

Universidad Centroccidental “Lisandro Alvarado”



*Laboratorio de Computación Evolutiva:
Aperos Teóricos, Numéricos y Seminuméricos de
Algoritmos Genéticos*

José Gregorio Sánchez

Trabajo de ascenso para optar a la categoría de
Profesor Titular

Abril 2011

Departamento de Sistemas
Decanato de Ciencias y Tecnología

Índice general

1. Introducción	1
1.0.1. ¿Cómo leer este documento?	3
I Bases Biológicas	5
2. Evolución	6
2.1. Introducción	6
2.2. Del Ápeiron a las cuerdas	7
2.3. Evolución Natural	8
2.3.1. Sobre el origen de las especies	9
2.4. El Paradigma Neo-Darwiniano	11
2.4.1. Topografía de Adaptación	12
3. Principios de Genética	15
3.1. Algunos conceptos básicos	16
3.2. Los experimentos de Mendel	17
3.3. Operadores de evolución genética	19
3.3.1. Selección	19
3.3.2. Cruce	19
3.3.3. Mutación	20
4. Búsqueda y Optimización	23
4.1. Búsqueda	24
4.1.1. Espacios de búsqueda	24
4.2. Heurística	26
4.2.1. Meta-heurística	26
4.3. Optimización	27

4.3.1. Optimización combinatoria	27
4.3.2. Optimización Multiobjetivo	28
II Modelos Seminuméricos	31
5. Números Aleatorios	32
5.1. Introducción	32
5.2. Aleatoriedad	34
5.2.1. Desviadas Uniformes	35
5.2.2. Evaluación de Generadores Aleatorios	39
III El Algoritmo Genético	49
6. Algoritmos Genéticos	50
6.1. Introducción	50
6.2. El Algoritmo Genético Simple	52
6.3. Elementos de inicio del Algoritmo Genético	55
6.3.1. Representación	55
6.3.2. Función de adaptabilidad	65
6.4. Operadores de variación	67
6.4.1. Selección	67
6.4.2. Cruce	74
6.4.3. Mutación	76
7. Experimentación	78
7.1. Introducción	78
7.1.1. Consideraciones Importantes	79
7.2. Comparando métodos de selección	80
7.2.1. Diseño experimental:	80
7.2.2. Descripción de los experimentos	81
7.2.3. Resultados experimentales	82
7.3. Comparando métodos de cruce	99
7.3.1. Diseño experimental	99
7.4. Comparando tipos de representación	104
7.4.1. Diseño experimental	104

7.4.2.	Descripción de los experimentos	107
7.4.3.	Resultados experimentales	109
7.5.	El Vendedor Viajero	119
7.5.1.	Extensión al Algoritmo Genético	119
7.5.2.	Diseño experimental y resultados	120
7.6.	Una prueba en Optimización multiobjetivo	124
7.6.1.	Resultados preliminares	126

IV Métodos Numéricos 132

8. Estimaciones Estadísticas 133

8.1.	Introducción	133
8.2.	Distribuciones y descriptores	134
8.3.	Comparando distribuciones	137
8.3.1.	La prueba $t - Student$	137
8.3.2.	Correlación lineal	139
8.3.3.	Diferencia en distribuciones	140
8.4.	Evaluación del AG	142
8.4.1.	Experimentación con el AG	145
8.5.	Importancia para el trabajo científico	146

9. Theorice 148

Índice de figuras

1.1. Estructura del documento	4
2.1. Topografía de Adaptación	12
3.1. Gametogénesis	17
3.2. Experimentos de Mendel	18
3.3. Cruce y recombinación	20
3.4. Tipos de mutaciones	22
4.1. Técnicas de búsqueda	25
4.2. Métodos convencionales de optimización	30
5.1. Distribución Uniforme	35
5.2. Distribución de Gauss	36
5.3. Generador ran16 en 2D	43
5.4. Generador ran16 en 3D	44
5.5. Generador randú en 3D	45
5.6. Generador randú en 3D punto de vista desplazado	45
5.7. Media aritmética y π	46
6.1. Codificación del cromosoma	56
6.2. Cromosoma con múltiples variables	58
6.3. Representación binaria	59
6.4. Representación real	63
6.5. Otras representaciones	65
6.6. Selección de rueda de ruleta	69
6.7. Probabilidad de selección ruleta-rango	72
6.8. Selección por torneo	74
6.9. Cruce de 1 punto	75

ÍNDICE DE FIGURAS

6.10. Cruce de n puntos	76
6.11. Mutación simple	77
7.1. Rueda de ruleta Métricas M1 y M2	87
7.2. Experimentación con Torneo $k = 2$	88
7.3. Experimentación con Torneo $k = 3$	89
7.4. Experimentación con Torneo $k = 4$	90
7.5. Experimentación con selección aleatoria	91
7.6. Comparación Métodos de selección M2	92
7.7. Comparación de métodos de selección M1	93
7.8. Gráfico de Percentil para prueba de 1 punto	100
7.9. Frecuencia y Percentil para cruce de 2 puntos	101
7.10. Frecuencia y Percentil para cruce uniforme	102
7.11. Función Jorobas de camello	106
7.12. Métricas M1, M2 para representación binaria con ruleta	110
7.13. Métrica M3 para representación binaria con ruleta	110
7.14. Métricas M1, M2 para representación binaria con torneo	111
7.15. Métrica M3 para representación binaria con torneo binario	111
7.16. Métricas M1, M2 para representación binaria Gray con ruleta	112
7.17. Métrica M3 para representación binaria Gray con ruleta	112
7.18. Métricas M1, M2 para representación binaria Gray con torneo	113
7.19. Métrica M3 para representación binaria Gray con torneo	113
7.20. Comparación de representaciones con métrica M1	114
7.21. Comparación de representaciones con métrica M4	115
7.22. Función de Ackley	117
7.23. Función madriguera de zorros de Shekel	117
7.24. TSP 50 Nodos	122
7.25. Función de prueba Optimización Multiobjetivo	129
7.26. Comparación de O.M. para 20,100 y 250 iteraciones	130
7.27. GVEGA con W_i variante	131
8.1. Ciclo experimental	147

