



PONTIFICIA UNIVERSIDAD CATOLICA DE CHILE
ESCUELA DE INGENIERIA
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

LOCALIZACIÓN ÓPTIMA DE CONDENSADORES EN UN SISTEMA DE DISTRIBUCIÓN ELÉCTRICO VIA ALGORITMOS GENÉTICOS

MIGUEL ÁNGEL LEIVA GÓMEZ

Memoria para optar al título de
Ingeniero Civil de Industrias, con Diploma en Ingeniería
Eléctrica

Profesor Supervisor:
SR. HUGH RUDNICK

Santiago de Chile, 2001



PONTIFICIA UNIVERSIDAD CATOLICA DE CHILE
ESCUELA DE INGENIERIA
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

LOCALIZACIÓN ÓPTIMA DE CONDENSADORES EN UN SISTEMA DE DISTRIBUCIÓN ELÉCTRICO VIA ALGORITMOS GENÉTICOS

MIGUEL ÁNGEL LEIVA GÓMEZ

Memoria presentada a la Comisión integrada por los profesores:

HUGH RUDNICK

LUIS CONTESSE

RODRIGO PALMA

Para completar las exigencias del título de
Ingeniero Civil de Industrias, con Diploma en Ingeniería Eléctrica

Santiago de Chile, 2001

A mis Padres, quienes me brindaron su apoyo incondicional durante el desarrollo de mi carrera.

AGRADECIMIENTOS

En primer lugar, deseo agradecer a Dios, por ser la luz que me ha guiado durante toda mi existencia.

Deseo agradecer a mi familia, especialmente a mis padres y hermana, por haberme brindado su apoyo tanto en los momentos alegres como en los difíciles.

En forma especial deseo agradecer a mi profesor guía, Hugh Rudnick, por su apoyo y orientación durante el desarrollo de este trabajo.

Deseo agradecer también al resto de la Comisión, quienes generaron un ambiente idóneo para el desarrollo de este trabajo a través de sus críticas siempre constructivas y acertados comentarios.

Finalmente, deseo dar las gracias a todas las personas que de alguna u otra forma han colaborado con sus ideas, sabios consejos o simplemente brindándome su apoyo durante todo el transcurso de mi carrera y especialmente en el desarrollo de esta Memoria.

ÍNDICE GENERAL

	Pág.
DEDICATORIA	ii
AGRADECIMIENTOS	iii
ÍNDICE DE TABLAS	vi
ÍNDICE DE FIGURAS.....	ix
RESUMEN.....	xii
I. INTRODUCCIÓN.....	1
1.1 Generalidades.....	1
1.2 Planteamiento del Problema.....	5
1.3 Revisión Bibliográfica	7
1.4 Objetivos de la Memoria.....	11
1.5 Estructura del Trabajo.....	12
II. ANÁLISIS TEÓRICO DEL MODELO	14
2.1 Características de los Sistemas de Distribución.....	14
2.2 Planteamiento del Problema de Optimización	15
2.2.1 Función de Costos.....	15
2.2.2 Restricciones Operacionales.....	18
2.2.3 Restricciones Técnicas.....	19
2.2.4 Formulación Global del Problema.....	20
2.3 Análisis de Sensibilidad	20
III. DESCRIPCIÓN DE MÓDULOS COMPONENTES DEL PROGRAMA	23
3.1 Lectura de Archivo de Datos.....	23
3.2 Parámetros e Inicialización de Variables	23
3.3 Análisis de Sensibilidad	24
3.4 Codificación Genética.....	24
3.5 Generación de Población Base.....	25
3.6 Decodificación de Individuos	25

3.7	Evaluación.....	26
3.8	Aplicación del Algoritmo Genético	27
3.9	Generación de la Nueva Población Base	28
3.10	Heurística Basada en la Sensibilidad	29
3.11	Resultado Final.....	31
IV.	PLANTEAMIENTO DE METODOLOGÍA DE SOLUCIÓN VIA ALGORITMOS GENÉTICOS.....	32
4.1	Introducción	32
4.2	Computación Evolutiva.....	33
4.3	Raíces de la Teoría de la Evolución.....	34
4.4	Algoritmos Genéticos Aplicados a los Sistemas de Distribución.....	36
4.4.1	Ventaja de los Algoritmos Genéticos frente a otros Métodos	37
4.4.2	AG en el Diseño de Sistema de Distribución multiperíodos	38
4.4.3	AG aplicados al Diseño de grandes Sistemas de Distribución	41
4.5	Método de Nichos para Algoritmos Genéticos	43
4.5.1	Algoritmos Genéticos	45
4.5.2	Métodos de Nichos	51
4.5.3	Clasificación de los Algoritmos de Nichos	52
4.5.4	Un Marco Formal para Escenarios Multimodales	54
4.5.5	Marco Analítico	56
4.5.6	Marco Empírico	61
4.5.7	Poblamiento Determinístico	69
V.	APLICACIÓN	79
5.1	Introducción	79
5.2	Estructura del Programa	80
5.3	Validación del Modelo.....	83
5.3.1	Sistema de Tres Barras	84
5.3.2	Sistema de Nueve Barras	89
5.3.3	Sistema de Treinta Barras.....	95
5.3.4	Conclusiones de los Sistemas de Prueba	101
VI.	CONCLUSIONES GENERALES	103

VII. DESARROLLOS FUTUROS	106
BIBLIOGRAFÍA.....	108
ANEXOS.....	117
ANEXO A: Teoría de los Algoritmos Genéticos	118
ANEXO B: Clasificación de los Algoritmos Genéticos.....	126
ANEXO C: Operadores Genéticos	157
ANEXO D: Manual de Uso del Programa	160
ANEXO E: Datos Sistema de 9 Barras	176
ANEXO F: Datos Sistema de 30 Barras.....	178

ÍNDICE DE TABLAS

	Pág.
Tabla 4.1: Clasificación de los Métodos de Nichos	54
Tabla 4.2: Clasificación de los Problemas	62
Tabla 5.1: Información General del Sistema de Tres Barras	84
Tabla 5.2: Parámetros AG, Sistema de 3 Barras	86
Tabla 5.3: Compensación óptima del Sistema de Tres Barras.....	86
Tabla 5.4: Pérdidas de Energía del Período analizado para el Sistema de 3 Barras ..	87
Tabla 5.5: Comparación de Tensiones, Sistema de Tres Barras	87
Tabla 5.6: Datos Sistema 9 Barras	89
Tabla 5.7: Parámetros AG, Sistema de 9 Barras	90
Tabla 5.8: Ahorro del sistema de 9 Barras compensado	91
Tabla 5.9: Compensación óptima para el Sistema de 9 Barras	91
Tabla 5.10: Compensación óptima Sistema de 9 Barras ($K_c=5$)	92
Tabla 5.11: Costos Totales Sistema de 9 Barras v/s n° Generaciones	94
Tabla 5.12: Datos del Sistema de 30 Barras	96
Tabla 5.13: Parámetros AG, Sistema de 30 Barras	97
Tabla 5.14: Ahorro del Sistema de 30 Barras Compensado	98
Tabla 5.15: Compensación óptima para Sistema de 30 Barras	99
Tabla 5.16: Localización de Condensadores en Sistema de 30 Barras	100

Tabla A.1: Clasificación de Métodos de Diversificación	123
Tabla B.1: Clasificación de los Métodos de Nichos	127
Tabla E.1: Impedancias del Sistema de 9 Barras	176
Tabla E.2: Cargas del Sistema de 9 Barras	176
Tabla E.3: Tensiones para Sistema de 9 Barras Compensado	177
Tabla F.1: Impedancias del Sistema de 30 Barras	178
Tabla F.2: Cargas del Sistema de 30 Barras	179

ÍNDICE DE FIGURAS

	Pág.
Figura 2.1: Discretización en Niveles de Carga.....	17
Figura 3.1: Modelo de Codificación	25
Figura 4.1: Estructura del ADN	35
Figura 4.2: Mitosis	36
Figura 4.3: Problema del Vendedor Viajero	47
Figura 4.4: Roulette Wheel Selection	49
Figura 4.5: Función de Prueba M1.....	63
Figura 4.6: Función de Prueba M2.....	64
Figura 4.7: Función de Prueba M3.....	65
Figura 4.8: Función de Prueba M4.....	66
Figura 4.9: Función de Himmelblau	67
Figura 4.10: Función "Shekel's Foxholes"	69
Figura 4.11: Resultado función de Prueba M1	75
Figura 4.12: Resultado función de Prueba M2	76
Figura 4.13: Resultado función de Prueba M3	76
Figura 4.14: Resultado función de Prueba M4	77
Figura 4.15: Pseudocódigo de "Poblamiento Determinístico"	78
Figura 5.1: Estructura del Programa	80

Figura 5.2: Diagrama de Bloques del Programa	81
Figura 5.3: Sistema de Tres Barras	84
Figura 5.4: Niveles de Carga Discretizados.....	85
Figura 5.5: Costo Condensadores	85
Figura 5.5: Convergencia AG, Sistema de 3 Barras	88
Figura 5.6: Sistema de 9 Barras	89
Figura 5.9: Convergencia AG, Sistema de 9 Barras	93
Figura 5.10: Composición de Costos Totales v/s n° de Generaciones.....	94
Figura 5.11: Sistema de 30 Barras	95
Figura 5.12: Convergencia AG, Sistema de 30 Barras	101
Figura D.1: Pantalla del Menú Principal.....	162
Figura D.2: Menú Principal	164
Figura D.3: Menús del Programa	164
Figura D.4: Abrir Base de datos.....	166
Figura D.5: Configuración Sistema.....	167
Figura D.6: Configuración de Líneas del Sistema	167
Figura D.7: Configuración de Subestaciones.....	168
Figura D.8: Configuración Nivel de Carga Base	168
Figura D.9: Configuración de Condensadores	169
Figura D.10: Configuración de Niveles de Carga.....	169

Figura D.11: Guardando Modelo	171
Figura D.12: Parámetros del Modelo	172
Figura D.13: Configuración AG	173
Figura D.14: Optimización del Sistema	174
Figura D.15: Referencias	175

RESUMEN

En este trabajo se ha desarrollado una metodología basada en Algoritmos Genéticos con el fin de determinar el tamaño, ubicación, tipo y número de condensadores a ser instalados en un sistema de distribución radial de topología conocida e invariante en el tiempo. El objetivo es minimizar las pérdidas de potencia de punta y las pérdidas de energía en un sistema de distribución considerando el costo de los condensadores.

Otro objetivo importante alcanzado a través de este trabajo es el desarrollo de una metodología de optimización de propósito general, altamente robusta que puede ser utilizada para resolver problemas complejos y difíciles de modelar por medio de los métodos tradicionales de optimización.

A partir de las ecuaciones de flujo de potencia y pérdidas totales se deriva un método basado en un análisis de sensibilidad ante la inyección de potencia reactiva para determinar localizaciones candidatas a ser compensadas.

Posteriormente, se ha desarrollado un Algoritmo Genético, capaz de encontrar el óptimo global de una función, a través de una técnica basada en la formación de nichos en la población. Este algoritmo es evaluado en diversas funciones de prueba para comprobar su robustez. Con el propósito de mejorar la convergencia del Algoritmo Genético off-line, se han implementado dos heurísticas basadas en el análisis de sensibilidad para relocalizar condensadores y localizar nuevos condensadores en el sistema de distribución.

La metodología se valida mediante su aplicación a tres sistemas de prueba que han sido analizados en el pasado por diversos autores, con el fin de establecer una comparación cualitativa y cuantitativa de los resultados obtenidos.

Finalmente, para hacer esta herramienta más atractiva en la planificación de sistemas de distribución, se ha desarrollado una interfaz gráfica que permite al usuario ingresar la información en forma sencilla, a través de un conjunto de menús y submenús, evitando que éste deba interiorizarse sobre la estructura interna de la información.

I INTRODUCCIÓN

1.1 Generalidades

El abastecimiento eléctrico se realiza a través de tres etapas, generación, transmisión y distribución, siendo esta última la encargada de llevar la energía eléctrica a los consumidores finales en forma eficiente y bajo estándares de calidad de servicio. Este proceso, realizado a través de redes de alimentación y subestaciones, debe ser continuamente modificado con el fin de mantener el sistema permanentemente adaptado producto de los constantes cambios de la demanda.

La planificación de la expansión de los sistemas de distribución eléctricos en el corto plazo es una tarea altamente compleja, que involucra altos costos de inversión y una gran diversidad de alternativas. Es por ello que en los últimos años se ha ido poniendo cada vez más énfasis en la necesidad de implementar esquemas de automatización en el área de la distribución de energía eléctrica, tanto para sistemas pequeños como de gran envergadura.

Es bien sabido que el negocio de la distribución eléctrica presenta economías de ámbito, o más bien economías de densidad, lo que equivale a decir que mientras más poblada es el área de concesión de la empresa, los costos de la firma por servir a un cliente disminuyen. Esto explica la existencia de un monopolio natural.

En el reglamento eléctrico vigente en Chile se establece que los distribuidores tienen la obligación de dar suministro a quien lo solicite en su zona de concesión, entregar un nivel mínimo de calidad de servicio y evaluar su cumplimiento, permitir el uso de líneas eléctricas para el transporte de energía de cualquier generador e interconectarse con otros concesionarios y por último mantener instalaciones seguras.

En la actualidad los concesionarios de servicio público de distribución son los responsables de que los estándares y normas de calidad de servicio sean cumplidos.

La calidad de servicio se define, en el reglamento vigente, como el conjunto de propiedades y estándares normales que son inherentes a la actividad de distribución de electricidad, y que constituyen las condiciones bajo las cuales dicha actividad debe desarrollarse.

Entre los parámetros incluidos en la calidad de servicio figuran:

- La seguridad de las instalaciones y de su operación, y el mantenimiento de las mismas,
- La satisfacción oportuna de las solicitudes de servicio, en los términos y condiciones establecidos en este reglamento,
- Correcta medición y facturación de los servicios prestados, y el oportuno envío a los usuarios y clientes,
- La utilización de adecuados sistemas de atención e información a los usuarios y clientes,
- Los estándares de calidad de suministro.

Respecto a este último punto, se debe aclarar, que por calidad de suministro se entiende el conjunto de parámetros físicos y técnicos, que conforme a la ley, al reglamento y las normas técnicas pertinentes debe cumplir el producto electricidad. Dichos parámetros son entre otros, tensión, frecuencia y disponibilidad.

Aunque la actual ley eléctrica no lo define explícitamente deja abierta la posibilidad para que se desarrolle la actividad de comercialización a través de empresas que compren energía y potencia en el mercado mayorista, y venda esa energía y potencia a usuarios finales regulados. En proyectos de reforma de la ley se introduce esta actividad de comercialización y las empresas asociadas. Para poder

transmitir esa energía y potencia, la empresa de comercialización deberá pagar un cierto peaje a la empresa propietaria de las redes de distribución por la prestación del servicio de alambres. Esto permitirá tener dos actividades separadas logrando de esta forma que sólo la actividad de distribución sea sometida a regulación, ya que es la porción no competitiva, debido a su condición de monopolio natural. Mientras que la actividad de comercialización, puede ser dejada en las manos de las fuerzas libres del mercado.

La principal actividad desarrollada por las empresas de comercialización es suministrar energía eléctrica a los usuarios finales regulados y los servicios de conexión de nuevos usuarios, compra, instalación, mantención y lectura de los equipos de medida, además del corte y reposición de suministro.

En este contexto la empresa de distribución, dueña de los alambres, será la encargada de llevar la electricidad desde las subestaciones hasta el punto deseado por el comercializador. Para ello la empresa de distribución cobrará una indemnización o peaje a las empresas de comercialización por la utilización de sus redes. Esta empresa también será la responsable de la operación del sistema distribución en forma eficiente y cumpliendo con los estándares de calidad.

Por razones de eficiencia en cada zona de concesión debe existir una sola empresa de distribución, ya que de lo contrario se produciría superposición de redes y equipos lo que llevaría por consiguiente a una duplicidad de costos, desaprovechándose las economías de ámbito o densidad en distribución.

Un tópico de gran relevancia en esta área lo constituye el control de la potencia reactiva a través de la adecuada localización de condensadores en el sistema de distribución. En los siguientes capítulos se desarrolla una técnica basada en los

Algoritmos Genéticos para determinar la ubicación, tamaño y clase de condensadores que se deben utilizar en el sistema.

Este estudio considera sistemas de distribución eléctrica preexistentes y escenarios estáticos, es decir, la aplicación desarrollada permite establecer la configuración de condensadores más adecuada a instalar en una red predefinida y cuya topología permanece invariante en el tiempo.

1.2 Planteamiento del Problema

Dentro del contexto actual, la distribución automatizada de energía es vista por la industria eléctrica como parte integrante de un sistema mayor que comprende elementos como el control automático de la generación, sistema de protección, control y monitoreo de subestaciones y líneas, administración de la carga etc. Los factores que incentivan este desarrollo son esencialmente económicos y están asociados al énfasis que se pone hoy día en lograr un uso eficiente de las instalaciones disponibles y aplazar mientras sea posible y económicamente justificable la realización de nuevas inversiones. Adicionalmente, ha aumentado la necesidad de ofrecer un suministro más seguro de energía así como mejorar las técnicas de control de los sistemas y la calidad del servicio eléctrico ofrecido.

Por otra parte, es universalmente conocido el hecho que el control de voltaje y potencia reactiva juegan un rol importante dentro de los esquemas de la distribución automatizada. La reducción de las pérdidas de potencia y energía es un tópico de interés para toda la empresa eléctrica, independiente del punto de la cadena generación-consumo final en que se encuentre. Desarrollos en distribución automatizada vía computadoras en las subestaciones y sistemas de recolección de información detallada sobre el sistema (SCADA), permiten reducir en forma efectiva las pérdidas que se producen en el sistema, mediante el control en tiempo real del perfil de voltajes y el flujo de corrientes reactivas a lo largo del sistema.

El uso de bancos de condensadores fijos y on/off permite reducir pérdidas de potencia y a su vez, mejorar el perfil de voltajes, especialmente cuando los condensadores son controlados de manera que pueden responder en forma diaria, semanal o con cualquier otra periodicidad a los cambios de carga reactiva que se producen en el sistema. Cuando la potencia reactiva es entregada únicamente por la planta de generación, cada uno de los componentes del sistema (es decir, generadores,

transformadores, líneas de distribución y transmisión, equipo de protección) deben ser dimensionados para dichos niveles de carga reactiva.

Los condensadores flexibilizan esta necesidad debido a la disminución de la demanda de potencia reactiva que enfrentan los generadores, al reducir las corrientes de línea desde el condensador hasta el equipo de generación. Como resultado se consigue una reducción de pérdidas y carga en líneas de distribución y transmisión, así como también en transformadores de subestaciones.

El nivel de compensación está estrechamente ligado a la localización de condensadores en el sistema de distribución, lo que esencialmente significa determinar la ubicación, tamaño, número y tipo de condensadores que van a ser utilizados en el sistema.

Resumiendo los beneficios económicos que se pueden obtener a partir de la instalación de bancos de condensadores son los siguientes:

- a) Capacidad liberada de las subestaciones; se requiere inyectar menos potencia reactiva al alimentador, así los transformadores de potencia ubicados en la subestación se ven liberados en un cierto margen.
- b) Capacidad liberada en el alimentador; existen menos exigencias térmicas sobre las líneas como consecuencia de la reducción de las corrientes, y por lo tanto se dispone de una mayor capacidad de transmisión de potencia.
- c) Mejor perfil de tensiones, lo que se refleja en un mejor servicio al cliente.
- d) Capacidad liberada en generación; al disminuir las pérdidas queda un excedente de energía disponible.
- e) Reducción de las pérdidas de capital que significarían perfeccionar o expandir el sistema.

En este trabajo se desarrolla una metodología basada en Algoritmos Genéticos, con la que se encontrarán configuraciones óptimas de localización de condensadores en un sistema de distribución. Esta técnica está inspirada en la teoría de la evolución de las especies de Charles Darwin, que centra su atención en la búsqueda del óptimo a través de la selección natural de individuos.

1.3 Revisión Bibliográfica

La problemática de localizar en forma óptima condensadores en un sistema de distribución es un tópico que ha sido estudiado con bastante énfasis por diversos autores en el pasado.

Los estudios anteriores difieren entre sí en la formulación del problema y el método utilizado para resolverlo. En la mayoría de estos desarrollos la función objetivo ha sido modelada como una maximización de los ahorros producto de las pérdidas de energía y de potencia de punta considerando además el costo de instalación y mantenimiento de los condensadores.

Las ventajas de poder variar la capacidad de compensación de un banco son reconocidas desde los años 1940.

Ponnaivaikko y Prakasa Rao (1983, [1]), Fawzi (1983, [2]) incluyeron la liberación de capacidad del sistema y el crecimiento de la carga en su formulación.

Cuttino (1944; 1947, [3,4]), por ejemplo, propone esquemas que utilizan combinaciones de condensadores de tap discreto, condensadores síncronos y transformadores con control de derivaciones, con el fin de obtener un control efectivo de pérdidas de potencia y caídas de voltaje.

A principios de la década de los '80, los autores Grainger y Lee modelaron el problema de control para el caso de bancos de condensadores de conexión on/off (1981;1982, [5,6]).

Algunos de los autores posteriores han planteado el problema como uno de optimización con restricciones donde han considerado restricciones de voltaje.

Una amplia variedad de metodologías de solución ha sido propuesta para resolver este problema. La mayoría de los trabajos anteriores utilizan métodos analíticos con alguna clase de heurística. De esta forma algunas metodologías simplificadas realizan supuestos que conllevan pérdidas de generalidad.

Ponnaivaikko y Prakasa Rao (1983, [1]) consideran factores adicionales y restricciones para el aumento de tensión y resuelven el problema utilizando el método de las variables locales.

Kaplan (1984, [7]) reformula el problema de los trabajos anteriores en lo que respecta a la localización, dimensionamiento y control de bancos de condensadores de switch y utiliza heurísticas para resolverlo.

Grainger et. al. formuló el problema como uno de programación no lineal considerando la ubicación y el tamaño de los condensadores como variables continuas y propuso un procedimiento iterativo simple basado en el método del gradiente.

Grainger y Lee (1981, [8]) desarrollaron un procedimiento para diseñar condensadores fijos y posteriormente introdujeron condensadores desconectables y consideraron la capacidad libre del sistema. También diseñaron un esquema para el control continuo de la compensación. En conjunto con El-Kib propusieron un

esquema para determinar el diseño y control óptimo de condensadores desconectables.

Por su parte El-Kib (1984, [9]) desarrolló un método que requiere de la conexión conjunta de grupos de condensadores. Más aún, para el caso de un número considerable de condensadores, el método propuesto requiere de un monto considerable de cálculos, lo que lo hace poco práctico.

El-Kib (1985, [10]) también propuso una metodología para diseñar condensadores para alimentadores desbalanceados.

Grainger y Civanlar (1985, [11]) consideraron la regulación de tensión y la localización de condensadores como un problema conjunto y presentaron un método que permite desacoplar el problema para su resolución.

Brown (1981, [12]) y Desai (1982, [13]) realizaron un análisis de las diversas fuentes que influyen en la potencia reactiva en un sistema de distribución.

Baran y Wu (1989, [14,15]) formularon el problema en forma análoga a Grainger pero consideraron el tamaño de los condensadores como una variable discreta y establecieron restricciones de tensión confluyendo en un problema de programación entera. Utilizaron la técnica de descomposición de fase I y fase II para resolver el problema. También incluyeron las ecuaciones de flujo de potencia en su modelo.

Chiang (1990, [16]) propuso una formulación similar a la de Baran y Wu pero además incluyó restricciones de carga adicionales en su modelo. Utilizó el método de Simulated Annealing con la capacidad de alcanzar el óptimo global en una función objetivo.

Un método de optimización es juzgado por su eficiencia, lo que se refleja en la calidad de la solución final y en el número de veces que debe ser evaluada la función objetivo para alcanzar una solución con una determinada calidad.

Considerando una base teórica, todos estos métodos, excepto, Simulated Annealing, entregan solamente soluciones que corresponden a óptimos locales consumiendo requerimientos computacionales bajos. El método de Simulated Annealing tiene la capacidad de encontrar soluciones cercanas al óptimo global pero los requerimientos computacionales son elevados.

La metodología basada en Algoritmos Genéticos propuesta en este estudio es capaz, por si sola, de determinar la solución correspondiente al óptimo global de una función considerando requerimientos computacionales razonables. Este algoritmo se denomina Algoritmo Genético con Nichos y ha sido probado en diversas funciones de prueba con diferentes niveles de dificultad. Finalmente, ha sido aplicado exitosamente en la resolución del problema de localización óptima de condensadores en un sistema de distribución eléctrico.

1.4 Objetivos de la Memoria

El problema general de localización de condensadores debe entregar como resultado el tipo, tamaño y número de condensadores a instalar en un sistema de distribución radial y el esquema de control de estos condensadores para diferentes niveles de carga.

El objetivo que se persigue a través de este estudio es reducir las pérdidas de energía y de potencia de un sistema de distribución para diferentes niveles de carga minimizando el costo total de instalación y mantenimiento de los condensadores, manteniendo las magnitudes de tensión dentro de ciertos límites preestablecidos para garantizar un nivel adecuado de calidad de servicio. En el desarrollo de este estudio se consideran sistemas de distribución cuya topología permanezca invariante en el tiempo.

Los objetivos puntuales a seguir son:

- Plantear y desarrollar una metodología de solución para el problema de localización de condensadores en sistemas de distribución basado en los Algoritmos Genéticos.
- Implementar un modelo computacional en un lenguaje determinado, Visual Fortran '90, que represente adecuadamente el problema de localización de condensadores y que permita obtener soluciones frente a diversos escenarios.
- Validar resultados con relación a los obtenidos mediante herramientas alternativas para un mismo escenario y para nuevos problemas con el fin de comprobar la robustez y flexibilidad del modelo.

1.5 Estructura del Trabajo

En el primer capítulo se entrega una descripción general de los sistemas de distribución eléctricos y de la importancia que posee, tanto del punto de vista técnico como económico, una adecuada compensación (capacitiva) del sistema. Se abordan las causas, consecuencias y en particular los beneficios que conlleva la instalación de bancos de condensadores en diversos puntos de la red. Además, se presentan los objetivos del presente trabajo y finalmente una breve revisión bibliográfica.

En el segundo capítulo se presenta la modelación teórica propuesta para resolver el problema de localización óptima de condensadores en un sistema de distribución y conjuntamente se describen las principales componentes y variables involucradas.

En el tercer capítulo se describen el funcionamiento del programa y sus módulos componentes. Se detallan sus funciones principales y las heurísticas utilizadas para relocalizar y localizar condensadores con el propósito de mejorar aún más la solución entregada por el AG.

En el cuarto capítulo se entrega una reseña histórica sobre la evolución en sistemas biológicos, que constituyen la base de inspiración de los modelos de algoritmos evolutivos y se describe el desarrollo realizado sobre algoritmos genéticos con nichos. Además, se presenta una descripción sobre las aplicaciones de AG en los sistemas de distribución.

En el quinto capítulo se valida la metodología desarrollada como medio de resolución de problemas de localización de condensadores en sistemas de distribución. Para ello se estudian tres sistemas de distribución y se comparan los resultados obtenidos a través del programa con otras metodologías existentes.

Las conclusiones globales son tratadas en el sexto capítulo.

Finalmente, en el séptimo capítulo se incluyen además, una serie de comentarios para futuros desarrollos tanto para el modelo propuesto como para la técnica de resolución.

II ANÁLISIS TEÓRICO DEL MODELO

2.1 Características de los Sistemas de Distribución

Los sistemas de distribución presentan un conjunto de características particulares:

- Topologías radiales
- Diversas y múltiples conexiones
- Cargas de distinta naturaleza
- Líneas sin transposiciones

El que típicamente sean radiales significa que el flujo de potencia nace en un sólo nudo. Ese nudo principal corresponde generalmente a la subestación que alimenta a toda la red a nivel de media tensión (12,15,23 [kV]). Esta característica presenta la ventaja de simplificar la operación del sistema de distribución y además facilita el despeje de fallas. En la actualidad, sin embargo, se están comenzando a implementar configuraciones de alimentadores con forma de anillo con el fin de aumentar la seguridad del sistema, a través de un mejor nivel de respaldo.

Se pueden encontrar otras conexiones además de la trifásica, a saber, bifásica y monofásica dependiendo del sistema, pero comúnmente son las menos.

Otro aspecto que se debe mencionar es la presencia de cargas de distinta naturaleza (residenciales, comerciales, alumbrado público, industriales y agroindustriales) las que definen un factor de potencia típico y un determinado comportamiento frente a variaciones de tensión.

En cuanto a los valores característicos de la impedancia de línea, no puede despreciarse la resistencia frente a la reactancia como sucede típicamente en los

algoritmos que trabajan con parámetros de líneas de transmisión. En sistemas de distribución la resistencia es mayor que la reactancia.

Basados en lo anterior, en la formulación y solución del modelo que sigue se harán los siguientes supuestos:

- La red constituye un sistema trifásico balanceado, de manera que es aplicable una representación por fase.
- No existen interconexiones entre las diferentes ramas del sistema (sistema radial).
- Todas las cargas varían de acuerdo a una curva de duración definida.
- El valor de la resistencia de línea puede ser mayor o comparable a la reactancia.

2.2 Planteamiento del Problema de Optimización

El problema de localización de condensadores consiste en la determinación de la ubicación, número, tipo y tamaño de los bancos de condensadores que van a ser instalados en el sistema de distribución en forma óptima.

2.2.1 Función de Costos

En esta sección se desarrolla la formulación del problema de localización óptima de condensadores y su control en un sistema de distribución considerando además, restricciones de carga, restricciones de la red y restricciones operacionales. La función de costo incluye tanto los costos asociados a las pérdidas de potencia real y energía como también el costo de instalación de condensadores.

La función de costos está compuesta por tres términos: el costo de los condensadores que incluye su costo de instalación, el costo total de las pérdidas de

energía en el sistema de distribución y el costo total de las pérdidas de potencia. El ahorro total de la red es la diferencia entre los costos totales antes de la compensación del sistema y los costos totales después de haberla realizado.

a) Costo de los condensadores

El costo fijo de los condensadores está compuesto por dos términos: el costo del equipo y el costo de instalación de los condensadores. Este costo generalmente depende del tamaño del banco de condensadores. De esta forma si C_j es el tamaño de un banco de condensadores localizado en el nodo j y K_c es el costo del kVar incluyéndose en éste el costo de instalación, el costo fijo total viene dado por la siguiente expresión:

$$C_{cond} = K_c \sum_{j=1}^M C_j \quad 2.1$$

b) Costo de las Pérdidas de Energía

Para calcular las pérdidas de energía del sistema, ha sido necesario considerar las variaciones de carga durante un determinado período de tiempo (T). Se ha hecho el supuesto que las variaciones de carga se puedan aproximar a través de niveles discretos y que toda la carga varía en forma proporcional según el nivel de carga que se trate. Considerando este supuesto, la curva de duración de carga está compuesta por segmentos lineales y el período de tiempo T es dividido en intervalos durante los cuales los niveles de carga son constantes. Esta representación se muestra en la figura 2.1.

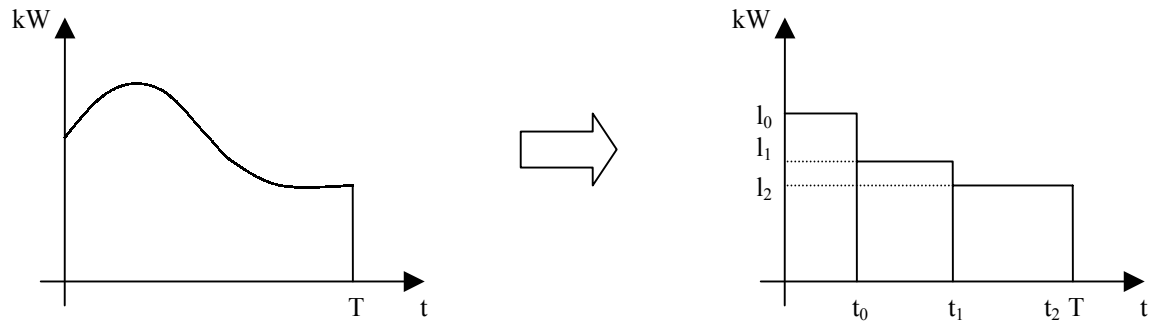


Figura 2.1: Discretización en Niveles de Carga

Además de lo mencionado anteriormente se desprende que los niveles de carga son proporcionales entre sí: $l_0 \propto l_1$, $l_0 \propto l_2$, $l_1 \propto l_2$.

Si P_i corresponde a las pérdidas totales en cualquier nivel de carga i con un tiempo de duración T_i , las pérdidas totales de energía vienen dada por la siguiente expresión:

$$E_{loss} = \sum_{i=1}^n T_i P_i \quad 2.2$$

El costo de las pérdidas de energía se calcula entonces, multiplicando las pérdidas de energía total por el factor de conversión de kWh a pesos, K_e .

$$C_{E-loss} = K_e \sum_{i=1}^n T_i P_i \quad 2.3$$

c) Costo de las Pérdidas de Potencia

Existe otro efecto de las pérdidas que corresponde a la ganancia marginal de la capacidad de distribución, como resultado de la compensación reactiva. En este modelo esto se calcula como función del valor máximo de las pérdidas (P_0) obtenidas para una configuración dada, en el valor peak de la curva de carga. Si K_p corresponde al factor de conversión de kW a pesos, entonces la siguiente expresión refleja los costos incurridos debido a la existencia de pérdidas de potencia:

$$C_{P-loss} = K_p \cdot P_0 \quad 2.4$$

d) Costo Total

El costo total viene dado por la suma de las tres expresiones anteriores, es decir, por los costos asociados a las pérdidas de potencia y energía, y por el costo de los condensadores que incluye además el costo de instalación de estos equipos.

$$C_{Total} = C_{Cond} + C_{E-loss} + C_{P-loss} \quad 2.5$$

2.2.2 Restricciones Operacionales

Restricciones de carga son expresadas en términos de las restricciones del flujo de potencia y las restricciones operacionales son expresadas en términos de las restricciones de magnitud de voltaje.

a) Restricciones de Flujo de Potencia

Dado un sistema de distribución trifásico balanceado, el flujo de las corrientes y voltajes en los nodos es gobernado por las leyes eléctricas. En una forma compacta, la restricción de flujo de potencia en un nivel cualquiera de carga i se puede modelar como:

$$F_i(x) = 0 \quad ; \quad i = 1, 2, \dots, n \quad 2.6$$

donde, x es el vector que contiene la magnitud y ángulo del voltaje de cada nodo del sistema.

b) Restricciones de Magnitud de Voltaje

Para redes de distribución bien diseñadas, cambios de las condiciones desbalanceadas de carga y de la demanda, sólo causan cambios sutiles en la variación de los voltajes. Por lo tanto, para un sistema debidamente equilibrado las restricciones de magnitud de voltaje pueden ser consideradas independientes del nivel de carga i , pudiendo ser establecidas de la siguiente forma:

$$V_{\min} \leq V_l \leq V_{\max} \quad ; \quad l = 1, 2, \dots, z \quad 2.7$$

donde l corresponde al número del nodo y z al número máximo de nodos en el sistema, mientras que V_{\min} y V_{\max} corresponden a la tensión mínima y máxima admisible respectivamente. De esta forma se garantiza que la tensión en cada una de las barras que conforman el sistema se va a encontrar dentro de ciertos límites preestablecidos.

2.2.3 Restricciones Técnicas Adicionales

a) Restricción de cantidad de condensadores

En muchas ocasiones las compañías experimentan limitaciones del número de equipos. Esta restricción está representada a través del siguiente conjunto de desigualdades:

$$C_j \leq r_j \quad ; \quad r_j > 0 \quad 2.8$$

donde C_j es un condensador de capacidad j y r es el número máximo de unidades disponibles.

2.2.4 Formulación Global del Problema

El problema global se puede definir como el de determinar óptimamente la localización para instalar condensadores, los tipos y tamaños de los condensadores a ser instalados y, durante cada nivel de carga, el esquema de control de estos equipos, sin perder de vista el objetivo de minimizar las pérdidas de energía y de potencia junto con el costo de los condensadores. Considerando que hay n niveles de carga y M posibles localizaciones de condensadores, la formulación del problema es la siguiente:

$$\min K_e \sum_{i=1}^n T_i P_i + K_p P_0 + K_c \sum_{j=1}^M C_j \quad 2.9$$

Sujeto a:

$$F_i(x) = 0 \quad ; \quad i = 1, 2, \dots, n$$

$$V_{\min} \leq V_l \leq V_{\max} \quad ; \quad l = 1, 2, \dots, z$$

$$C_j \leq r_j \quad ; \quad r_j > 0$$

donde K_e , K_p , K_c son constantes que representan los costos de energía, de potencia de punta y de kVAr respectivamente. P_i corresponde a las pérdidas de potencia para cualquier nivel de carga i , mientras que P_0 corresponde a las pérdidas de potencia de punta.

2.3 Análisis de Sensibilidad

Un análisis de sensibilidad es utilizado para determinar las barras que se van a compensar. En estudios anteriores se han definido las posibles localizaciones de los condensadores a través de heurísticas o basándose en la experiencia de los ingenieros.

La predeterminación de la ubicación de los condensadores sirve para disminuir el espacio de búsqueda del algoritmo de optimización, y por otra parte, también se consigue un ahorro significativo del tiempo empleado para encontrar soluciones óptimas.

