 15 iteraciones. Resulta interesante observar que el número de actualizaciones de este conjunto con Path Relinking fue mayor que con la Búsqueda Dispersa lo cual se observa en la Tabla 4.9. Se generaron el mismo número de trayectorias y de combinaciones, lo cual refleja que las trayectorias ayudan a encontrar más rápidamente soluciones de mejor calidad. Por este motivo, se construyen trayectorias en cada iteración.

Nuevamente en el ejemplo 3, de la Tabla 4.9 se observa que el número de trayectorias que actualizaron al *Refset1* es más de la mitad con respecto a las combinaciones, y es el ejemplo en el que se realizaron más actualizaciones del *Refset1*, con un total de 38, a diferencia del ejemplo 2 donde el número de trayectorias y de combinaciones difieren por una unidad, en total para éste ejemplo se actualiza únicamente 11 veces este subconjunto.

Una vez terminado el proceso de optimización, en la Tabla 4.12 se observa que el subconjunto *RefSet1*, en los 5 ejemplos al final del proceso se cumple la restricción de la k_{∞} , el valor más alto que se presenta es de 1.19 y el más bajo es de 1.5, ambos valores satisfacen la restricción del parámetro. En el caso del *FPP* como se observa en la Tabla 4.13, los ejemplos mostrados cumplen con lo establecido, excepto el ejemplo 2 el cual se encuentra ligeramente por arriba del valor

requerido. Es importante mencionar que en los ejemplos 1, 4 y 5 de la misma tabla, se encuentran los valores más bajos de este parámetro.

Estos hechos se ven reflejados en las Gráficas 4.1, 4.2 y 4.3, donde se observa la evolución de la Función objetivo y de los parámetros en cada uno de los ejemplos a lo largo del proceso iterativo, aunque inician con valores que no cumplen con las restricciones de los parámetros, al final del proceso éstos se estabilizan. En cuanto a los valores de la k_{∞} , en todos los ejemplos los valores finales quedan entre 1.15 y 1.2, en el *FPP* es más notoria la disminución de los valores de este parámetro, la mayoría de los ejemplos cumplen con la restricción impuesta para dicho parámetro, obteniendo el valor mínimo en 1.39 el cual corresponde al ejemplo número 1. Es interesante observar que en este ejemplo, al inicio del proceso la distribución de la Gadolinia en la semilla, se encontraban en los primeros renglones del diseño. La celda semilla no se considera con una buena distribución, sin embargo al final del proceso de búsqueda se encontró un valor significativamente bajo del *FPP*.

Por lo que se observa en los resultados, la restricción para la k_{∞} no tiene mayor problema, sin embargo, para el *FPP* es más difícil que se encuentre en el intervalo impuesto. Al final del proceso, en la mayoría de los elementos del subconjunto *RefSet1* los parámetros analizados (k_{∞} y *FPP*) son bastante similares, lo que puede verse como una tendencia a una buena solución.

Una observación importante es el tiempo de CPU invertido para cada diseño, el cual todavía es considerable, esto puede ocurrir por diversos factores, uno de ellos es por el equipo de cómputo que se utiliza para realizar los cálculos, si se contara con un equipo más veloz el tiempo disminuiría.

Referencias

Artículos:

1. F. C. García-López, B. Melián, J. A. Moreno-Pérez, and J. M. Moreno-Vega. "Hybrid metaheuristics based on the scatter search". *Proceedings of the European Symposium on Intelligent Technologies, EUNITE*. 2001.
2. Cecilia Martín del Campo, Juan Luis François, Ramiro François. "Aplicación de búsqueda tabú a la optimización de la distribución radial de enriquecimiento en un ensamble de combustible para BWR". *XIII Congreso Anual SNM/ XX Reunión anual SMRS*, Ixtapa Zihuatanejo. 2002.
3. Cecilia Martín del Campo, Juan Luis François, Linda Avendaño, Mario González, Jaime González. "Optimización integrada del diseño axial de combustible y del diseño de patrón de recarga en BWR". *XIII Congreso Anual SNM / XX Reunión Anual SMSR*, Ixtapa Zihuatanejo. 2002.
4. Jaime González Contreras y Cecilia Martín del Campo Márquez. "Evaluación del diseño radial de celdas de combustible en un ciclo de operación de un reactor BWR". *XIV Congreso anual de la SNM/XXI*, Guadalajara Jalisco. 2003.
5. Jaime González Contreras, Cecilia Martín del Campo Márquez y Juan Luis François Lacouture. "Estudio de sensibilidad sobre reglas heurísticas

aplicadas a la optimización neutrónica de celdas para BWR”. *XV Congreso Anual de la SNM*, Cancún, México. 2004.