Índice de cuadros

5.1. Comparación de desviadas uniformes	47
6.1. Tabla de verdad del XOR	59
6.2. Tabla de adaptabilidad	66
6.3. Tabla de adaptabilidad para selección de ruleta	68
7.1. Parámetros de los experimentos con selección	84
7.5. Métodos de selección: M1	85
7.6. Métodos de selección: M2	86
7.7. Prueba t -Student para M1	86
7.8. Prueba t -Student para M2	86
7.2. Resultados de M1 y M2 Rueda de ruleta y selección aleatoria	94
(a). Rueda de ruleta M1 y M2	94
(b). Selección aleatoria M1 y M2	94
7.3. M1y M2 para Selección de Torneo $k = 2, k = 3$	95
(a). Selección de Torneo $k = 2$, M1 y M2	95
(b). Selección de Torneo con $k = 3$, M1 y M2	95
7.4. Torneo con $k = 4$, M1 y M2	96
7.9. Resumen comparativo bajo t -Student	97
7.10. Métodos de cruce a analizar	99
7.11. Cruce de un punto	100
7.12. Kolmogorov-Smirnov para cruce 1 punto	101
7.13. Kolmogorov-Smirnov para cruce 2 puntos	102
7.14. Kolmogorov-Smirnov para cruce uniforme	102
7.15. Parámetros experimentales	108
7.16. Métricas de representación binaria	109
7.17. Métricas de representación binaria Gray	114
7.18. Comparación binaria-Gray con prueba t	118

ÍNDICE DE CUADROS

7.19. Vendedor viajero comparación en M1	123
7.20. VV tolerancia a ruido en M2	123
7.21. Descripción de GVEGA	127
7.22. Comparación de Optimización Multiobjetivo para 20 iteraciones . . .	128
7.23. Comparación de Optimización Multiobjetivo para 100 iteraciones . .	129
7.24. Comparación de Optimización Multiobjetivo para 250 iteraciones . .	129

Lista de algoritmos

1.	El Algoritmo Genético Simple	54
2.	Conversión de binario a Gray	60
3.	Conversión de Gray a binario	60
4.	Selección de rueda de ruleta	70
5.	Selección por truncado	71
6.	Selección por torneo	74

... las cosas están llenas de Dioses.

Anaximandro de Mileto
610 - 546 A.C.

1

Introducción

La I.A. o Inteligencia Artificial (área que cobija al cómputo evolutivo) es una disciplina que desde sus comienzos ha transitado un camino lleno de controversias, muchas de ellas quizás originadas por el descriptor mas elemental de todos: su nombre. Desde sus comienzos a mediados del siglo XX, los debates entre los alcances, límites y naturaleza de lo que llamamos *inteligencia* han sido muy activos entre defensores de posiciones opuestas y de si es filosóficamente posible para un sistema *inanimado* tener un atributo calificativo de *inteligente*. Desde el enfoque de Marvin Minsky, uno de los padres de la I.A., que plantea la posibilidad del cerebro humano como una máquina cuasi-determinista para la que aún no poseemos la capacidad tecnológica de paralelismo ni los modelos matemáticos completos para simular sus bloques básicos, hasta las posiciones de pensadores mas *humanistas* quienes argumentan que el *sentido común* por ejemplo, es mas que la simple aplicación de reglas de producción y/o árboles complejos de decisión. Todos estos argumentos han tenido sus altos y bajos en los activos debates en el mundo de la inteligencia artificial, y son significantes los aportes que ésta le ha brindado a la ingeniería y ciencias tecnológicas en general.

Dejando de lado los antagónicos debates (posiblemente bizantinos) y las discusiones de las posibilidades filosóficas de la inteligencia artificial, un aspecto que

pareciera tener amplia aceptación es que la capacidad de adaptación de un objeto a su entorno denota (al menos hasta cierto punto) algún grado de inteligencia [49, 37], y es en esta línea que la computación evolutiva basa sus principios y guías. La estrategia de adaptación ha sido usada por los seres vivos para resolver el arduo y complejo problema de sobrevivir.

Los algoritmos genéticos son métodos computacionales basados en la dinámica evolutiva de los seres vivos, y por lo tanto, modelan aspectos biológicos de naturaleza compleja, continua y al menos a nuestro actual entendimiento: no determinista. Este modelado se realiza a través de herramientas de programación discretas y en su totalidad deterministas, no obstante, el éxito que han tenido tanto en el mundo académico como en el empresarial e industrial los han dado a conocer como poderosas herramientas en la solución de problemas no triviales. Este amplio historial de aplicaciones exitosas los han hecho muy populares en los trabajos de investigación y desarrollo en una gran diversidad de áreas del saber.

En el estudio de los algoritmos genéticos es importante el conocimiento de los mecanismos naturales que se modelan, así como también, los métodos numéricos sobre los que se apoyan para simular estos procesos biológicos y para comparar el rendimiento del algoritmo propuesto con algún o algunos parámetros de calidad establecidos. Siendo estos algoritmos procesos estocásticos, el investigador debe tener sumo cuidado con las conclusiones que deriva de su trabajo experimental, y esto significa también que especial cuidado debe haber sido puesto en el diseño de los experimentos, de manera de reducir el grado de especulación en las observaciones y reportar resultados con un respaldo apropiado de *tendencia* estadística. No es difícil encontrar trabajos de investigación donde algunas conclusiones no están suficientemente sustentadas, lo que las convierte en el mejor de los casos, en observaciones especulativas, y en el peor caso, en un error o una falacia. Todo investigador y en especial aquellos que trabajan o se inician en la experimentación con procesos estocásticos, debe conocer los aspectos teóricos importantes en la observación de comportamientos y tendencias experimentales y reportar estas observaciones con un lenguaje técnico apropiado.

Este texto se diseña como una guía referencial de aperos teóricos de modelos seminuméricos y de programación numérica de algoritmos genéticos, orientada a estudiantes de postgrado e investigadores en el área, en especial a los participantes de la mención de Inteligencia Artificial del programa de Maestría en Ciencias de la

Computación del Decanato de Ciencias y Tecnología de la UCLA. Contiene lo que estimo son las herramientas necesarias para la construcción de un algoritmo genético simple con base no solo en los aspectos teóricos y filosóficos que lo soportan sino también en los elementos de programación numérica y seminumérica que permitirán evaluar y medir su desempeño en términos de un análisis estadístico apropiado. Cada capítulo se centra en la discusión teórica y estudio práctico de los principales constructores del algoritmo genético de manera de brindar al lector una base de fundamentos, o un marco referencial sobre el cual pueda derivar sus propios argumentos y conclusiones.