El análisis de sensibilidad es un procedimiento sistemático para seleccionar ubicaciones o barras que presenten un mayor impacto sobre las pérdidas de potencia activa respecto a una inyección de potencia reactiva en dicho nodo.

En este contexto, las pérdidas de potencia compleja vienen dadas por:

$$P_L + jQ_L = VI^* = VY^*V^* \quad 2.10$$

a partir de esta ecuación se pueden expresar las pérdidas de potencia activa como:

$$P_L = \sum_i \sum_j V_i V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) \quad 2.11$$

donde G_{ij} y B_{ij} son la conductancia y la susceptancia de la línea entre los nodos i y j , mientras que $\theta_{ij} = \theta_i - \theta_j$ es la diferencia de ángulos entre las tensiones de las barras i y j .

Las pérdidas incrementales vienen expresadas por el siguiente sistema de ecuaciones:

$$\begin{bmatrix} \frac{\partial P_L}{\partial P} \\ \frac{\partial P_L}{\partial Q} \end{bmatrix} = \begin{bmatrix} \frac{\partial \theta}{\partial P} & \frac{\partial V}{\partial P} \\ \frac{\partial \theta}{\partial Q} & \frac{\partial V}{\partial Q} \end{bmatrix} \begin{bmatrix} \frac{\partial P_L}{\partial \theta} \\ \frac{\partial P_L}{\partial V} \end{bmatrix} \quad 2.12$$

en que la relación entre las pérdidas de potencia activa y una inyección de potencia reactiva viene dada por la siguiente expresión:

$$\frac{\partial P_L}{\partial Q} = \frac{\partial \theta}{\partial Q} \frac{\partial P_L}{\partial \theta} + \frac{\partial V}{\partial Q} \frac{\partial P_L}{\partial V} \quad 2.13$$

en donde a su vez:

$$\frac{\partial \theta}{\partial Q} = \frac{1}{\frac{\partial Q}{\partial \theta}} \quad y \quad \frac{\partial V}{\partial Q} = \frac{1}{\frac{\partial Q}{\partial V}} \quad 2.14$$

Además los elementos de la ecuación anterior se pueden expresar como:

$$\frac{\partial Q_i}{\partial \theta_j} = -V_i V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) \quad ; \quad i \neq j \quad 2.15$$

$$\frac{\partial Q_i}{\partial \theta_j} = -G_{ii} V_i^2 + P_i \quad ; \quad i = j \quad 2.16$$

$$\frac{\partial Q_i}{\partial V_j} = V_i (G_{ij} \sin \theta_{ij} + B_{ij} \cos \theta_{ij}) \quad ; \quad i \neq j \quad 2.17$$

$$\frac{\partial Q_i}{\partial V_j} = -B_{ii} V_i + \frac{Q_i}{V_i} \quad ; \quad i = j \quad 2.18$$

y finalmente derivando la ecuación 2.11 se pueden obtener los términos:

$$\frac{\partial P_L}{\partial \theta_i} = -2V_i \sum_j V_j (G_{ij} \sin \theta_{ij}) \quad 2.19$$

$$\frac{\partial P_L}{\partial V_i} = 2 \sum_j V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) \quad 2.20$$

Utilizando las ecuaciones anteriores se pueden ordenar las barras de acuerdo a su sensibilidad ante una inyección de potencia reactiva.

III. DESCRIPCIÓN DE LOS MÓDULOS COMPONENTES DEL PROGRAMA

En el siguiente acápite se describen los diversos módulos que componen el programa.

3.1 Lectura de Archivo de Datos

En el primer bloque se da lectura al archivo de datos que contiene la información del problema, almacenándola en diversas listas y matrices según los requerimientos del programa.

3.2 Parámetros e Inicialización de Variables

En este bloque se mencionan las variables que utiliza el modelo y los parámetros que son ingresados por el usuario.

Parámetros del Modelo de Distribución

En esta etapa son configurados los parámetros del sistema de distribución, adecuándolos a las exigencias del modelo. Se definen la tensión base, la tensión mínima y máxima admisible, la frecuencia del sistema, la cantidad y tamaño de condensadores con los que se cuenta, el costo asociado a la energía y a la potencia de punta como también el valor de los kvar instalados. En este punto también se define el número de localizaciones (o barras) con las que va a trabajar el Algoritmo Genético. Además, se define el número de localizaciones que va a ser considerado en la heurística de localización de condensadores.

Parámetros del Algoritmo

En esta sección se definen los parámetros que va a utilizar el Algoritmo Genético. Se definen el número de generaciones, la cantidad de individuos que conformarán cada población, la probabilidad de ocurrencia del Crossover, el porcentaje de la población que va a ser seleccionado por el operador de Elitismo, la probabilidad de ocurrencia de la mutación y el número de competidores que van a participar en el torneo de selección estocástico. Además, se debe elegir la variante de Algoritmo Genético que se desea utilizar. Existen tres posibilidades: Algoritmo Genético Clásico, Algoritmo Genético con SUS (Selección Estocástica Universal) o Algoritmo Genético con Nichos.

3.3 Análisis de Sensibilidad

La tercera etapa consiste en realizar el análisis de sensibilidad a través del cual se definen las barras cuyas pérdidas de potencia activa son más sensibles una inyección de potencia reactiva. Estas barras, entonces son las candidatas a ser compensadas.

Este análisis básicamente sirve para reducir el espacio de búsqueda de soluciones del algoritmo Genético.

3.4 Codificación Genética

La codificación genética es una etapa crucial dentro del Algoritmo ya que determina la forma en que se va a almacenar la información. En este caso se utilizó una codificación binaria y su estructura se detalla a continuación. Suponiendo que existe una barra seleccionada se deben incluir todos los niveles de carga (que

corresponden a segmentos discretos de la curva de duración) y en cada nivel de carga deben existir variables que representen cada uno de los tamaños de condensadores preestablecidos. Por ejemplo, si se han definido 3 tamaños de condensadores (C1, C2, C3) y 3 niveles de carga, además de un conjunto de localizaciones para instalarlos, la codificación se puede representar a través del siguiente diagrama:

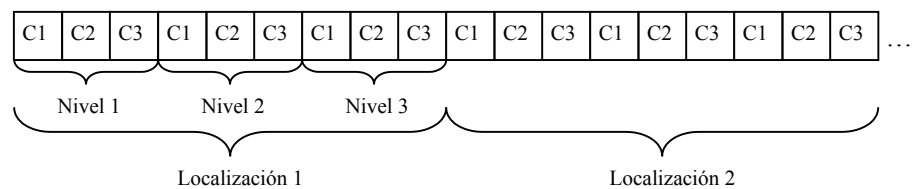


Figura 3.1: Modelo de Codificación

De esta manera un 1 en la primera casilla significa que existe un condensador de tamaño C1 activado en el nivel de carga 1 en la localización 1.

3.5 Generación de la Población Base

La población base es generada en forma aleatoria siguiendo una distribución normal.

3.6 Decodificación de los Individuos

En esta etapa, la información contenida en cada string (o individuo) debe ser decodificada a base decimal, de manera que la función de evaluación pueda interpretar los valores de los diversos parámetros incluidos en cada individuo.

3.7 Evaluación

El quinto bloque realiza el proceso de evaluación, el que se compone de dos partes, donde la primera corresponde al análisis técnico, y la segunda se refiere al análisis económico.

Análisis Técnico

Primeramente se corre un flujo de potencia que en este caso corresponde a un Newton-Raphson con el fin de obtener las tensiones en cada una de las barras, además de verificar, por supuesto, si la configuración en cuestión converge o no.

Enseguida se verifica la restricción de tensión, en la que se constata que ésta se mantiene en los rangos preestablecidos que garanticen una adecuada calidad de servicio.

Finalmente se verifica si el número de condensadores instalados no excede los indicados por el usuario.

Análisis Económico

Este consiste en determinar los costos asociados a la configuración actual del sistema. Se calculan en este punto los costos correspondientes a las pérdidas de energía del sistema y de potencia de punta como también los costos asociados a la instalación y la mantención de los condensadores. El fitness que se le asigna a cada individuo corresponde a la resta de todos estos costos a un número establecido por el programa como máximo (Big M).

Una vez que se ha obtenido el fitness que corresponde a un número real es adicionado al final de cada string como un índice de adaptabilidad que presenta cada individuo.

3.8 Aplicación del Algoritmo Genético

En este bloque se aplica el Algoritmo Genético a la población anterior, que consiste básicamente en someter esta población a los diversos operadores genéticos obteniéndose como resultado una nueva población de individuos con un mayor grado de adaptabilidad. Como ya se mencionó anteriormente existen tres variantes del Algoritmo Genético que se detallan a continuación:

Algoritmo Genético Clásico

Corresponde a un Algoritmo Genético que utiliza 4 operadores genéticos: selección, crossover, mutación y elitismo. El proceso de selección es a través de un torneo estocástico donde el usuario ingresa el número de competidores que desea que participen en el torneo. El *crossover* es del tipo multipunto, y el usuario debe indicar la cantidad de cortes que se le van a realizar a cada individuo y la probabilidad de que se produzca el crossover. Generalmente esta probabilidad toma un valor de 0.8. El proceso de mutación corresponde a un operador binario que intercambia un 1 por un 0, o viceversa, en forma aleatoria, dada una cierta probabilidad de ocurrencia que el usuario debe especificar. Típicamente esta probabilidad es baja, de aproximadamente un 0.001. El operador Elitista por su parte selecciona un porcentaje con los mejores individuos de la población que son incluidos directamente en la nueva población. Este porcentaje también es un dato que debe ser ingresado por el usuario.

Algoritmo Genético con SUS

Esta variante incluye tres operadores genéticos: selección, crossover y mutación.

Los operadores crossover y mutación son análogos a los anteriormente descritos, sin embargo el de selección corresponde a la selección estocástica universal, donde la probabilidad de que cada individuo sea seleccionado es directamente proporcional a su fitness.

Algoritmo Genético con Nichos

Este algoritmo cuenta con dos operadores genéticos: crossover y mutación que actúan en forma similar a los anteriormente descritos. La modalidad de nichos le entrega a este algoritmo la capacidad de encontrar el óptimo global de una función, además de un conjunto de óptimos locales. Garantía que hasta ahora el resto de los Algoritmos Genéticos no podían entregar.

3.9 Generación de la nueva Población Base

Una vez que la población ha sido manipulada genéticamente, los individuos que la componen son ordenados según su fitness. Finalizado este proceso entonces, se obtiene la nueva población base.

3.10 Heurística basada en la Sensibilidad

Una vez que el Algoritmo Genético ha terminado, es utilizado un algoritmo que considera la sensibilidad de las pérdidas de potencia activa respecto de una inyección de potencia reactiva. Con este método se trata de mejorar (off-line) la convergencia de la solución entregada por el Algoritmo Genético. La heurística de sensibilidad procede como sigue:

1. Primero se relocalizan condensadores que fueron instalados por el Algoritmo Genético.
2. Después se procede a instalar condensadores en este nuevo sistema.

Heurística de Relocalización de Condensadores

Partiendo de la mejor solución entregada por el Algoritmo Genético, la heurística trata de relocalizar condensadores en el sistema. El procedimiento es el siguiente:

1. Calcular el nuevo esquema de sensibilidades.
2. Determinar posibles ubicaciones para la relocalización de condensadores, que corresponden a un número de barras especificados por el usuario y que presentan las mayores sensibilidades.
3. Un condensador es elegido para ser relocalizado sólo si la sensibilidad de la barra en la que se encuentra es menor que la de la nueva localización. Se deben seleccionar las localizaciones que cumplan con este criterio. Se debe mover el condensador más grande primero. Este procedimiento continúa hasta que un condensador es aceptado para ser reemplazado o la lista con los posibles condensadores o localizaciones se agote. Si no hay condensadores elegibles se debe terminar la heurística de relocalización y comenzar la de localización.

4. Con este nuevo esquema se corre un flujo de potencia. Si el flujo converge se evalúan las restricciones de desigualdad. Si alguna de las restricciones es violada, se debe ir al paso 6.
5. Comparar los valores de la función de fitness. Si el nuevo esquema tiene un mejor fitness, la relocalización es aceptada y el condensador es sacado de la lista. Cada condensador sólo se puede mover una vez. Volver al paso 1. En caso contrario, la relocalización es rechazada y se debe tratar con el siguiente condensador más pequeño. Volver al paso 3.
6. No se debe realizar la relocalización y se debe tratar con el siguiente condensador más pequeño. Volver al paso 3.

Heurística de Localización de Condensadores

Después de haber realizado la heurística de relocalización y si aún quedan condensadores disponibles esta heurística trata de localizar nuevos condensadores en el sistema. Esto se realiza eligiendo la barra o las barras que presenten mayor sensibilidad y localizando allí el condensador más pequeño disponible. El procedimiento es el siguiente:

7. Evaluar las sensibilidades del nuevo sistema.
8. Localizar el condensador disponible más pequeño en la barra de mayor sensibilidad. Si no hay localizaciones disponibles parar y terminar.
9. Enseguida se deben verificar las restricciones. Si alguna es violada se debe parar y terminar.
10. En este paso se deben comparar los valores de la función de fitness. Si el fitness del nuevo esquema es mayor la localización es aceptada temporalmente. Se debe probar con el siguiente condensador más grande disponible. Volver al paso 9.

Si con el siguiente tamaño de condensador no se consigue un mejor fitness que con el esquema anterior, se debe aceptar la localización anterior y se debe volver al paso 7.

De esta forma entonces se trata de mejorar la solución entregada por el Algoritmo Genético aprovechando el análisis de sensibilidad.

3.11 Resultado final

Una vez que se han alcanzado el número de generaciones definidas por el usuario y ha concluido la heurística de sensibilidad, el programa entrega el resultado final. Este consiste en el despliegue en pantalla de una lista con las barras que deben ser compensadas, junto con el tamaño de los condensadores para cada uno de los niveles de carga. Además se incluye información sobre las pérdidas de energía y potencia en el sistema con y sin compensación junto con el costo asociado a la inversión que se debe realizar en condensadores.

También se crea un archivo, 'resultados.dat', que contiene información más detallada sobre el resultado de la optimización. Se incluye información sobre el sistema en general, la compensación que se debe realizar, junto con los niveles de pérdidas asociados. Además se incorpora información sobre los ahorros de pérdidas y un listado de todas las tensiones para cada una de las barras del sistema.

IV. PLANTEAMIENTO DE METODOLOGÍA DE SOLUCIÓN VÍA ALGORITMOS GENÉTICOS

4.1 Introducción

Los condensadores son usualmente instalados en los sistemas de distribución para reducir las pérdidas de energía y de potencia, mejorar la regulación de voltaje y la seguridad del sistema con la ventaja adicional de liberar en cierto margen la capacidad del sistema. Los beneficios económicos dependen en gran medida de dónde y cuánta capacidad sea instalada, además por supuesto, de un adecuado esquema de control para los diferentes niveles de carga del sistema de distribución.

Una gran cantidad de métodos han sido desarrollados para resolver el problema de localización de condensadores. En los trabajos más tempranos la mayoría de los investigadores utilizaron métodos analíticos convencionales en conjunto con algún tipo de heurística. Durán (1983, [17]) utilizó métodos de programación matemática para resolver el problema. En la actualidad sin embargo se han desarrollado otras técnicas que se adaptan mejor a los requerimientos del problema. Entre las principales están la Optimización Numérica, Tabú Search y Simulated Annealing, siendo las más recientes las basadas en los Algoritmos Evolutivos.

Bajo estos supuestos, se presenta en el siguiente capítulo el desarrollo de un modelo utilizando un Algoritmo Evolutivo, a través del cual se busca obtener una solución no sólo óptima económicamente, sino que además confiable, flexible y por sobre todo robusta, es decir, fácilmente ajustable a diversos escenarios.

4.2 Computación Evolutiva

La computación evolutiva pertenece a una amplia gama de métodos que tratan de optimizar una función utilizando una estrategia esencialmente independiente del problema que se trate. Estos métodos requieren muy poco conocimiento sobre la estructura del espacio de búsqueda o del problema en particular. Es por ello que son naturalmente aplicados a problemas cuya estructura es poco conocida.

Los Algoritmos evolutivos (AEs) son estrategias de búsqueda basadas en la población, que mantienen la localización de un conjunto de *pruebas* en el espacio paramétrico de la función. La comparación estándar a través de la cual nuevas pruebas son generadas o las antiguas son descartadas, es una función de la población de pruebas existentes. Combinaciones de pruebas individuales en la población son usadas para derivar nuevas pruebas. La población de prueba es inicializada arbitrariamente, y evoluciona hacia mejores regiones del espacio de búsqueda a través de procesos aleatorios de *selección* (determinísticos en algunos casos), *mutación* y *recombinación*. El medio ambiente por su parte entrega una valiosa información, el valor de la *función de fitness* (valor de la función objetivo) del intento o prueba de búsqueda. El proceso de selección favorece aquellos individuos (pruebas) de alto fitness (“sobrevivencia del más adaptado”) para reproducirse más frecuentemente que los peores individuos. El mecanismo de recombinación permite la mezcla de la información de los padres para transmitírsela a sus descendientes, y la mutación introduce la innovación en la población de prueba. Esto conduce durante el proceso de evolución a que la calidad promedio de la población (pruebas, soluciones) aumente, con la esperanza de alcanzar la solución óptima.

En la actualidad se cuenta con diversos algoritmos evolutivos, entre los que destacan: Programación Evolutiva, Programación Genética, Estrategia de Evolución y los Algoritmos Genéticos. Todos ellos comparten una base conceptual común. Estos algoritmos son suficientemente complejos para proveer un mecanismo de búsqueda lo

suficientemente robusto y poderoso. Aunque no puede afirmarse que un AE, como simulación de un proceso genético, no sea una búsqueda aleatoria de una solución para un problema, el resultado es claramente no aleatorio.

4.3 Raíces de la Teoría de la Evolución

Con el fin de facilitar la comprensión sobre los algoritmos evolutivos y sus orígenes, se presenta a continuación una breve reseña histórica sobre las raíces de la teoría de la evolución.

La teoría de la evolución a través de la selección natural fue publicada simultáneamente por Charles Darwin (1809-1882) y Alfred Rusel Wallace (1823-1913) en 1858. Darwin y Wallace son ambos naturalistas que invirtieron varios años coleccionando y clasificando especies en expediciones alrededor del mundo. Su experiencia los convenció que tanto plantas como animales cambian a través del tiempo.

El mecanismo de herencia fue pobremente entendido cuando Darwin y Wallace propusieron su teoría. Era sabido que muchos rasgos o características eran heredados. El austriaco Monk y el botánico Gregor Johann Mendel (1822-1884) publicaron su trabajo sobre genética en 1886 donde se demostraba que los rasgos heredados eran discretos y que además existían dos sets de instrucciones para cada característica. En 1909 Wilhelm Johanssen acuñó el término *gen* para referirse a una unidad discreta de herencia. El contenido completo de material genético almacenado en una célula es conocido como *genoma*.

En el año 1953 Watson y Crick anunciaron la estructura del ADN. Propusieron que el ADN está formado por dos cadenas unidas a través de una base complementaria, conformando la famosa doble hélice. El modelo demuestra un claro código compuesto por cuatro letras y sugiere un mecanismo simple para ser copiado. Las hélices se pueden separar y los extremos libres pueden conformar copias de ellas mismas.

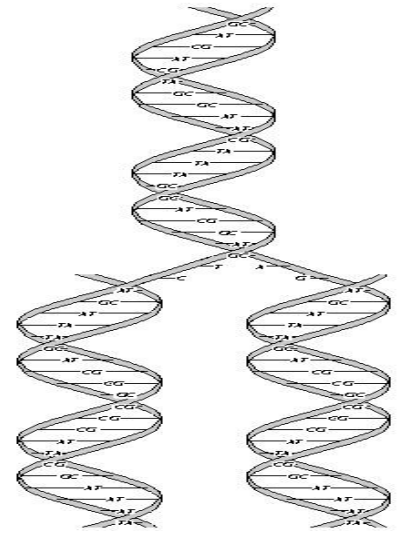


Figura 4.1: Estructura del ADN

El mecanismo principal de la evolución lo constituye la mutación de ADN mientras éste es copiado. La tasa de mutación natural se estima alrededor de una mutación por 10^5 - 10^6 nucleótidos. Una mutación que afecte un aminoácido de una proteína puede cambiar radicalmente su funcionamiento, dejándola algunas veces incluso totalmente disfuncional.

Diferentes genes que codifican la misma proteína son denominados *alelos*. Frecuentemente un alelo va a dominar la población, éste es conocido como del *tipo salvaje*.

La reproducción sexual es un proceso complicado e involucra dos fases. La fase *haploide* que esta compuesta por una copia de cada uno de los genes y la fase *diploide* compuesta por dos copias de cada gen. Dos haploides (usualmente gametos) se combinan en el proceso de *singamia* para producir la fase diploide del organismo. El proceso inverso es conocido como *meiosis*. Durante el proceso de la meiosis dos ‘padres’ o líneas de ADN se unen en ciertos puntos a lo largo del cromosoma. La fase haploide se forma al elegir uno de los hilos de ADN de uno u otro padre. Se cree que la selección ocurre en forma aleatoria y por ello se denomina *clasificación aleatoria*.

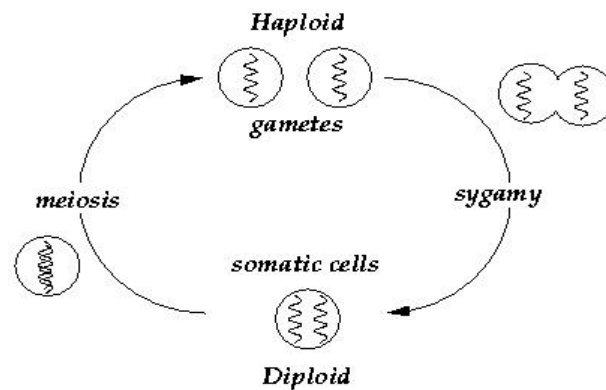


Figura 4.2: Mitosis

El genoma o el conjunto completo de material genético define el *genotipo* de un organismo, mientras que el estado físico de un organismo es conocido como *fenotipo*.

Después de esta descripción histórica de la evolución y de los medios que la promueven se abordará el tema de los algoritmos genéticos propiamente tal.

4.4 Algoritmos Genéticos aplicados a los Sistemas de Distribución

El presente capítulo ilustra la aplicación de AGs en el diseño óptimo de grandes sistemas de distribución, para ello se requiere determinar el tamaño óptimo de subestaciones al igual que las características y localización de alimentadores utilizando para ello los correspondientes costos fijos y variables (que generalmente corresponden a expresiones no lineales). Esta técnica puede ser aplicada tanto a problemas que consideran un sólo período como a problemas que consideran múltiples períodos en el diseño de sistemas de distribución.

Los AGs corresponde a un método de búsqueda y optimización basado en la optimización natural. Ellos consisten en una *población* de strings (de bits) que son

transformados a través de tres operadores genéticos: *selección*, *crossover* y *mutación*. Cada string (cromosoma) representa una solución posible al problema que se está optimizando y cada bit (o grupo de ellos) representa un valor para alguna de las variables del problema (gen). Estas soluciones son clasificadas a través de una función de evaluación, que a su vez permite determinar los mejores valores, o *fitness*, para las mejores soluciones. Cabe señalarse que la población inicial es generada en forma aleatoria.

Este comportamiento sencillo en realidad esconde uno muy poderoso. De hecho la combinación del operador de selección y crossover fomentan la proliferación de individuos que poseen bloques compactos de bits que presentan un buen desempeño, estos bloques son denominados *schemata*, y son replicados a través de la selección y combinados o separados por el crossover. Por otro lado, la mutación se encarga de expandir la búsqueda a otros lugares evitando la convergencia prematura (óptimos locales) y generando nuevos individuos o recobrando especies que se habían perdido.

De esta forma los AGs tienden a seleccionar individuos con buen comportamiento y a recombinar alguno de sus bloques que los constituyen, creando de esta manera un mayor número de copias de los bloques que presentan un mejor desempeño.

4.4.1 Ventaja de los AGs respecto a otros Métodos

Algunas de las ventajas que presentan los AGs respecto a otras técnicas de optimización clásicas son:

- Los AGs no requieren que la función objetivo tenga un “buen comportamiento”, ya que son capaces de tolerar sin ningún problema discontinuidades o funciones polinomiales.
- Los AGs se prestan para implementaciones distribuidas o en paralelo.
- Debido a que los AGs realizan una búsqueda en una población de puntos y están basados en reglas de transición probabilísticas identificando un mayor número de peaks, es menos factible que converjan a un máximo o mínimo local.
- Son simples de implementar ya que la única información necesaria es la función objetivo y las restricciones correspondientes.

4.4.2 Algoritmos Genéticos en el Diseño de Sistemas de Distribución Multi-períodos

El objetivo de planificar un sistema de distribución es generar un diseño que permita obtener una expansión ordenada en infraestructura para proveer en forma económica, confiable y segura energía eléctrica a los usuarios.

La necesidad de una expansión estratégica está ligada al crecimiento de la demanda. El período de planificación debe ser dividido en diversas etapas. El resultado es una lista de inversiones que se debe realizar en cada etapa.

Para determinar la expansión óptima de un sistema de distribución, se han propuesto un sinnúmero de modelos matemáticos en el pasado, muchos de los cuales son modelos estáticos que generan una solución para un set fijo de datos y que consideran un período simple. Por otra parte el planteamiento dinámico considera la evolución de la demanda a través del tiempo y los consecuentes cambios topológicos en la red (inserción de nuevos alimentadores, nuevas subestaciones, reforzamientos, etc.)

Los modelos que se pueden realizar son lo suficientemente flexibles como para considerar características realistas y condiciones naturales de operación como:

- Inclusión de múltiples alimentadores en el mismo trecho, lo que se refleja en un aumento de seguridad del sistema.
- La posibilidad de no considerar líneas en alguna etapa o período y su posible reinsertión en alguna etapa posterior.
- Establecer configuraciones radiales y en algunos casos se permite la existencia de loops con el propósito de reforzar la seguridad del sistema.
- Consideración de múltiples objetivos.
- Manejo en forma natural de diversos factores de diversidad de carga.
- Posibilidad de no especificar inmediatamente, o de antemano, en que año debe ser construida una subestación o debe ser puesta en servicio.

Los objetivos que se persiguen a través de la planificación de distribución consisten en proveer un diseño y sus respectivos planes de implementación necesarios para una expansión ordenada de las construcciones (como subestaciones y alimentadores), minimizando los costos asociados a estas nuevas construcciones y sus costos operacionales, todo esto sujeto a niveles aceptables de confiabilidad.

En los modelos implementados se han considerado las siguientes restricciones:

- Operación de la red bajo una configuración radial (sin embargo pueden existir algunos loops).
- Restricciones de caída de voltaje.
- Especificaciones de la demanda.
- Flujo de potencia sujeto a límites térmicos en las líneas.
- Ubicación posible para subestaciones y líneas.

Los costos de las construcciones se pueden dividir en tres elementos: costos de las subestaciones, costos de la capacidad de expansión de las subestaciones y costos de los nuevos alimentadores. Las pérdidas en la red se consideran en los costos operacionales.

Finalmente, los elementos que ya existen (subestaciones o alimentadores) no son incluidos en los costos de inversión, sin embargo las pérdidas de potencia de éstos equipos si son considerados dentro de los costos operacionales.

En la elaboración de modelos se han considerado los siguientes supuestos:

- a) Una demanda máxima ha sido considerada para cada etapa del planeamiento de la expansión del sistema. Ha sido necesario considerar información acerca de la curva de carga (como el factor de carga) con el objetivo de evaluar las pérdidas y para implementar algunos índices que permitan medir la confiabilidad del sistema, como por ejemplo, la carga media desconectada anualmente (correspondiente a un período) o la energía promedio anual no suministrada.
- b) Las nuevas instalaciones son conocidas de antemano en lo que respecta a su localización y costo estimado.

La estrategia de expansión está guiada por el crecimiento de la demanda (o la carga). Por ello el período considerado en el análisis es dividido en diversas etapas y el resultado que se debe obtener corresponde a una lista de las inversiones que se van a realizar en cada etapa.

La aplicación de AGs al problema de planificación de un sistema de distribución se bosqueja a través del siguiente lineamiento general:

- a) Un conjunto de variables es elegido para representar una solución de la red en el período analizado (que comprende todas las etapas). Estas variables son codificadas en un cromosoma.
- b) Un AG es aplicado a una familia de soluciones, produciéndose el nacimiento de nuevas generaciones.
- c) Cada solución de la nueva generación es evaluada a través de una función objetivo (o fitness function) que incluye costos de inversión, costos por pérdidas, confiabilidad, desviaciones de la caída de tensión; no cumplimiento de otras restricciones relacionadas con la función objetivo.
- d) De esta forma al final del proceso se obtiene una familia que presenta un buen desempeño.

Características especiales de algunos modelos implementados:

- Cálculo de flujo DC ha sido utilizado como primer filtro para infactibilidades.
- También ha sido incluido un cálculo de flujo AC.
- Consideración de factores de diversidad de la demanda.

4.4.3 Algoritmos Genéticos aplicados al Diseño de grandes Sistemas de Distribución

A continuación se detalla la utilización de un AG para el diseño óptimo de un gran sistema de distribución, resolviendo para ello el tamaño óptimo y localización de alimentadores y subestaciones utilizando los correspondientes costos fijos así como también la verdadera función de costos variables no-lineal (y no una aproximación lineal).

A pesar que se trata de un método que permite determinar la expansión óptima de un sistema de distribución en un solo período, a través de una técnica conocida

como “metodología pseudodinámica” (pseudodynamic methodology) se pueden resolver problemas que consideran múltiples períodos.

Como se mencionó anteriormente este algoritmo considera una función de costos variables no-lineal, así como también la caída de tensión en los nodos y un índice que entrega una medida de confiabilidad del sistema. Este algoritmo incluye además una codificación entera, que permite una mayor flexibilidad, tomando en cuenta diversos aspectos, como por ejemplo:

- Diferentes tipos de conductores.
- Diferentes tamaños de subestaciones.
- Construcción apropiada de alimentadores adicionales para incrementar la confiabilidad del sistema.

Como un desafío futuro se propone este último aspecto para ser aplicado en un modelo multiobjetivo en el planeamiento de un sistema de distribución. Para ello se pueden implementar dos funciones objetivos, una referente a los costos y otra que considere la confiabilidad del sistema de distribución de tal manera que ambas sean evaluadas en forma simultánea por el modelo.

En este caso el modelo de diseño es un problema de optimización mixto no-lineal de localización y tamaño de alimentadores y subestaciones que puede ser utilizado para la expansión de sistemas de distribución en un período o considerando múltiples períodos, a través de la metodología pseudodinámica.

El modelo contiene una función objetivo que considera el costo de inversión de futuros alimentadores y subestaciones, así como también, una función de costos variables no-lineal asociada a la operación del sistema de distribución.

Las restricciones consideradas en el modelo son las siguientes:

- Especificaciones de potencia demandada.
- Límites de capacidad de los alimentadores.
- Límites de capacidad de las subestaciones.

Además, se incluye en el modelo el cálculo de caída de voltaje en todos los nodos.

El algoritmo permite determinar la localización y tamaños óptimos de subestaciones y alimentadores que deben ser construidos para la expansión futura de la red de distribución cumpliendo con los requerimientos de demanda futura y considerando los límites de caída de voltaje en los nodos. Más aún, el algoritmo también obtiene un índice usado para evaluar la confiabilidad del sistema de distribución bajo una configuración radial de la red, pero sin descartar la posibilidad de una topología no radial, con alimentadores “en reserva” que usualmente no están en operación pero que pueden utilizarse en reconfiguraciones de la red.

4.5 Método de Nichos para Algoritmos Genéticos

Los Algoritmos Genéticos pertenecen a la familia de los Algoritmos Evolutivos, que corresponden a modelos computacionales inspirados en la naturaleza.

Los Algoritmos Genéticos (AGs) constituyen un método de búsqueda para resolver problemas complejos, han sido exitosamente aplicados a funciones de optimización, a estructuras físicas y de programación. Los AGs han mostrado resultados iniciales prometedores en clasificación y aprendizaje de máquina como también en la simulación de sistemas biológicos, ecológicos, económicos, financieros y sociales.

Los AGs están basados en los principios genéticos de la biología y operan en forma análoga a la evolución. Sin embargo mientras que los procesos naturales de la evolución mantienen una variedad de especies, cada una de las cuales ocupa un nicho ecológico separado, los AGs tradicionales empujan a la población artificial hacia la convergencia. Esto significa que todos los individuos de la población tempranamente comienzan a ser idénticos. Lo que conlleva a que aún cuando un problema presente diversas soluciones, un AG tradicional sólo va a localizar una de ellas.

Los métodos de nichos permiten a los AGs mantener una población de diversos individuos y por ende son capaces de localizar múltiples soluciones óptimas dentro de una población. Técnicas efectivas de nichos son cruciales para el éxito de los AGs particularmente en aprendizaje de máquina, optimización de funciones multimodales, optimización de funciones multiobjetivos y en simulaciones de sistemas complejos y adaptativos.

El objetivo de esta sección es diseñar una técnica para el mantenimiento estable de subpoblaciones, o nichos, en los AGs. Para alcanzar este objetivo primero se desarrollará un marco para modelar AGs que incorporan el método de nichos. Después será utilizado este marco para construir modelos individuales de AGs que incorporan la metodología de nichos.

No se puede realizar un estudio acabado de los AGs que incorporen la metodología de nichos, sin considerar el tópico de la diversidad, que es un tema recurrente en la literatura de los AGs. La motivación original para desarrollar la metodología de nichos fue, de hecho, promover la diversidad en los AGs tradicionales. La diversidad puede servir a dos propósitos en los AGs clásicos. El primero de ellos es retrasar la convergencia con el fin de promover la exploración y así poder localizar una mejor solución única. El segundo propósito es localizar múltiple soluciones finales.

Cabe señalar que ambos propósitos no son mutuamente excluyentes; técnicas que localizan múltiples soluciones finales también son altamente efectivas para controlar la convergencia. Sin embargo, técnicas que son desarrolladas para retrasar la convergencia no suelen ser útiles para obtener múltiples soluciones.

Los AGs capaces de formar nichos serán primeramente examinados en base a funciones de optimización multimodales. Esto es, dada cierta función de optimización con múltiples óptimos el objetivo será localizar varios de los mejores óptimos, incluyendo por supuesto el óptimo global.

4.5.1 Algoritmos Genéticos

Los AGs son procedimientos de propósito general, para búsqueda paralela basados tanto en los principios de evolución como genéticos. Un AG trabaja modificando repetidamente una población compuesta de estructuras artificiales a través de la aplicación de operadores genéticos. Típicamente los AGs son una “caja negra” que utiliza exclusivamente la información de fitness; sin la necesidad de requerir información del gradiente o algún otro conocimiento interno del problema.

Componentes de los Algoritmos Genéticos

El objetivo en una optimización es encontrar la mejor solución o soluciones para un determinado problema, respecto a uno o más criterios. Para poder utilizar los AGs primero se debe definir una estructura adecuada para representar las soluciones. Una instancia de esta estructura de datos, en lo referente al espacio de búsqueda, representa un estado o punto dentro del espacio de todas las posibles soluciones.

Esta estructura básica de un AG consiste en uno o más cromosomas (usualmente uno). Un cromosoma es típicamente un *string* de bits, debido a ello el

término *string* es generalmente utilizado para referirse a un cromosoma. Los AGs sin embargo, no se limitan sólo a representaciones de strings binarios, existen otras posibilidades como vectores de números reales y programas computacionales de alto nivel. A pesar que estructuras de largo variable son apropiadas para diversos problemas, la estructura de largo fijo es prácticamente una norma.

Cada cromosoma (string) constituye una concatenación de un número de subcomponentes denominados *genes*. Los *genes* se encuentran en diversas posiciones del cromosoma denominadas *loci* y toman un conjunto de valores conocidos como *alelos*. En una representación binaria un *gen* es un bit, *locus* es su posición dentro de un string y un *alelo* es su valor (0 ó 1). El término biológico *genotipo* se refiere a la composición genética de un individuo y corresponde a una estructura en un AG. El término *fenotipo* se refiere a la característica externa de un individuo y corresponde a una estructura decodificada de un AG.

Para fijar conceptos, se utilizará un ejemplo sencillo pero muy ilustrativo de un problema de optimización de maximización de una función de dos variables:

$$f(x_1, x_2) = x_1 + x_2 \quad ; \quad 0 \leq x_1 \leq 1 \quad (4.1)$$

$$0 \leq x_2 \leq 1$$

Una técnica común para codificar variables reales es transformarlas en enteros binarios de largo suficiente para proveer el grado de precisión deseado. Asumiendo que una codificación de 8-bits es suficiente para x_1 y x_2 , éstas variables son decodificadas a través de la normalización binaria entera correspondiente, dividiendo el entero por 2^8-1 . Por ejemplo, 00000000 representa $0/127$ ó 0, mientras que 11111111 representa $127/127$ ó 1. La estructura de datos a ser optimizada es un string de 16-bits, que representa diversas combinaciones de x_1 y x_2 . La variable x_1 está

representada por los 8-bits más a la izquierda del string, mientras que la variable x_2 corresponde a los 8-bits más a la derecha. El genotipo de un individuo corresponde al string de 16-bits mientras que el fenotipo es una instancia de la dupla (x_1, x_2) . El genotipo es un punto en el espacio de 16 dimensiones de Hamming que rastrea el AG. El fenotipo es un punto en el espacio bi-dimensional compuesto por las variables decodificadas.