6. R. Perrusquia, J. L. Montes, J. J. Ortiz. “Desarrollo de un programa de cómputo de cálculo rápido para el prediseño de celdas de combustible nuclear avanzado 10x10 para reactores de agua en ebullición”. *XVI Congreso Anual de la SNM y XXIII Reunión anual de la SMSR*, Oaxaca, Oaxaca. 2005.
7. Cecilia Martín del Campo Márquez, Roberto Carmona Hernández, Ivonne Patricia Camargo. “Optimización radial de una celda de combustible BWR usando algoritmos genéticos”. *XVII Congreso Anual de la SNM*, Acapulco México. 2006.

Libros:

8. Zbigniew Michalewicz. “Genetic algorithms + Data structures = Evolution programs”. *Springer*. 1992.
9. Manuel Laguna, Rafael Martí. ”Scatter search, methodology and implementations in C”. *Kluwer Academic Publishers*, Boston, USA. 2003.

Revistas:

10. J.K. Hao, E. Lutton, E. Ronald, M. Schoenauer, D. Snyers (Eds.). “A template for scatter search and path relinking”, *Lecture Notes in Computer Science* 1363, pp. 1-53. 1997.
11. Fred Glover, Manuel Laguna. “Fundamentals of scatter search and path relinking”. *Control and Cybernetics* 29, pp. 653-684. 2000.

12. Fred Glover, Manuel Laguna y Rafael Marti. "Scatter search and path relinking: Advances and applications". *Working Papers Series Hearin Center for Enterprise Science*, the University of Mississippi, USA. 2002.
13. B. Melián, J. A. Moreno-Pérez, y J. M. Moreno-Vega. "Metaheurísticas: Una visión global". *Revista Iberoamericana de Inteligencia Artificial* 19, pp. 7-28. 2003.
14. Fred Glover y Belén Melián. "Búsqueda Tabú". *Revista Iberoamericana de Inteligencia Artificial* 19, pp. 29-48. 2003.
15. J.L. François, C. Martín del Campo R. François, L.B. Morales. "A practical optimization procedure for radial BWR fuel lattice design using tabú search with a multiobjective function". *Annals of Nuclear Energy* 30, pp. 1214-1229. 2003.
16. Carlos A.S. Oliveira, Panos M. Pardalos and Mauricio G. C. Resende. "GRASP with path-relinking for the quadratic assignment problem". AT&T Labs Technical Report, 2004.
17. José Alejandro Castillo, Juan José Ortiz, Gustavo Alonso, Luis B. Morales, Edmundo del Valle. "BWR control rod design using tabú search". *Annals of Nuclear Energy* 32, pp 741-754. 2005.
18. Michel Gendreau y Jean-Yves Potvin. "Metaheuristics in combinatorial optimization". *Annals of Operations Research* 140, pp189–213. 2005.

Manuales:

19. Studsvik Scandpower, "Aurora Methods".(1998).
20. Studsvik Scandpower, "Helios Methods".(1998).
21. Studsvik Scandpower, "Zenith Methods".(1998).

Material consultado en Internet:

22. <http://www.cfe.gob.mx/es/LaEmpresa/generacionelectricidad/>.
23. http://es.wikipedia.org/wiki/Combustible_nuclear#Combustibles_de_fisi.C3.B3n_nuclear
24. http://www.sagangea.org/hojared_radiacion/paginas/Planta%20Laguna_Verde.html
25. http://www.fip.unam.mx/simposio_investigacion/estudio_nuevo.html
26. http://es.wikipedia.org/wiki/Combustible_nuclear#Combustibles_de_fisi.C3.B3n_nuclear
27. http://www.fi-p.unam.mx/simposio_investigacion/aplicacion.html
28. <http://triga.ga.com/fuel.html>

Agradecimientos

Al M. en C. José Alejandro Castillo Méndez por compartir sus conocimientos y guiarme para concluir una meta importante en mi vida.

Al Dr. Edmundo del Valle, por su colaboración para la realización de mi trabajo.

Al Instituto Nacional de Investigaciones Nucleares, por haberme brindado la oportunidad de utilizar las instalaciones dentro del Instituto.

Al Instituto Politécnico Nacional por la formación académica que obtuve durante mi estancia en este centro de enseñanza.

Al Departamento de Gestión de Combustible de la Comisión Federal de Electricidad por las facilidades brindadas para la elaboración del presente trabajo.

Se desea agradecer de manera especial al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el apoyo brindado para el desarrollo del presente trabajo a través del proyecto SEP-2004-C01-46694.