Como muestra la Figura 1.1, esta guía de útiles comprende cuatro grandes partes generales desarrolladas en varios capítulos: Los fundamentos teóricos, donde se discuten los principales elementos que dan base a los algoritmos genéticos como la teoría de la evolución, la genética, los operadores biológicos de cambio, etc. Los Modelos seminuméricos (término acuñado por Donald Knuth para referirse a los algoritmos y elementos de interfaz entre la programación simbólica y la programación numérica), el algoritmo genético en sí y los métodos numéricos de apoyo que incluyen desde la descripción de las desviadas uniformes para la simulación de aleatoriedad, hasta las pruebas estadísticas de análisis comparativo que permitan evaluar el rendimiento. En cada caso, se plantean los elementos filosóficos básicos y las descripciones teóricas de manera de complementar y expandir la visión del lector sobre el elemento de estudio, y tenga éste la base necesaria y suficiente no solo para proponer extensiones a los operadores de cambio u otros constructores del algoritmo, sino también para diseñar sus propias métricas y pruebas de rendimiento en la evaluación de una aplicación en un dominio específico .

1.0.1. ¿Cómo leer este documento?

Este documento esta diseñado con contenido independiente que no necesita la lectura de un capítulo previo para la comprensión de uno siguiente, aunque por supuesto, algunas referencias cruzadas son planteadas. Se asume que el lector posee cierto conocimiento sobre el paradigma computacional evolutivo o al menos conocimiento de programación, matemática y estadística a nivel de ingeniería básica. Este compendio de aperos de algoritmos genéticos busca brindar apoyo al estudiante/investigador durante todo el proceso de desarrollo y aplicación de un A.G., desde su diseño hasta la evaluación de resultados y formulación de conclusiones.

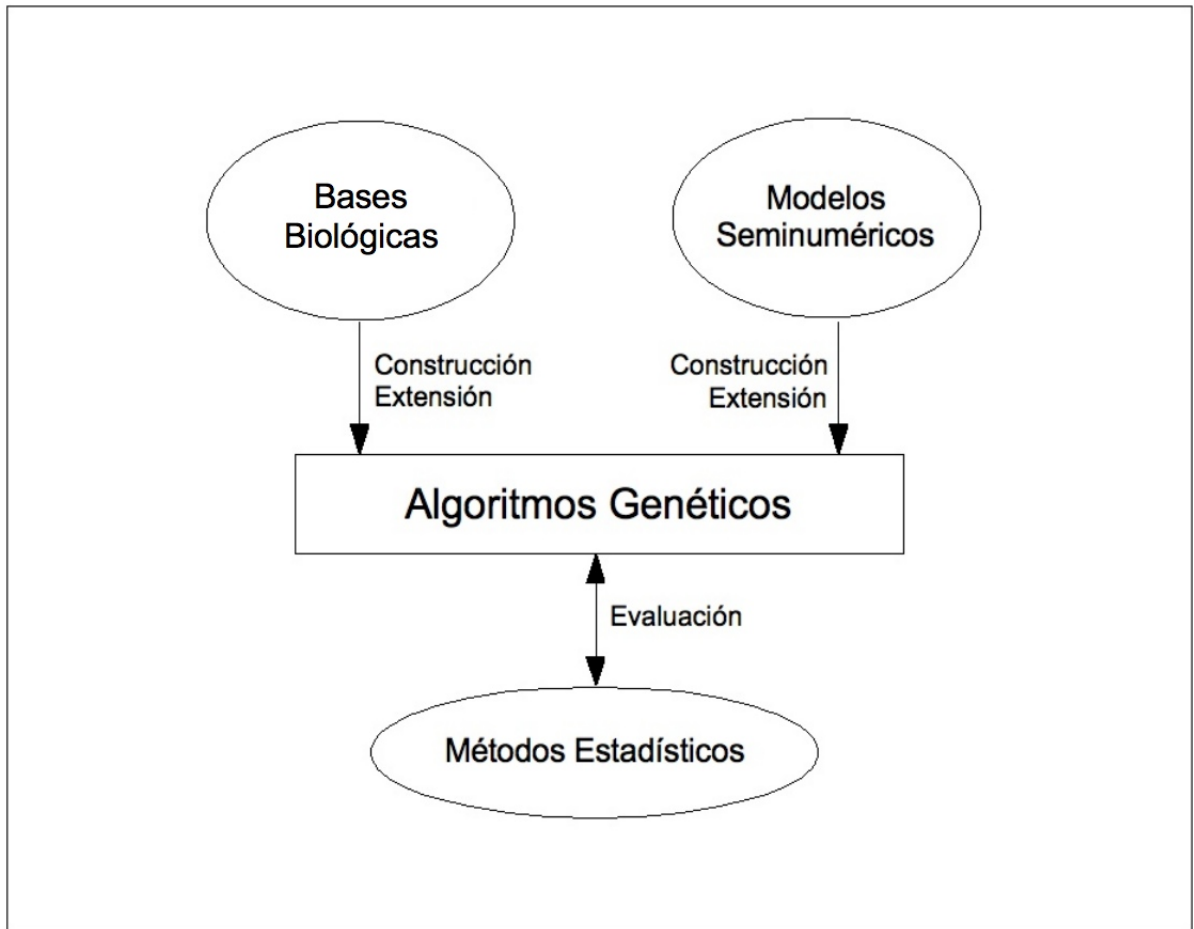


Figura 1.1: Estructura general del texto. Los fundamentos teóricos y modelos semi-numéricos de apoyo para la elaboración y propuestas de nuevos algoritmos genéticos, y los métodos estadísticos necesarios para la evaluación y formulación de conclusiones del trabajo experimental.

Parte I
Bases Biológicas

La teoría de evolución por selección natural acumulada es la única teoría que tenemos, que en principio es capaz de explicar la existencia de la complejidad organizada.