Para optimizar una estructura utilizando AGs se debe establecer una medida que permita designar un índice de calidad a cada estructura en el espacio de búsqueda. La *función de fitness* es responsable de esta tarea. En una función de maximización, la función objetivo usualmente es utilizada como función de fitness.

Funciones de fitness existen en diversas aplicaciones, no sólo matemáticas. Para problemas de optimización combinatorial como el del vendedor viajero donde la longitud del tour tratada adecuadamente constituye una buena función de fitness.

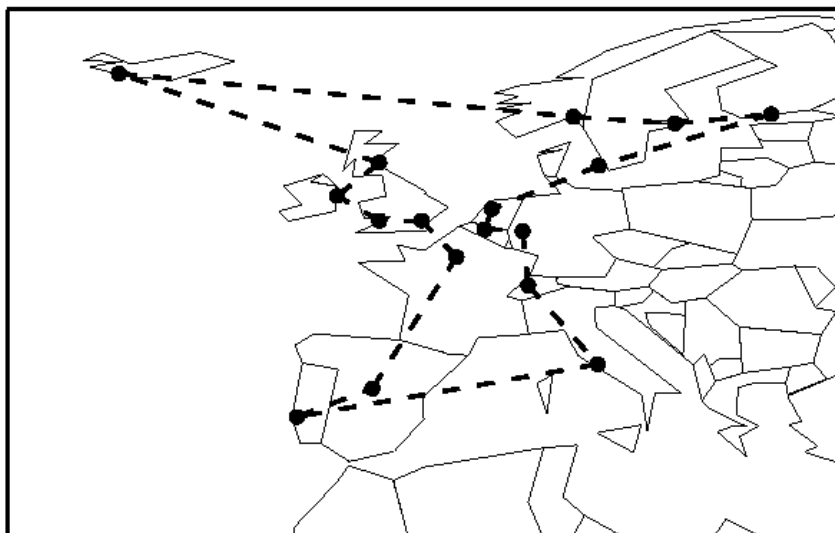


Figura 4.3: Problema del Vendedor Viajero

En problemas de optimización ponderada en redes neuronales donde puede ser utilizado la suma de los errores al cuadrado de un set de ejemplos de entrenamiento como función de fitness.

Mecánica de los Algoritmos Genéticos

Un Algoritmo Genético simple genera aleatoriamente una población inicial de n estructuras. El AG procede para un número fijo de generaciones o hasta que algún criterio de término sea satisfecho. Durante cada generación, el AG lleva a cabo una selección proporcional al fitness, seguida de un crossover de punto simple y finalmente se realiza el proceso de mutación.

Primero el sistema de selección proporcional al fitness asigna a cada estructura individual i en la población una probabilidad de selección $p_s(i)$, de acuerdo a la razón entre el i -ésimo fitness y el fitness total acumulado de la población:

$$p_s(i) = \frac{f(i)}{\sum_{j=1}^n f(j)} \quad 4.2$$

Entonces son seleccionados un total de n individuos para un posterior procesamiento genético, de acuerdo a la distribución definida por $p_s(i)$. El modelo más simple de selección proporcional al fitness es *roulette-wheel selection*, en que n individuos son elegidos simulando n juegos de ruleta. La ruleta contiene una casilla para cada elemento de la población y el tamaño de cada casilla es directamente proporcional a su respectivo $p_s(i)$. Cabe señalar, que un miembro de la población que posea un fitness elevado va a ser elegido con mayor frecuencia que uno que tenga un fitness menor.

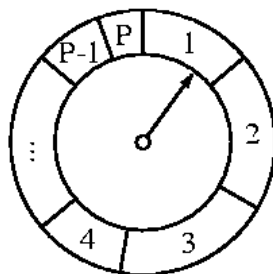


Figura 4.4: Roulette Wheel Selection

Después de la selección, los n individuos son sometidos al crossover (también denominado recombinación) con una probabilidad fija de ocurrencia p_c por cada string. Los n strings elegidos son emparejados en forma aleatoria, generándose $n/2$ parejas. Para cada pareja el crossover puede como no puede ocurrir. Con una probabilidad $1-p_c$, el crossover no ocurre, y ambos individuos avanzan a la etapa de mutación. En otro caso la pareja produce dos hijos producto del crossover y sólo ellos continúan en la etapa de mutación.

El crossover de punto simple opera de la siguiente manera. Primero una de las potenciales $l-1$ posiciones de cruce del string es elegida aleatoriamente. Los dos padres elegidos son cortados en la posición determinada, obteniéndose dos segmentos por cada uno de ellos. Los segmentos apropiados de los distintos padres son concatenados generándose dos hijos. Por ejemplo, suponiendo que un padre está compuesto por 16 ceros, mientras que el otro está compuesto por 16 unos. Suponiendo además, que el sitio 12 es elegido aleatoriamente como punto de corte de entre los 15 posibles. Los padres y sus hijos se muestran a continuación:

Padres		Hijos
0000000000000000	→	0000000000001111
1111111111111111	crossover	1111111111110000

Después que la etapa de crossover ha concluido, comienza la de mutación. Para cada string que avanza a esta etapa, cada uno de sus bits puede ser mutado con una probabilidad p_m . La población resultante de la etapa de mutación entonces reemplaza a la población anterior completándose de esta forma una generación. Las generaciones subsecuentes siguen el mismo ciclo compuesto por la selección, crossover y mutación.

Existen diversas alternativas para los operadores de selección, crossover y mutación presentados anteriormente. A continuación se destacan algunas de ellas.

El *torneo de selección* es una alternativa al método de selección basado en una ruleta (Roulette wheel selection) y es más sensitivo a un fitness relativo que a uno absoluto. El método de torneo de selección (Brindle, 1981; Goldberg & Deb, 1991) realiza n torneos y selecciona n individuos. En cada torneo compiten k elementos de la población y es elegido el más adaptado, o el que presente un mayor fitness. La variación más común es el torneo de selección binaria, con $k = 2$.

El *método de selección elitista* (1975, [65]) asegura que el o los mejores elementos de una población sobrevivirán de generación en generación. La estrategia más básica de elitismo consiste en copiar el mejor elemento de la población actual directamente en la siguiente población, si este elemento no ha sido transferido a través del proceso normal de selección, crossover y mutación. En general cualquier método estándar de selección puede convertirse en uno elitista.

En el *crossover de doble punto* dos segmentos son elegidos en forma aleatoria y los padres intercambian el segmento que se encuentra entre los dos puntos de corte. Considerando un mayor número de puntos de corte se obtiene el *crossover multipunto* como el utilizado en el desarrollo de este trabajo.

En el *crossover uniforme* (1989, [63]) cada bit del primer padre es transferido al primer hijo con una probabilidad fija; en caso contrario es transferido al segundo hijo. El resto de las localizaciones (Loci) de ambos hijos son rellenadas utilizando los bits del segundo padre. Para mayor detalle sobre operadores genéticos ver Anexo C.

4.5.2 Métodos de Nichos

Los métodos de nichos son técnicas que promueven la formación y mantención de subpoblaciones estables en los AGs. Además, los métodos de nichos se pueden aplicar en la formación y mantención de subpoblaciones intermedias orientadas a la obtención de una solución final única. Sin embargo, tradicionalmente son utilizados en el contexto de formar y mantener múltiples soluciones finales. La atención se centrará entonces en este último punto para el resto de este desarrollo. Para una introducción a la teoría que gobierna el comportamiento de los AG ver Anexo A.

Un método de nichos debe ser capaz de formar y mantener múltiples y diversas soluciones finales, sin importar si estas soluciones poseen igual fitness o diferentes. Además, debe ser capaz de mantener estas soluciones durante un período infinito de tiempo, con respecto al tamaño de la población.

Es sabido que reducir la presión de selección, el ruido de selección y la alteración provocada por los operadores generalmente no resulta en un AG capaz de formar y mantener nichos. Lo que se requiere no es sólo disminuir la velocidad del proceso de selección, el ruido de este proceso o el grado de alteración provocado por los operadores. Lo que se busca es un nuevo tipo de algoritmo, uno que promueva la diversidad a través de dimensiones útiles permitiendo la convergencia en diversas dimensiones. Con respecto al espacio de búsqueda, la convergencia puede ocurrir en algunas regiones locales, pero la diversidad debe orientar esta búsqueda a las regiones

más prominentes o adaptadas. Los métodos de nichos alteran el algoritmo de selección para generar una presión de selección dentro de las regiones y no entre las regiones del espacio de búsqueda. La presión de selección dentro de regiones individuales es substancial, pero además, deben preservarse las propiedades de anichamiento. En adelante se centrará la atención en el dominio general de la aplicación; optimización de funciones multimodales. Los métodos de nichos deben ser aplicables sin modificaciones radicales en múltiples y diversos campos.

Dado un conjunto de múltiples óptimos y una capacidad limitada para localizar un óptimo, el mejor AG con nichos deberá preferir el óptimo más alto. Algunos AGs pueden preferir también algún óptimo que se encuentre en otro punto parcialmente óptimo. El mejor método de nichos debe ser capaz de localizar los peaks más altos ante la presencia de un gran número de peaks inferiores, aún cuando se trate de problemas altamente engañosos. Más aún, el mejor método de nichos no siempre será altamente selectivo, sino también tendrá la habilidad de formar y mantener óptimos globales y no globales. Se centrará la atención en este análisis en la mantención en vez de la formación de soluciones óptimas; ya que se asume de facto que un AG simple es capaz de formar buenas soluciones, y esta capacidad, por supuesto, será transmitida a un AG con nichos. Si esta capacidad no es transferible a un algún algoritmo en particular, se considerará que este algoritmo no corresponde a un método de nichos.

4.5.3 Clasificación de los Algoritmos de Nichos

Generalmente los métodos de nichos pueden ser clasificados en dos dimensiones según su comportamiento. La primera dimensión es espacio v/s tiempo. Hasta ahora se ha discutido los métodos de nichos como algoritmos espaciales; forman y mantienen subpoblaciones en el tiempo. En este estudio los métodos de

nichos espaciales son frecuentemente denominados, *métodos de nichos paralelos*, debido a que conceptualmente desarrollan nichos en paralelo dentro de una única población. Los métodos de nichos temporales son usualmente denominados nichos secuenciales, debido a que conceptualmente desarrollan nichos en forma secuencial a través del tiempo.

La segunda dimensión corresponde a los nichos que se desarrollan en un único ambiente versus los que lo hacen en múltiples ambientes. De hecho es posible clasificar el ambiente general como único o múltiple y espacial o temporal. Muchos métodos de nichos son capaces de formar y mantener nichos estables en un ambiente único. (El ambiente en funciones multimodales de optimización está compuesto por una única función de adaptación). Esos son los métodos de nichos más generales debido a que pueden ser aplicados en situaciones compuestas de múltiples ambientes. Existen otras técnicas de nichos que requieren múltiples ambientes ya sean espaciales (múltiples funciones de adaptación) o temporales (funciones de adaptación no estacionarias). En algunos casos el ambiente general varía a través de ambos: tiempo y espacio.

En el siguiente cuadro se clasifican los diversos tipos de métodos de nichos según su comportamiento en las dimensiones antes descritas. El método de nichos de cada cuadrante de la tabla es similar al tipo de problema que es aplicable.

	AMBIENTE SIMPLE	MÚLTIPLES AMBIENTES
TEMPORAL	Localización Secuencial	Sobre Especificaciones AGs Ecológicos
ESPACIAL	Ventaja Heterocigota Crowding Competición Restringida Fitness Sharing	AGs Ecológicos Sistemas Inmunes

Tabla 4.1: Clasificación de los Métodos de Nichos

Por ejemplo, técnicas temporales en un sólo ambiente, son más adecuadas para funciones de optimización multimodales. Técnicas espaciales en un sólo ambiente son apropiadas para funciones de optimización multimodales, funciones de optimización multiobjetivo, y para clasificación. Técnicas espaciales en múltiples ambientes son ventajosas para resolver funciones de optimización multiobjetivo. Finalmente, técnicas temporales en múltiples ambientes son apropiadas para resolver problemas de simulación adaptativa. Para mayor detalle ver Anexo B

4.5.4 Un Marco formal para Escenarios Multimodales

Los escenarios multimodales constituyen el terreno donde los métodos de nichos van a correr. La optimización de escenarios multimodales o funciones de fitness involucra tópicos como el número total de óptimos en el espacio de búsqueda, la cantidad y calidad de los óptimos que se desean localizar, y el grado de dificultad del problema debido a aislamiento u óptimos confusos. No se van a examinar todos los posibles tipos de funciones multimodales debido a que son infinitas. En vez de

ello se concentrará en resolver un tipo general de problemas de optimización de funciones multimodales que incluye (de alguna forma) los problemas prácticos típicos a los que se les aplican los métodos de nichos.

El objetivo final en la optimización de una función multimodal es encontrar varias soluciones, tanto globales como locales. Antes de continuar es necesario definir formalmente la optimización. Se debe asumir un espacio de búsqueda S y una función objetivo f que asigna un número real a cada elemento de S ($f : S \rightarrow R$, donde R es el conjunto de los números reales). Se debe asumir además sin pérdida de generalidad, que el objetivo es la maximización de f . Se define una función de vecindad, $N : S \rightarrow 2^S$, que asigna a cada elemento i de S un set de elementos de S que de alguna forma son cercanos a i (i no necesita estar en su propia vecindad).

Para un $i \in S$ arbitrario, se define $N(i) \subseteq S$ como la *vecindad* de i ; i es un *máximo local* si $f(i) \geq f(j)$ para todos los $j \in N(i)$; i es un *máximo global* si $f(i) \geq f(j)$ para todos los máximos locales j . Se llama c al número de máximos locales y z al número de máximos globales.

Diversos tipos de problemas de optimización de funciones multimodales pueden resultar de interés para personas que resuelven problemas reales. A continuación se presenta un conjunto de estos problemas:

1. Encontrar cualquier $b < c$ máximos;
2. Encontrar todos los c máximos;
3. Encontrar por lo menos los $b \leq c$ máximos mayores;
4. Encontrar cualquier $b < z$ máximo global;
5. Encontrar los z máximos globales.

Como ejemplo los problemas del tipo 5, son importantes en la optimización de funciones multiobjetivos. En la optimización de funciones multiobjetivos las

soluciones Pareto-óptimas son óptimos globales, mientras que las soluciones dominadas no lo son. Cabe destacarse, que al resolver un problema equivalente del tipo 3, se encuentra una solución a cualquiera de los otros tipos. Por ello se va a centrar el esfuerzo en encontrar los b mejores máximos en un escenario que consta de c máximos. Existen otros tipos de problemas que no se encuentran en la lista pero que son menos interesantes como encontrar el menor máximo. Una manera de manejar criterios secundarios es adicionarlos en la función de fitness. En general no se puede predeterminar si un criterio que no esta basado en el fitness tiene alguna aplicación relevante. Entonces se restringe la atención a la optimización de funciones multimodales basadas en el fitness y más específicamente a los problemas del tipo 3.

4.5.5 Marco Analítico

Consiste de abstracciones, definiciones y supuestos simplificadorios, de los que más tarde se derivarán modelos específicos. Estas abstracciones, definiciones y suposiciones permiten la extracción de los mejores comportamientos de cada método. Como se demostrará en simulaciones posteriores, el valor predictivo de los modelos no se ve aminorado por la presencia de estas suposiciones. De hecho la realización de los modelos no sería posible sin estas suposiciones.

El marco analítico representa una nueva aproximación en la modelación de los AGs. Sin embargo esta nueva aproximación no es completamente novedosa ya que toma prestados supuestos explícitos e implícitos de otros estudios acerca de la formación de nichos en AGs.

Clases equivalentes

El análisis tradicional de esquema de un AG hace uso de la partición de esquema (schema partitions) del espacio de búsqueda. Esto es, si uno escoge un set

de posiciones de bits fijos y se está interesado en cómo el AG asigna valores a los bits de esas posiciones, entonces una partición natural es una en la que se considere todo esquema posible teniendo las posiciones predeterminadas fijas. Esa partición natural define una clase equivalente. Si han sido designados k bits como fijos, el espacio de búsqueda se subdivide en 2^k particiones de esquemas o clases equivalentes.

Muchos investigadores partiendo por Holland (1975, [78]) han sugerido un marco más general en el que las particiones no necesariamente correspondan a posiciones de bits. Pueden ser consideradas particiones arbitrarias, consistentes en clases equivalentes arbitrarias que no necesitan basarse en el esquema. Estas clases equivalentes arbitrarias son conocidas como *esquema generalizado* o *formae*.

Debido a que se está estudiando la optimización de funciones multimodales, se desea que las clases correspondan a óptimos locales en el espacio de búsqueda. Para algunas funciones bien definidas, los óptimos locales corresponden a particiones del esquema, sin embargo el modelo no se limitará a tales funciones solamente. En muchos casos, particiones basadas en los óptimos no poseen una relación uno a uno con las particiones de esquema. Sin embargo existe un traslape significativo entre las clases de equivalencia basadas en los óptimos y las correspondientes basadas en los esquemas.

Si se pudiera examinar el espacio de búsqueda en tres dimensiones o menos, intuitivamente se podría conocer los peaks a los que cada elemento pertenece. Definiciones de clases basadas en bits y en esquemas, no siempre son lo más adecuados y naturales para un problema dado. Más aún, un gran número de óptimos locales que no interesan pueden estar presente. En ese caso surge la pregunta, si es necesario representar todos los óptimos extraños como clases o agruparlos a todos en una clase conocida como “ninguno de los de arriba”. Surge otra duda cuando se trata de un punto que no se encuentra muy cerca de ningún máximo y en el peor de los

casos se trata de un mínimo equidistante de dos o más máximos. ¿A qué clase se debe entonces asignar ese punto? Más aún, muchos estudios consideran un punto perteneciente a un peak sólo si posee cierto porcentaje de fitness del fitness del peak ó si es que se encuentra a cierto número de pasos desde el peak de acuerdo a algún criterio de vecindad.

La primera definición que se realiza en este marco es decidir qué constituye una clase equivalente, y a que clase pertenece cada punto del espacio de búsqueda. En estos términos se define una partición X del espacio de búsqueda utilizando los conceptos de “atracción” y “base de atracción”. Se considera que cada máximo local es un centro de atracción y se asigna a cada clase todos los puntos que se encuentren en la base máxima de atracción. Cada elemento de la población es considerado miembro de la clase correspondiente al peak por el que es atraído y todos los puntos del espacio de búsqueda son atraídos por exactamente un peak.

Se define una *base local de atracción máxima* en términos de un algoritmo de escalamiento de cerros (hillclimbing) bajo un operador de vecindad apropiado. Un punto se encuentra en la base de atracción de un máximo local si escala al máximo utilizando un operador de vecindad dado y un algoritmo determinístico para escalar. Un algoritmo para escalar es determinístico si la probabilidad de que un punto escale a dos peaks diferentes, en dos ocasiones, es cero. La definición de clases equivalentes basadas en un algoritmo para escalar, es consistente con la noción de que los AGs son métodos de optimización globales que operan mejor combinados con métodos de optimización locales como el “hillclimbing”. En la práctica el método para escalar es invocado por el AG en la población final, con el fin de forzar a los elementos de la población a escalar al óptimo local más cercano.

Fitness Representativo

Se asume además, que cada clase i tiene un fitness representativo, f_i , que se define como el más alto del peak correspondiente; todos los elementos de i tendrán idéntico fitness f_i . La validez de este supuesto va creciendo a medida que el programa progresa hacia el equilibrio. Esto es relevante debido a que se está interesado en el comportamiento del algoritmo en el equilibrio. Es posible relajar este supuesto de manera que f_i sea el fitness medio de los elementos de la clase i , cada clase presenta cierta varianza en el fitness de sus elementos, que conforman cierto tipo de distribución predefinida.

Peaks deseables

Algunos de los modelos asumen que se está interesado en localizar los c peaks. Aquellos modelos corresponden a los del tipo 2, vistos anteriormente. Otros modelos más generales asumen que se está interesado en encontrar por o menos los b peaks más altos de un total de c . Esos modelos corresponden al tipo más general de problemas de optimización: los del tipo 3.

Mantenimiento de Clases

Desde un punto de vista fijo, los modelos operan manteniendo las clases más que formando clases. Debido a ello se asume que todas las clases deseadas se encuentran presentes desde el principio en la población. Se debe hacer notar que este es un requerimiento para el modelo y no para el AG de nichos que se correrá. Si la formación de clases es exitosa en un AG de nichos, entonces cuando el AG se aproxime al equilibrio, las perspectivas de mantención serán mayores.

El proceso de localizar un máximo local está ligado, según la literatura estándar de AGs, a la optimización de funciones unimodales. El tamaño de la población requerido para localizar un simple peak está relacionado con el ruido presente en la señal, dentro y entre los “schematas” (esquemas). Por ello el proceso para localizar y mantener peaks puede incorporar estimadores basados tanto en la relación señal a ruido como en consideraciones de mantención de clases. Como era de esperarse la mantención de clases está sujeta a requerimientos de formación como en un tamaño de la población suficientemente grande para mantener el número de clases deseado.

Cruza (Crossover)

El marco presentado no asume una variación particular del operador de cruza. Modelos individuales, dado un operador de cruza y la clase a la que pertenecen ambos elementos que van a ser cruzados, pueden asignar probabilidades señalando si los hijos van a pertenecer a varias clases.

Un supuesto que no varía a lo largo de este estudio es que al cruzar dos elementos de una misma clase se generan dos hijos pertenecientes a la misma. Radcliffe denomina este supuesto: *respeto*. Además agrega, que la presencia de *respeto* permite que los AGs converjan a clases equivalentes adecuadas.

Discriminación perfecta

Los supuestos de discriminación perfecta establecen que un AG con nichos es capaz de determinar, sin error, a qué clase pertenecen todos los puntos del espacio de

búsqueda. Más aún, la pertenencia de clases por si sola determina la proximidad relativa de todos los puntos del espacio de búsqueda.

El supuesto de discriminación perfecta posee diversos nombres y diversas consecuencias. Los nombres son *discriminación perfecta*, *repartición perfecta*, *nichos no traslapados* (la partición de nichos en clases equivalentes asegura que no se van a traslapar), *comparación perfecta* y *comparación ausente de ruido*. Una consecuencia es que un elemento de una clase dada, sin importar el método de comparación o la medida de distancia empleada, se encuentra siempre más cerca de un elemento de su misma clase. Una segunda consecuencia es que una medida de diferencia, exenta de error, es capaz de determinar cuándo dos elementos son miembros de una misma clase, y cuál de los dos se encuentra más cerca de un tercer elemento.

4.5.6 Marco Empírico

La parte empírica del marco de modelación desarrollado consta de problemas de prueba para la optimización de funciones multimodales. Algunos problemas tienen el propósito de verificación: fueron diseñados para ajustarse perfectamente al modelo correspondiente, con el fin de probar la validez del modelo. Otros problemas prueban los límites de los modelos correspondientes para determinar cuán bien se aplican a problemas artificiales de gran dificultad y a problemas representativos de los “casos promedios”. Diversos problemas de pruebas han sido utilizados en el proceso de análisis y diseño. Finalmente algunos problemas de prueba han sido incluidos como complementos, debido a que estudios anteriores de AGs han establecido estos problemas como metas preliminares.

La tabla 4.2 resume cada problema y su grado de dificultad. La tabla señala el número de peaks que un problema posee, separándolos en deseables e indeseables. También muestra si un problema es engañoso y su grado de dificultad

correspondiente. Además señala si el problema posee un fenotipo accesible por el usuario.

PROBLEMA	PEAKS DESEABLES	PEAKS EXTRAÑOS	GRADO DE DIFICULTAD	MEDIDA DE DISTANCIA
M1	5	0	Bajo	Fenotípica
M2	5	0	Bajo	Fenotípica
M3	5	0	Bajo	Fenotípica
M4	5	0	Bajo	Fenotípica
M5	4	0	Mediana	Fenotípica
M6	25	0	Alta	Fenotípica

Tabla 4.2: Clasificación de los Problemas

Antes de examinar los problemas en detalle, se señalan los parámetros por defecto utilizados en el marco. La medida de distancia fenotípica utilizada corresponde a la distancia euclidiana en el espacio variable.

La inicialización de la población del AG es aleatoria y uniforme, a menos que se especifique o contrario. Todas las variables han sido codificadas en forma binaria.

M1-M4: Funciones Sinusoidales

Los primeros problemas de prueba (M1-M4) corresponden a funciones sinusoidales de una dimensión y que contienen 5 picos. Funciones similares fueron primeramente utilizadas por Goldberg y Richardson en 1987 y después en dos estudios subsecuentes. Estas cuatro funciones constituyen una meta preliminar para los métodos de AGs con nichos. A pesar de la simplicidad de estas funciones, muchos de los métodos de AGs con nichos han experimentado problemas en el pasado para

encontrar y mantener los cinco picos. De hecho los AGs tradicionales han mostrado converger rápidamente a uno de los cinco picos (1987, [44]). Se han incluido estas funciones en el marco empírico por dos razones: para establecer una primera barrera para los métodos de nichos y para comparar algunos resultados con estudios anteriores. La única variable x en M1-M4 se encuentra restringida al valor real $[0,1]$ y está codificada utilizando 30 bits. Es decodificada a través de la normalización de los 30 bits resultantes, dividiendo el número por $2^{30}-1$.

La primera función, M1, consiste de peaks uniformemente separados y de igual altura. Los máximos se localizan en los siguientes valores de x : 0.1, 0.3, 0.5, 0.7, y 0.9. todos los máximos son de altura 1.0. M1 se define a continuación y se despliega en la figura 4.5.

$$M1(x) = \sin^6(5\pi x) \quad (4.3)$$

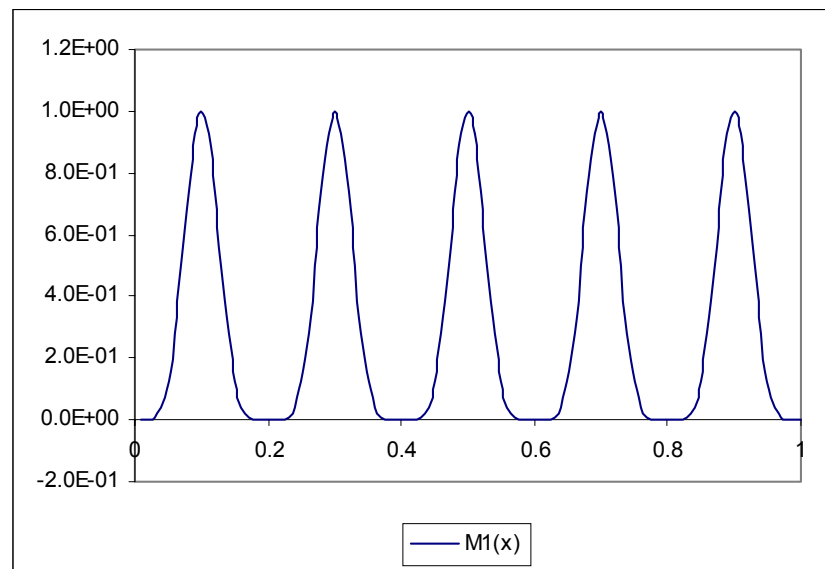


Figura 4.5: Función de Prueba M1

La función M2 consiste de peaks igualmente separados de altura no uniforme. Los máximos se localizan en los valores de x 0.1, 0.3, 0.5, 0.7, y 0.9. Los máximos poseen una altura aproximada de 1.000, 0.917, 0.707, 0.459, y 0.250 respectivamente. M2 se define a continuación y se despliega en la figura 4.6.

$$M2(x) = e^{-2 \ln(2) \left(\frac{x-0.1}{0.8}\right)^2} \sin^6(5\pi x) \quad (4.4)$$

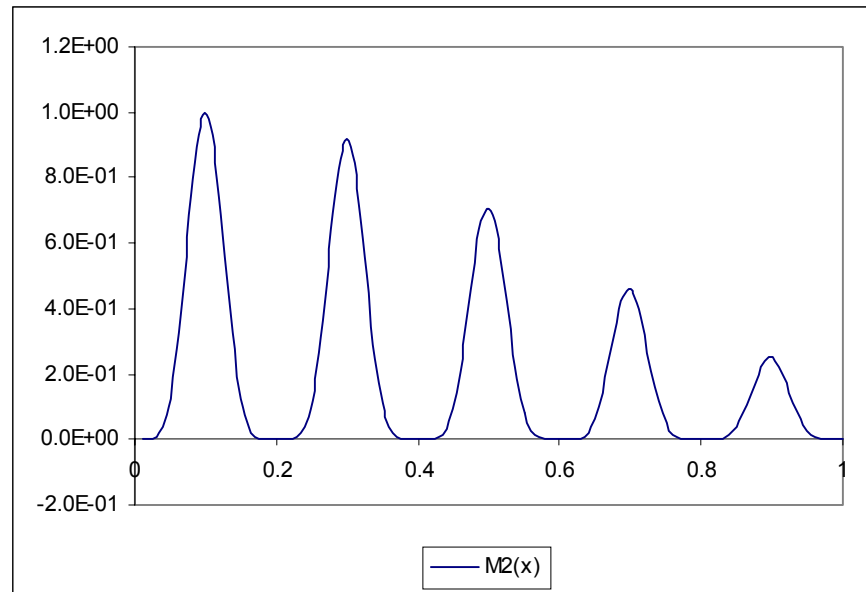


Figura 4.6: Función de Prueba M2

M3 consiste de peaks separados desigualmente que tienen una altura uniforme. Los máximos se localizan en los valores de x 0.080, 0.247, 0.451, 0.681 y 0.934. Todos los máximos son de altura 1.0. M3 se define a continuación y se despliega en la figura 4.7.

$$M3(x) = \sin^6(5\pi[x^{0.75} - 0.05]) \quad (4.5)$$

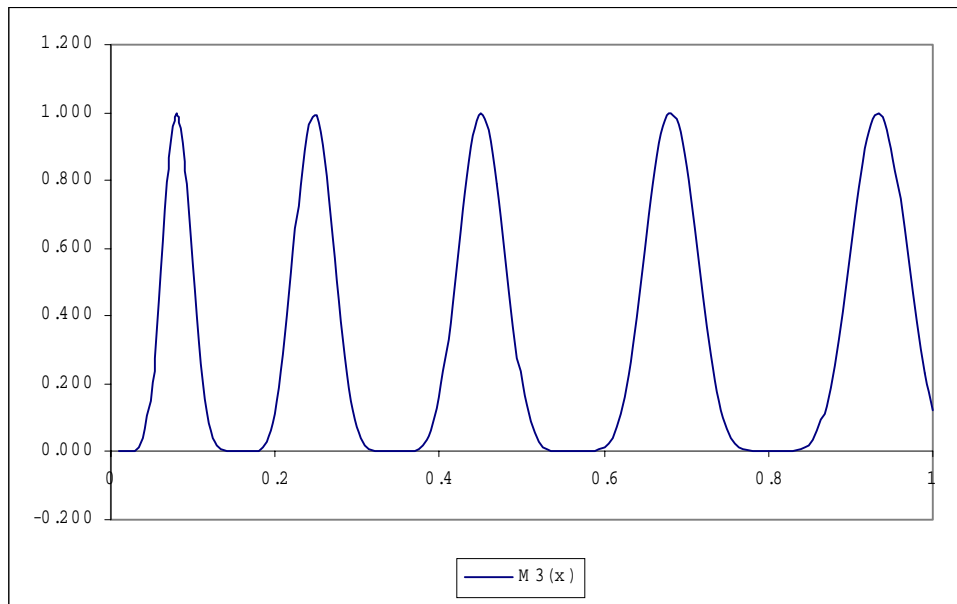


Figura 4.7: Función de Prueba M3

M4 está compuesta por peaks espaciados desigualmente y de altura no uniforme. Los máximos se localizan en los valores de x 0.080, 0.247, 0.451, 0.681 y 0.934. Las alturas aproximadas de los máximos son 1.000, 0.948, 0.770, 0.503, y 0.250 respectivamente. M4 se define abajo y se muestra en la figura 4.8.

$$M4(x) = e^{-2 \ln(2) \left(\frac{x-0.08}{0.854} \right)^2} \sin^6(5\pi[x^{0.75} - 0.05]) \quad (4.6)$$

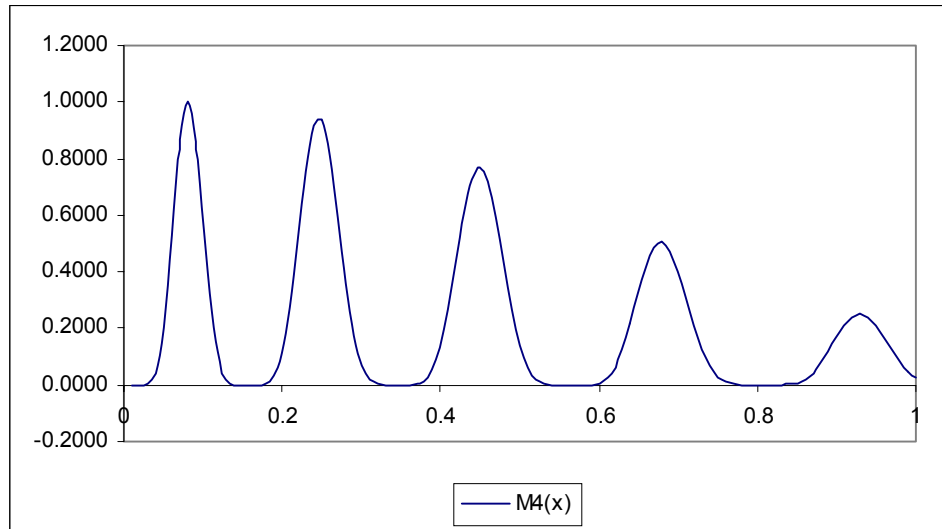


Figura 4.8: Función de Prueba M4

M5: Función de Himmelblau

M5, que se muestra en la figura 4.9, es la función modificada de Himmelblau. La función de Himmelblau es bi-dimensional y cuenta con cuatro peaks de igual altura. Al igual que las funciones M1-M4 ha sido utilizada en estudios anteriores de AGs con nichos. Esta función se incluye para ser consistente con estudios anteriores y como hito multidimensional preliminar. Las variables x é y de la función de Himmelblau se restringen al rango de valores reales $[-6,6]$ y se encuentran codificadas utilizando 15 bits por variable. El AG opera a través de un string de 30 bits, que es decodificado dividiéndolo primeramente en mitades (en los componentes x é y correspondientes) y después transformando cada resultado binario-entero en un número real del intervalo $[-6,6]$.

Los máximos de M5 se localizan en las coordenadas (x,y) : $(3.000, 2.000)$, $(3.584, -1.848)$, $(-3.779, -3283)$, y $(-2.805, 3.131)$. Todos los máximos son de altura 1.000. M5 se define a continuación:

$$M5(x, y) = \frac{2186 - (x^2 + y - 11)^2 - (x + y^2 - 7)^2}{2186} \quad (4.7)$$

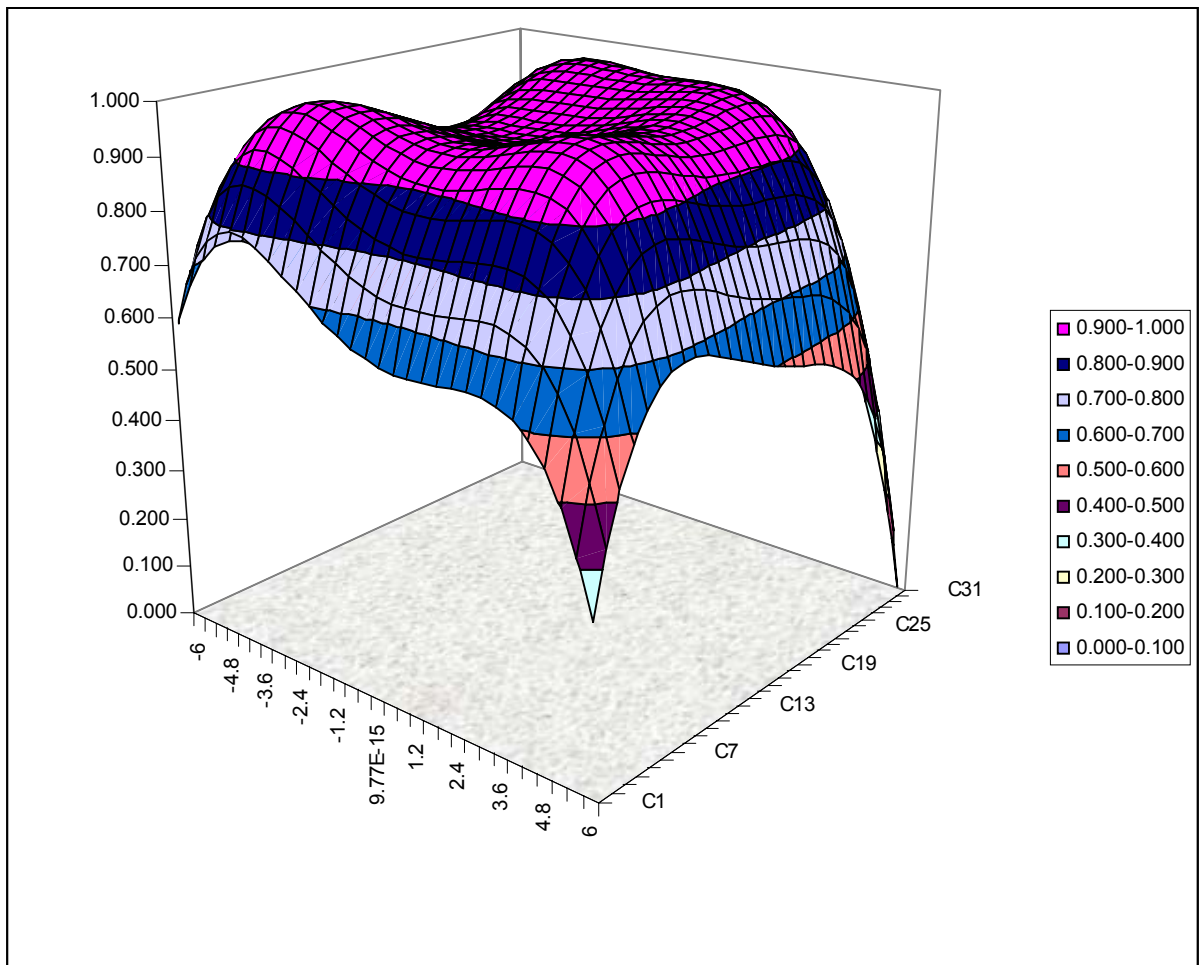


Figura 4.9: Función de Himmelblau

M6: Shekel's Foxholes

M6, mostrada en la figura 4.10, es el problema Shekel's Foxholes de la disertación de De Jong (1975, [65]). M6 es una función bi-dimensional que cuenta con 25 peaks. Ha sido utilizada en otros estudios pero siempre con el propósito principal de localizar un óptimo global simple. M6 es una función complicada debido a la gran cantidad de óptimos que presenta. Las variables x é y se encuentran restringidas al intervalo real $[-65.536, 65.536]$ y se codifican utilizando 17 bits por cada variable. De esta manera el AG opera sobre un string de 34 bits, que es decodificado primero dividiendo el string en los dos componentes x é y , para posteriormente transformar cada resultado entero binario en un número real perteneciente al intervalo $[-65.536, 65.536]$.