Richard Dawkins

2

Evolución

2.1 Introducción

En la Europa del siglo XVII, la teoría mas ampliamente aceptada sobre el origen de la vida era la *creación especial*, en la que un Dios todopoderoso creaba en un mismo instante de tiempo todo el arreglo de seres vivos que vemos hoy. En 1664 un clérigo irlandés hizo pública una aseveración ciertamente particular, en su libro *Annalium pars posterior* James Ussher aseguraba que:

“Cielo y tierra, centro y substancia fueron hechos en el mismo instante de tiempo, y las nubes llenas de agua y el hombre fueron creados por la Trinidad el 26 de octubre de 4004 A.C. a las 9:00 de la mañana.”

La arqueología moderna ha demostrado que al menos en el año esta aseveración es incorrecta, pues nuestro registro fósil proporciona evidencia de vida en la tierra con mucha anterioridad a esta fecha, no obstante, estas evidencias no estaban disponibles en la Europa de ese tiempo y la aseveración de Ussher encontró seguidores en todos los estratos de la sociedad. El paradigma de la creación especial gobernaba el pensamiento de la época, aún y cuando decenas de siglos atrás algunos eruditos de distintas escuelas griegas de pensamiento habían propuesto teorías mas racionales

2.2. DEL ÁPEIRON A LAS CUERDAS

sobre el origen del universo, como Demócrito y Leucipo de la escuela atomista, Tales y Anaximandro de la escuela jónica milesia entre otros.

En este capítulo se presentarán y discutirán las ideas que dieron origen a la teoría de la evolución y los elementos principales de la biología evolutiva. Desde las primeras ideas de Anaximandro de Mileto, a los postulados de Darwin y los fundamentos de la genética moderna. Los operadores biológicos evolutivos también serán discutidos en detalle de manera de facilitar una base teórica para la mejor comprensión de estos operadores como aperos de algoritmos evolutivos en general.

2.2 Del Ápeiron a las cuerdas

Desde antes del siglo VI previos a nuestra era, el pensamiento humano ya se enfocaba en una teoría racional sobre la mecánica y los principios fundamentales de la vida, de lo animado e inanimado, de todo lo existente; del universo. La escuela jonia ya debatía en términos mas “racionales” lo que cerca de 5000 años antes, los *vedas*¹ describían como el punto de partida del universo: un momento de vibración al que describieron como el “*om*”, el sonido primordial, el origen, el principio *Brahman*. Uno de los mas destacados filósofos de la escuela jonia, Anaximandro de Mileto, postuló una teoría sobre el principio fundamental del todo, el “*arjé*” de todos los mundos (los “*kosmoi*”), el origen del cosmos, de la vida y los elementos inertes. Anaximandro, fiel a su línea de pensamiento, no atribuía a los Dioses del Olimpo el fenómeno de la creación, en lo que hemos derivado de sus trabajos y de lo que a él se le acredita a través de otros ensayos, propone que el origen del mundo vino de un tipo de materia, sin límites (eterna), dimensiones (infinita), ni naturaleza definida de la que se formaron a través de procesos de separación, todos los componentes que observamos [43]. A este primordial elemento le llamó *ápeiron* (indefinido, ilimitado, infinito).

Anaximandro es considerado por muchos como el primer evolucionista, el padre de toda teoría de evolución. Su principio de la especialización o desarrollo de lo complejo a través de la organización y sinergia de elementos mas básicos hasta llegar al ápeiron, es la generalización mas antigua de la teoría que cerca de 2300 años mas tarde comenzaría a plantear Erasmus Darwin y que en 1859 publicaría su nieto

¹Textos sagrados de la filosofía hindú

Charles en “*El origen de las especies*”. Casi 2500 años después de su ápeiron, los planteamientos de Anaximandro de un elemento primordial, sin límites, no tangible, no divisible, no determinista y omnipresente, se mantiene en vigencia en algunos ejes de la física teórica. El *om* de los vedas y el ápeiron del filósofo jónico, son para algunos, las *cuerdas*² que buscan y discuten los físicos de hoy.

2.3 Evolución Natural

¿Quiénes somos?, ¿De dónde venimos?, estas preguntas han acompañado a la humanidad desde su principio. En el quehacer intelectual humano siempre ha estado activa la indagación de estas interrogantes. A través de la historia, algunos han buscado respuesta en dominios no formales, mas allá de la razón y el entendimiento, otros como Anaximandro, sugirieron alternativas mas acorde a los principios de la mecánica universal que habían generalizado por observación, solo que a niveles de abstracción mas complejos para los que no teníamos (ni tenemos), leyes determinantes definidas.

Hoy por hoy, la biología evolutiva y la genética son disciplinas ampliamente estudiadas. Muchas teorías se han desarrollado en estas áreas pero aún no existe un meta-modelo que resuelva los fundamentos filosóficos que estructuran éstas ciencias; ¿Exactamente cuál es la causa natural de la mutación?, ¿Por qué aparentemente se observa que algunas bacterias mutan hacia rasgos beneficiosos dependiendo de las condiciones del ambiente? [32], ¿Qué exactamente es lo que nos hace elegir una pareja por sobre otra?, ¿Son todas estas cosas atribuibles solo a la casualidad o habrán patrones deterministas hasta cierto grado que describan su ocurrencia? En 1831 el buque H.M.S. Beagle zarpó desde Plymouth, Inglaterra, con rumbo al Pacífico sur. Uno de los pasajeros era un naturalista que algunos años después publicaría sus conclusiones y teorías basadas en las observaciones y análisis de la información que recogió en este viaje. Este naturalista era Charles R. Darwin siguiendo los pasos del también naturalista Erasmus Darwin, su abuelo, y quien ya había avanzado algunos estudios en la línea que Charles seguía. La publicación en 1859 de *Sobre el Origen de las Especies*; representaría un revolucionario, y controversial punto de cambio en la forma como los humanos nos miraríamos a nosotros mismos en lo adelante.

²Referente a la teoría de cuerdas y teoría de supercuerdas de la física moderna.

2.3. EVOLUCIÓN NATURAL

Este controversial libro y mas tarde el descubrimiento de los trabajos de Gregor Mendel, los experimentos de Thomas Morgan y otros, fundarían las bases de una gran revolución en la actividad científica de la biología, detonando la chispa que encendería el motor de la experimentación sobre los *cambios observados en las características de las poblaciones en el tiempo*, o mas técnicamente, *los cambios en la frecuencia de los rasgos cromosómicos observados en el tiempo*, en otras palabras, evolución.

2.3.1. Sobre el origen de las especies

Aún y cuando los temas sobre el comienzo de la vida fueron tratados con anterioridad a la publicación de Darwin, y aunque estas observaciones fueron hechas bajo el escrutinio de reglas de la lógica, no existe, o no se tiene conocimiento de un registro de recolección de información organizada, sistematizada y supervisada en esta área, como tampoco se conoce la práctica de una metodología e inspección cartesiana en el desarrollo de las conclusiones sino hasta el trabajo oficialmente presentado por Charles Darwin en 1859. Luego de un meticuloso, paciente y largo período de observación de fenómenos naturales, Darwin argumentó que la diversidad de tipos y especies dispersas por todo el mundo, son consecuencia de un proceso paulatino y gradual de diferenciación genética bajo presiones de condiciones del ambiente. En sus propias palabras, tomadas de [10]:

“Considerando el origen de las especies, es concebible que un naturalista reflexionando sobre las afinidades mutuas de seres orgánicos, sus relaciones embriológicas, su distribución geográfica, sucesión geológica y otros factores similares, pueda llegar la conclusión que las especies no fueron independientemente creadas, sino que descendieron, como variables de otras especies...”

Esto significa que tiempo atrás las diferencias entre grupos de especies eran menos marcadas y que incluso, aún mas atrás en el tiempo, algunas especies hoy cercanas no existían de por sí sino que vivían como la misma especie ancestral. En el laboratorio natural de las Islas Galápagos donde Darwin realizó sus investigaciones, encontró diferencias importantes entre algunas especies de aves, reptiles y mamíferos locales con sus congéneres en tierra firme. Estas diferencias en sus características fenotípicas permitían a las especies insulares desenvolverse mejor en su ambiente, es decir, se habían adaptado al medio ambiente donde por generaciones habían estado.

Postulados de Darwin

El estudio de Darwin se pueden resumir en cuatro conclusiones fundamentales:

1. Los individuos dentro de un grupo de especie son variables.
2. Algunas de estas variaciones son transmitidas a la descendencia.
3. En cada generación, un mayor número de crías son mas aptas para sobrevivir y reproducirse.
4. La supervivencia y la reproducción no son eventos aleatorios. Aquellos que sobreviven y se reproducen, o aquellos que se reproducen mas, son los que poseen las variaciones mas favorables (según el ambiente), en otras palabras, son beneficiados por la *selección natural*.

Para el momento en que esta teoría se publicó, nada se sabia sobre la mutación o la genética en general, por lo que Darwin no pudo responder satisfactoriamente algunos de los planteamientos de los críticos, específicamente en lo referente a la variación. Estas críticas argumentaban que el volumen de variabilidad en las poblaciones era limitado, y que por lo tanto la selección natural disminuiría hasta acabarse cuando esta variabilidad fuera nula.

Hoy, estudios científicos alrededor del mundo apoyan la validez de estos cuatro postulados. Numerosas observaciones en muchas especies diferentes han confirmado pequeñas variaciones dentro de grupos presentes en áreas geográficas separadas [23, 18, 5]. Gregor Mendel introdujo los términos de *dominancia* y *recesividad* genética [35], y por primera vez dio un indicio de cómo los rasgos y variaciones en organismos eran pasadas a sus crías. Los trabajos de Mendel sobre sus plantas de frijoles fueron los pioneros que abrieron el camino y ofrecieron suelo firme a los científicos genetistas y biólogos evolucionistas.

Las leyes de Mendel, los experimentos y observaciones científicas hechas en diversos laboratorios del mundo sumado a nuestro actual conocimiento de la genética, de la mutación, y de otros operadores de cambio, nos permiten asegurar la validez de los postulados uno y dos. Similarmente, nos brindan las bases para aceptar la factibilidad de los postulados tres y cuatro.

2.4 El Paradigma Neo-Darwiniano

Desde los postulados de Darwin y los trabajos subsiguientes, se ha “configurado” lo que en la actualidad es probablemente la teoría más ampliamente aceptada de entre toda la colección de teorías evolutivas; el paradigma Neo-Darwiniano [16]. Desde este enfoque, se argumenta que las observaciones en los cambios en la línea cronológica de la historia de la vida se le puede acreditar a cuatro procesos básicos: [26]

1. Reproducción
2. Mutación
3. Competencia
4. Selección

La reproducción se completa con la transferencia de la carga genética de un individuo a otro y si todos los individuos de una población son exitosos reproduciéndose, entonces ésta aumentaría de manera exponencial. La mutación aparece en los inevitables errores de transferencia de información genética y tiene un jurado evaluador en la selección, que en la medida en que hayan más individuos compitiendo por los recursos disponibles (como por ejemplo otros individuos de género opuesto en el caso de especies con reproducción sexual), será o ejercerá mayor presión sobre los individuos contendientes.

Estos cuatro procesos componen la meta-estrategia evolutiva [16, 26], que en un proceso gradual, paulatino y continuo acercan a la población hacia niveles de aptitudes cada vez mayores, es decir, la interacción de estos cuatro procesos (que actúan a nivel del individuo, no de la población) paulatinamente desplazan la masa poblacional a niveles de mayor adaptabilidad. La evolución no es un proceso de perfección, es decir, no está intrínsecamente orientado a alcanzar un nivel de excelencia predeterminado debido a que el proceso en sí es “ciego”, o sea, no establece metas ni planificación de rutas ni estándares de calidad ni niveles de excelencia para alcanzar las metas propuestas bajo algún criterio de calidad. Desde la perspectiva neo-Darwiniana, la evolución es un proceso de solución de problemas donde una respuesta de adaptación es favorecida en la medida que brinde al individuo ventajas en su capacidad reproductiva y/o de supervivencia (para la reproducción).

2.4.1. Topografía de Adaptación

El genetista Sewall Wright planteó en sus trabajos sobre teoría evolutiva una manera de describir la adaptación de un individuo y la “migración” de una población hacia niveles de adaptabilidad mayores. Propuso lo que denominó *topografía de adaptación*³ también traducido como *paisajes adaptativos*, donde partiendo de la premisa que el *genotipo* de un individuo configura su *fenotipo*, y que éste a su vez establece la relación de ese individuo con el ambiente en términos de su adaptabilidad, representó los niveles de adaptación en una especie de terreno o mapa donde los picos representan los niveles de adaptación (mas alto → mayor adaptabilidad), de manera que mientras mas alto estuviera un individuo en el terreno, mas alto su nivel de adaptación a este terreno.