Los máximos de M6 están localizados en las coordenadas (x,y) : $(16i,16j)$, donde i y j representan todos los enteros del intervalo $[-2,2]$. Los 25 máximos son todos de diferente tamaño, partiendo desde 476.191 hasta 499.002. El óptimo global se localiza en $(-32,32)$. M6 se define a continuación:

$$M6(x, y) = 500 - \frac{1}{0.002 + \sum_{i=0}^{24} \frac{1}{1+i + (x-a(i))^6 + (y-b(i))^6}}, \quad (4.8)$$

donde $a(i) = 16[(i \bmod 5) - 2]$ y $b(i) = 16(\lfloor \frac{i}{5} \rfloor - 2)$.

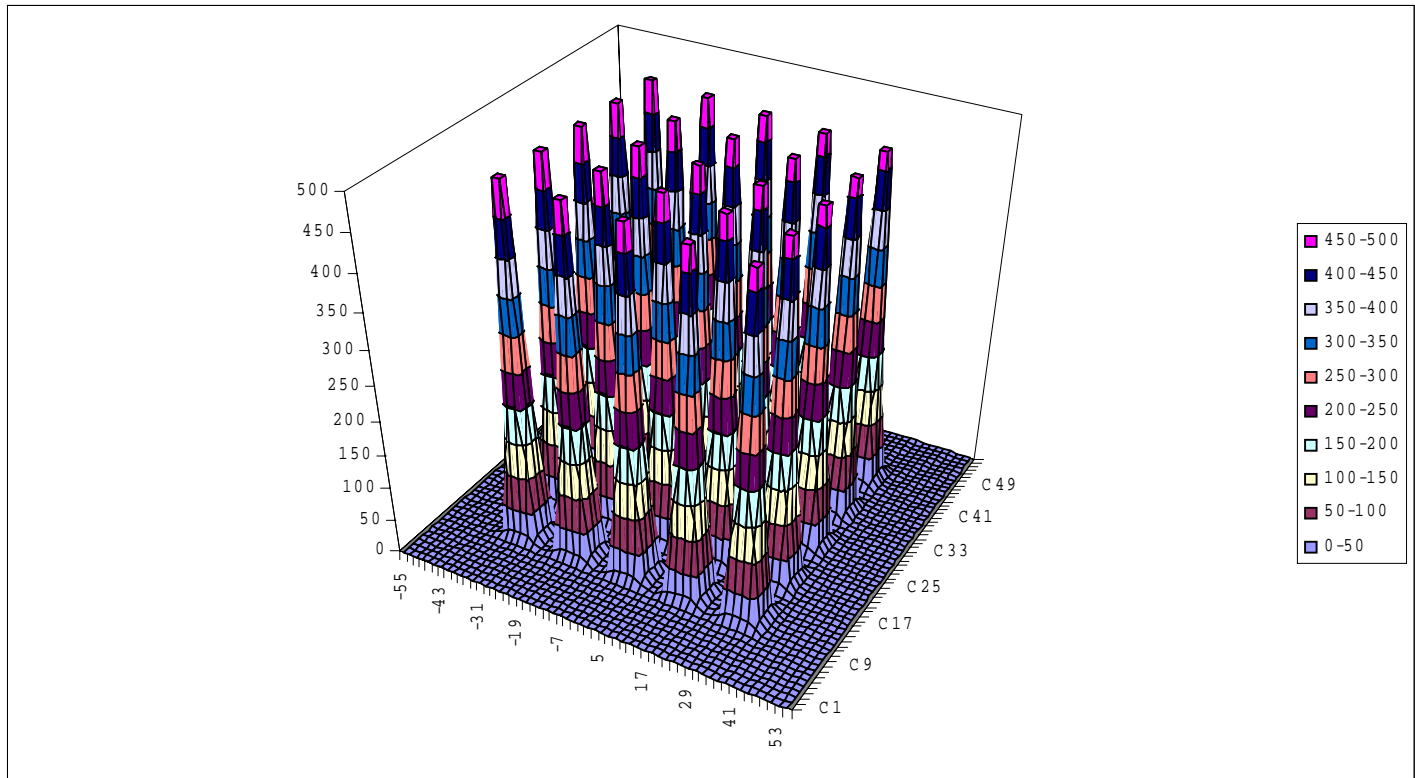


Figura 4.10: Función “Shekel’s Foxholes”

4.5.6 Poblamiento Determinístico

Un método de poblamiento, de acuerdo a lo anteriormente hablado, es un esquema de selección que inserta un nuevo elemento en la población al sobrescribir uno similar. Esta estrategia de reemplazar elementos similares de la población parece ser, intuitivamente, una fundación sólida sobre la cual desarrollar un método efectivo de nichos. Sin embargo 17 años después que De Jong (1975, [65]) introdujo el concepto de poblamiento, no se ha logrado mantener consistentemente más de dos peaks de una función multimodal, sin importar lo simple que sea.

Antes de De Jong, Cavicchio introdujo diversos esquemas en los que los hijos reemplazaban directamente a sus padres. Debido a que los padres son similares a sus

hijos, estos denominados *esquemas de preselección* califican como métodos de poblamiento. Nuevamente, intuitivamente pareciera que las técnicas de reemplazo de padres son capaces de preservar representantes de múltiples peaks. Sin embargo, como el método de poblamiento de De Jong, los esquemas de preselección nunca han exhibido capacidades para formar nichos.

A continuación se busca implementar un método de poblamiento que sea capaz de preservar exitosamente la diversidad de la población. Otro factor de motivación, desde el punto de vista ecológico, lo constituye el hecho que en poblaciones naturales puedan convivir exitosamente especies distintas y separadas a través de métodos similares al de poblamiento.

Según Mahfoud, el método de poblamiento de De Jong y el de preselección de Cavicchio no fueron exitosos debido a un conjunto de factores entre los que destacan, la creación de ruido o varianza en el proceso de selección, que conlleva la eliminación de material genético. Como se ha visto con anterioridad a pesar de la ausencia de presión de selección, la población de un AG simple converge debido al ruido del proceso de selección. Esta evidencia indica que el principal culpable del fracaso de los métodos de poblamiento es el error estocástico en el reemplazo de miembros de la población. Se puede combatir la pérdida de material genético en los métodos de poblamiento introduciendo alternativas de diseño que reduzcan o eliminen el error de reemplazo.

Al introducir mejoras potenciales al diseño de un AG se debe tener en mente que el espacio de diseño es altamente no lineal. Esto ha significado que en estudios anteriores de AGs cambios bien intencionados tengan consecuencias imprevistas, generando un desempeño más pobre que el del algoritmo original. Inclusive se han visto mejoras en un tipo de problemas, pero un comportamiento peor en otros. Cuando se va a realizar una alteración de diseño es importante tener claro el objetivo

en términos del desempeño deseado del algoritmo. Debido a la no linealidad del diseño, en algunas ocasiones se dará un paso hacia atrás que de alguna forma ayudará a alcanzar el objetivo deseado. Debido a que la principal premisa a la que están sujetos los métodos de poblamiento, es reemplazar elementos similares, se define el objetivo final en términos del proceso de reemplazo. El primer criterio es la reducción del error de reemplazo.

En el resto de este capítulo efectúan cambios sobre un método de poblamiento básico de pobre desempeño y se van analizando los resultados generado por los cambios sobre las funciones de prueba. El modelo define reemplazos correctos e incorrectos, permitiendo utilizar el número de errores de reemplazo como criterio de diseño.

Criterios de desempeño

Los dos criterios para medir el desempeño utilizados en este estudio son el número de peaks o clases que el algoritmo es capaz de mantener y el número de errores de reemplazo que se realiza. Se considera que un peak es mantenido si por lo menos existe un elemento de la población (que posee un fitness de por lo menos un 80% de la altura del peak) en la base de atracción del peak. En el caso de las funciones de prueba M1 y M2 las bases de atracción son fáciles de visualizar. Poseen una base de ancho 0.2 a lo largo del eje x en la base de cada peak y están definidas utilizando un operador de vecindad fenotípico de tamaño $\varepsilon = 1/(2^{30} - 1)$ que corresponde al incremento mínimo de la representación en el eje x . M1 y M2 son suficientemente simples como para que la calidad de las soluciones finales tienda a ser similar entre los algoritmos. Lo que se busca es mantener un alto grado de diversidad y a su vez encontrar buenas soluciones finales. La cantidad de soluciones corresponde al número de peaks mantenidos; uno de los criterios de desempeño.

El otro criterio de desempeño lo constituye el número de errores de reemplazo. Se define un error de reemplazo cuando se reemplaza un miembro de una clase o peak por un miembro de otra. El primer objetivo entonces, es desarrollar un método de poblamiento que minimice el número de errores de reemplazo y que, como resultado, mantenga el número máximo de peaks.

Algoritmos y Resultados

Se ha trabajado sobre un algoritmo básico al que se le han realizado modificaciones con el fin de eliminar parámetros innecesarios y reducir el número de errores de reemplazo. Todos los algoritmos fueron corridos durante 500 generaciones en las funciones M1 y M2 utilizando una población inicial de 100 individuos. La población fue inicializada aleatoriamente, según una distribución normal.

En un primer intento se pudo constatar que sólo se lograron mantener 2 peaks. Esto se explica por errores realizados en la etapa de reemplazo, ya que se eligieron una gran cantidad de individuos de otras clases para ser reemplazados. Esto constituye una invitación abierta para que se produzca la pérdida de material genético.

En el segundo intento se seteó la probabilidad de crossover a “1”. En teoría un método efectivo de poblamiento también es capaz de operar bajo una estructura de crossover completo. Como era de esperarse los resultados generados son muy parecidos a los anteriores, logrando conservarse tan sólo 2 peaks.

Una fuente de error mayor lo constituye el *factor de poblamiento* (FP). Hasta el momento se ha utilizado un FP muy bajo, lo que implica que sólo se está tomando una pequeña muestra de la población. De Jong notó en su estudio que mientras mayor

sea el FP, menor es el número fijo de posiciones de bits. En este caso se corrió el AG con $FP=n$. A pesar que con este cambio se incrementa el orden de complejidad del problema ($O(n^2)$ en vez de $O(n)$ comparaciones por generación) se logra obtener un algoritmo intermedio útil. Este último algoritmo es similar en complejidad al algoritmo de repartición de fitness (1987, [44]), que también realiza un ciclo a través de toda la población para calcular el fitness compartido de cada elemento. Se pudo constatar al correr este nuevo AG que el número de errores de reemplazo bajo drásticamente (casi a cero). Además se logró mantener elementos de la población de los cinco peaks del espacio de búsqueda. Sin embargo también se distribuyeron elementos a lo largo del espacio y particularmente en los valles de menor fitness de las funciones. Este último hecho se debe principalmente a que el nuevo AG presenta una *presión de selección* muy baja. La ausencia de presión de selección se puede explicar por lo siguiente. Mientras que el algoritmo selecciona dos buenas soluciones para reproducirse en cada generación, también sobrescribe los elementos más similares de la población, sin importar si son buenos, malos, o si tienen un fitness equivalente. Debido a que este algoritmo reemplaza el elemento de la población que es más parecido, lo más probable es que se reemplace una solución con fitness muy parecido. Como consecuencia entonces del método de selección se produce una cancelación de la presión de selección. El método de poblamiento selecciona elementos para la reproducción de acuerdo a su fitness. Si el elemento más cercano de la población es continuamente reemplazado, el método de poblamiento va a reemplazar al elemento que posea un fitness similar, resultando en una mejora prácticamente nula. Por el contrario, si en algún porcentaje del tiempo un elemento distinto es reemplazado por error, es altamente probable que se produzca una mejora si el elemento reemplazado tiene un fitness bajo. Este descubrimiento asombroso lleva a la conclusión que el método de poblamiento de De Jong presenta problemas de convergencia debido a los errores de reemplazo.

La siguiente modificación al algoritmo surge de una nueva observación. Al aumentar el “FP” (factor de poblamiento) a n un alto porcentaje del tiempo uno de ambos padres es reemplazado. Esto sugiere un método aproximado al de De Jong utilizando una técnica muy similar a la de preselección de Cavicchio: reemplazar cada padre con el hijo más cercano o parecido, tomando la precaución que el mismo hijo no reemplace a ambos. Este nuevo algoritmo tiene el efecto inmediato de reducir la complejidad al nivel de un AG simple, debido a que previamente se requería de $O(n^2)$ comparaciones y ahora sólo se realizan $O(n)$. En adición el FP desaparece, debido a que ahora no se trabaja con una muestra sino sólo con los padres.

El nuevo algoritmo es idéntico al anterior excepto que éste en vez de trabajar con toda la población sólo lo hace con los padres de los nuevos elementos. Cada par de hijos es insertado simultáneamente a la población de la siguiente manera. Existen dos métodos posibles para reemplazar dos padres por sus dos hijos: hijo 1 reemplaza a padre 1 é hijo 2 reemplaza a padre 2 ó hijo 1 reemplaza a padre 2 é hijo 2 reemplaza a padre 1. El reemplazo utilizado es el que entrega la menor suma de distancias absolutas entre un hijo y el padre reemplazado.

En este caso a pesar que el número de errores de reemplazo permanece muy bajo, la mayoría de las soluciones se han desplazado hacia los valles en vez de los picos. Actualmente el algoritmo exhibe una tenue presión de selección inversa, con efecto acumulativo. Algo sorprendente en un principio, pero que se puede explicar de la siguiente manera. En el algoritmo anterior los elementos tenían la posibilidad de no ser reemplazados por sus hijos, mientras que en el actual los padres son siempre reemplazados por sus hijos. De hecho se produce un efecto de eliminación de los mejores elementos de la población, reemplazándolos por elementos que a pesar de ser parecidos, frecuentemente resultan ser peores. Esto conlleva a una disminución marcada de la presión de selección.

Finalmente lo que queda pendiente en el proceso de diseño es adicionar presión de selección al algoritmo. En este caso se van a emparejar a todos los miembros de la población en forma aleatoria, sin reemplazo. Esto significa que cada miembro de la población se cruzará una vez durante cada generación. Para retornar la presión de selección, sólo se reemplazará un padre si el hijo es mejor, al igual que en la técnica de preselección. La diferencia radica en que en este caso se procesan dos padres y dos hijos al mismo tiempo y se emplea una medida de similitud para determinar qué hijo compite contra qué padre. Este método es conocido como selección de reemplazo y ha sido empleado en estudios previos de AGs.

Este algoritmo tiene la capacidad adicional verdadero paralelismo, debido a que todos los elementos de la población pueden ser procesados independientemente en cada generación. Como se muestra en las figuras a continuación se han formado agrupaciones en torno a todos los peaks.

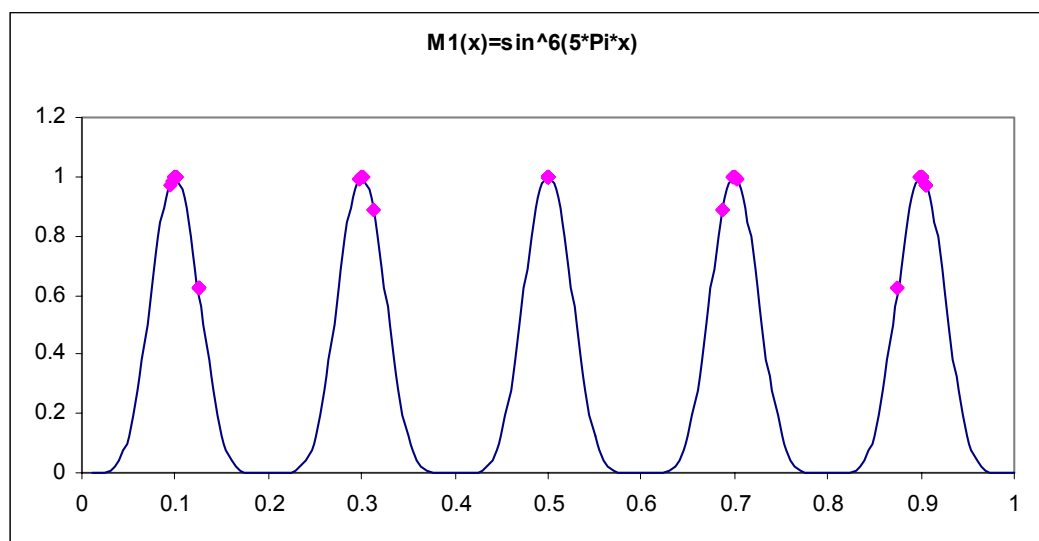


Figura 4.11: Resultado función de Prueba M1

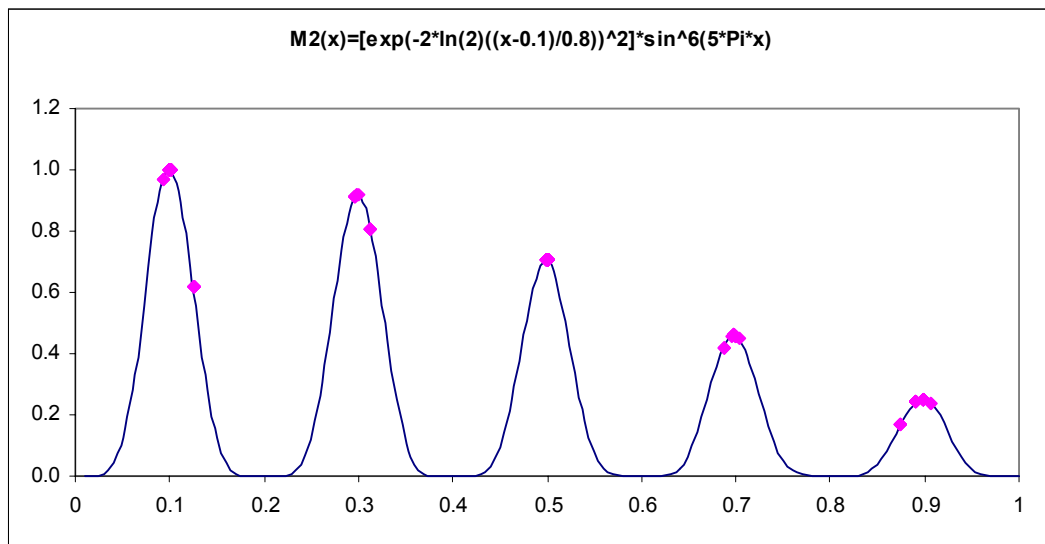


Figura 4.12: Resultado función de Prueba M2

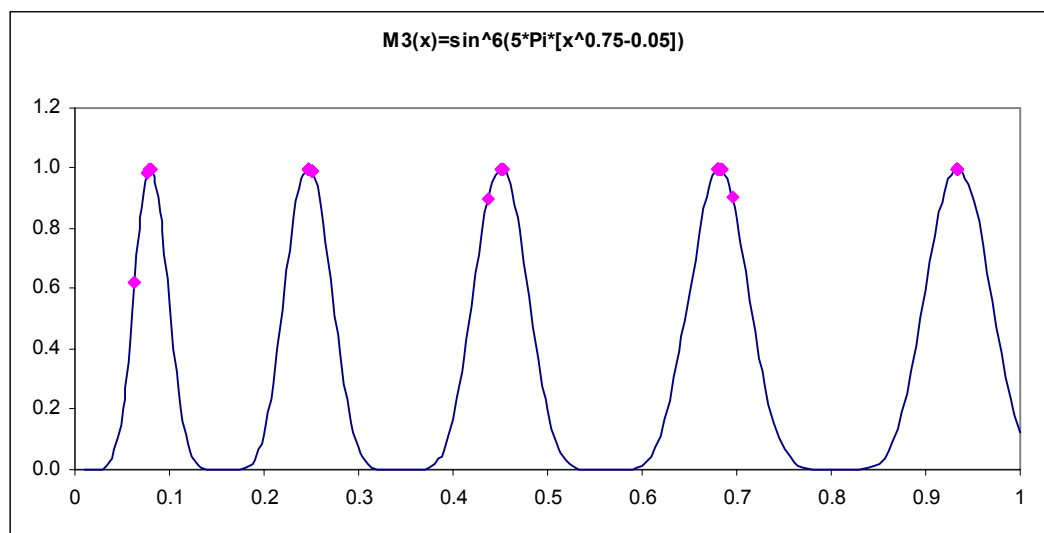


Figura 4.13: Resultado función de Prueba M3

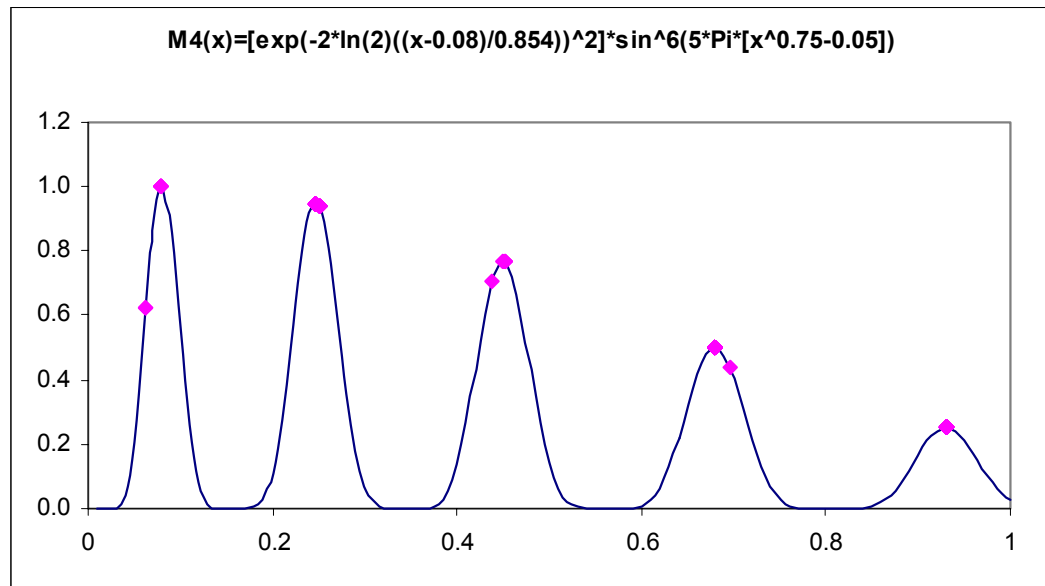


Figura 4.14: Resultado función de Prueba M4

El desempeño de este algoritmo en las funciones de prueba tri-dimensionales, M5 y M6, es absolutamente satisfactorio, dado que en la primera localizó los cuatro óptimos y los mantuvo durante todo el proceso, mientras que en la segunda localizó cerca de 16 de los 25 peaks y entre ellos se encontraba por supuesto el óptimo global.

Este nuevo método desarrollado se denomina “*poblamiento determinístico*”, porque en realidad realiza muchos de los procesos aleatorios del algoritmo de De Jong, en forma determinística.

Es ilustrativo resumir las propiedades del “*poblamiento determinístico*”. El algoritmo incorpora una medida de similitud parecida a la utilizada por el método de De Jong, pero en este caso se reemplazan los padres como en la técnica de preselección. Por otra parte, a diferencia del método de De Jong, se pueden utilizar las medidas de distancia genotípica y fenotípica indistintamente, siendo preferible, cuando es posible, utilizar la fenotípica. En M1 y M2 virtualmente se eliminaron todos los errores de reemplazo y se localizaron y mantuvieron los cinco picos. El

algoritmo es simple de implementar y más rápido que sus predecesores, debido a que realiza la selección de reemplazo a través de torneos binarios y no muestrea la población para la selección ni el reemplazo, además resulta ser inherentemente paralelo. El algoritmo completo es presentado en pseudocódigo a continuación:

Poblamiento Determinístico
(REPETIR durante g generaciones)

DO $n/2$ veces

1. Seleccionar dos padres, p_1 y p_2 , aleatoriamente sin reemplazo
2. Cruzarlos, generando c_1 y c_2
3. Aplicar mutación y otros operadores posibles, generando c'_1 y c'_2
4. IF [Distancia(p_1, c'_1) + Distancia(p_2, c'_2)] \leq [Distancia(p_1, c'_2) + Distancia(p_2, c'_1)]
 - Si $f(c'_1) > f(p_1)$ reemplazar p_1 con c'_1
 - Si $f(c'_2) > f(p_2)$ reemplazar p_2 con c'_2

ELSE

- Si $f(c'_2) > f(p_1)$ reemplazar p_1 con c'_2
- Si $f(c'_1) > f(p_2)$ reemplazar p_2 con c'_1

END IF

Figura 4.15: Pseudo código de “*poblamiento determinístico*”

V. APLICACIÓN

5.1 Introducción

Definido el problema de localización de condensadores en un sistema de distribución y la metodología de resolución basada en la aplicación de los algoritmos genéticos, se procede al desarrollo de un algoritmo computacional. En el siguiente capítulo se explica la estructura que compone el modelo.

El programa ha sido desarrollado utilizando el lenguaje de programación Visual Fortran 97, debido principalmente a la versatilidad y rapidez de éste en el manejo numérico, a lo que se suma la nueva interface visual que permite al programa crear un ambiente interactivo. El manejo del programa requiere un conocimiento elemental de Windows en sus versiones 95 ó 98, pues es el ambiente para el cual fue desarrollado.

La estructura interna es de tipo modular compuesto principalmente por tres bloques, Base de Datos, Modelación y Optimizar. Cada una de ellos compuestos a la vez por una serie de subrutinas y procedimientos libres o asociados a eventos generados por el usuario ya sea mediante el mouse o el teclado.

El programa requiere del sistema Windows 95 o superior, con una capacidad de 32 MGB de memoria R.A.M disponible en un P.C. compatible.

5.2 Estructura del Programa

El programa ha sido estructurado en un conjunto de menús y submenús que permiten al usuario interactuar con él, seleccionando diversas opciones como por ejemplo: el método de optimización a emplear y la inclusión o exclusión de las heurísticas entre otras. Además estos menús dan la posibilidad de definir muchos de los parámetros utilizados por el programa, tanto del sistema eléctrico propiamente tal (tensión máxima, tensión mínima, frecuencia, costos de equipos, etc.), como los relacionados con el desempeño del AG (n° de generaciones, probabilidad de reproducción, probabilidad de mutación, etc.).

Esto le entrega gran versatilidad al programa dando la posibilidad al usuario de ajustar el programa a sus requerimientos. Para mayor detalle se puede consultar el Anexo D. En la figura 5.1 se presenta la estructura principal del programa; menús principales y submenús asociados.

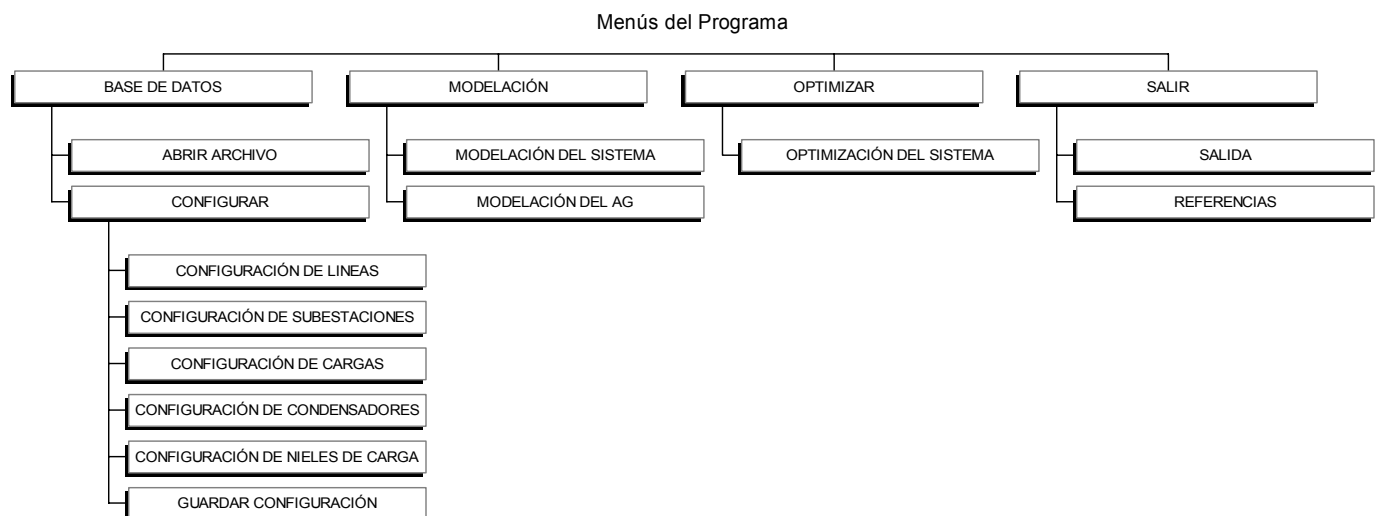
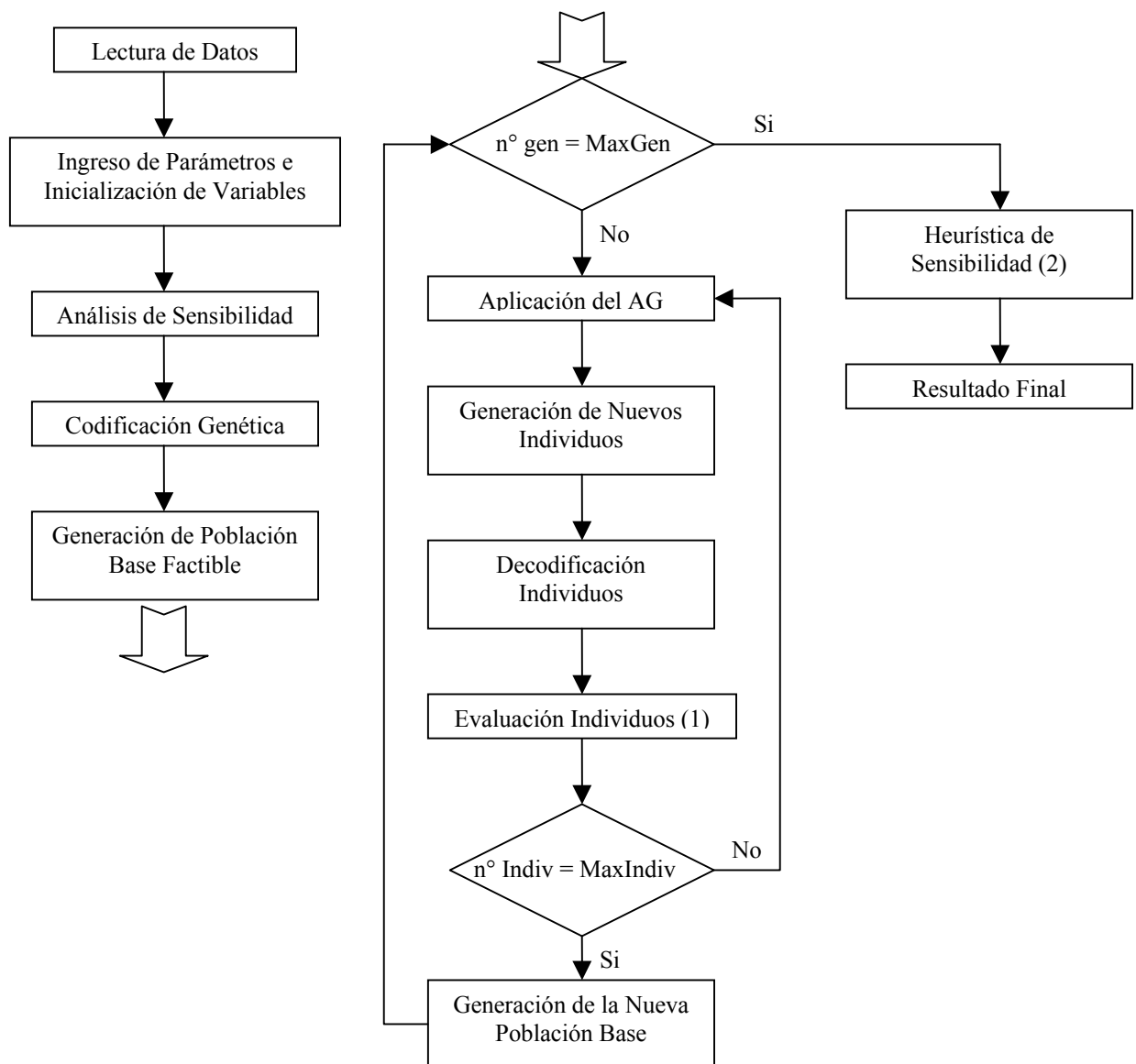
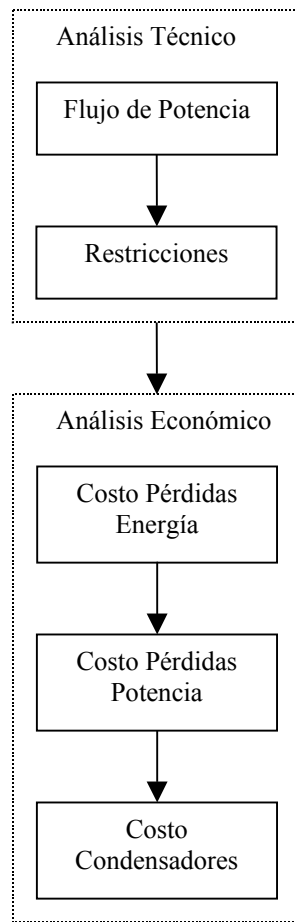


Figura 5.1: Estructura del Programa

El diagrama de bloques del programa se presenta en la figura 5.2. En él se describen todas las tareas desarrolladas por el programa en forma secuencial.



(1) Evaluación de Individuos



(2) Heurística de Sensibilidad

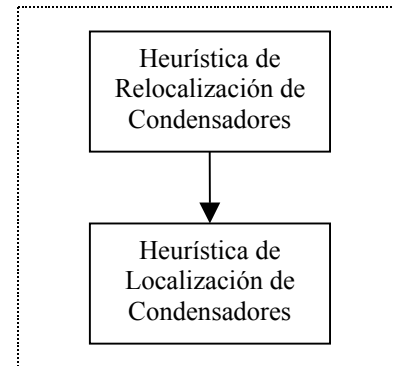


Figura 5.2: Diagrama de Bloques del Programa

En la primera parte del programa se lee el archivo de datos que contiene información del problema. En el segundo bloque se inicializan parámetros y variables que utiliza el programa y que son definidos por el usuario. Enseguida, en la tercera etapa se realiza un análisis de sensibilidad que permite determinar las barras cuyas pérdidas de potencia activa son más sensibles a una inyección de potencia reactiva. De esta forma entonces, se reduce el espacio de búsqueda del AG.

Después que se han seleccionado las barras para ser compensadas se implementa un sistema de codificación binaria para las posibles soluciones o

“individuos” que contienen toda la información necesaria sobre la localización, tamaño y tipo de condensadores. Una vez determinada la estructura de los individuos se da comienzo a la generación de la población base inicial. Esto se realiza en forma aleatoria siguiendo una distribución normal. Enseguida se da comienzo a la etapa de decodificación de los individuos a base decimal para realizar el proceso de evaluación que está compuesto por dos etapas; la primera corresponde a un análisis técnico y la segunda a uno económico.

Una vez determinada la primera familia de soluciones se procede con el AG propiamente tal. En este bloque se somete a la población anterior a diversos operadores genéticos obteniéndose como resultado una nueva población de individuos que cuentan con un mayor grado de adaptabilidad. Es aquí donde comienza el ciclo principal del programa retornando a la etapa de decodificación y evaluación de los individuos. Finalmente, éstos son ordenados según su fitness cerrándose una generación y dando paso al comienzo de una nueva.

Una vez que se ha alcanzado el número máximo de generaciones se da comienzo a las heurísticas basadas en la sensibilidad de las barras. La primera de ellas está encargada de relocalizar condensadores que inicialmente fueron instalados por el AG, siempre y cuando se logre una mejora del fitness. La segunda, y última etapa del programa consiste en localizar nuevos condensadores en el sistema eléctrico.

Finalmente entonces, es procesada la información de salida o “output” del programa y es almacenada en un archivo de salida. Además es preparada una síntesis de los resultados más relevantes del estudio que es desplegada en la pantalla.