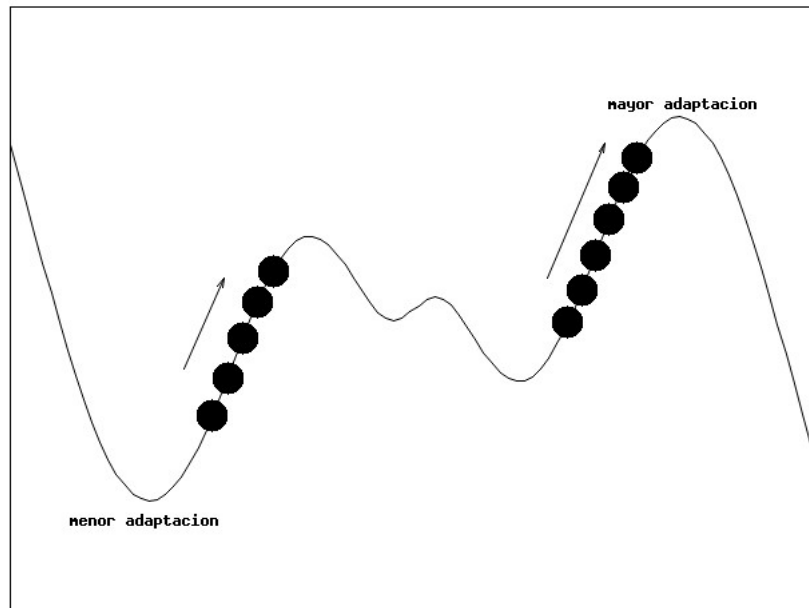


Figura 2.1: Una topografía de adaptación. Los puntos oscuros representan individuos en la población, las flechas denotan la dirección en que el proceso evolutivo lleva la tasa colectiva de adaptación.

³Del inglés *adaptation landscapes*

2.4. EL PARADIGMA NEO-DARWINIANO

Cada pico en la figura 2.1 corresponde a una colección de fenotipos optimizados lo que implica una o mas colecciones de genotipos optimizados también, puesto que los primeros son codificados a partir de información de los últimos. El proceso evolutivo “avanza” de manera probabilística escalando las pendientes del terreno hacia la cima de los picos, la competencia estimula este movimiento recibiendo esporádico impulso de la mutación, la reproducción garantiza que estas iniciativas continúen en generaciones subsiguientes y la selección va descartando las muestras poblacionales que ejercen fuerza negativa en el ascenso, es decir, la selección va descartando aquellas fórmulas fenotípicas que se alejan de los picos o que no van “marchando” hacia ellos.

Esta dinámica tiene como consecuencia que en un período de tiempo lo suficientemente largo, la población debe estar, en términos de una tendencia central, ubicada en los picos mas altos de la topografía de adaptación, no obstante, esto seria solo si el ambiente es estático, es decir, si las condiciones que formaron el perfil topográfico no han cambiado durante todo el lapso de tiempo estipulado, y eso en principio nos plantea un conflicto con la mecánica universal, como lo plantearía otro antiguo filósofo también de la escuela jonia, Heráclito de Éfeso, que el principio universal es el fuego, el movimiento, el cambio. El ambiente (sea éste físico, conceptual, virtual, etc.) está en constante cambio, un pequeño estímulo en una variable o interacción de variables pueden ocasionar un cambio importante en la topografía del terreno de adaptación, por ejemplo, un cambio de estado en las condiciones que forman el ambiente puede ocasionar que lo que antes era un pico pase a ser ahora un valle y que otra montaña resurja en un punto en sentido opuesto al avance evolutivo que se traía. La selección y mutación frenarán la inercia en el avance hacia el antiguo pico haciendo el giro correspondiente hacia el nuevo destino, la competencia y la reproducción seguirán a esto aportando la energía que dará los primeros impulsos hacia la nueva cima.

Desde esta perspectiva, podemos apreciar entonces como la evolución no es en realidad un proceso que lleve a la perfección, es decir, no es un proceso que se plantee metas y estrategias para alcanzarlas bajo algún criterio de calidad determinado. La evolución es una efectiva estrategia de resolución de problemas en el aquí y el ahora de un contexto determinado y que, siendo un proceso “ciego”, permite la flexibilidad de operar en cualquier entorno, es decir, de adaptarse a el. La evolución es un proceso independiente del dominio en el que actúa, ocurre a nivel del individuo y por las interacciones de este con sus congéneres, ejerce influencia a nivel de grupo.

2.4. EL PARADIGMA NEO-DARWINIANO

En [34] (citado en [16]) el autor resume las características principales del paradigma neo-Darwiniano, entre estas:

- El individuo es el objetivo primario de la selección.
- La variación genética es principalmente un fenómeno aleatorio. Los procesos estocásticos juegan un papel significativo en la evolución.
- La variación genotípica es principalmente un producto de la recombinación y mutación.
- La evolución gradual puede incorporar discontinuidades fenotípicas.
- No todos los cambios fenotípicos son necesariamente consecuencia de una selección natural *ad hoc*.
- La evolución es un cambio en la adaptación y diversidad, no solo un cambio en la frecuencia de los genes.
- La selección es probabilista no determinista.

El individuo y no el grupo, es el objetivo primario de la selección. La descripción de un fenómeno (como por ejemplo el altruismo animal) basado en una tesis de "... por el bien de la especie...", o "... para preservación de la especie por encima del individuo...", no tienen mucho sentido desde una perspectiva evolucionista [11]. La preservación de la especie es consecuencia de la preservación del ego del individuo, es la preservación de éste con fines de incrementar su éxito reproductivo, lo que mantiene la especie en sí. El objetivo de la reproducción es pasar el material genético a generaciones siguientes, y en este sentido, algunas paradojas como la del altruismo animal (que Darwin no pudo responderse a su satisfacción), tienen sentido cuando se analizan desde el enfoque de la selección sexual, por ejemplo, en [25], el autor describe y plantea un modelo que en concordancia con la teoría evolucionista, da cabida al altruismo animal. En sus estudios describe un escenario basado en la relación genética de los individuos donde el altruismo es factible, y plantea un modelo matemático-estadístico para predecir este comportamiento. Para mayor referencia en el comportamiento social y la evolución, ver [46, 25, 11, 55].

La genética es la ciencia que explica el por qué te pareces a tu padre, y si no te pareces, el por qué deberías ...

Anónimo

3

Principios de Genética

La genética es uno de los mas jóvenes y controvertidos campos de la ciencia moderna que se ocupa en principio del estudio de los bloques fundamentales del código de la vida; su gestión y transmisión, y es esto lo que la hace un campo de gran importancia para el entendimiento de diversos fenómenos en áreas susceptibles al comportamiento humano como la psicología, la economía, la publicidad, el mercadeo, etc, así como también un entrometido campo que toca delicadas fibras en áreas como la religión, la teología y la filosofía entre otras.

Los numerosos y fuertes debates en temas de ética, moralidad, teología, deontología, espiritualidad, etc. alrededor del campo de acción de la genética, en especial desde que a finales del siglo XX fue “clonado” con éxito un ser vivo superior (la oveja Dolly), son reflejo del fuerte impacto que este campo ha tenido en las diversas ramas del saber, desde las ciencias sociales y humanísticas, hasta las ciencias naturales, la ingeniería y la tecnología.

Este capítulo introduce los principios y conceptos básicos de la genética, sus procesos fundamentales y sus principales postulados, sus mecanismos de transmisión y reparación de errores de la información bioquímica haciendo mayor énfasis en aquellos sobre los que la computación evolutiva se inspira de manera mas directa.

3.1 Algunos conceptos básicos

En el argot de la computación evolutiva comúnmente se encuentran términos de la genética para referir ideas similares dentro del contexto de la simulación del proceso adaptativo biológico. En particular los Algoritmos Genéticos emulan el proceso adaptativo dentro de una misma especie y por tanto incluyen el operador de cruce con reproducción sexual. En el resto del capítulo se toma como base este tipo de reproducción y los términos asociados a ella.

En la reproducción por vía sexual, el núcleo de una célula masculina (esperma), se funde con una célula femenina (óvulo) cada uno aportando un 50% de la información del código de diseño de un nuevo ser vivo. Este ser pasa entonces a formar parte de la *prole* o descendencia. Tanto el óvulo como el espermatozoide, es decir, las células masculinas y femeninas o los *gametos* llevan la información en forma de un complemento simple de material nuclear dispuesto y arreglado en forma de cuerda llamado *cromosoma* [41]. Son complementos simples porque llevan solo una parte de la información, un paquete sencillo definido como un conjunto *haploide* (abreviado como n) que al fundirse uno con otro, forman una célula llamada *cigoto* que tiene un complemento doble llamado conjunto *diploide* ($2n$).

La formación de gametos, o células sexuales ocurre en la *gametogénesis* gracias a un proceso especial de división celular llamado *meiosis* [41]. Este proceso (también llamado división de reducción) permite crear células haploides (n) dejando solo una copia de cada cromosoma en cada gameto. En la fusión de los gametos haploides, las dos copias del cromosoma forman el conjunto diploide del cigoto. Por esta razón, las células diploides contienen 2 copias de cada cromosoma, no obstante, para los cromosomas del sexo, denotados como X para el género femenino y Y para el masculino, este no es el caso. El sexo de un individuo es determinado por un par de cromosomas. Cuando un gameto femenino (X) se funde con un gameto masculino con la copia X del mismo cromosoma, entonces el cigoto formado tiene el conjunto diploide $X + X$, es decir, XX lo que hace que se desarrolle en un individuo femenino. Si por otro lado el gameto masculino (creado en la gemtogénesis) quedó con el conjunto haploide Y , entonces la célula cigoto resultante tendrá el conjunto diploide $X + Y = XY$, y en el desarrollo el individuo será de género masculino. Por esta razón el cromosoma sexual del género masculino no puede ser *homólogo* puesto que el par de cromosomas presentes son diferentes.

3.2. LOS EXPERIMENTOS DE MENDEL

En la Figura 3.1 se observa los pasos en el que dos individuos M y F con células diploides $2n$, crean gametos haploides n a través de la meiosis (división de reducción), y al fundirse en el proceso reproductivo crean una célula diploide (el cigoto) que lleva 50% de la información de cada uno de los gametos de los padres.

La configuración genética del individuo, es decir, los patrones y formas del código genético heredado es llamado el **genotipo**, mientras que el conjunto y acomodo de las características observables en ese individuo constituyen su **fenotipo**. La unidad funcional hereditaria es el **gen** que puede tener varias formas en una ubicación específica o **locus** del cromosoma. A estas diferentes posibilidades de configuración del gen se le llama **alelos**, es decir, un alelo es una de dos o mas secuencias específicas de ácido desoxirribonucleico (ADN) de un gen en particular.

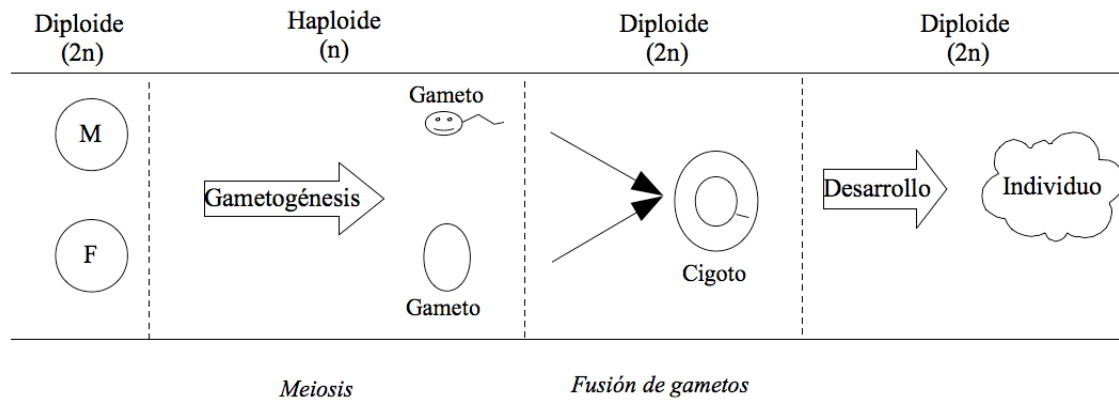


Figura 3.1: Formación de células haploides (gametos) a partir de células diploides a través de meiosis y formación de células diploide (cigoto) a partir de los gametos luego de la fusión.