5.3 Validación del Modelo

Parte fundamental del estudio consiste en validar el modelo propuesto, para ver su desempeño frente a diversos sistemas, tanto de prueba como reales. Esto se hará a través de un análisis comparativo entre los resultados arrojados por el programa desarrollado y un algoritmo de flujo de potencia. En la gran mayoría de los

casos los resultados se expresarán en dólares con el fin de poder compararlos con los resultados obtenidos por otros autores.

5.3.1 Sistema de Tres Barras

El sistema a estudiar se muestra en la figura 5.3. Las características de este sistema se presentan en la misma figura. Información general del sistema se entrega en la tabla 5.1.

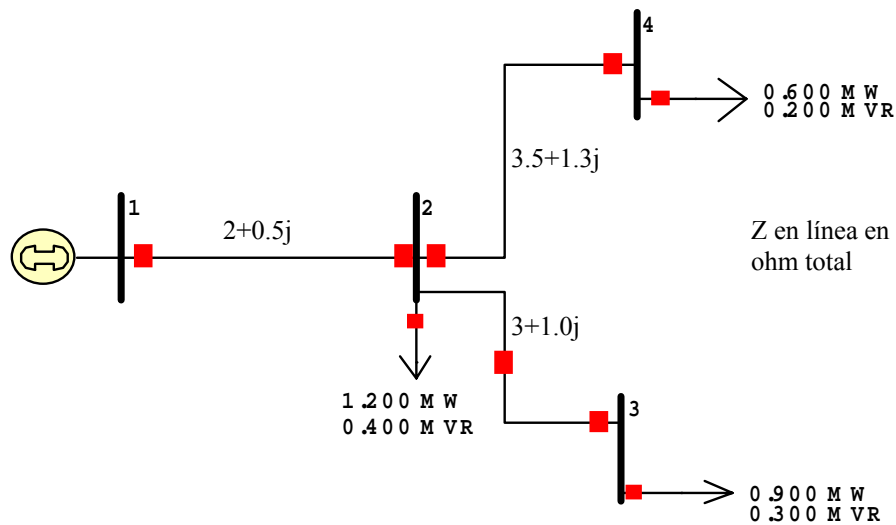


Figura 5.3: Sistema de Tres Barras

Tensión Base (KV)	23
Frecuencia Sistema (Hz)	50
Costo Energía (Mills/KWh)	60
Costo Potencia (\$/KW)	120
Costo KVAR (\$/año)	5

Tabla 5.1: Información general del Sistema de 3 Barras

El horizonte de análisis corresponde a un año que ha sido dividido en tres períodos con diversos niveles de carga. Enseguida se presenta la curva de niveles de carga discretizada que fue utilizada para este problema.

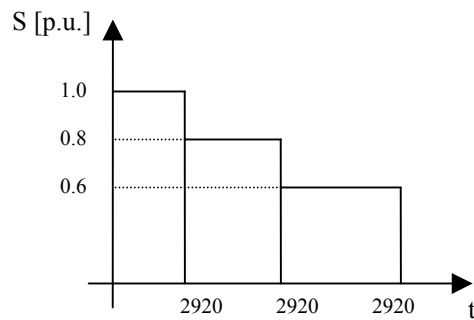


Figura 5.4: Niveles de Carga discretizados

En este estudio se considera un costo de instalación de condensadores directamente proporcional al tamaño (kVAr), tal como se indica en la figura 5.5. Se consideraron unidades de 200, 300, 500 y 1000 [kVAr].

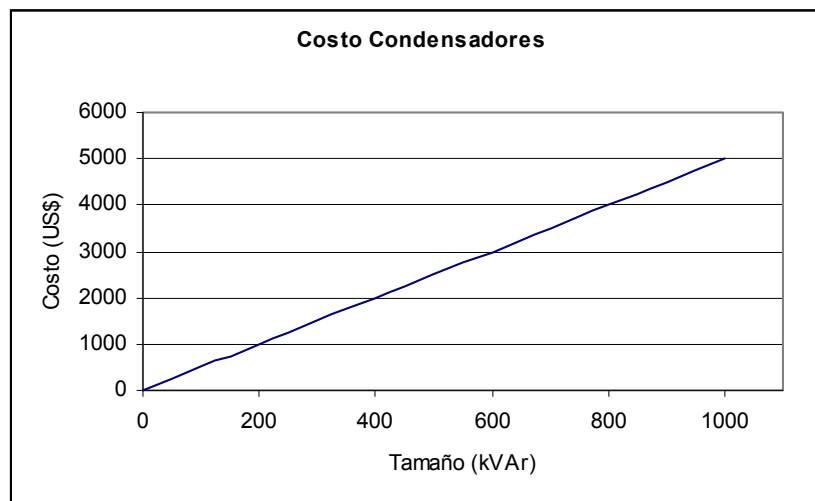


Figura 5.5: Costo Condensadores

En este caso se corrió un AG con operadores de elitismo, selección (torneo estocástico), crossover y mutación. Los parámetros del AG utilizado se pueden ver en la tabla 5.2.

Número máximo de Generaciones	50
N° Localizaciones utilizadas por el AG	3
N° Localizaciones Heurística Relocalización	4
N° Individuos por Generación	100
Probabilidad Crossover	0.8
Probabilidad Mutación	0.01
Porcentaje Elitismo	5%
N° Competidores Torneo Estocástico	3

Tabla 5.2: Parámetros AG, Sistema 3 Barras

Después de 50 generaciones se obtuvo como resultado que tan sólo la barra n° 3 debe ser compensada con el condensador más pequeño disponible (200 [kVAr]) para cada uno de los niveles de carga analizados. El resumen de los resultados se presenta a continuación:

N° Barra	Nivel 1 (1.0 p.u.)	Nivel 2 (0.8 p.u.)	Nivel 3 (0.6 p.u.)
3	200	200	200

Tabla 5.3: Compensación óptima del Sistema de Tres Barras

Enseguida se presenta un cuadro resumen con las pérdidas de energía experimentadas en el horizonte de análisis, antes y después de la compensación:

Nivel (p.u.)	Sin Compensación (MW)	Con Compensación(MW)
1.0	127.820206	119.568947
0.8	79.682144	73.925934
0.6	43.653465	40.071667

Tabla 5.4: Pérdidas de Energía del Período analizado para el Sistema de Tres Barras

Los ahorros en costos debido a la disminución de pérdidas en el sistema, después de ser compensado fueron de US\$ 1394.44. Como restricción del problema se impuso que las tensiones estuvieran dentro del rango [0.95-1.05].

Análisis de Resultados

Debido a que diversos autores utilizan distintos métodos para tratar el problema y generalmente la modelación de la función objetivo difiere según el autor que se trate, se hace difícil realizar una comparación directa de los resultados. Sin embargo, es posible realizar una comparación ordenes de magnitud.

En una primera instancia se realiza una comparación de tensiones con respecto a otro programa de control de potencia reactiva (PCR) [Castillo & Pardo, '89] para el sistema en estudio. Además se comparan las tensiones del sistema compensado con las obtenidas a través de un flujo de potencia (FP).

Barra	FP	PCR	Programa	Diferencia FP(%)	Diferencia PCR (%)
1	23.00	23.00	23.00	0.00%	0.00%
2	22.75	22.76	22.74	0.06%	0.11%
3	22.62	22.64	22.61	0.05%	0.14%
4	22.65	22.67	22.63	0.10%	0.19%

Tabla 5.5: Comparación de Tensiones, Sistema de 3 Barras

Como se puede apreciar en la tabla anterior la diferencia de tensiones en el sistema compensado es mínima. Los valores entregados por el programa son levemente inferiores, lo que se explica debido a que el objetivo principal del programa es minimizar los costos sujeto a restricciones de calidad de suministro. Por ende mientras los voltajes estén dentro de un rango razonable se buscará la solución que minimice los costos.

Respecto a la convergencia del método de solución se puede ver que es prácticamente monótona, estabilizándose a partir de la generación 20 en adelante aproximadamente.

La evolución de la convergencia puede apreciarse en la figura 5.6.

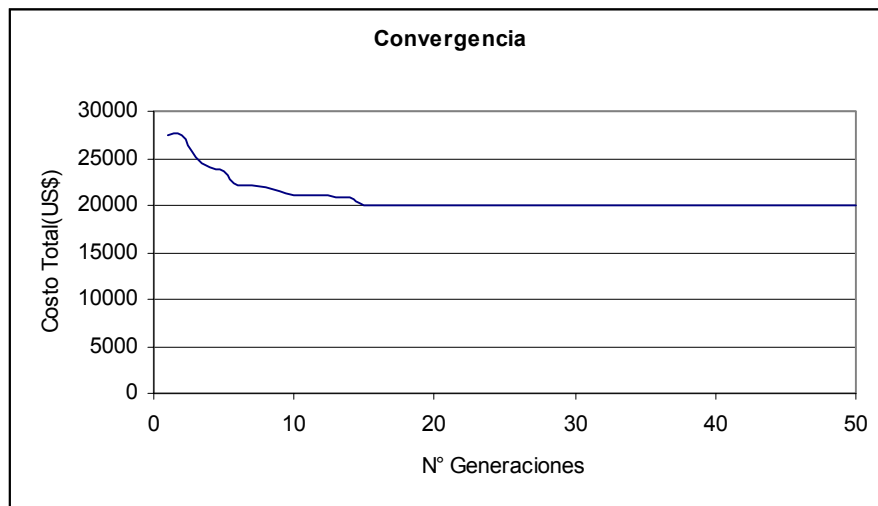


Figura 5.6: Convergencia AG, Sistema de 3 Barras

5.3.2 Sistema de nueve Barras

Este sistema fue analizado por los autores J.J. Grainger, S. Cilvanar, y K. Clinard mediante un método alternativo de compensación óptima, llamado “Criterio de las áreas iguales”. Este método sin embargo, no considera ramas laterales y su aplicación se realiza en un sistema de una sola rama.

Los datos del sistema se encuentran en el Anexo G y el sistema se muestra en la figura 5.7.

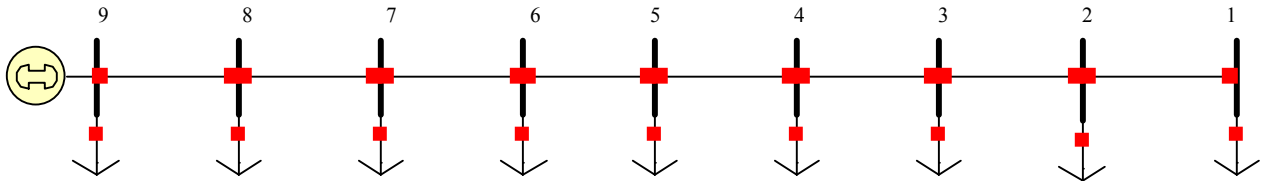


Figura 5.7: Sistema de 9 Barras

Información general del sistema se presenta en la tabla 5.6.

Tensión Base (KV)	23
Frecuencia Sistema (Hz)	50
Costo Energía (Mills/KWh)	30
Costo Potencia (\$/KW/año)	200
Costo KVAR (\$/año)	1.5

Tabla 5.6: Datos Sistema 9 Barras

La curva de carga utilizada para estudiar la compensación de este sistema es idéntica a la utilizada en el sistema anterior y viene dada por la figura 5.4.

Los condensadores utilizados en este estudio son de cuatro capacidades distintas; 200 [kVAr], 300 [kVAr], 500 [kVAr] y 1000 [kVAr]. Se cuenta con 30 unidades de cada tipo.

El programa se hizo correr durante 100, 150, 200 y 300 generaciones y se consideraron 5, 7 y 9 posibles localizaciones o barras a ser compensadas. Los parámetros utilizados por el AG se presentan en la siguiente tabla:

Número máximo de Generaciones	100,150,200,300
N° Localizaciones utilizadas por el AG	5,7,9
N° Localizaciones Heurística Relocalización	4,2,1
N° Individuos por Generación	100
Probabilidad Crossover	0.8
Probabilidad Mutación	0.01
Porcentaje Elitismo	5%
N° Competidores Torneo Estocástico	3

Tabla 5.7: Parámetros AG, Sistema 9 Barras

Los resultados arrojados por el programa para los tres casos se despliegan en la tabla 5.8. Como se puede apreciar se logran ahorros sustanciales en el sistema, especialmente cuando se compensan 9 barras.

N° Localizaciones	Disminución Pérdidas (US\$)	Ahorro (US\$)
5	35620.71	26920.71
7	38364.89	28914.987
9	42873.35	33423.35

Tabla 5.8: Ahorro del Sistema de 9 Barras compensado

La configuración óptima de condensadores para el último caso (9 barras) se logró después de 300 generaciones y se presenta en la tabla 5.9. Los tamaños corresponden a kVAr instalados.

N° Barra	Nivel 1 (1.0 p.u.)	Nivel 2 (0.8 p.u.)	Nivel 3 (0.6 p.u.)
7	1300	1000	300
9	200	200	200
8	300	300	300
4	700	200	200
6	2000	1700	1000
5	500	500	300
3	300	0.0	0.0
2	500	200	200
1	300	500	200

Tabla 5.9: Compensación óptima para Sistema de 9 Barras

La cantidad total de kVAr instalados asciende a 6300 y su costo es de US\$ 9450. De la tabla anterior se desprende además el tipo de condensadores que es necesario adquirir. Por ejemplo, para la barra n° 7 se necesita un condensador fijo de 1000 [kVAr] y otro desconectable de 300 [kVAr].

Análisis de los Resultados

Al comparar los resultados con el programa de control de reactivos de Castillo y Pardo [18], la cantidad de kVAr instalados para el caso de 9 condensadores, que es el más cercano al analizado aquí, se pudo apreciar una diferencia de tan sólo 22 kVAr, que se puede explicar principalmente a diferentes tamaños de condensadores utilizados en ambos estudios.

Además se puede ver que el sistema fue compensado en cada una de sus barras, esto se debe principalmente al costo bajo de los kVAr anuales empleados. Se desprende entonces que al subir el costo del kVAr (anual) ya no sería económicamente óptimo compensar el sistema en todas sus barras. De hecho, aumentado el costo del kVAr a 5 dólares se obtiene este nuevo esquema de compensación.

Barra n°	Nivel 1 (1.0 p.u.)	Nivel 2 (0.8 p.u.)	Nivel 3 (0.6 p.u.)
4	700	0.0	200
6	800	800	800
5	1000	1000	700
3	800	500	0.0
2	300	300	300
1	200	200	200

Tabla 5.10: Compensación óptima sistema de 9 Barras ($K_c=5$ [US\$/kVar])

En este caso sólo se han instalado 3800 kVAr en seis barras del sistema.

En este sistema, para efectos de comparación de resultados, prácticamente no se consideró regulación de tensión y se impuso que ésta se encontrara dentro del

rango [0.85-1.05]. El perfil de tensiones para cada uno de los niveles se muestra en el Anexo G.

Con respecto a la convergencia, se puede apreciar en la figura 5.9, que el algoritmo en un principio oscila levemente, para después tomar un comportamiento monótonamente decreciente. La figura muestra la convergencia del algoritmo para tres valores de localizaciones distintas y para un total de 100 generaciones.

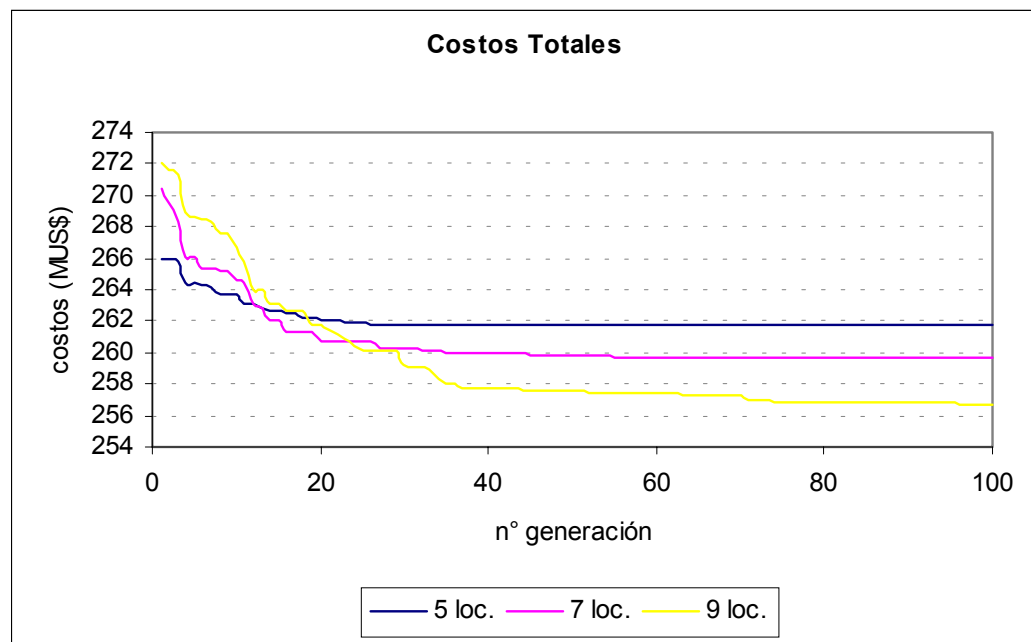


Figura 5.9: Convergencia AG, Sistema 9 Barras

Además, se puede observar que el menor costo se obtiene cuando el AG considera las 9 barras. Por ello, para este caso particular se analizó lo que ocurre al aumentar el número de generaciones. Los resultados se pueden ver en la tabla 5.11 y en la figura 5.10.

N° Generaciones	Costos (MUS\$)		
	Condensadores	Pérdidas	TOTAL
100	10.80	245.915	256.715
150	7.20	248.072	255.272
200	9.45	245.762	255.212
300	9.45	245.758	255.208

Tabla 5.11: Costos Totales v/s n° Generaciones

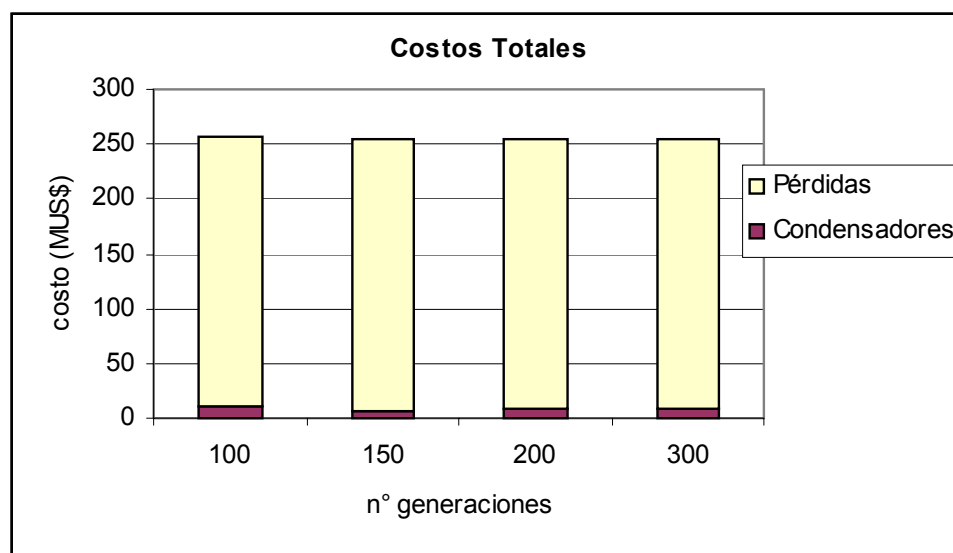


Figura 5.10: Composición de Costos Totales v/s n° Generaciones

Se puede apreciar en el gráfico anterior que el impacto de los costos de los condensadores es mínimo frente a la magnitud de los costos de las pérdidas de potencia y energía. Además se pudo constatar que con 300 generaciones el sistema alcanza su mínimo costo. Este valor se mantiene de la generación 249 en adelante.

5.3.3 Sistema de treinta Barras

Corresponde a un sistema de distribución de 23 [kV] que posee treinta ramas, cinco de las cuales son laterales. El diagrama unilineal se puede observar en la figura 5.11. La información correspondiente a la resistencia y reactancia de cada sección se encuentra en [11]

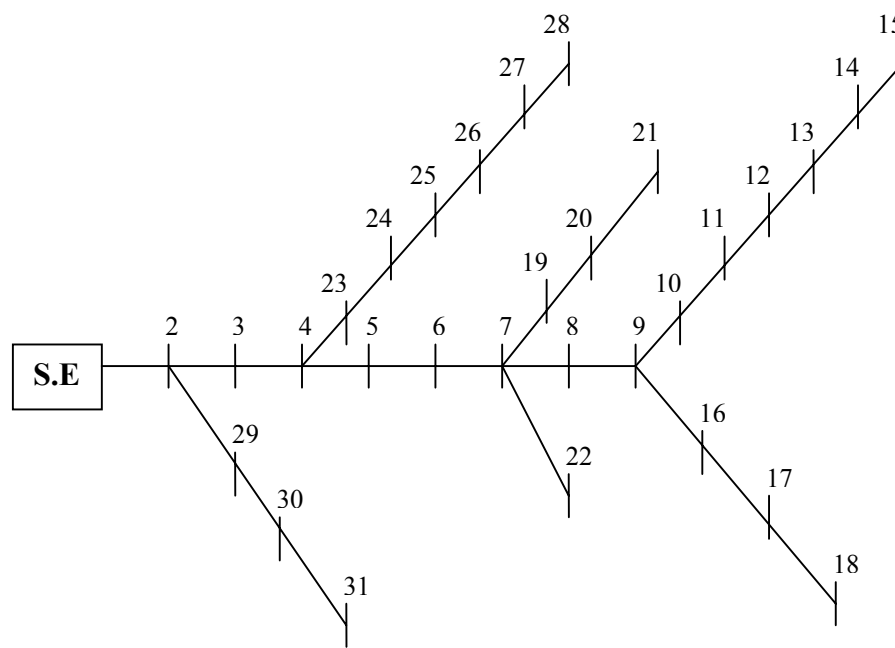


Figura 5.11: Sistema de 30 Barras

Los coeficientes de costo para las pérdidas de potencia de punta y energía son, respectivamente, $K_p = 120$ [US\$ /kW/año] y $K_e = 30$ [mills/kWh]. Además, se consideró un costo para los condensadores proporcional al kVAr de 0.75 [US\$/kVAr/año].

Este valor considera una vida útil de 20 años para los equipos y una tasa de descuento de un 15% anual.

La importancia de analizar este sistema radica en que es similar al empleado por los autores Srinivasan Sundhararajan y Anil Pahwa (1994, [19]) que trabajaron con un algoritmo similar al desarrollado en este estudio. Además, los autores J.J. Grainger y S. Civanlar (1985, [11]) utilizaron este sistema para probar la eficacia de su metodología desarrollada para control de reactivos. Lamentablemente, los autores no incluyen toda la información del sistema necesaria para trabajar con él. Específicamente, no incluyen información sobre el largo de las ramas, parámetro de gran incidencia sobre los resultados finales que entrega el programa en cuanto a la localización y dimensionamiento de los bancos de condensadores. Debido a ello fue necesario suponer los valores. Como consecuencia, se realizará una comparación considerando rango de valores similares y no exactos entre los resultados entregados por diversos autores y los del programa desarrollado.

Enseguida se presenta un cuadro resumen con los parámetros más relevantes del sistema estudiado.

Tensión Base (KV)	23
Frecuencia Sistema (Hz)	50
Costo Energía (Mills/KWh)	30
Costo Potencia (\$/KW/año)	120
Costo KVAR (\$/año)	0.75

Tabla 5.12: Datos del Sistema de 30 Barras

Para la compensación del sistema fueron utilizados cuatro tamaños distintos de condensadores 300, 600, 900 y 1200 [kVAR]. Además, se consideró una existencia de 30 unidades de cada tamaño de condensador. Cabe señalar que se impuso una

regulación de tensión de un 5%, lo que obliga que las tensiones de cualquier solución encontrada estuvieran en el rango [0.95-1.05].

Los parámetros del AG utilizado se resumen en la siguiente tabla:

Número máximo de Generaciones	50, 100, 200
Nº Localizaciones utilizadas por el AG	30
Nº Localizaciones Heurística Relocalización	0
Nº Individuos por Generación	100
Probabilidad Crossover	0.8
Probabilidad Mutación	0.01
Porcentaje Elitismo	5%
Nº Competidores Torneo Estocástico	3

Tabla 5.13: Parámetros AG, Sistema de 30 Barras

Análisis de Resultados

Con el programa se evaluaron 50, 100 y 200 generaciones considerando 30 localizaciones o barras a ser compensadas.

Los resultados arrojados por el programa para los tres casos se despliegan en la tabla 5.14. Como se puede apreciar se logran ahorros sustanciales en el sistema, especialmente cuando se compensan 9 barras.

Caso	N° Localizaciones	N° Generaciones	Disminución Pérdidas (US\$)	Ahorro (US\$)
1	10	10	22706.259	18656.259
2	20	20	27201.884	18651.884
3	30	50	27667.956	18217.956
4	10	100	23451.29	20526.290
5	20	100	28136.656	24086.656
6	30	100	28373.198	24323.198

Tabla 5.14: Ahorro del Sistema de 30 Barras compensado

Como se puede apreciar en la tabla anterior el ahorro máximo se alcanza en 100 generaciones y considerando 30 localizaciones. Además, se puede ver en los tres primeros casos que a pesar que el número de generaciones aumenta, el ahorro disminuye levemente. Esto se explica, debido a que el número de localizaciones también se ha ido aumentando gradualmente, lo que conlleva a que exista un mayor número de posibles esquemas de compensación y por lo tanto si se aumenta el número de localizaciones, se requiere de un mayor número de generaciones para que el algoritmo encuentre una mejor solución.

Debido a que en la implementación del sistema las distancias debieron ser supuestas, tan sólo fue posible evaluar un nivel de carga, ya que al variar éste el flujo de potencia divergía, sin embargo los resultados pueden ser comparados con los obtenidos por otros autores.

Antes que todo, se presenta el esquema de compensación obtenido para el caso que considera 100 generaciones en la tabla 5.15. Los tamaños de los condensadores están en kVAr.

N° Barra	Nivel 1 (1.0 p.u.)
3	300
23	1200
26	300
28	900
8	900
19	300
20	300
18	600
13	300
15	300

Tabla 5.15: Compensación óptima para Sistema de 30 Barras

La cantidad total de kVAr instalados por el programa es de 5400 [kVAr], mientras que los autores Grainger y Civanlar compensan el sistema con 3639 [kVAr]. Esta diferencia se puede explicar debido a que los autores utilizan una curva de carga que varía en el período de análisis, alcanzando el nivel peak (1.0 p.u.) sólo al final de éste y por un pequeño lapso de tiempo. Además, su modelación incluye costos de los condensadores no lineales y tan sólo analizan casos que abarcan hasta 5 condensadores.

Por otra parte, al comparar los resultados con los obtenidos por los autores S. Sundhararajan y A. Pahwa que también utilizan un AG para resolver el problema, se pudo constatar que la compensación obtenida por ellos para el mismo caso es de 5400 [kVAr], considerando niveles de carga discretos y costos de condensadores proporcionales al kVAr ($K_c=0.75$ [US\$/kVAr/año]). Al analizar este caso con el costo de equipos antes señalado, el programa obtiene una compensación óptima de 5400 [kVAr] para el nivel máximo de carga. Resultado que concuerda exactamente con los obtenidos por los autores antes mencionados. Las mayores diferencias se

presentan en la localización de los condensadores, tal como se puede apreciar en la tabla 5.16.

Modelo Base	Programa
13	3
14	8
15	13
18	15
20	18
21	19
22	20
27	23
28	26
31	28

Tabla 5.16: Localización de Condensadores en sistema de 30 Barras

Esta diferencia con respecto a las localizaciones se puede deber fundamentalmente a la distinta magnitud de las impedancias totales de cada una de las ramas, producto del supuesto realizado.

Con respecto a la convergencia del AG, se puede ver en la figura 5.12 que el AG converge monótonamente decreciente durante 100 generaciones y para tres valores distintos de localizaciones.

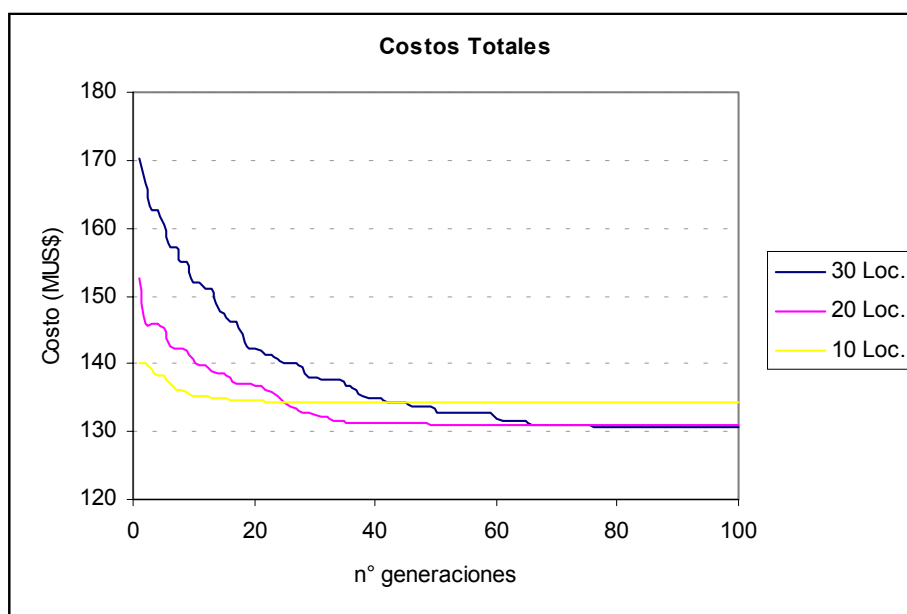


Figura 5.12: Convergencia AG, Sistema de 30 Barras

Además, se puede apreciar que al considerar un mayor número de localizaciones, se pueden conseguir costos totales más bajos. Otro hecho importante de destacar es que al considerar un mayor número de localizaciones las primeras iteraciones del AG generan resultados pobres debido al mayor número de variables en la modelación. Se puede observar además, que para 20 y 30 localizaciones el AG converge a un valor muy parecido desde la generación 70 en adelante.

5.3.4 Conclusiones de los Sistemas de Pruebas

Como se ha podido constatar en los sistemas de prueba el programa presenta gran versatilidad para analizar diversos casos con diferente grado de dificultad. Tal vez una de las restricciones más fuertes es la del flujo de potencia, específicamente la relacionada con la convergencia del sistema para las diversas soluciones. Se pudo apreciar en el último caso analizado que el sistema presentaba gran sensibilidad con

respecto a los parámetros de las líneas, lo que provocaba que para muchas de las posibles soluciones entregadas por el AG el flujo de potencia divergiera.

Es importante destacar que a pesar que se utiliza una heurística de sensibilidad para determinar los nodos más sensibles a una inyección de potencia reactiva, si se evalúa un mayor número de localizaciones por el AG la solución final mejora notablemente. Esto en gran medida, se debe a la adecuada implementación del AG que le permite analizar un amplio espectro del espacio de búsqueda. Además el algoritmo presenta una gran presión de selección lo que provoca que el AG sea capaz de alcanzar los óptimos locales.

Otro hecho relevante de mencionar es que al considerar costos proporcionales al kVAr bajos el algoritmo tiende a encontrar soluciones que localizan bancos de condensadores en un gran número de barras. Este hecho se puede evitar de dos formas. La primera de ellas consiste en elevar el costo de los condensadores con el fin de aumentar su participación en la función objetivo y la segunda en introducir un nuevo término en la función objetivo que penalice la compensación de un gran número de barras.

Una característica importante del programa es que fue desarrollado con una estructura modular lo que facilita futuras modificaciones y la implementación de mejoras. A modo de ejemplo, se puede incluir una alternativa que considere costos discretos de condensadores para diversos tamaños.

Como conclusión final, es importante señalar que una adecuada compensación puede reducir considerablemente los costos de operación del sistema, hecho que se refleja en una disminución notable de las pérdidas de potencia activa que circula por las ramas.

VI. CONCLUSIONES GENERALES

En este estudio se ha propuesto una metodología de solución basada en AG para resolver el problema de localización de condensadores en sistemas de distribución.

Los AGs constituyen un método bien adaptado para la resolución de problemas de naturaleza combinatorial. Son capaces de manejar eficientemente, tanto variables discretas como continuas, sin realizar ningún cambio en el mecanismo de búsqueda. Debido al gran número de evaluaciones de la función objetivo, esta técnica es lenta comparada con algoritmos de optimización tradicionales. Sin embargo, la evaluación de la función objetivo es independiente para cada string, lo que presenta la posibilidad de que pueda ser procesada en paralelo. Este paralelismo implícito hace que este método sea el más adecuado para el desarrollo de la optimización computacional paralela.

Además, se ha desarrollado una heurística rápida que permite medir la sensibilidad de las pérdidas de potencia activa frente a una inyección de potencia reactiva, con ello se puede disminuir el espacio de búsqueda del AG y acelerar el proceso general.

Con anterioridad se han planteado metodologías análogas basadas en AGs, pero ninguna ha sido capaz de garantizar de alguna forma alcanzar el óptimo global del problema. En este caso, dado el estudio realizado sobre AGs se presenta la posibilidad cierta de lograr encontrar el óptimo global de un problema dado, gracias a la implementación de la técnica de nichos que evita la pérdida de material genético y con ello minimiza la posibilidad de una convergencia prematura. Además, presenta la cualidad, que dada una población inicial suficientemente grande es capaz de muestrear prácticamente todo el espacio de búsqueda.

Una gran ventaja de la metodología planteada es la capacidad que posee de considerar tanto la función objetivo como restricciones sin ningún tipo de aproximación, lo que se traduce en una mayor exactitud de las soluciones encontradas.

Una característica importante del programa es la facilidad del ingreso y modificación de información, a través de interfaces gráficas que evitan que el usuario deba interiorizarse en como debe estar estructurada la información. Más aún, es posible diseñar un sistema de distribución completo con estas interfaces.

Un aspecto que hay que considerar al evaluar el costo de instalación de un banco de condensadores, es que aquellos de conexión fija son más baratos que los on/off. Como resultado del programa se obtiene el esquema de compensación para diversos niveles de carga, donde se puede ver los condensadores que deben ser instalados en forma fija y cuáles deben ser on/off. Como consejo práctico se recomienda asignar aquellos bancos que están más de un 95% del tiempo conectados como fijos. De todas formas, existe un compromiso entre la reducción de ahorro por energía y menor costo de dichos bancos que puede ser evaluado matemáticamente.

A medida que aumenta el número de localizaciones a considerar por el AG se ha podido constatar en los casos analizados que los costos totales del sistema disminuyen considerablemente. Sin embargo, existe un compromiso entre la calidad de la solución y los recursos computacionales requeridos que debe ser evaluado a la hora de buscar una solución.

Las heurísticas de relocalización y localización de condensadores logran mejorar las soluciones encontradas por el AG, en la medida que se considere un número bajo de iteraciones.

La combinación de la heurística de sensibilidad, un algoritmo de propósito global y un conjunto de heurísticas hacen que este método sea lo suficientemente robusto como para encontrar soluciones de alta calidad.

Finalmente se puede señalar que este algoritmo se puede aplicar exitosamente en otros problemas de optimización pertenecientes al campo de la distribución de energía eléctrica como la planificación de sistemas eléctricos considerando múltiples etapas, control de potencia reactiva, programación de la generación, etc.

VII. DESARROLLOS FUTUROS

El programa ha sido desarrollado considerando sólo el caso de bancos de condensadores de conexión on/off. Sin embargo, a medida que el control que se ejerce sobre los condensadores es más próximo al continuo, la efectividad de éste se aproximará más al caso ideal de control continuo. En el caso de bancos que permitan un esquema de conexión distinta al caso típico considerado (on/off), como por ejemplo sería el caso de tres taps de conexión, habría que modificar el programa para que considere esta nueva posibilidad.

En lo que respecta al AG, la calidad de las soluciones encontradas dependen fuertemente de los parámetros genéticos (probabilidad crossover, probabilidad mutación, etc.). En el programa estos parámetros deben ser determinados por el usuario, sin embargo sería de gran interés implementar una metodología que permita definir el conjunto de valores más adecuados de los parámetros a ser empleados por el AG con el fin de que el usuario no necesariamente deba interiorizarse en el funcionamiento de este tipo de algoritmos para poder definir los valores de los parámetros. Para profundizar sobre este tema se recomienda ver [20].

Respecto a la determinación de la población inicial, la técnica de fitness compartido permite determinar el tamaño óptimo de la población inicial de manera de minimizar las pérdidas de representantes de diversos nichos hasta una cierta generación.

La estructura modular del programa permite facilitar el proceso de comprensión de su código con el fin de introducir modificaciones que podrían hacerlo aplicable a una mayor variedad de condiciones. Por ejemplo, se puede modificar en el sentido que considere un mayor rango de posibilidades de representación de la carga sobre el sistema (curva de carga), de hecho se podría implementar un sistema que

considere una curva de carga continua, otra posibilidad es que consideren cargas P, Q dependientes del voltaje. En fin, existe una gran variedad de modificaciones que es posible implementar con el fin de abarcar un mayor número de casos y mejorar el rendimiento del programa.

BIBLIOGRAFÍA

[51] Adachi, N. & Matsuo, K. (1991), “**Ecological dynamics under different selection rules in distributed and iterated prisoner’s dilemma game**”. Lecture notes in computer science: Parallel problem solving from nature, 496, 388-394.

[14] M.E. Baran and F.F. Wu, “**Optional capacitor placement on radial distribution systems**”, IEEE Trans. on Power Delivery, vol.4, pp 725-734, January 1989.

[15] M.E. Baran and F.F. Wu, “**Optional sizing of capacitor placed on a radial distribution system**”, IEEE Trans. on Power Delivery, vol.4, pp 735-743, January 1989.

[12] D.R. Brown, “**Performance analysis of a variable source of reactive power on distribution system primary feeders**”, IEEE Trans. On Power Apparatus and Systems, vol. PAS-100, pp. 4364-4372, November 1981.

[43] Bäck, T. (1994), “**Selective Pressure in Evolutionary Algorithms: A characterization of selection mechanism**”. Proceeding of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, 57-62.

[18] M. Castillo, G. Pardo, “**Control de Voltaje y Potencia Reactiva en sistemas de Distribución**”, Memoria para optar al título de Ingeniero Civil de Industrias mención Eléctrica, Pontificia Universidad Católica de Chile, 1989.

[66] Cavicchio, D.J., Jr. (1970), “**Adaptive search using simulated evolution**”. Unpublished doctoral dissertation, University of Michigan, Ann Arbor.

[74] Cedeño, W., & Vemuri, V. (1992). ”**Dynamic multimodal function optimization using genetic algorithms**”. Proceedings of the XVIII Latin-American Informatics Conference, 292-301.

- [16] H.D. Chiang, J.C. Wang, O. Cockings H.D. Shin, “**Optimal capacitor placements in distribution systems: Part I and Part II**”, IEEE Trans. on Power Delivery, vol.5, pp 634-649, January 1990.
- [21] H. Chiang, J. Wang, J. Tong, G. Darling, “**Optimal Capacitor placements and Control in Large-Scale unbalanced distribution Systems: System modeling and new Formulation**”, IEEE Trans. On Power Systems, vol. 10, N° 1, pp. 356-362, February 1995.
- [26] W. Chin, L. Proenca, V. Miranda, “**Demonstrating an efficient capacitor location and sizing method for distribution systems – application to the Macau Network**”, INESC- Macau and INESC, Portugal
- [76] Culberson, J. C. (1992). “**Genetic invariance: A newparadigm for genetic algorithm design (Technical Report N° TR92-02)**”. Edmonton: University of Alberta, Department of Computing Science.
- [3] W.H. Cuttino, “**Extending the Use of Shunt Capacitors by Means of Automatic Switching**”, AIEE Transactions, vol. 63, pp. 674-678, 1944.
- [4] W.H. Cuttino, “**Automatic Swiching Schemes for Capacitors**”, AIEE Transactions, vol. 66, pp. 311-314, 1947.
- [49] Dasgupta, D., & McGregor, D.R. (1992), “**Nonstacionary function optimization using the structured genetic algorithm**”. In R. Männer & B. Manderick (Eds.), Parallel problem solving from nature, 2 8pp. 145-154). Amsterdam: Elsevier.
- [65] De Jong, K.A., (1975), “**An analysis of the behaviour of a class of genetic adaptive systems**”. (Doctoral dissertation, University of Michigan). Dissertation abstracts international, 36(10), 5140B. (University Microfilms N° 76-9381).

[25] M. Delfanti, G. Granelli, P. Marannino, M. Montagna, “**Optimal Capacitor Placement using Deterministic and Genetic Algorithm**”, IEEE Trans. On Power Systems, vol. 15, N° 3, pp. 1041-1046, August 2000.

[13] K. Desai and D.R. Brown; “**Muliple variable sources of reactive power on distribution system primary feeders**”, IEEE Trans. On Power Apparatus and Systems, vol. PAS-100, pp. 4674-4680, december 1982.

[17] H. Duran, “**Optimum number, location, and size of shunt capacitors in radial distribution feeder: A dinamic programming Aproach**”, IEEE Trans. On Power Apparatus and Systems, vol. 87,N° 9, pp. 1769-1774, January 1983.

[9] A.A. El Kib, “**Capacitive-Compensation Planning and Operation for Primary Distribution Feeders**”, Disertación de Doctorado,1984, Universidad de Carolina del Norte.

[10] A.A. El Kib, J.J. Grainger, K.N. Clinard, L.J. Gale, “**Placement of fixed and/or non-simultaneosly switched capacitors on unbalanced tree-phase feeders involving laterals**”, IEEE Trans. On Power Apparatus and Systems, vol. PAS-104, pp. 3298-3305, November 1985.

[56] Elo, S. (1994), “**A parallel genetic algorithm on the CM-2 for multi-modal optimization**”. Proceeding of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, 818-822.

[2] T.H. Fawzi, S.M. El-Sobki, M.A. Abel-Halim, “**New Approach for the Aplication of Shunt Capacitors to the Primary Distribution Feeders**”, IEEE Trans. On Power Apparatus and Systems, vol. 102, pp. 10-13, January 1983.

- [37] F.García Merayo, “**Lenguaje de Programación Fortran 90**”, ITP-Paraninfo, 1998.
- [34] David E. Goldberg, “**The Design of Innovation: Lessons from Genetic Algorithms, Lessons for the Real Life**”, Illigal Report n° 98004, University of Illinois at Urbana-Champaign, February 1998.
- [38] Goldberg, D.E. (1989c), “**Genetic Algorithms in search, optimization, and machine learning**”. Reading, MA: Addison-Wesley.
- [40] Goldberg, D.E. (1983), “**Computer - aided gas pipeline operation using Genetic Algorithms and rule learning**”, (Doctoral dissertation, University of Michigan). Dissertation Abstract International, 44(10), 3174B. (University Microfilms N° 8402282).
- [41] Goldberg, D.E., & Rudnick, M, (1991), “**Genetic Algorithms and the variance of fitness**”. Complex Systems, 5, 265-278.
- [44] Golberg, D.E., and Richardson, J. (1987), “**Genetic Algorithms with sharing for Multimodal Function Optimization**”. Genetic Algorithms and there Applications: Proceedings of the Second International Conference on Genetic Algorithms, 41-49.
- [47] Goldberg, D.E., and Smith, R.E. (1987), “**Nonstationary function optimization using genetic algorithms with dominance and diploidy**”. Genetic Algorithms and there Applications: Proceedings of the Second International Conference on Genetic Algorithms, 59-68.
- [48] Goldberg, D.E., Korb, B. & Deb, K. (1989), “**Messy genetic algorithms: Motivation, analysis and first results**”. Complex Systems, 3, 493-530.
- [5] J.J. Grainger, S.H. Lee, A.A. Kib, “**Design of a Real-Time Control Scheme for Capacitive Compensation of Distribution Feeders**”, IEEE Trans. On Power Apparatus and Systems, vol. 101, pp. 1012-1020, 1981.

- [6] J.J. Grainger, S.H. Lee, “**Capacity Release by Shunt Capacitor Placement on Distribution Feeders**”, IEEE Trans. On Power Apparatus and Systems, vol. 101, pp. 1236-1244, 1982.
- [8] J.J. Grainger and S.H. Lee, “**Optimum size and location of shunt capacitors for reduction of losses on distribution feeders**”, IEEE Trans. On Power Apparatus and Systems, vol. PAS-100, pp. 1105-1116, March 1981.
- [11] J.J. Grainger and S. Civanlar, “**Volt/Var control on Distribution Systems with Lateral Branches Using Shunt Capacitors and Voltage Regulators: Part I, Part II and Part III**”, IEEE Trans. On Power Apparatus and Systems, vol. PAS-104, pp. 3278-3297, November 1985.
- [60] Greene D.P., & Smith S.F. (1993), “**Competition-based induction of decision models for examples**”. Machine Learning, 13(2/3), 229-257.
- [61] Greene D.P., & Smith S.F. (1994), “**Using coverage as a model building constraint in learning classifier systems**”. Evolutionary Computation, 2(1), 67-91.
- [46] Grosso, P.B. (1985), “**Computer simulation of genetic adaptation: Parallel subcomponents interaction in a multilocus model**”, (Doctoral dissertation, University of Michigan). (University Microfilms N° 8520908).
- [30] V. Hanagandi, M. Nikolau, “**A Hybrid approach to Global Optimization using a Clustering Algorithm in a Genetic Search Framework**”, Computers and Chemical Engineering, Abril 1995.
- [29] H.H. Happ, “**Optimal Power Dispatch**”, IEEE Trans. On Power Apparatus and Systems, vol. 93.N° 3, pp.820-830, May/June 1974.
- [24] M.H. Haque, “**Capacitor placement in radial Distribution Systems for Loss reduction**”, IEE Proc.-Gener. Transm. Distrib., Vol 146, N° 5, September 1999.
- [75] Harik, G. (1994). “**Finding multiple solutions in problems of bounded difficulty (IlliGAL Report N° 94002)**”. Urbana: University of Illinois, Illinois Genetic Algorithms Laboratory.

- [72] Hatjimihail, A.T. (1993). “**Genetic algorithms-based design and optimization of statistical quality-control procedures**”. *Clinical Chemistry*, 39(9), 1972-1978.
- [50] Hillis, W.D. (1990), “**Co-evolving parasites improve simulated evolution as an optimization procedure**”. *Physica D*, 42,228-234.
- [39] Holland, J.H. (1992), “**Adaptation in Natural and Artificial Systems**”. Cambridge, MA: MIT Press.
- [67] Holland, J.H., & Reitman, J.S. (1978). “**Cognitive systems based on adaptive algorithms**”. In D. A. Waterman & F. Hayes-Roth (Eds.), *Pattern-directed inference systems* (pp. 313-329). New York: Academic Press.
- [78] Holland, J.H. (1975). “**Adaptation in natural and artificial systems**”. Ann Arbor: University of Michigan Press.
- [23] Y. Huang, H. Yang, C. Huang, “**Solving the Capacitor problem in a radial Distribution System using Tabu Search Approach**”, *IEEE Trans. On Power Systems*, vol. 11, N° 4, pp. 1868-1873, November 1996.
- [7] M. Kaplan, “**Optimization of number, location, size, control type, and control setting for shunt capacitors on radial distribution feeders**”, *IEEE Trans. On Power Apparatus and Systems*, vol. PAS-103, pp. 2659-2663, September 1984.
- [53] Langton C.G. (Ed.) (1989), “**Artificial life**”. Redwood City, CA: Addison-Wesley.
- [54] Langton C.G., Taylor, C. Farmer, J.D., & Rasmussen, S. (Eds.) (1992), “**Artificial Life II**”. Redwood City, CA: Addison-Wesley.
- [20] F.G. Lobo, “**The Parameter less Genetic Algorithm: Rational and Automated Parameter selection for simplified Genetic Algorithm Operation**”, Illigal report n° 200030, University of Illinois at Urbana Champaign, July 2000.
- [31] Samir W. Mahfoud, “**Simple Analytical Models of Genetic Algorithms for Multimodal Function Optimization**”, Illigal Report n° 93001, University of Illinois at Urbana-Champaign, February 1993.

- [70] Mahfoud, S. W. (1992). “**Crowding and preselection revisited**”. In R. Männer & B. Manderick (Eds.), *Parallel problem solving from nature*, 2 (pp. 27-36). Amsterdam: Elsevier.
- [71] Mahfoud, S. W. (1994b). “**Crossover interactions among niches**”. *Proceeding of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, 188-193.
- [35] B.L. Miller, D.E. Goldberg, “**Genetic Algorithms, Tournament Selection, and the Effects of Noise**”, Illigal Report n° 985006, University of Illinois at Urbana-Champaign, July 1995.
- [77] Goldberg, D. E., Deb, K., & Korb, B. (1990). “**Messy genetic algorithms revisited: Studies in mixed size and scale**”. *Complex Systems*, 4, 415-444.
- [28] V. Miranda, J.V. Ranito, L.M. Proenca, “**Genetic Algorithms in optimal multistage distribution Network Planning**”, INESC and FEUP/DEEC-Facultade de Engenharia da Univ. do Porto.
- [22] K. Miu, H. Chiang, G. Darling, “**Capacitor placement, replacement and control in Large-Scale Distributions Systems by a GA-Based Two-Stage Algorithm**”, *IEEE Trans. On Power Systems*, vol. 12, N° 3, pp. 1160-1166, August 1997.
- [73] Pál, K.F. (1994). “**Selection schemes with spatial isolation for genetic optimization**”. *Lecture Notes in Computer Science: Parallel Problem Solving from Nature-PPSN III*, 866, 170-179.
- [1] M. Ponnavaiko, K.S. Prakasa Rao, “**Optimal choice of Fixed and Switched Capacitors on Radial Distributors by the Method of Local Variations**”, *IEEE Trans. On Power Apparatus and Systems*, vol. 102, pp. 1607-1615, June 1983.
- [36] Adam Prügel-Bennett, “**A Course on Micro-Evolution and Genetic Algorithms**”, Speech and Itelligent Systems research group, Department of Electronics and Computer Science, University of Southampton, May 17, 1998.
- [32] Rajendra B. Patil, “**Intervals in Evolutionary Algorithms for Global Optimization**”, Los Alamos National Laboratory, Los Alamos, email: rbp1@lanl.gov.

- [27] I.J. Ramírez-Rosado, J.L. Bernal-Agutián, “**Genetic Algorithm applied to the design of large power Distribution Systems**”, IEEE Trans. On Power Systems, vol. 13, N° 2, pp. 696-703, May 1998.
- [59] Richardson J.T., Palmer M.R., Liepins, G., & Hilliard M. (1989), “**Some guidelines for genetic algorithms with penalty functions**”. Proceedings of the third international conference on genetic algorithms, 191-197.
- [33] F. Rothlauf, D. Goldberg, A. Heinzl, “**Network Random Keys – A Tree representation Scheme for Genetic and Evolutionary Algorithms**”, Illigal Report n° 2000031, University of Illinois at Urbana-Champaign, July 2000.
- [42] Rudnick, M. & Goldberg, D.E., (1991), “**Signal, noise, and genetic algorithms**” (IlligAL Report N° 91005). Urbana: University of Illinois, Illinis Genetic Algorithms Laboratory.
- [52] Sannier, A.V., II & Goodman, E.D. (1987), “**Genetic learning procedures in distributed environments**”. Genetic Algorithms and there Applications: Proceedings of the Second International Conference on Genetic Algorithms, 162-169.
- [57] Schaffer, J.D. (1984), “**Some experiments in machine learning using vector evaluated genetic algorithms**”. Unpublished doctoral dissertation, Vanderbilt University, Nashville.
- [58] Schaffer, J.D. (1985), “**Multiple objective optimization with vector evaluated genetic algorithms**”. Proceedings of an international conference on genetic algorithms and their applications, 93-100.
- [69] Sedbrook, T.A., Wright, H., & Wright, R. (1991). “**Appilcation of a genetic classifier for patient triage**”. Proceedings of the Fourth International Conference on Genetic Algorithms, 334-338.
- [62] Shorrocks B. (1979), “**The genesis of diversity**”. Baltimore: University Park Press.
- [45] Smith, R.E., (1988), “**An Investigation of diploid genetic algorithm of adaptive search of nonstationary functions**”, (Marsters Thesis and TCGA Report

N° 88001). Tuscaloosa: University of Alabama, The Clearinghouse for genetic algorithms.

[19] Srinivasan Sundhararajan, A. Pahwa, “**Optimal Selection of Capacitors for Radial Distribution Systems using a Genetic Algorithm**”, IEEE Trans. On Power Systems, vol. 9, N° 3, pp. 1499-1507, August 1994.

[68] Stadnyk, I. (1987). “**Schema recombination in a pattern recognition problem**”. Genetic algorithms and their applications: Proceedings of the Second International Conference on Genetic Algorithms, 27-35.

[63] Syswerda, G. (1989), “**Uniform crossover in genetic algorithms**”. Proceedings of the third international conference on genetic algorithms, 2-9.

[64] Syswerda, G. (1991), “**A study of reproduction in generational and steady-state genetic algorithms**”. In G.J.E. Rawlings (Ed.), Foundations of genetic algorithms (pp. 94-101). San Mateo: Morgan Kaufmann.

[55] Varela, F.J., & Bourguine, P. (Eds.) (1992), “**Toward a practice of autonomous systems**”: Proceedings of the First European Conference On Artificial Life. Cambridge, MA: MIT Press.

[79] Yin, X., & Gernay, N. (1993). “**A fast genetic algorithm with sharing scheme using cluster analysis methods in multimodal function optimization**”. In R. F. Albrecht, C. R. Reeves, & N.C. Steele (Eds.), Artificial neural nets and genetic algorithms: Proceedings of the international conference in Innsbruck (pp. 450-457). Berlin: Springer-Verlag.

ANEXOS

ANEXO A: TEORÍA DE LOS ALGORITMOS GENÉTICOS

Clásicamente se tiene la impresión de que los Algoritmos Genéticos simplemente mejoran poblaciones de strings. A continuación se estudiará el funcionamiento interno de un AG simple, incluyendo el mecanismo a través del cual el AG alcanza la solución final. Además se hará un repaso de la teoría de un AG básico, incluyendo el concepto de esquema (schemata), bloques componentes (building blocks), paralelismo implícito (implicit parallelism), el teorema del esquema (schema theorem) y la hipótesis de los bloques componentes (building blocks hipótesis).

Esquema (Schemata)

Mientras que los AGs superficialmente procesan strings, implícitamente están procesando shematas, que representan similitudes entre strings (1989c, [38]; 1992, [39]). Un AG no puede, por un problema práctico, visitar todo punto en el espacio de búsqueda. Sin embargo puede muestrear un número suficiente de hiperplanos en regiones “altamente adaptadas” del espacio de búsqueda. Cada uno de estos hiperplanos corresponde a un conjunto altamente adaptado de substrings.

Un schema es un string de largo l (que corresponde al largo que poseen todos los strings de la población), tomado del conjunto $\{0,1,*\}$, donde “*” es un carácter ‘don’t care’ ó ‘wild card’ que por definición puede ser cualquier cosa. Cada schema representa el set de todos los strings binarios de longitud l , cuyas correspondientes posiciones de bits contienen bits idénticos a aquellos “0” y “1” del schema. Por ejemplo, el schema, $10**1$, representa el set de strings de compuestos por cinco bits, $\{10001, 10011, 10101, 10111\}$. El schemata también es denominado “*subconjunto de similitudes*”, debido a que representa un subconjunto de strings que poseen similitudes en ciertas posiciones. El schemata presenta dos propiedades, una de *orden*

y otra denominada *largo definido*. *Orden* es el número de posiciones de bits fijos (se excluyen evidentemente los don't care) en el schema. El *largo definido* es la longitud entre los bits fijos extremos del schema. Por ejemplo, el schema anterior es de orden 3, que se escribe como $o(10^{**}1)=3$, y posee un largo definido de 4, o equivalentemente $\delta(10^{**}1) = 4$. Cada string en la población es un elemento de 2^l schemata.

Bloques Componentes (Building Blocks)

Concisamente, los bloques componentes corresponden a schematas bien adaptados de orden menor que poseen una longitud definida baja (1989c, [38]), donde el fitness del schema corresponde al fitness promedio del elemento que lo contiene. Bloques componentes representan similitudes (entre strings) que son significativas en la solución que el AG entrega de un problema particular.

La selección elige strings con alto fitness para un posterior procesamiento. Los strings que poseen un schema con un alto grado de adaptación, son seleccionados un mayor número de veces. El crossover por su parte, pocas veces descompone schematas que poseen un largo definido bajo y la mutación generalmente no altera schematas de orden bajo. Debido a ello, schematas de orden bajo, altamente adaptados y de longitud definida pequeña, conocidos también como bloques componentes, proliferan de generación en generación. De lo anterior se desprende que los AGs procesan bloques componentes o schematas útiles. Holland (1992, [39]) estimó que mientras un AG procesa n strings cada generación, procesa además un orden de n^3 schematas útiles. El denominó a este fenómeno *paralelismo implícito*. Para la solución de problemas reales, la presencia de paralelismo implícito significa que una población más grande tiene la posibilidad de localizar una solución en un tiempo polinomialmente más rápido que una población más pequeña.

Teorema del Schema (Schema theorem)

Un AG simple antes de una convergencia significativa destina un número de intentos exponencialmente creciente enfocados a schematas útiles o bloques componentes. Esto se ilustrará a través de la derivación del siguiente teorema. Si $m(H,t)$ corresponde al número de instancias del schema H presente en la población en la generación t . Se puede calcular el número de instancias de H en la siguiente generación ó $m(H,t+1)$, en términos de $m(H,t)$. Recordando que un AG simple, asigna a un string una probabilidad de selección directamente proporcional a su fitness. Se desprende de la ecuación 4.1 que H debe ser seleccionada $m(H,t) \cdot (f(H)/f_{prom})$ veces, donde f_{prom} es el fitness promedio de la población y $f(H)$ es el fitness promedio de aquellos strings de la población que son elementos de H .

La probabilidad que un crossover de punto simple destruya el esquema es precisamente la probabilidad de que el punto de corte caiga dentro de las posiciones definidas por el schema. La probabilidad de que H sobreviva al crossover es mayor o igual al término, $1 - p_c \cdot \delta(H)/(l-1)$. Esta probabilidad de supervivencia es una desigualdad debido a que un schema que ha sido alterado puede recuperar su composición si se cruza con otro schema similar. La probabilidad de que H sobreviva la mutación es $(1 - p_m)^{o(H)}$, que puede aproximarse por $1 - o(H)p_m$ para un p_m y un $o(H)$ pequeños. El producto del número esperado de selecciones y la probabilidad de sobrevivir entrega una expresión conocida como el teorema del schema:

$$m(H, t+1) \geq m(H, t) \cdot \frac{f(H)}{f} \cdot \left(1 - p_c \frac{\delta(H)}{l-1} - o(H)p_m\right) \quad (\text{a.1})$$

El teorema del schema establece que los bloques componentes crecen exponencialmente en le tiempo, mientras que los schematas que se encuentran bajo el promedio decaen en una tasa similar.

Goldberg (1983, [40], 1989c, [38]), en su presentación detallada sobre el teorema del schema, desarrolló la hipótesis de los bloques componentes, que establece: “Bloques componentes se combinan para generar mejores strings”. Esto es, la recombinación y el crecimiento exponencial de los bloques componentes lleva a la formación de mejores bloques componentes, que a su vez resultan en una solución final altamente adaptada.

Mientras que el teorema del schema es adecuadamente predictivo para el crecimiento de éste, sobresimplifica en algunos casos la descripción del comportamiento de los AGs. Antes que todo, $f(H)$ y \bar{f} no se mantienen constantes de generación en generación. Los fitnesses de los elementos de la población varían significativamente después de las primeras generaciones. Segundo, el teorema de schema considera las pérdidas del schema pero no las ganancias de éste. Schematas tradicionalmente son construidos a partir del crossover y la mutación. Además a medida que el AG progresa, los elementos de la población son cada vez más parecidos, de esta forma los schematas que son alterados por el crossover tienden a reconstituirse inmediatamente. Finalmente, el teorema del schema corresponde a una teoría de expectativas y no considera la varianza; en muchos problemas de interés la varianza del (fitness del) schema puede resultar relativamente alta, convirtiendo la detección del schemata que contiene el óptimo global en un proceso altamente ruidoso (1991, [41]; 1991, [42]). Una varianza significativa del fitness del schema puede provocar una “convergencia prematura”.

A pesar de la simplicidad, el teorema del schema describe un conjunto de aspectos importantes del comportamiento de los AGs. Una alta probabilidad de mutación incrementa las alteraciones de los schematas de alto orden, mientras que probabilidades altas de crossover pueden alterar schematas que poseen un largo definido alto.

Cuando se considera el proceso de selección, una población converge a una tasa proporcional al fitness del mejor individuo respecto del fitness promedio de la población; esta tasa es una medida de la *presión de selección* (1994, [43]). Ya sea incrementando p_c ó p_m , o decrementando la presión de selección, se aumenta el muestreo o la exploración del espacio de búsqueda, pero se inhibe la explotación de un mejor schemata. Al disminuir p_c ó p_m , o incrementar la presión de selección se consigue explotar un mejor schemata, en desmedro de una exploración más profunda en busca de mejoras schematas. Como conclusión se puede decir que un AG debe mantener un usualmente delicado balance entre la *exploración* y la *explotación*.

Diversidad en los algoritmos Genéticos

En la literatura anterior sobre los AGs, los métodos de nichos han sido utilizados como una técnica para promover la diversificación de la población. El mecanismo simple de selección de los AGs replica las soluciones que poseen un fitness alto y descarta las soluciones con un fitness bajo, orientando la convergencia de la población. Sin embargo, dadas soluciones alternativas que poseen un idéntico fitness, la población aún seguirá convergiendo. Un AG simple pierde soluciones y subsoluciones debido a tres efectos: *Presión de selección* (selection pressure), *ruido de selección* (selection noise), y *alteración producida por operadores* (operator disruption).

La *presión de selección* es el resultado del valor esperado del proceso de selección: es esperable que las soluciones con un fitness bajo desaparezcan de una población finita.

El *ruido de selección* resulta de la varianza del proceso de selección: en una población finita elecciones aleatorias de subsoluciones competitivas de idéntico

fitness, generan ruido al valor de cada individuo, forzando eventualmente a la población a entregar buenas soluciones.

La *alteración de los operadores* resulta de la aplicación del crossover y la mutación, que son capaces de destruir, en forma directa, las buenas soluciones.

Las técnicas para diversificar una población típicamente reducen la presión de selección, el ruido de selección, o la alteración de operadores (o alguna combinación de ellas). Existen algunos métodos que por si sólo no clasifican como técnicas de nichos pero que logran aumentar la diversidad de la población. A continuación se presenta una clasificación de estos métodos según su categoría:

CATEGORÍA	RUIDO DE SELECCIÓN	PRESIÓN DE SELECCIÓN	ALTERACIÓN DE OPERADORES
Infusión directa			
Selección con reducción de ruido	X		
Corridas secuenciales múltiples	X		
Emparejamiento restringido			X
AGs termodinámicos		X	X
Ajuste de parámetros de control	X	X	X
Corridas paralelas múltiples	X	X	X

Tabla A.1: Clasificación de métodos de diversificación

La diferencia entre los mecanismos de nichos y diversificación suele ser muy sutil y depende de la utilidad de la diversificación que promueva el mecanismo. Por ejemplo, muchos esquemas de diversificación retardan la convergencia del AG, pero son incapaces de mantener establemente múltiples soluciones, aún cuando estas soluciones correspondan al óptimo global. Por otro lado, otros esquemas son algo más exitosos para mantener múltiples óptimos globales, pero fallan cuando tratan de mantener múltiples óptimos locales.

Utilidad de la Diversidad

Los AGs que son inicializados aleatoria y uniformemente y detenidos inmediatamente presentan, en la mayoría de los casos, una diversidad en cada *locus* cerca de la máxima. Anteriormente fueron expuestos métodos de diversificación que son capaces de reducir los tres criterios más relevantes, (ruido de selección, presión de selección y alteración de operadores) a niveles arbitrariamente bajos. Al realizar esto, sin embargo, resulta un AG cuyo desempeño es poco beneficioso desde el punto de vista de la búsqueda. Goldberg al estudiar esos AGs que se basan en la mutación como un mecanismo de diversificación, instauró el concepto de *diversidad útil*. Goldberg & Richardson (1987, [44]) establecieron que “mantener la diversidad por si sola no es el punto, sino se requiere de la mantención de una diversidad apropiada, diversidad que de alguna manera ayude a obtener mejores strings”. En un contexto más general, la diversidad es útil, en la medida que ayude a alcanzar alguna meta.

En la literatura de los AGs, la diversidad es conocida por desempeñar dos roles potenciales: permitir la exploración de gran parte del espacio de búsqueda con el fin de generar una mejor y única solución final; y por permitir explorar múltiples soluciones. Estos roles se pueden establecer en términos de la formación y mantenimiento de una diversidad de subsoluciones que apuntan hacia la obtención de una única subsolución, versus la posibilidad de formar y mantener un conjunto de soluciones diversas. Los dos roles no son enteramente independientes. En la búsqueda de una única solución en el espacio de búsqueda, el AG encuentra múltiples soluciones. De la misma forma un algoritmo para mantener múltiples soluciones es indudablemente aplicable, de alguna manera, en la mantención de la diversidad con el fin de alcanzar una única solución.

Muchos de los estudios sobre diversidad se han focalizado en la diversidad individual de los alelos. Sin embargo esta perspectiva microscópica puede estar errada. De hecho estos métodos generalmente no mantienen una variedad de

subsoluciones. Se hace énfasis en que un mecanismo que se base en mantener la diversidad entre alelos promueve una diversidad útil, si es capaz de alcanzar mejores soluciones finales para un n fijo, en problemas moderados a difíciles, al ser comparado con un AG simple.

El método desarrollado en este estudio es capaz de formar y mantener múltiples soluciones finales, y además es efectivo al formar y mantener múltiples subsoluciones que sirven para determinar una única solución final. Aún en casos donde la solución final y las subsoluciones intermedias son de diferente fitness, se espera que este método sea capaz de encontrarlas y mantenerlas. Esas son las características de un buen método de nichos. Por ende el test más importante para un método de nichos consiste en verificar si posee la capacidad para encontrar múltiples soluciones finales en un monto de tiempo razonable y mantenerlas por un amplio período de tiempo. Un método de nichos debe ser efectivo en problemas donde las soluciones posean diferentes fitness, como también en aquellos en que las soluciones tengan un fitness idéntico. De hecho se asume que un método de nichos es capaz, tal vez con alguna leve modificación, de formar y mantener múltiples y útiles subsoluciones, cuando el objetivo es encontrar una única solución final.

La atención se centra entonces en la formación y mantención de múltiples soluciones en una población. Existe un amplio conjunto de soluciones en las que un usuario de AGs puede estar interesado. Algunos usuarios, por ejemplo, pueden estar interesados en las mejores k soluciones de una función multimodal. Otros, pueden estar interesados en encontrar la k -ésima solución óptima, siempre y cuando ésta sea lo suficientemente diferente.

ANEXO B: CLASIFICACIÓN DE LOS ALGORITMOS GENÉTICOS

Generalmente los métodos de nichos pueden ser clasificados en dos dimensiones según su comportamiento. La primera dimensión es espacio v/s tiempo. Hasta ahora se ha discutido los métodos de nichos como algoritmos espaciales; forman y mantienen subpoblaciones en el tiempo. En este estudio los métodos de nichos espaciales son frecuentemente denominados, *métodos de nichos paralelos*, debido a que conceptualmente desarrollan nichos en paralelo dentro de una única población. Los métodos de nichos temporales son usualmente denominados nichos secuenciales, debido a que conceptualmente desarrollan nichos en forma secuencial a través del tiempo. Cabe señalar que la designación de paralelo versus secuencial es independiente del número de procesos físicos empleados.

La segunda dimensión corresponde a los nichos que se desarrollan en un único ambiente versus los que lo hacen en múltiples ambientes. De hecho es posible clasificar el ambiente general como único o múltiple y espacial o temporal. Muchos métodos de nichos son capaces de formar y mantener nichos estables en un ambiente único. (El ambiente en funciones multimodales de optimización está compuesto por una única función de adaptación). Esos son los métodos de nichos más generales debido a que pueden ser aplicados en situaciones compuestas de múltiples ambientes. Existen otras técnicas de nichos que requieren múltiples ambientes ya sean espaciales (múltiples funciones de adaptación) o temporales (funciones de adaptación no estacionarias). En algunos casos el ambiente general varía a través de ambos: tiempo y espacio.

La segunda dimensión de comportamiento es similar a la dicotomía de la evolución *simpátrica* versus *alopátrica* de las especies de una población ecológica, esto si se permite que el tiempo en este caso como la geografía actúe como una barrera. La evolución *simpátrica* se refiere a la diferenciación de las especies que coexisten geográficamente, pero que al evolucionar explotan diferentes recursos o

nichos ecológicos dentro del mismo medio ambiente. La evolución *alopátrica* se refiere a la evolución de las especies en un ambiente geográfico aislado. De hecho diversos lugares geográficos en el mundo poseen diferentes ambientes. En cada ambiente puede surgir una especie que se adapte adecuadamente a éste.

En el siguiente cuadro se clasifican los diversos tipos de métodos de nichos según su comportamiento en las dimensiones antes descritas. El método de nichos de cada cuadrante de la tabla es similar al tipo de problema que es aplicable.

	AMBIENTE SIMPLE	MÚLTIPLES AMBIENTES
TEMPORAL	Localización Secuencial	Sobre Especificaciones AGs Ecológicos
ESPACIAL	Ventaja Heterocigota Crowding Competición Restrictiva Fitness Sharing	AGs Ecológicos Sistemas Inmunes

Tabla B.1: Clasificación de los Métodos de Nichos

Por ejemplo, técnicas temporales en un sólo ambiente, son más adecuadas para funciones de optimización multimodales. Técnicas espaciales en un sólo ambiente son apropiadas para funciones de optimización multimodales, funciones de optimización multiobjetivo, y para clasificación. Técnicas espaciales en múltiples ambientes son ventajosas para resolver funciones de optimización multiobjetivo. Finalmente, técnicas temporales en múltiples ambientes son apropiadas para resolver problemas de simulación adaptativa.

B.1 Sobreespecificación

Un alto porcentaje de los genomas de organismos de orden superior no poseen una función conocida. Este hecho ha promovido a investigadores sobre AGs a investigar el potencial del exceso de material genético en la diversificación y la formación de nichos. De esta forma se han implementado métodos de nichos que contienen exceso de material genético en cantidades suficientes como para rastrear un solo óptimo dado un ambiente determinado, a través del tiempo. La sobreespecificación combinada con un ambiente dinámico genera un método de nichos temporal y multidinámico. Muchos investigadores desean adaptar las técnicas de sobreespecificación en la búsqueda de múltiples óptimos en un ambiente simple. Sin embargo en un ambiente estacionario y dado un tiempo suficiente, toda la diversidad desaparece del material genético expresado y del redundante.

Muchos AGs utilizan alguna forma de sobreespecificación o redundancia en sus representaciones. Una aproximación estimulada por la biología es el uso de múltiples hebras de cromosomas. En el caso de tratarse de dos hebras se conoce como diploidía o poliploidía si se trata de varias. Muchos estudios que se realizaron tempranamente en el campo de los AGs utilizaron estructuras de cromosomas diploides. La razón principal de esos estudios fue simular sistemas biológicos genéticos; convirtiéndose la resolución de problemas en un objetivo secundario. Estos estudios analizados por Smith (1988, [45]), no presentaron ninguna ventaja en la optimización de funciones estáticas debidas a la diploidía. Una excepción es el AG de Grosso (1985, [46]) que exitosamente logró mantener múltiples peaks de una función estática.

Goldberg y Smith (1987, [47]) propusieron que mientras los AGs con diploidía no presentan ninguna ventaja en ambientes estáticos, si la tienen frente a ambientes dinámicos. Un cromosoma diploide es capaz de conservar diversidad que

fue útil en el pasado y que en el futuro puede volver a hacerlo. Después de todo los AGs fueron concebidos por Holland como sistemas adaptativos. Goldberg y Smith aplicaron exitosamente un AG diploide para rastrear un óptimo global que se alternaba entre dos valores a intervalos regulares. La población logró mantener cromosomas con ambos miembros y consecuentemente la localización de ambos óptimos. Sin embargo sólo una localización se expresaba cada vez.

La expresión ocurre a través de un mapa de *dominación*. Cada locus (o emplazamiento) de un cromosoma diploide contiene dos alelos, pero sólo uno de ellos se expresa en cada ocasión con el fin de computar el fenotipo y consecuentemente el valor de la función. Si algún tipo de alelo es dominante (ya sea un “0” o un “1” en codificación binaria) en cada locus y ambos tipos se presentan, se expresa el dominante. Cuando el locus contiene dos alelos dominantes iguales (caso homocigoto), entonces ese alelo se expresa. El mapa de dominación para cada individuo señala que tipo de alelo es dominante en que localización. El mapa de dominación evoluciona junto con el resto del genotipo.

Cuando el medio ambiente esta constantemente cambiando una solución previamente útil es conservada en un bit (que no se expresa) de un cromosoma diploide de la población. De esta forma si el ambiente posterior alcanza un estado similar al anterior, la solución almacenada retorna. Cabe señalarse que la población como un todo almacena soluciones útiles y estas soluciones deben ser construidas utilizando piezas de varios individuos.

En una función que oscila entre dos óptimos no queda claro en los estudios que se han realizado si con cromosomas diploides es suficiente para rastrear más de un óptimo ó cuántos cromosomas diploides son necesarios para ello. Dado un ambiente estacionario simple, existe evidencia que apunta a la conclusión que una

población perteneciente a un AG diploide va a converger completamente a pesar que le tome más tiempo que a un AG haploide.

Además, es claro que la redundancia en la ciencia computacional o en los organismos vivos evita la pérdida de información. Hasta ahora se ha analizado los beneficios de la redundancia a través de la diploidía. Otra alternativa es mantener la redundancia a pesar de los conflictos de información que se puedan gestar producto de un segmento extra en un cromosoma haploide. Goldberg (1989c, [38]) notó que existe la posibilidad de manipular adecuadamente estos cromosomas al utilizar strings de largo variable en conjunto con operadores de duplicación y eliminación. Para resolver este conflicto se hace necesario la presencia de un procedimiento de *dominio Inter-cromosoma*.

Un AG que permite la sobre especificación dentro de un cromosoma haploide es denominado *Messy GA* (AG “confuso”). Un AG messy incorpora strings de largo variable y maneja la sobre especificación utilizando métodos de precedencia posicional. Esto es si un gen reside en más de un locus, el locus más a la izquierda toma la precedencia. Después que un cromosoma es alterado por los operadores genéticos, los genes anteriormente ocultos ganan precedencia. Goldberg, Korb y Deb (1989, [48]) notaron que otros esquemas de precedencia son posibles, como por ejemplo, la precedencia adaptativa. Hasta ahora no se han aplicado AGs messy a funciones no estacionarias.

Un *AG redundante estructurado* (1992, [49]) codifica el fenotipo de un individuo completo o parte de éste, un número predeterminado de veces. *Bits de activación* son responsables de activar o desactivar un bit codificado. Los bits de activación son ordenados jerárquicamente de tal forma que se activen o desactiven unos a otros. Bits redundantes nunca son activados simultáneamente. Dasgupta y McGregor restringieron su atención en jerarquías de hasta dos niveles, consistentes en la activación de bits y de bits codificados. Resolviendo el mismo problema de

Goldberg y Smith los autores mostraron que un AG estructurado es efectivo en el rastreo de dos óptimos alternos.

B.2 Algoritmos Genéticos Ecológicos

Algunos mecanismos inducen la formación de nichos utilizando y usualmente creando múltiples ambientes. Estos tipos de AG son conocidos como *AG Ecológicos*. A diferencia de los AGs que presentan sobre-especificación, como los señalados en la sección anterior, los AGs ecológicos interactúan extensamente en el ambiente en que se encuentran y en algunas ocasiones hasta lo modifican. Las categorías de AGs ecológicos incluyen métodos como la coevolución simbiótica o parásita, modelación de recursos, descomposición de la función de adaptación y simulación ecológica a escala.

Los múltiples ambientes requeridos por un AG ecológico se presentan en el tiempo o espacio o en ambas dimensiones. Un ejemplo de ambientes temporales múltiples es una función de adaptación no estacionaria. Ejemplos de ambientes espaciales múltiples incluyen funciones de adaptación multiobjetivo y funciones de adaptación múltiples que residen en diferentes regiones geográficas.

Un AG ecológico es similar a un AG diploide, pero en el primero la función de adaptación varía en el tiempo, éste es conocido como el modelo de Hilis (1990, [50]) de “coevolución de parásitos”. El AG de Hilis simultáneamente realiza la coevolución de una población de individuos seleccionados y de una población compuesta por un set de pruebas para los seleccionados. Los seleccionados y los sets definen funciones de adaptación para ellos que evolucionan en el tiempo. A los seleccionados se les asigna cierto grado de adaptabilidad según su desempeño en un set de casos de prueba. El set de 10 a 20 casos le asigna un valor de adaptación al

seleccionado basado en cómo éste los desempeñe. A modo de esquematizar el procedimiento se utiliza una malla en la que cada elemento contiene un seleccionado y un conjunto de casos de prueba. Ambos, tanto el seleccionado como los casos de prueba son representados a través de una estructura diploide y son operados en paralelo por medio de dos AGs separados.

La formación de nichos en el AG de Hillis es producido por diversos motivos, por un lado se encuentra el ambiente temporal creado por la evolución de la función de adaptabilidad y por otro lado se encuentra la habilidad de escalamiento de los cromosomas diploides. La geografía es un mecanismo de diversificación secundario. Simulaciones realizadas por Hillis demuestran la habilidad de adaptación de AGs diploides a cambios graduales en el medio ambiente.

En el AG de Hillis, los elementos de la población (elegidos) coevoluciona junto a la función de adaptabilidad (set de casos de prueba), pero no realizan ninguna contribución directa para mejorar el grado de adaptabilidad de otros individuos de la población. Un tipo de coevolución más general es aquel en el que cualquier elemento de la población puede contribuir directamente sobre el “fitness” de algún otro. En algunos modelos de coevolución, el grado de adaptabilidad de un individuo depende enteramente de la interrelación que tenga con el resto; en esos casos la función de “fitness” se distribuye espacialmente a través de la población, y temporalmente a medida que otros individuos evolucionan. Algunos modelos de coevolución hacen uso además de la geografía..

Un modelo de coevolución en el que los individuos obtienen su fitness del resto es el “Game World” (1991, [51]). El “World Game” es un sistema geográfico bidimensional en el que múltiples estrategias para un juego en particular son utilizadas por los individuos para competir entre ellos. El fitness de un individuo está determinado por el número, tipo y localización de los otros individuos. Este juego

inicialmente contiene estrategias generadas aleatoriamente con algunos sitios en blancos que corresponden a estrategias nulas. A la población se le está permitido aumentar o disminuir. Una generación consiste en que cada individuo juega una serie de juegos con sus vecinos y va acumulando puntaje o equivalentemente mejorando su fitness. Entonces la selección, mutación y opcionalmente la cruce ocurren en un vecindario local. Varios comportamientos dinámicos y estados cuasi-estables evolucionan. Se ha observado que en ese tipo de escenarios tanto la competencia como la cooperación se desarrollan, y frecuentemente dos o más especies (diferentes bits o estrategias) coexisten en un cuasi-equilibrio. Es poco probable que un sistema se mueva a otro con un equilibrio muy diferente.

Sannier y Goodman (1987, [52]) describen el sistema Asgard, en el que el fitness de un individuo está determinado por la habilidad que éste posea para utilizar recursos (alimentos) en un terreno bi-dimensional. Individuos que son incapaces de utilizar adecuadamente los recursos fallecen. El medio ambiente varía tanto en el espacio (terreno) como en el tiempo debido a la distribución de recursos alimenticios, pero los individuos que se encuentran relativamente cerca tienden a experimentar el mismo medio ambiente. El terreno corresponde a un toroide que está dividido en cuatro cuadrantes iguales. Cada cuadrante contiene dos “granjas” rodeadas por “desiertos”. El alimento sólo existe en las granjas. Cada granja experimenta dos estaciones: verano e invierno. Las estaciones están coordinadas de modo que cuando es verano en un cuadrante en el otro es invierno. La producción de alimentos depende del consumo y de la mantención del campo en el período de tiempo inmediatamente anterior, así como también de la estación en que se esté. El consumo y la mantención se encuentran determinados solamente por el número de individuos que ocupan un terreno en particular. La producción potencial de alimentos alcanza un máximo en una región durante la estación de verano. Lo inverso ocurre durante el invierno.

Asgard trabaja de la siguiente manera: los individuos son replicados o se cruzan con otros que se encuentren geográficamente cerca. Los operadores de mutación e inversión son aplicados a ambos sistemas de reproducción. Los retoños o hijos inicialmente son localizados cerca de sus padres. En cada instante de tiempo un individuo quema una unidad de su energía almacenada para mantenerse y consume energía adicional para reproducirse. Individuos que pierden toda su energía mueren. El tamaño de la población va variando con el tiempo.

Los individuos están compuestos por una lista de instrucciones. Sólo existen dos instrucciones: *Mover-X* y *Alimento-X-Y*, donde *X* corresponde a una de ocho posibles posiciones vecinas, e *Y* es la dirección de una instrucción. La instrucción *Mover*, mueve a un individuo a una de las ocho direcciones. La instrucción *Alimento* verifica si en la posición *X* existe alimento y salta a la posición *Y* en el caso que esto sea afirmativo. Un individuo consume el alimento que se encuentra en cierta posición en el momento de moverse hacia ella.

Sannier y Goodman inicializaron aleatoriamente las localizaciones en el terreno con 100 elementos generados también aleatoriamente. De esta manera se pudo observar que el comportamiento de Asgard en las primeras generaciones se asemejaba más a una caminata aleatoria (random walk) y fallecieron todos los individuos que no se localizaron en una granja. Después de 1500 generaciones se pudieron distinguir ocho clases de individuos. Algunas se encontraban totalmente contenidas en la misma granja, mientras que otras estaban esparcidas en diferentes granjas. Muchos individuos se desplazaron en una sola dirección hasta encontrar alimento. Después de 3000 generaciones evolucionaron dos clases de individuos, que fueron denominados como granjeros y nómades. Los granjeros se tendieron a mover según un patrón circular en cada granja. Los nómades se desplazaban a través de dos granjas de un mismo cuadrante, consumiendo el excedente de cada granja en la estación de verano.

Sannier y Goodman estudiaron el comportamiento de los granjeros extrayendo a todos los nómades del terreno. En aislamiento los granjeros exhibieron un alto grado de eficiencia en el consumo durante el invierno, mientras que en el verano la eficiencia disminuía a la mitad. Sin embargo con la presencia de los nómades, la eficiencia en su consumo era alta en ambas estaciones. Por otro lado, los nómades aislados no sobreviven debido a que no permanecen el tiempo suficiente en un área como para aumentar su eficiencia.

Sannier y Goodman comprobaron que antes de la generación 4000, los individuos resultantes de la cruce entre granjeros y nómades eran débiles. Sin embargo, una vez alcanzado este umbral, surgió un genotipo que eventualmente acaparó a toda la población. Este genotipo fue denominado *compuesto* y se caracterizaba por encontrar tempranamente alimento incurriendo seguidamente en un loop infinito. De esta forma un individuo recién nacido que encontraba prontamente alimento se transformaba en un granjero, en caso contrario se convertía en un nómade.

Los autores notaron que Asgard resultó altamente sensitivo a los valores de los parámetros.

Holland (1992, [39]) desarrolló un sistema adaptativo complejo que denominó Echo. Echo consiste en un terreno geográfico, dividido en localidades o ciudades. Cada emplazamiento produce diversos tipos de recursos que varían con el transcurso del tiempo (de intervalo de tiempo en intervalo de tiempo) y la localidad geográfica. Los individuos o *agentes* en cada intervalo de tiempo ocupan cierto lugar y consumen los recursos que allí se encuentren y pueden interactuar entre ellos a través de combates, transacciones o emparejándose. Cualquiera sea el tipo de la interacción, se genera como resultado el traspaso de recursos de un individuo a otro. Por ejemplo, el perdedor de un combate debe entregar todos sus recursos al ganador. Sin embargo, el

usuario, tiene la facultad para gobernar el sistema y definir las reglas de interacción entre individuos. Echo no utiliza una función de fitness explícita, pero la muerte de individuos se produce en la medida que éstos pierdan sus recursos. Los agentes que mejor utilicen los recursos naturales sobrevivirán.

Echo realiza iteraciones que se encuentran sincronizadas entre todos los lugares. Primeramente se seleccionan dos agentes en cada lugar para que interactúan ya sea a través de combate, transacción o emparejándose. Después de cada interacción los agentes consumen algo de los recursos del sector en particular. Los agentes deben pagar un “costo de mantención”, de manera tal que si no lo pueden hacer, morirán. Si un agente ha acumulado suficientes recursos del tipo adecuado, se replicará posiblemente a través de la mutación (los hijos se pueden generar a través de la mutación o el emparejamiento). Si un agente no encuentra recursos en el lugar actual, emigrará hacia algún sector vecino. Después de esto cada sector renueva sus recursos. Holland observó que comportamientos complejos, como comunidades cooperativas de agentes específicos, surgen del modelo.

Simulaciones ecológicas como las de Sannier y Goodman y Holland son representativas de numerosos estudios en el creciente campo de la vida artificial (1989, [53]; 1992, [54]; 1992, [55]). Holland afirma que actualmente el propósito principal de estos modelos, como Echo, es conducir experimentos racionales.

A continuación se centrará la atención en la búsqueda de métodos para dividir la función de fitness en múltiples ambientes espaciales. La visión preliminar de tal partición surge de la teoría de formación de nichos ecológicos. En un marco ecológico común, el medio ambiente está compuesto por múltiples recursos. Los diversos tipos de recursos definen ejes multidimensionales en el espacio de posibles ambientes. En este marco un nicho es cualquier subespacio del espacio formado por el ambiente completo. Los nichos pueden presentarse disjuntos o traslapados.

Algunos ecologistas sostienen la conjetura, que existe una relación uno a uno entre las variables fenotípicas y los recursos que existen en el ambiente. Por ejemplo, la variable fenotípica tolerancia a la temperatura, corresponde al recurso del ambiente temperatura. Las variables fenotípicas son aquellas que indican la habilidad de un individuo para utilizar recursos del medio ambiente. Un problema que posea una relación uno a uno es la condición subjetiva requerida para la rotulación de los ejes. Se debe tratar de llevar las variables fenotípicas y ambientales a una correspondencia 1 a 1.

El subespacio de los posibles ambientes correspondientes a una especie en particular se denomina *nicho fundamental*. La porción de un medio ambiente real que una especie ocupa en la actualidad se denomina *nicho realizado*. El nicho realizado de una especie se puede expandir o contraer, dependiendo de las fuerzas competitivas y de las fluctuaciones medio ambientales. El principio de la exclusión competitiva establece que si dos especies ocupan el mismo nicho, una de ellas eventualmente va a desaparecer. Para que coexistan ambas especies, éstas deben ocupar nichos distintos. Si dos especies ocupan nichos traslapados, alguna de las dos eventualmente se va a diferenciar o a extinguir.

En una primera instancia lo establecido anteriormente sobre medio ambientes, nichos y recursos no parece ser útil en un AG estándar. Ambientes, nichos y recursos se encuentran implícitamente codificados en la función de fitness de un AG. La dificultad principal, sin embargo, es extraer los recursos de la función de fitness. En los casos en que la función de fitness se puede descomponer en criterios independientes, cada uno de ellos constituye un recurso separado.

Una aproximación más directa la constituye el *Algoritmo de División Dinámica* (1994, [56]), que va creando un mayor número de subpoblaciones a medida que va progresando, a través de divisiones de la población en mitades. Cada

subpoblación es independiente, no existe migración u otro tipo de interacción entre las subpoblaciones existentes. Cuando las subpoblaciones son suficientemente diversas y grandes se dividen. Durante la división el individuo mejor adaptado (mejor fitness) sirve como semilla para la nueva subpoblación. El resto de los individuos son asignados a las subpoblaciones que los representen mejor. La división dinámica alcanza un grado aproximado de subdivisión de la función objetivo en tantas categorías como subpoblaciones existan. El algoritmo requiere una población total de aproximadamente 100 veces el número de peaks, para localizar cerca de los 100 peaks. Sin embargo, este múltiplo parece aumentar polinomialmente en la medida que el número de peaks crece.

Existen otros dos tipos de AGs que merecen mención. Uno desarrolla la optimización de funciones *multiobjetivos* (o multicriterios); el otro aprendizaje de máquina. En ambos AGs la función objetivo es dividida en componentes independientes, y elementos diferentes de la población optimizan cada componente.

En la optimización de funciones *multiobjetivos* el objetivo es localizar soluciones que simultáneamente optimicen un set de objetivos, sin combinar los objetivos en una suma ponderada u otra función. El fitness de cada individuo, en este caso es un vector que contiene un elemento por cada función objetivo. La finalidad entonces, es encontrar individuos que sean miembros del conjunto óptimo de Pareto. Un individuo *Pareto-Óptimo* es aquel, que no es dominado por ningún otro individuo. Un individuo *domina* a otro si por lo menos es igualmente adaptado en todas las dimensiones, y más adaptado en por lo menos una de ellas.

El primer sistema de este tipo fue llamado VEGA (diminutivo de “vector evaluated genetic algorithm”; Schaffer, 1984, [57], 1985 [58]). VEGA trabajaba dividiendo la población en tantas subpoblaciones como objetivos hallan y asignando un objetivo a cada subpoblación. Cada función objetivo asigna fitness a todos los

individuos de su subpoblación. Después de la asignación de fitness, VEGA mezcla la población completa y aplica los operadores genéticos de selección, cruza y mutación. Debido a que en cada generación VEGA asigna individuos en forma aleatoria a cada subpoblación, un individuo puede ser evaluado respecto a diversas funciones objetivos de una generación a la próxima. Este comportamiento genera una suerte de promedio. Un análisis demostró que el efecto de VEGA es el mismo que si el fitness fuera una combinación lineal de sus atributos (1989, [59]). A pesar de este efecto promedio, Schaffer reporto algunos éxitos al encontrar soluciones óptimas desde el punto de vista de Pareto.

El segundo sistema llamado COGIN (diminutivo de coverage-based genetic induction; D.P.Greene & Smith, 1993, [60], 1994, [61]) está enfocado al aprendizaje de máquina. COGIN asigna ejemplos de entrenamiento a reglas que los cubren. Las reglas más adecuadas (o adaptadas) reciben ejemplos que están cubiertos por más de una regla. Las reglas son ranqueadas según el número de ejemplos de prueba que cubran, considerando también la exactitud con que los cubran. COGIN emplea un proceso de selección denominado *cobertura basada en filtración*, que repetidamente selecciona la mejor regla de la población, eliminando los ejemplos de entrenamiento que cubre y generando un nuevo ranking para el resto de las reglas. Este proceso se realiza hasta que todos los ejemplos de entrenamiento se agoten. Todas las reglas no seleccionadas son eliminadas, lo que resulta en un tamaño de población variable. Debido a que el fitness de una regla se basa en el número de ejemplos de prueba que ésta cubre, COGIN efectivamente crea múltiples ambientes al particionar el set de ejemplos entre las reglas sobrevivientes. Cada regla es adaptada para un único medio ambiente dado por los ejemplos de entrenamiento que la regla cubre. A pesar que el conjunto de ejemplos de entrenamiento permanece estático, los límites de las particiones varían de generación en generación.

B.3 Ventaja Heterocigota

A continuación se analizarán los métodos espaciales que forman y mantienen nichos dentro de un único ambiente estático. Estos métodos son directamente aplicables a problemas de optimización de funciones multimodales, como también a otro tipo de problemas. Antes de explorar este método en profundidad se establecerá otro teorema ecológico denominado ventaja heterocigota.

Recordando que en cromosomas diploides cada locus (o localización) contiene un par de alelos, y que si ambos alelos no calzan, se dice que el individuo es heterocigoto en ese locus. En sistemas de dominancia completa, el alelo dominante en teoría va a guiar eventualmente al alelo recesivo a la extinción (1979, [62]). Sin embargo si los heterocigotos son más adaptados que los homocigotos (o sea, tienen una ventaja), la población mantendrá ambos alelos en proporciones estables. Algunos biólogos sostienen que la ventaja heterocigota es un factor mayor en la diversidad natural de las poblaciones.

Grosso (1985, [46]) mantuvo peaks separados en un espacio de búsqueda utilizando diploidía, la ventaja heterocigota y la migración. A pesar de que no está tan claro qué métodos llevan a la formación exitosa de nichos, un candidato principal es la combinación de diploidía con la ventaja heterocigota. Grosso implementó la ventaja heterocigota asignando en forma directa diferentes fitness a diferentes combinaciones de alelos en un locus.

A pesar que pareciera que la ventaja heterocigota no se pudiera aplicar a poblaciones haploides, en realidad no es así, ya que individuos haploides se convierten en diploides en un punto en cada generación, durante la cruce. Debido a ello se propone el siguiente mecanismo para implementar la ventaja heterocigota en poblaciones haploides. Se debe realizar la cruce antes de la selección utilizando

emparejamiento aleatorio. Se debe elevar el fitness de ambos hijos por una función de la distancia entre sus padres; esta función debe crecer a medida que aumenta la distancia. Una alternativa a directamente elevar los fitnesses, es utilizar la función de distancia como un criterio secundario de fitness. El método de ventaja heterocigota demanda una mayor exploración en ambos tipos de población tanto haploides como diploides.

B.4 Poblamiento: Reemplazo Restringido

En AGs generacionales como el AG simple, la población entera es reemplazada cada generación por los hijos formados a partir de la cruce y de la mutación (a excepción de individuos suficientemente afortunados para pasar sin alteraciones las etapas de selección, cruce y mutación). AGs de estado estable (1989, [63], 1991, [64]), por otro lado, sólo procesan algunos individuos en cada ocasión e insertan los nuevos hijos en la población. Algunas técnicas en esta etapa tratan de preservar la diversidad. En vez de realizar un reemplazo generacional completo o reemplazar a través de un proceso de selección (reemplazando el peor), tratan de reemplazar miembros de la población en la medida que aporten a la mantención de la diversidad. Los métodos que insertan nuevos elementos en la población al reemplazar elementos similares son denominados métodos de poblamiento (*Crowding methods*).

De Jong (1975, [65]) presentó un algoritmo que llamó “*modelo de factor de poblamiento*” (crowding factor model). El modelo de poblamiento de De Jong corresponde a un AG de estado estable, debido a que sólo una fracción de la población se reproduce y muere en cada generación. Cada nuevo miembro generado reemplaza uno ya existente, preferentemente el más similar. Para reemplazar el miembro más similar, se toma una pequeña muestra de la población existente y el nuevo miembro reemplaza el elemento de la muestra más cercano.

Específicamente el modelo de De Jong funciona de la siguiente manera. Una proporción de la población determinada por el *intervalo generacional (IG)* es elegida a través de una selección proporcional al fitness a la que seguidamente se le aplica la cruce y la mutación. Después de la mutación $IG*n$ individuos de la población son elegidos para morir (para ser reemplazados por los nuevos hijos). Cada hijo encuentra el elemento que debe reemplazar tomando una muestra aleatoria de FP individuos de la población, donde FP es conocido como el *factor de poblamiento*. Los hijos reemplazan a los individuos más similares de la muestra. La similitud es calculada utilizando la medida de paridad de Hamming, conocida como *comparación genotípica*.

Poblamiento (Crowding) está inspirado en un fenómeno ecológico: la competencia entre miembros similares de una población natural por recursos limitados. Esto es, individuos similares en una población natural, frecuentemente de la misma especie, tienden a ocupar los mismos nichos y por ello deben competir entre ellos por los recursos. Individuos diferentes, frecuentemente de diversas especies, tienden a ocupar nichos distintos, por ello típicamente no compiten por los recursos. Cuando un nicho ha alcanzado su capacidad total, los miembros débiles son expulsados de la población por los más fuertes. El resultado final es que nuevos miembros de una especie particular reemplazan a los individuos más viejos. El número total de miembros de cada especie no varía a menos que cambie el medio ambiente.

El método de “poblamiento determinístico” no es un modelo que permita que una población logre diversificarse y encontrar una mezcla estable de especies, sino simplemente mantiene la diversidad existente en la mezcla. El objetivo de De Jong al diseñar este método, fue mantener la diversidad de los alelos con el fin de evitar la convergencia prematura. De hecho el algoritmo fue exitoso en cierto grado al ser capaz de mantener la diversidad de los alelos, sin embargo errores estocásticos

introducidos por bajos valores del factor FP y por otros factores forzaron que el algoritmo gradualmente cayera en una fijación de las posiciones de cada bit. El método de “poblamiento determinístico” de De Jong fue empíricamente testado y se pudo comprobar que es limitado en la optimización de funciones multimodales, donde una preservación selectiva y una diversidad útil son esenciales.

Cinco años antes del trabajo de De Jong, Cavicchio (1970, [66]) introdujo muchos de lo conocidos *métodos de preselección*, y afirmaba que uno de ellos lograba mantener la diversidad de la población exitosamente. Al igual que De Jong, los métodos de preselección buscan prevenir la convergencia prematura. Cavicchio sostenía que realizar comparaciones con cada hijo resultaba ser una estrategia de reemplazo demasiado cara. En vez de esto proponía que como usualmente los hijos resultan ser similares a sus padres, se podía simplemente reemplazar uno de ellos.

El mejor esquema de preselección trabaja de la siguiente manera: si uno de los hijos tiene un fitness mayor que el del peor padre, entonces el hijo reemplazará ese padre. Desafortunadamente el efecto de este esquema no es evidente en el trabajo de Cavicchio, debido a que un conjunto de otros factores, incluyendo la modificación propia de los parámetros, también estaban presentes. Si se trata de optimizar funciones multimodales simples usando preselección, generalmente no se tendrá éxito en preservar representaciones de más de un óptimo.

Antes de 1992, la preselección pasó un largo tiempo sin ser tocada, excepto en efímeras menciones realizadas en algunos estudios, y el método de poblamiento de De Jong se ha utilizado sólo en un par de aplicaciones esporádicas, mayoritariamente en sistema de clasificación (1983, [40]; 1978, [67]). Leves modificaciones respecto al método de De Jong con la adición de presión de selección en la etapa de reemplazo, han sido implementadas en dos estudios. Stadnyk (1987, [68]) con su estrategia de reemplazo del más cercano al peor fue una. Esta estrategia selecciona FP candidatos

inversamente proporcionales a su fitness de una población y reemplaza al candidato más cercano con el nuevo elemento. La estrategia de reemplazo del más cercano al peor de Sedebrook, Wright, y Wright (1991, [69]) reemplaza al individuo más cercano del tercio inferior de la población. (La población fue ordenada según fitness en forma ascendente.) Desafortunadamente, ningún estudio fue capaz de modificar el método de “poblamiento determinístico” de forma tal que fuese capaz de optimizar funciones multimodales o formar nichos efectivamente.

En 1992, a través del análisis del método de poblamiento de De Jong y el modelo de preselección de Cavichio, Mahfoud fue capaz de determinar por qué ninguno de los métodos anteriores tuvo éxito y realizó modificaciones en el diseño que permitieron desarrollar un algoritmo de poblamiento exitoso. El algoritmo resultante fue denominado “poblamiento determinístico” (1992, [70], 1994b, [71]) y exhibió capacidades extensivas en la formación de nichos cuando fue aplicado a problemas de optimización multimodal y a problemas de clasificación. Desde entonces, este método ha sido utilizado en optimización y diseños estadísticos, métodos de control de calidad químicos (1993, [72]), como también en el problema de “spin-glass” (1994, [73]).

El método de “poblamiento determinístico” funciona de la siguiente manera. Primero, se agrupa a todos los elementos de la población en $n/2$ pares. Entonces se cruzan todos los pares y opcionalmente se mutan sus hijos. Cada hijo compite en un torneo contra uno de sus padres. Dado un par de padres y sus dos hijos, se pueden realizar las siguientes combinaciones de torneos padre-hijo: padre1 contra hijo1 y padre2 contra hijo2; ó padre1 contra hijo2 y padre2 contra hijo1. El grupo de torneos que fuerce la competencia más cercana es el que se mantiene, donde la cercanía es la distancia promedio entre las duplas padre-hijo en un grupo. La cercanía se calcula de acuerdo a alguna medida de distancia apropiada, preferentemente la distancia fenotípica.

Desde la introducción del método de “poblamiento determinístico”, dos métodos nuevos pero similares han sido propuestos y han mostrado un éxito preliminar en la optimización de funciones multimodales. Los dos métodos emplean la estrategia de reemplazo del peor más cercano. Al igual que en el método de “poblamiento determinístico” no se utiliza ningún sistema de selección, sólo de reemplazo. Un tercer método también ha sido propuesto, que a pesar que no corresponde a un método de poblamiento como fue planteado, puede ser potencialmente modificado para emular el comportamiento del método “poblamiento determinístico”.

El primer método (1992, [74]) le entrega a cada elemento en la población exactamente un intento para ser el primer padre en la etapa de cruce. El segundo padre es elegido en forma aleatoria de una muestra de MS elementos de la población. El elemento de la muestra que sea fenotípicamente más cercano al primer padre se convertirá en el segundo padre. Después de la cruce y la mutación cada uno de los dos hijos es insertado en la población de la siguiente manera. Primeramente, CP grupos que contienen CS individuos cada uno son aleatoriamente seleccionados de la población. Segundo, el elemento más similar de cada grupo respecto al hijo que se encuentra esperando avanza al proceso de selección. Cedeño y Vemuri indican que la estrategia de reemplazar al peor más cercano (como la de Stadnyk, 1987, [68]) se orienta demasiado a favorecer los peaks más elevados. Los autores no indican con qué frecuencia un padre es reemplazado bajo este criterio ni tampoco si la restricción de emparejamiento es esencial en este algoritmo.

Cedeño y Vemuri probaron su método de poblamiento en poblaciones compuestas por 200 o más individuos en diversas funciones de optimización cuyos espacios eran bi-dimensionales y contenían múltiples peaks. Los autores utilizaron una codificación de cromosomas compuesta por 30 bits para cada dimensión. Otros parámetros fueron, $p_c=0.95$, $p_m=0.06$, $MS=4$, $CF=5$, y $CS=10$. En una función que

contenía dos peaks de una altura uniforme, su método logró mantener ambos en una estricta proporción de igualdad. En funciones de 5 peaks de diversas alturas, su método, después de 10 generaciones, comenzó a transferir individuos del peak más bajo al más elevado. Después de 100-150 generaciones varios de los peaks menores se estabilizaron, pero otros perdieron todos sus representantes.

El segundo método de poblamiento (1994, [75]) agrega elitismo a su estrategia de reemplazo. El método de poblamiento comienza con la elección de dos elementos de la población en forma aleatoria, que posteriormente son cruzados y mutados. Después de la generación de dos hijos, se toma una muestra aleatoria de *FP* individuos de la población. Cada hijo compete con el elemento más cercano de la muestra. Finalmente, el ganador avanza a la siguiente generación.

Harik utilizó este método en una variada gama de funciones multimodales, que tienen entre 5 y 32 peaks. En cada simulación su algoritmo mantuvo representantes de todos los peaks por 100 generaciones. Mientras que la distribución de elementos de la población es uniforme al principio, en la generación 100, algunos peaks pierden elementos que son transferidos a otros peaks. Harik calcula que La expectativa del tiempo de mantención se comporta exponencialmente respecto a *FP*.

El tercer método es denominado GIGA (diminutivo de “gene invariant genetic algorithm”, Culberson, 1992, [76]) y emplea cualquier tipo de cruce, pero ningún sistema de mutación. La selección ocurre sólo durante el proceso de reemplazo, y el emparejamiento se restringe a elementos de fitness similar. GIGA selecciona dos padres adecuados para la etapa de cruce. Dos padres se cruzan *s* veces, generando *s* pares de hijos. El mejor par de hijos reemplaza a sus padres. Si el elitismo es opcional, el mejor par de hijos reemplaza a sus padres siempre y cuando éstos sean mejores. El fitness de un par de individuos está definido como el máximo fitness entre ambos individuos.

Se debe destacar además que GIGA conserva todos los alelos. Debido a que no existe mutación y a que tanto hijos como padres compiten en parejas, todos los alelos permanecen en las mismas proporciones con que se presentaron en la población inicial. El objetivo de GIGA es por lo tanto intercambiar bits entre soluciones de forma beneficiosa. Una característica interesante de GIGA, desde el punto de vista de la formación de nichos, es que crea dos subpoblaciones dentro de la población. La primera población es altamente adaptada y converge a una solución óptima. La segunda subpoblación contiene mayoritariamente complementos de subsoluciones de la primera subpoblación. Se podría decir que contiene toda la basura que ha sido expulsada de la primera subpoblación.

GIGA, tal como fue concebido, no desarrolla nichos. Sólo puede formar dos subpoblaciones y una de ellas es muy poco significativa. Sin embargo con algunas modificaciones, un híbrido de GIGA y el método “poblamiento determinístico” puede potencialmente formar y mantener múltiples subpoblaciones. Como también conservar todos los alelos.

B.5 Competencia Restringida

Una aproximación diferente para alcanzar subpoblaciones estables es restringir la competencia entre individuos muy diferentes durante el proceso de selección. Recordando parte de la teoría de la evolución ecológica, especies que coexisten tienden a ocupar diferentes nichos en el medio ambiente, de esta forma típicamente no compiten por recursos.

Goldberg, Deb, y Korb (1990, [77]) implementaron en un *AG desordenado o confuso* (“messy genetic algorithm”) una competencia restringida que denominaron “*Thresholding*”. Thresholding primero toma un competidor en forma aleatoria de la

población, sin reemplazo, para realizar un torneo de selección binario. Para elegir el segundo competidor, thresholding elige competidores en forma aleatoria en la población, sin reemplazo, hasta que encuentra un elemento que posea por lo menos un umbral preespecificado de genes en común con respecto al primero (por supuesto que los dos individuos no pueden ser el mismo individuo). Si no se encuentra un segundo competidor después de un número fijo de intentos, el primer competidor avanza a la siguiente etapa sin cruzarse. Cabe señalarse además, que los alelos de ambos competidores no necesariamente deben calzar; sino basta con que el “loci” correspondiente este presente. Los autores recomendaron esta técnica para la resolución de problemas cuyas sub-funciones variaran y no para localizar múltiples soluciones.

Una modificación análoga se le puede realizar a un AG estándar que cuente con el sistema de selección de torneo estocástico binario. La idea básica es evitar la competencia entre individuos desiguales, acordando para ello algún criterio de desigualdad. La competencia puede restringirse en un torneo de selección binario de la siguiente manera. El primer competidor se puede seleccionar en forma aleatoria de la población, con reemplazo. Los potenciales segundos competidores son seleccionados repetidamente en forma aleatoria de la población, también con reemplazo, hasta que uno esté fenotípicamente (o alternativamente, genotípicamente) dentro de una distancia específica θ del primer competidor. El competidor con mayor fitness avanza.

Se pueden realizar un conjunto de modificaciones opcionales. La primera es seleccionar al primer competidor sin reemplazo, entregándole a cada individuo de la población exactamente un intento al igual que el primer competidor. La segunda modificación es seleccionar candidatos para ser el segundo competidor sin reemplazo, pero limitando la cantidad que pueden ser rechazados. Si el límite es alcanzado entonces el primer competidor continúa a la siguiente etapa. La tercera modificación

es seleccionar una muestra de la población y realizar la competencia con el individuo más similar de la muestra.

B.6 Fitness Compartido (fitness sharing)

La utilización del fitness como recurso compartido único ha suscitado el mayor éxito en la actualidad en lo referente a la formación de nichos.

Holland (1975, [78], 1992, [39]) discute el concepto de limitar el número de individuos que ocupan un nicho hasta una capacidad determinada. Establece además que si un nicho tiene asociado un “pago” fijo por cada intervalo de tiempo y cada individuo que allí se encuentre debe compartir este pago equitativamente, entonces se genera una situación estable donde cada nicho contiene un número de individuos proporcional al “pago” que debe realizar. (Pago en este contexto es equivalente al fitness de un AG.) Si algún nicho llegase a estar sobrepoblado es debido a la ventaja que presentan ciertos individuos para emigrar hacia otros nichos. Los métodos para formar nichos que utilizan este concepto fueron denominados por Holland, *métodos de repartición de fitness*. Los métodos de repartición se pueden definir como algoritmos que requieren de elementos similares de la población para compartir el fitness. Para inducir la repartición estos algoritmos típicamente alteran el fitness de cada elemento de la población basados en la proximidad respecto a otros elementos.

El método de repartición fue explícitamente introducido por Goldberg y Richardson (1987, [44]) como el “método de las funciones de repartición”. Su método que en la actualidad es conocido simplemente como “*repartición*”, es directamente aplicable a funciones de optimización multimodales. La repartición reduce el fitness de un individuo en un monto que depende del número de individuos similares en la población. Específicamente, el nuevo fitness repartido, f' , es igual a su

antiguo fitness, f , dividido por la *cuenta del nicho*. La cuenta del nicho correspondiente a un individuo es la suma de los valores de la función de repartición (sh) entre el individuo y cada individuo de la población (incluyéndolo a él). El fitness compartido de un elemento i de la población es:

$$f'(i) = \frac{f(i)}{\sum_{j=1}^n sh(d(i, j))} \quad (\text{b.1})$$

La función de repartición sh es una función de la distancia entre dos elementos de la población; entrega un “1” si los elementos son idénticos, un “0” si cierto umbral de desigualdad es excedido y un valor intermedio para niveles de desigualdad medios. El umbral de desigualdad está especificado por una constante, σ_{share} ; si la distancia entre dos elementos de la población es mayor o igual a σ_{share} , no afectan el fitness compartido de otros elementos. Las funciones de repartición más usuales son de la forma:

$$sh(d) = \begin{cases} 1 - \left(\frac{d}{\sigma_{share}}\right)^\alpha, & \text{si } d < \sigma_{share} \\ 0, & \text{otro caso} \end{cases} \quad (\text{b.2})$$

En la ecuación anterior α es una constante, típicamente seteada en “1” usada para regular la forma de la función de repartición. Tanto la medida de distancia fenotípica como genotípica pueden ser empleadas; la elección apropiada depende del tipo de problema.

Goldberg y Richardson reportaron agrupaciones de elementos de la población estables cerca de los peaks de dos funciones sinusoidales. Una función cuenta con cinco peaks de igual altura mientras que la otra tiene cinco peaks de diferente altitud. El número de individuos aglomerados en torno a cada peak es proporcional a su altura. Sólo cuando el número esperado de individuos decae, cerca a “1”, el peak pierde toda su representación en la población.

La *repartición genotípica* utiliza la distancia de Hamming como medida de distancia, mientras que la *repartición fenotípica* emplea la distancia euclidiana en el espacio paramétrico decodificado. La repartición fenotípica entrega resultados levemente mejores debido al menor ruido generado en el espacio paramétrico decodificado. Además han sido probadas estrategias que restringen el emparejamiento en conjunto con modelos de repartición. Ellas han reducido exitosamente el número de cruzas letales entre individuos que pertenecen en diferentes peaks.

Modelos de repartición pueden ser implementados utilizando cualquier método de selección, pero la elección del método va a incrementar o decrementar la estabilidad del algoritmo. El muestreo universal estocástico es el sistema más popular en la actualidad. El sistema de torneo de selección es otra posibilidad, pero se deben tomar provisiones adicionales para mantener la estabilidad.

Yin y Gerny (1993, [79]) propusieron que un algoritmo de aglomeración fuera implementado antes de comenzar con la etapa de repartición de fitness, con el objetivo de dividir la población en diversos nichos. Cada individuo comparte sólo con los elementos que se encuentran en su nicho. Debido a que el fitness compartido de cada individuo no es calculado con respecto a toda la población, el algoritmo resulta ser más rápido que los sistemas de repartición tradicionales. Yin logró aplicar exitosamente métodos de repartición y aglomeración tanto en problemas de prueba como en uno real de “flujo de carga en un sistema eléctrico de potencia”.

Spears, propuso un “esquema simple de sub-población”, que incorpora una forma de repartición de fitness. Este método cuenta con sólo una restricción de emparejamiento, emplea un rotulado de k-bits para cada individuo e inhibe el emparejamiento entre individuos de distinto rotulado, además de la mutación de los bits marcados. El sistema de repartición divide el fitness de cada individuo por el

tamaño de la sub-población. Spears notó que esta metodología mantiene subpoblaciones estables en casi todos los peaks de una función de varios peaks, excepto en los dos menores. Algunas subpoblaciones presentan individuos en más de un peak, creando competencia dentro de la subpoblación y estimulando la emigración hacia los peaks más altos.

En el campo de aprendizaje de máquina, se probaron explícitamente la repartición de fitness en su “clasificador de estímulo-repuesta”. Demostraron teóricamente los beneficios de los métodos para formar nichos. Kargupta y Smith emplearon la repartición de fitness para desarrollar “redes polinomiales”. Packard utilizó un “mecanismo impulsor de diversidad” que disminuye el fitness de elementos de la población que se encuentran genotípicamente más cercanos al elemento de mayor fitness (mecanismo de repartición efectiva).

En la optimización de funciones multiobjetivos, la repartición de fitness ha demostrado ser muy efectiva. Fonseca y Fleming implementaron la repartición de fitness entre individuos que poseen valores similares de la función objetivo. A pesar que un individuo domine a otro, ambos no comparten la función de fitness. Los autores asignan un fitness a cada individuo basados en el ranking del individuo. El ranking es “1” más el número de elementos que domina un individuo. Todos los individuos no dominados reciben un ranking de “1”. Los autores notaron que si un método de nichos es bueno para mantener la diversidad en funciones de fitness planas, entonces también debe ser bueno para resolver los problemas con funciones de optimización multiobjetivo.

El *Algoritmo Genético de Pareto Anchado*, funciona a través del torneo de selección con un sistema de repartición de fitness continuamente actualizado. Después de tomar aleatoriamente dos competidores de la población, el AG selecciona una muestra adicional de tamaño k . Si un competidor es dominado por algún

elemento de la muestra y el otro no, el competidor no dominado gana el torneo. Si ninguno es dominado, el sistema de repartición selecciona un ganador. El ganador es el competidor que posea el menor conteo de nicho, donde el conteo de nicho se estima tomando una muestra de la población. Eligiendo como ganador al que tenga menor conteo de nicho se logran encontrar soluciones más adecuadas desde el punto de vista óptimo de Pareto. Los autores sostienen que el tamaño de la muestra k sirve para que el usuario pueda controlar la presión de selección. Valores pequeños de k resultan en una presión de selección mayor.

La mayor desventaja para utilizar los sistemas de repartición de fitness es el tiempo adicional requerido para recorrer toda la población con el fin de calcular el valor del fitness compartido. Muchos autores han sugerido calcular el fitness compartido de una muestra de tamaño fijo de la población. Métodos de repartición y muestreo han sido aplicados exitosamente en la resolución de problemas multimodales complejos y en la optimización de problemas multiobjetivos. Como lo han demostrado Yin y Gernay la agrupación es otra alternativa potencial para disminuir el tiempo requerido.

B.7 Modelos de Sistemas Inmunológicos

En algún lugar entre las técnicas que utilizan múltiples funciones de fitness, esquemas de poblamiento, y métodos de repartición de fitness se encuentran los patrones en que se basan los modelos de sistemas inmunológicos. Un sistema inmunológico es capaz de diferenciar un objeto extraño (antígeno) de aquellos que son nativos de un organismo. Para conseguirlo forman *anticuerpos* cuyo trabajo es reconocer los *antígenos*. Hasta el momento modelos de sistemas inmunológicos han sido sólo aplicados en simulaciones en las cuales la población de strings (de bits), denominados anticuerpos, evolucionan para calzar o cubrir un conjunto de patrones de strings, denominados antígenos. Los antígenos definen un medio ambiente espacial múltiple que asignan un fitness creciente a elementos de la población que se vayan pareciendo cada vez más a los antígenos. La tarea de patrones y coincidencias en los sistemas inmunológicos es similar a las de los sistemas de clasificación, excepto por el hecho que no existe un código explícito en los anticuerpos que les permita ser generalizados. La investigación de los modelos de sistemas inmunológicos se encuentra recién en sus primeras etapas. Hasta ahora nadie ha demostrado una técnica que les permita optimizar funciones multimodales.

Stadnyk (1987, [68]) introdujo uno de los primeros sistemas inmunológicos con patrones coincidentes. Ella probó dos funciones de fitness diferentes, la primera de ellas es el promedio de bits coincidentes en un conjunto de patrones. La segunda función consiste en un promedio menos generoso: cuenta cada bit coincidente sólo si los dos bits adyacentes también coinciden. Esto fue diseñado para prevenir asignaciones de alto fitness a reglas que pueden ser destruidas por un crossover de punto simple. El autor empleó una muestra aleatoria de k patrones para evaluar cada elemento de la población.

El AG de Stadnyk emplea un método de selección proporcional al fitness, cruza, mutación y un esquema de reemplazo. El autor probó con diversos esquemas de reemplazo que fallaron en inducir la creación de nichos, aún en problemas simples con dos patrones complementarios que figuraban en proporciones iguales en el conjunto de patrones. Esos esquemas incluyen el reemplazo del peor elemento de la población, el sistema de poblamiento de De Jong con un FP pequeño, y otro caso en el que el factor de poblamiento es n . Finalmente, se escogió un esquema de poblamiento que reemplaza el más cercano de los peores elementos de la población. Es decir, FP candidatos son elegidos de la población inversamente proporcional a su fitness, y el más cercano al nuevo elemento es reemplazado.

Stadnyk encontró que subpoblaciones estables evolucionaron en torno a diferentes patrones de un conjunto de patrones. Usando $k=12$, $FP=10$, y $n=50$ se logró mantener exitosamente subpoblaciones de strings que coinciden con un Conjunto de 10 patrones, en por lo menos 5 posiciones de bits por patrón. La formación de nichos observada no se debe sólo al aislamiento del esquema de poblamiento, sino también a los patrones de muestreo.

Forrest, Jarvonik, Smith, y Perelson presentaron un modelo de sistema inmunológico extensivo que le asignaba fitness a la población entera a través de la aplicación repetida del siguiente procedimiento. (El procedimiento se aplicaba $3n$ veces.) Primero el algoritmo selecciona aleatoriamente un antígeno. Después aleatoriamente selecciona una muestra de tamaño σ de anticuerpos sin reemplazo. El anticuerpo más coincidente gana la competencia y se le adiciona un puntaje de coincidencia a su fitness. Los autores mostraron que para valores muy bajos de σ , el algoritmo exhibe una forma de generalización. También determinaron experimentalmente que su algoritmo requiere de por lo menos 15 elementos en la población por cada Solución que se mantenga.

Se debe notar que Stadnyk comenzó a tener éxito en la formación de nichos al disminuir el número de antígenos utilizados para calcular el fitness de cada elemento de la población a 3.

ANEXO C: OPERADORES GENÉTICOS

Los operadores genéticos son reglas estocásticas de transición empleados por los Algoritmos Genéticos. Estos operadores son aplicados a cada string durante cada generación con el fin de obtener una nueva y mejorada población a partir de la ya existente.

Los operadores genéticos utilizados en este algoritmo son los siguientes:

Selección

Este operador participa en la creación de una nueva población (o generación) seleccionando los mejores individuos de una población anterior. Esto significa que habrá un mayor número de copias de los mejores individuos, pero sin embargo, también existirán algunas de los peores. En este algoritmo fueron implementadas 2 variantes de este operador: la primera corresponde al *Torneo Estocástico (stochastic tournament)* que básicamente selecciona un individuo para ser reproducido, comparando dos individuos que se eligen en forma aleatoria donde el mejor de ambos será el ganador; esto generalmente se realiza considerando una probabilidad fija (en caso de que la probabilidad sea variable este método se conoce con el nombre de *Blotzman Tournament*). Este esquema puede ser ampliado utilizando un mayor número de individuos en la competencia. La segunda, es conocida como Selección Estocástica Universal (SUS: Stochastic Universal Selection) y fue desarrollada por Baker en el año 1987, [90], por lo que también se la conoce simplemente como Baker Selection y consiste básicamente en utilizar una ruleta con tantas casillas como individuos haya (n) donde cada casilla tiene una longitud proporcional al fitness del individuo y la suma de todas las casillas vale 1. Además se cuenta con n indicadores distribuidos uniformemente alrededor de la ruleta, entonces con un sólo intento se seleccionan los n individuos necesarios.

Crossover (Cruza)

Es el operador genético principal y consiste en intercambiar parte de cromosomas entre dos individuos. El operador de este tipo más sencillo consiste en seleccionar un punto aleatorio para la cruza del cromosoma e intercambiar los genes que se encuentran entre el punto seleccionado y el término del cromosoma.

El crossover no se lleva a cabo con cada par de cromosomas de hecho su frecuencia es controlada a través de una probabilidad de crossover. Esta probabilidad generalmente es bastante elevada (típicamente es de 0,8). En este caso se utilizó un crossover multipunto en el que el usuario determina el número de cortes que se le van a realizar a cada individuo.

Mutación

Este operador consiste en cambiar el valor de un gen (de un cromosoma) en forma aleatoria. La mutación debe ser utilizada con cautela y con una baja probabilidad (típicamente del orden de 0,001) para poblaciones normales.

La importancia de este operador es que actúa como un *preservador de la vida*, ya que algunos genes (valores de bits) importantes son perdidos en el proceso de selección, y la mutación los rescata si es necesario. Sin embargo como se señaló anteriormente, abusar de este operador puede conducir a una *caminata aleatoria* (random walk) independiente de la probabilidad de ocurrencia de los otros operadores.

Elitismo

Es un operador genético que selecciona los individuos más adaptados o con mejor fitness de una población y los transfiere automáticamente a la siguiente, de esta forma se garantiza que los mejores individuos no se van a perder producto de la manipulación genética de los otros operadores. Este operador ha sido incluido en el algoritmo implementado donde el número de individuos seleccionados en forma

directa está determinado por un porcentaje ingresado por el usuario. La utilización de porcentajes elevados puede generar una convergencia prematura del algoritmo.

Epidemia

Este operador elimina todas las soluciones excepto las que poseen mejor desempeño o valor de fitness. En el algoritmo el número de soluciones que debe ser salvado puede ser especificado, así como también la ocasión entre generaciones en que este operador se va a aplicar. Las soluciones eliminadas son entonces reemplazadas por nuevas generadas con el fin de refrescar la información procesada por el algoritmo y para eliminar las soluciones correspondientes a óptimos locales.

ANEXO D: MANUAL DE USO DEL PROGRAMA

A continuación se describen los aspectos básicos para el uso del programa de Localización Óptima de Condensadores en un Sistema de Distribución Eléctrico. Programa desarrollado en Visual Digital Fortran, Standar Edition, Versión 5.0.

Cabe consignar, que las explicaciones e instrucciones acerca del manejo de esta herramienta están dirigidas a personas con cierta familiaridad con el ambiente WINDOWS.

D.1 Descripción General

El programa se ejecuta directamente en ambiente WINDOWS haciendo doble clic sobre el ícono del programa. Enseguida se presenta al usuario un menú con diversas posibilidades **BASE DE DATOS**, **MODELACION**, **OPTIMIZAR** y **SALIR**, cada una de las cuales cuenta con subtareas, las que a continuación se presentan y explican detalladamente.

BASE DE DATOS: Permite al usuario la creación de un nuevo sistema de distribución eléctrico, es decir, ingresar un conjunto de alimentadores, subestaciones y cargas para futuros estudios de compensación capacitiva. Además se debe ingresar la información sobre los condensadores y niveles de carga para los cuales se desea estudiar el sistema. Las alternativas que se derivan de este menú son las siguientes:

- **ABRIR ARCHIVO:** El cuadro de diálogo **ABRIR** permite al usuario abrir documentos que se encuentran en distintas ubicaciones, ya sea en el disco duro del equipo, un disco extraíble o en una unidad de red a la que se este conectado. El modelo permite la apertura de cualquier archivo que cuente con

una estructura de información adecuada para el programa. Por lo tanto, el usuario tiene la posibilidad de definir el nombre y extensión del archivo que más le acomode.

- **CONFIGURAR:** El cuadro de diálogo **Configurar Sistema** permite al usuario mediante el uso del mouse (ratón) y el teclado, crear un nuevo sistema de distribución eléctrico, para lo cual debe ingresar la información necesaria para los alimentadores, subestaciones y cargas. Además se debe ingresar la información relacionada con los condensadores y niveles de carga. Terminada la configuración, el nuevo sistema puede ser almacenado en forma de archivo utilizando el cuadro de diálogo **Guardar Configuración**.

MODELACION: Permite al usuario ajustar los principales parámetros del sistema de distribución y del AG.

- **MODELACION DEL SISTEMA:** Este cuadro de diálogo permite configurar los parámetros del sistema de distribución, tensión base, tensión máxima (en p.u.), tensión mínima (en p.u.), frecuencia, costos de potencia (\$/kW-año) y energía (\$/kWh) y costo de los condensadores (puede incluir el costo de operación y mantenimiento). Además se debe indicar el número máximo de localizaciones que va a considerar el AG y el número de localizaciones que va a utilizar la heurística de relocalización. Para ello el usuario, mediante el mouse (ratón) y el teclado, proporciona los parámetros del sistema.
- **MODELACIÓN ALGORITMO GENÉTICO:** este menú permite la definición de los parámetros del AG como: el número máximo de generaciones e individuos por generación, la probabilidad de crossover y el número de cortes que se va a realizar a cada individuo (crossover multipunto) y la probabilidad de mutación. Además se puede optar entre tres modalidades

de AGs. El AG clásico (AGC), AG(SUS) que utiliza el método de muestreo estocástico universal y AG(Nichos) en el que no se pierde material genético y que es capaz de crear nichos en la población.

OPTIMIZAR: opción que despliega el menú con las alternativas para optimizar el sistema de distribución eléctrico modelado. Además se puede elegir si se desean correr las heurísticas al término de la optimización.

SALIR : Cuadro de diálogo que permite ya sea terminar la aplicación o bien ver la referencia del programa.

D.2 Funcionamiento

El programa se activa al hacer doble clic sobre el icono correspondiente. Enseguida se despliega la pantalla central del programa (Figura D.1) que contiene el menú principal.

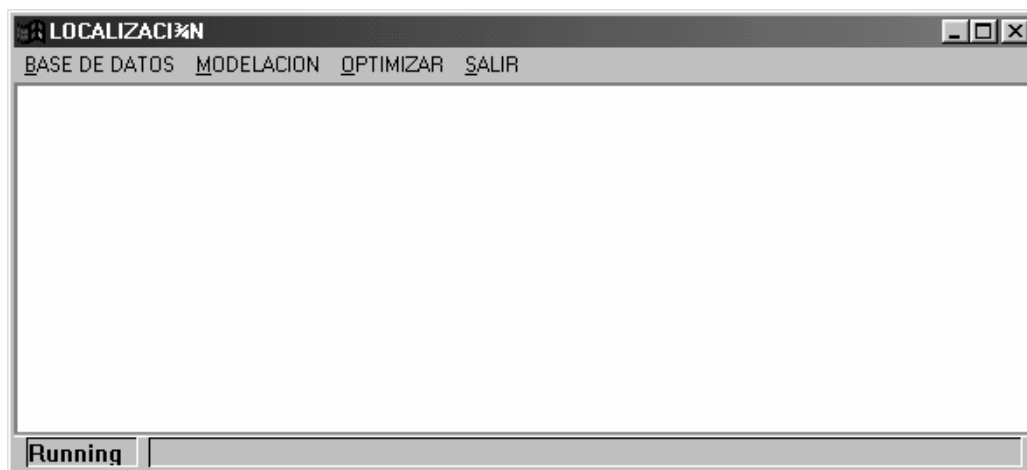






Figura D.1: Pantalla del Menú principal

Pueden observarse, en el extremo superior derecho de la figura N°4.3 los comandos de menú control de windows,  **Minimizar**, que reduce la ventana de la aplicación o programa a una barra de tareas. Para restablecer el tamaño y ubicación de la ventana debe hacer click en el botón de la barra de tareas.  **Maximizar**, permite aumentar el tamaño de la ventana de la aplicación hasta llenar la pantalla. Para restaurar la ventana a su tamaño y posición inicial debe hacer click en , o doble click en la barra de título. Y  **Cerrar**, que como su nombre indica, cierra la aplicación, es una vía alternativa a SALIR del menú principal.

MENU PRINCIPAL

Una vez abierto el programa se tiene inmediatamente acceso al menú principal, (Figura D.2), el cual posee para cada una de las opciones posibles un submenú.



Figura D.2: Menú Principal

Cada submenú cuenta con diversas funciones que constituyen las tareas que puede realizar el programa. Enseguida se detallan los menús y sus respectivos submenús.

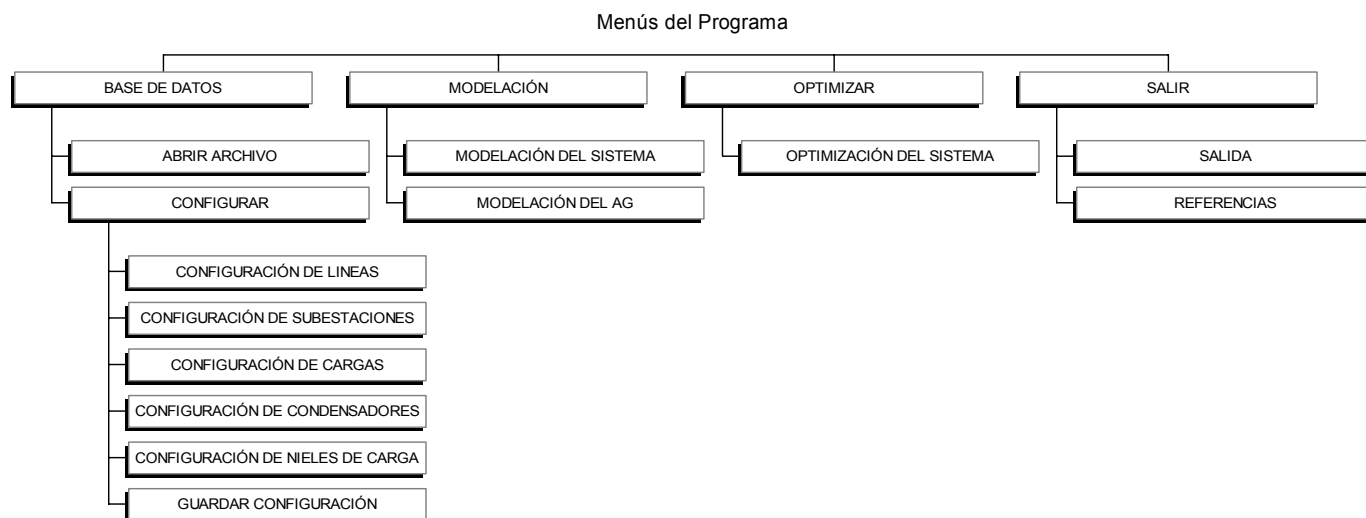


Figura D.3: Menús del Programa

Para activar el menú principal y/o sus distintas opciones basta con situar el puntero del mouse (ratón), sobre la opción deseada y luego hacer click. De igual

forma se logra el mismo objetivo mediante la utilización del teclado, donde para activar el menú principal basta con presionar la tecla **ALT**. Luego presionando la letra subrayada correspondiente a la opción esta se activará, o bien mediante las teclas de direccionamiento **←**, **↑**, **↓**, **→**, en combinación con la tecla **ENTER** para ejecutar las funciones o desplegar un submenú. Es importante notar que el tratamiento de los submenús es idéntico al del menú principal.

BASE DE DATOS

Esta opción agrupa las funciones **ABRIR ARCHIVO** y **CONFIGURAR**, que permiten ingresar al sistema datos, ya sea desde una archivo existente, o bien en forma manual mediante las ventanas de diálogos diseñadas en el programa. A continuación se presenta una completa reseña del modo de operación de éstas.

ABRIR ARCHIVO

La primera opción es **ABRIR ARCHIVO**, la que al ser ejecutada, ya sea mediante una combinación de teclas o el mouse (ratón), crea el cuadro de diálogo que muestra la figura D.4.



Figura D.4: Abrir Base de Datos

Este permite buscar cualquier tipo de documentos que se encuentren tanto en la red como en el disco duro del sistema. Para abrir un cierto documento, basta con localizarlo en la unidad o carpeta correspondiente y hacer doble clic sobre éste o presionar el botón abrir que aparece en el cuadro de diálogo. Si se hace click sobre CANCELAR, el cuadro se cerrará automáticamente volviendo al menú principal.

CONFIGURAR

La opción configurar permite diseñar un sistema de distribución eléctrico compuesto por líneas, generadores y cargas, además del ingreso de información relacionada con los condensadores como tamaño y costo. El cuadro de diálogo correspondiente se presenta en la figura D.5.

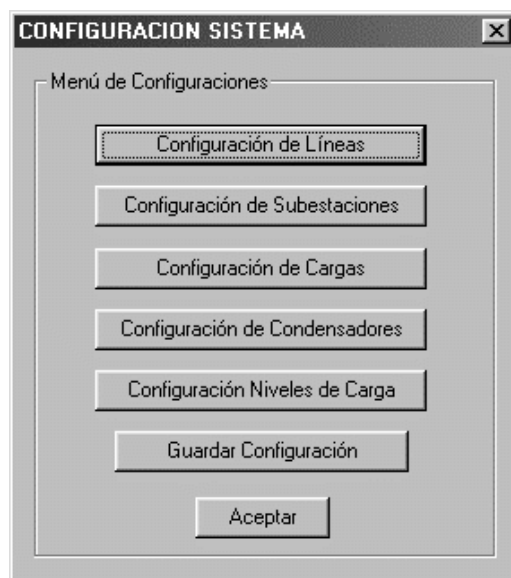


Figura D.5: Configuración Sistema

La selección de cada una de las alternativas anteriores genera una nueva ventana de diálogo, (Figuras D.6, D.7, D.8, D.9, D.10), donde el usuario debe ingresar la información requerida.



Figura D.6: Configuración de Líneas del Sistema

The dialog box is titled "CONFIGURACION DE SUBESTACIONES" and contains the following elements:

- Section: "Configuración Subestaciones"
- Input fields: "N° Nodo Subestación" and "Capacidad Subestación"
- Buttons: "Agregar", "Modificar", "Borrar", "Aceptar", and "Cancelar"
- A large empty rectangular area for a list or table.

Figura D.7: Configuración de Subestaciones

The dialog box is titled "CONFIGURACION NIVEL CARGA BASE" and contains the following elements:

- Section: "Configuración Nivel Carga Base"
- Input fields: "Nodo Carga", "Consumo P [KW]", and "Consumo Q [KVAR]"
- Buttons: "Agregar", "Modificar", "Borrar", "Aceptar", and "Cancelar"
- A large empty rectangular area for a list or table.

Figura D.8: Configuración nivel de Carga base

The dialog box is titled "CONFIGURACION DE CONDENSADORES" and contains the following elements:

- Section: "Configuración Condensadores"
- Field: "Capacidad Condensador [kvar]" with an empty text input box.
- Field: "N° de Condensadores disponibles" with an empty text input box.
- Area: A large empty rectangular box for a list of capacitors.
- Buttons: "Agregar", "Modificar", and "Borrar" are stacked vertically to the right of the list area.
- Buttons: "Aceptar" and "Cancelar" are at the bottom of the dialog.

Figura D.9: Configuración de Condensadores

The dialog box is titled "CONFIGURACION DE NIVELES DE CARGA" and contains the following elements:

- Section: "Nivel de Carga Base"
- Field: "Nivel de Carga [p.u.]" with a text input box containing the value "1.0".
- Field: "Intervalo de Duración Nivel Carga Base [Hrs]" with an empty text input box.
- Section: "Otros Niveles de Carga"
- Field: "Nivel De Carga [p.u.] (Respecto a Nivel de Carga Base)" with an empty text input box.
- Field: "Intervalo de Duración Nivel de Carga [Hrs]" with an empty text input box.
- Area: A large empty rectangular box for a list of load levels.
- Buttons: "Agregar", "Modificar", and "Borrar" are stacked vertically to the right of the list area.
- Buttons: "Aceptar" and "Cancelar" are at the bottom of the dialog.


Figura D.10: Configuración de Niveles de Carga

Todos los diálogos presentados anteriormente tienen un funcionamiento análogo. Se debe ingresar la información en las unidades correspondientes y posteriormente se debe presionar el botón **AGREGAR**. De esta forma entonces, la información de la línea (subestación, carga, condensador o nivel de carga) aparecerá en la pantalla inferior del cuadro de diálogo.

Para ingresar un nuevo elemento se deben sobrescribir las casillas superiores y nuevamente presionar el botón **AGREGAR**. Esta operación se debe realizar tantas veces como elementos se deseen ingresar. Finalmente se debe presionar la tecla **ACEPTAR**.

Si se desea modificar la información ya ingresada basta posicionarse en el cuadro inferior de la pantalla y hacer clic sobre la línea que contiene la información que desea modificarse. Enseguida se debe presionar el botón **MODIFICAR**, lo que provocará que la información aparezca en las casillas superiores respectivas, donde se pueden modificar los valores. Una vez terminada la modificación se debe presionar nuevamente la tecla **AGREGAR**.

Para eliminar una línea de información de la pantalla inferior basta posicionarse sobre ésta haciendo clic con el botón del mouse y presionando la tecla **BORRAR**.

Finalizada las tareas de configuración, el usuario tiene la opción de almacenar el sistema creado haciendo click en **GUARDAR CONFIGURACION**, lo que creará una nueva ventana de diálogo (Figura D.11). Para guardar el documento en una carpeta distinta, se debe hacer click en una unidad diferente del cuadro **Guardar en** o doble click en otra carpeta de la lista de carpetas. Si se desea crear una nueva carpeta, se debe hacer click en **Crear nueva carpeta**  y luego en el cuadro **Nombre de archivo**, escribir el nombre asignado a la nueva base de datos, cuya extensión debe

ser cualquiera. Finalmente hacer click en **GUARDAR** con lo que se retorna al menú principal.

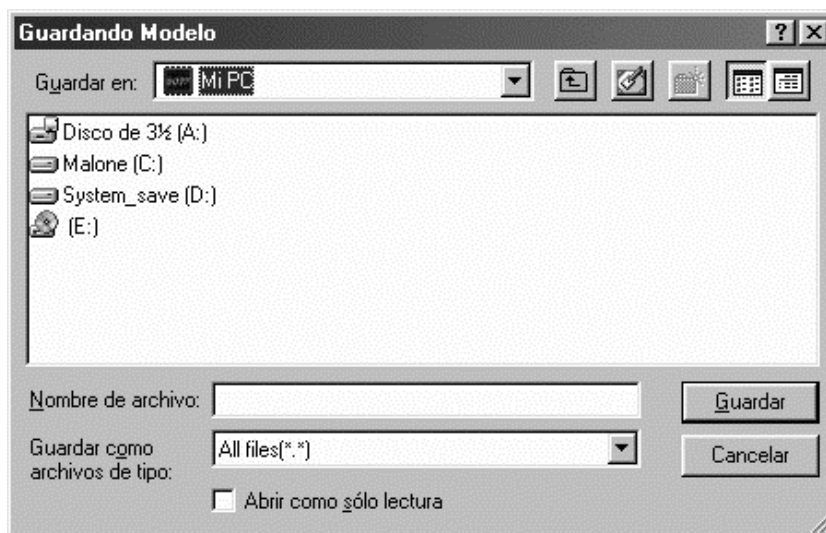


Figura D.11: Guardando Modelo

Si no se almacena la información del modelo, ésta queda de todas maneras disponible para la ejecución de la optimización. Sin embargo si posteriormente se sale del programa el modelo analizado se perderá.

MODELACIÓN

Una vez que se ha terminado con la etapa de configuración se deben definir los parámetros del sistema eléctrico y del AG. Para ello el menú **MODELACION** consta de dos submenús: **MODELACION DEL SISTEMA** y **MODELACION AG**. Ambos procedimientos se detallan a continuación.

MODELACION DEL SISTEMA

Al ejecutar esta función aparece el panel de la figura D.12, donde se distinguen cuatro bloques. El primero corresponde a la configuración de los parámetros eléctricos del sistema: tensión base, tensión mínima, tensión máxima y frecuencia. El segundo corresponde a los costos involucrados en la modelación. Estos son: costo energía, costo potencia y el costo de los condensadores. En el tercer bloque se debe ingresar el número de localizaciones que deberá manejar el AG. Finalmente, en el cuarto bloque se debe especificar el número de localizaciones que deberá considerar la heurística de relocalización.

PARAMETROS DEL MODELO	
Parámetros del Sistema	
Tensión Base [KV]	23
Tensión mínima [p.u.]	0.95
Tensión máxima [p.u.]	1.05
Frecuencia [Hz]	50
Costos	
Costo Energía [Mills/kWh]	60
Costo Potencia [kW]	120
Costo Condensadores [kVAR]	5
Localización de Condensadores	
N° máximo de Localizaciones	
Heurística de Relocalización	
N° máximo de localizaciones a considerar por la heurística	4
OK Cancel	

Figura D.12: Parámetros del Modelo

MODELACIÓN DEL ALGORITMO GENÉTICO

El menú de modelación del AG (figura D.13) permite elegir entre tres AGs distintos. El primero es el AG Clásico que está compuesto por los operadores genéticos de selección (torneo estocástico de selección), crossover (que puede ser del tipo multipunto o simple), mutación y elitismo (que copia los mejores individuos de una generación en la siguiente). El segundo corresponde al AG (SUS) que difiere del anterior en el método de selección empleado. En este caso se utiliza el muestreo universal estocástico ó método de selección de Baker. Además, no utiliza el operador de elitismo. Finalmente, se encuentra disponible la opción de AG (Nichos) que tiene la capacidad formar nichos en la población evitándose la pérdida de material genético.

The image shows a software dialog box titled "CONFIGURACION ALGORITMO GENETICO". It is divided into several sections for configuring genetic algorithm parameters:

- Parámetros Generales:** "Número Máximo de Generaciones" is set to 50, and "Número de Individuos por Generación" is set to 100.
- Crossover:** "Probabilidad Crossover" is 0.8. There is an unchecked checkbox for "Corte Automático" and "Número de Cortes" is set to 1.
- Mutación:** "Probabilidad Mutación" is set to 0.01.
- Métodos:** Three radio buttons are present: "Algoritmo Genético Clásico" (selected), "Algoritmo Genético (SUS)", and "Algoritmo Genético (Nichos)". A "Configurar" button is located to the right of the "Algoritmo Genético Clásico" option.

At the bottom of the dialog are "OK" and "Cancel" buttons.

Figura D.13: Configuración AG

Luego para finalizar el proceso de modelación, basta con hacer click en **ACEPTAR** quedando la información almacenada y regresando al menú principal.

OPTIMIZAR

Terminada la etapa de definición de parámetros del sistema, se procede con la optimización seleccionando OPTIMIZAR del menú principal y haciendo click en OPTIMIZACIÓN DEL SISTEMA.

OPTIMIZACIÓN DEL SISTEMA.

Al ejecutar esta subrutina aparecerá la ventana de diálogo OPTIMIZACIÓN DEL SISTEMA. (Figura D.14).

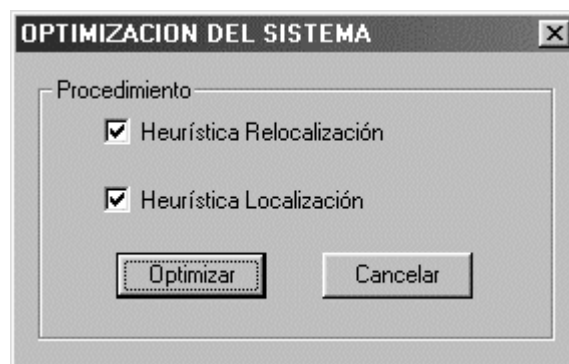


Figura D.14: Optimización del Sistema

El panel presenta un recuadro donde el usuario debe especificar si se desean ejecutar las heurísticas de Relocalización y Localización una vez finalizada la etapa de optimización. Presionándole botón **OPTIMIZAR** se da comienzo a la optimización del sistema eléctrico.

Una vez terminada la optimización se despliega un resumen de resultados en la pantalla principal del programa. Además, se genera un archivo en el mismo directorio donde se ejecuta el programa con un detalle de los resultados.

Finalmente, el usuario puede realizar una nueva optimización modificando algunos de los parámetros antes mencionado y haciendo click en **OPTIMIZAR**.

SALIR

La opción salir presenta dos alternativas, la primera corresponde a la salida propiamente tal, lo que implica el término de la ejecución del programa, mientras que la segunda permite visualizar una breve referencia del programa (ver Figura D.15).

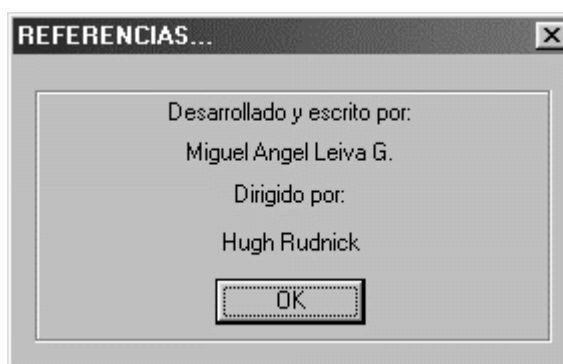


Figura D.15: Referencias

ANEXO E: DATOS SISTEMA DE 9 BARRAS

E.1 Información del Sistema de 9 Barras

Este Sistema de Prueba es de 23 [kV], cuenta con 9 barras y 9 líneas. La subestación se encuentra localizada en el nodo n° 10. A continuación se entregan las impedancias y cargas que componen el sistema.

Nodo i	Nodo j	Long [km]	R [ohm/km]	X [ohm/km]
1	2	6.27	0.85134	0.4826
2	3	5.63	0.85134	0.4824
3	4	2.41	0.85134	0.4829
4	5	1.68	0.5357	0.4694
5	6	3.7	0.5357	0.4669
6	7	1.3	0.5357	0.468
7	8	2.73	0.2727	0.4414
8	9	1.41	0.01	0.4291
9	10	1.01	0.1216	0.4086

Tabla E.1: Impedancias del Sistema de 9 Barras

Nodo	P [kW]	Q [kVAr]
1	1640	200
2	980	130
3	1150	60
4	780	110
5	1610	600
6	1598	1840
7	1790	446
8	980	340
9	1840	460

Tabla E.2: Cargas del Sistema de 9 Barras

E.2 Tensiones para el Sistema de 9 Barras Compensado

Barra	Nivel 1 (1.0 p.u.)	Nivel 2 (0.8 p.u.)	Nivel 3 (0.6 p.u.)
10	23.00	23.00	23.00
9	22.95	22.96	22.96
8	22.97	22.97	22.95
7	22.72	22.76	22.75
6	22.48	22.57	22.60
5	21.97	22.13	22.29
4	21.81	22.00	22.18
3	21.47	21.72	21.99
2	20.93	21.31	21.69
1	20.54	21.03	21.47

Tabla E.3: Tensiones para Sistema de 9 Barras compensado

ANEXO F: DATOS SISTEMA DE 30 BARRAS

F.1 Información del Sistema de 30 Barras

Este Sistema de Prueba es de 23 [kV], cuenta con 30 barras y 29 líneas. La subestación se encuentra localizada en el nodo n° 1. A continuación se entregan las impedancias y cargas que componen el sistema.

Nodo i	Nodo j	Longitud [km]	Resistencia [Ω /km]	Reactancia [Ω /km]
1	2	1.2	0.196	0.655
2	3	0.6	0.279	0.015
3	4	0.6	0.444	0.439
4	5	0.4	0.864	0.751
5	6	0.4	0.864	0.751
6	7	0.7	0.1374	0.774
7	8	0.8	0.1374	0.774
8	9	0.8	0.1374	0.774
9	10	0.4	0.1374	0.774
10	11	0.5	0.1374	0.774
11	12	0.6	0.1374	0.774
12	13	0.5	0.1374	0.774
13	14	0.6	0.1374	0.774
14	15	1	0.1374	0.774
9	16	0.7	0.864	0.751
16	17	1	0.1374	0.774
17	18	1.1	0.1374	0.774
7	19	1	0.864	0.751
19	20	1.2	0.864	0.751
20	21	1.2	0.1374	0.774
7	22	0.5	0.864	0.751
4	23	0.5	0.444	0.439
23	24	0.5	0.444	0.439
24	25	0.5	0.864	0.751
25	26	0.4	0.864	0.751
26	27	0.4	0.864	0.751
27	28	1	0.1374	0.774
2	29	0.5	0.279	0.015
29	30	1	0.1374	0.774
30	31	1.4	0.1374	0.774

Tabla F.1: Impedancias del Sistema de 30 Barras

Nodo	P [kW]	Q [kVAr]
3	522	174
5	936	312
10	189	63
12	336	112
13	657	219
14	783	261
15	729	243
16	477	159
17	549	183
18	477	159
19	432	144
20	732	244
21	495	165
22	207	69
23	522	174
24	1917	639
26	1116	372
27	549	183
28	792	264
29	882	294
31	882	294

Tabla F.2: Cargas del Sistema de 30 Barras