3.2 Los experimentos de Mendel

Gregor Mendel fue un monje checo que a mediados del siglo *XIX* realizó una serie de experimentos en plantas que llevarían posteriormente a la formalización de

3.2. LOS EXPERIMENTOS DE MENDEL

la genética como un campo de estudio per se.

Mendel tomó plantas de granos con semillas corrugadas y semillas lisas y las cruzó entre ellas hasta conseguir solo semillas del mismo fenotipo, es decir, cruzó corrugadas con corrugadas y lisas con lisas descendencia tras descendencia, hasta conseguir plantas de linaje puro, o sea, hasta que todas las semillas producidas por las plantas de granos corrugados fueran corrugadas y las producidas por los cruces de lisas fueran todas lisas. Una vez obtenido un linaje puro, Mendel cruzó corrugadas con lisas y obtuvo una primera generación de híbridos (llamada en términos experimentales F1) donde todas eran lisas. Luego cruzó los híbridos F1 entre sí y obtuvo una segunda generación; los híbridos F2. En esta generación obtuvo corrugadas y lisas en una relación de 3:1, es decir, por cada 3 lisas había una corrugada (ver Figura 3.2). En cada experimento, encontró que los F1 eran siempre de un mismo tipo y que ambos tipos aparecían en F2, llamó a la forma que aparecía en F1 como *dominantes* y la forma que re-aparecía en F2 *recesivas* [41].

En estos experimentos, cuando Mendel repetidamente cruzó las plantas para obtener un linaje puro, obtuvo organismos con cromosomas en sus células diploide con pares homólogos que eran idénticos para cada gen, es decir, eran *homocigoto*, mientras que los híbridos en la descendencia F_n con copias diferentes del mismo gen (una forma dominante y otra recesiva) eran *heterocigoto*.

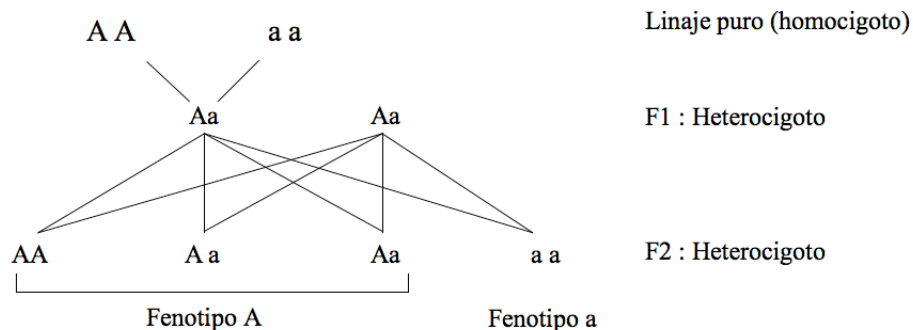


Figura 3.2: Representación genérica de un experimento de Mendel para una característica fenotípica. Formas dominante (A) y recesiva (a) de una rasgo cualquiera. La relación entre formas A y a en la generación F2 es de 3:1

3.3 Operadores de evolución genética

Los cambios en la frecuencia de alelo en el tiempo son posibles gracias a la acción de los operadores genéticos de evolución. Estos operadores permiten al individuo adaptarse a su entorno facilitándoles así la ardua tarea de supervivencia. La selección de fórmulas genéticas exitosas, el cruce y recombinación de éste material y los sutiles cambios esporádicos (y hereditarios) en el código genético, llevan a un equilibrio entre el individuo y su ambiente. Esta sección cubrirá tres de los principales operadores de cambio como lo son la selección, el cruce y la mutación.

3.3.1. Selección

El proceso de selección se refiere a la escogencia de un elemento de entre un grupo o conjunto de elementos. Es también descrito como el *proceso evolutivo que determina qué tipos de organismos prevalecerán* [42]. Varios tipos de selección han sido descritas, entre ellas la selección natural, selección sexual, selección familiar, entre otras, y algunos modelos teóricos de sus principios han sido propuestos. La característica esencial de todos estos modelos de selección, es la escogencia del individuo más apto o mejor dotado según algún criterio o política previamente establecida. El perfil del banco genético de una población es en gran parte moldeado por este operador puesto que es a través de él que las formas genéticas “exitosas” serán tomadas en cuenta para contribuir en la composición genotípica de las siguientes generaciones.

3.3.2. Cruce

El cruce es el operador evolutivo donde el material genético de dos individuos es combinado en la creación de un nuevo organismo. Ocurre cuando *dos cromosomas, normalmente dos instancias homólogas del mismo cromosoma, se rompen y luego reconectan pero a una porción diferente* [9]. Como se mencionó en la sección 3.1, la ploidie refiere al número de juegos de cromosomas en una célula. Si posee un solo juego es un haploide, y si posee dos juegos es diploide. Los humanos como todo mamífero, somos organismos diploides pero nuestros gametos son haploides gracias a la meiosis en la gametogénesis. En el cruce, los cromosomas se “rompen” en un punto llamado *quiasma* y si esta ruptura ocurre en el mismo locus en la secuencia de pares base, se produce un intercambio de genes. Este intercambio hace posible que el código genético de los padres sea combinado en la cría en una relación 50/50

3.3. OPERADORES DE EVOLUCIÓN GENÉTICA

(mitad padre, mitad madre), y garantiza la diversidad genética de la población debido a que diferentes genotipos son combinados en uno solo. Esto igualmente puede ayudar a “desaparecer” algún rasgo no ventajoso que haya aparecido por causa de una mutación (en el caso que la cría herede la versión no mutada del otro padre, del mismo gen).

El cruce ocurre en la primera fase meiótica, donde una parte de un **cromátido**, es decir, un tira del cromosoma replicado, se intercambia físicamente con otro en el quiasma formando entonces un cromosoma con nuevas combinaciones de alelo (ver Figura 3.3). Este es un proceso complejo donde múltiples elementos llevan a cabo meticulosas tareas, por ejemplo, en la pro-fase I de la meiosis, el apareado de cromosomas homólogos ocurre en un *proceso aún por descubrir* [50], sin embargo, para el contexto de este trabajo es suficiente entenderlo como el proceso que hace posible la creación de nuevos individuos mediante la combinación del material genético de los padres.

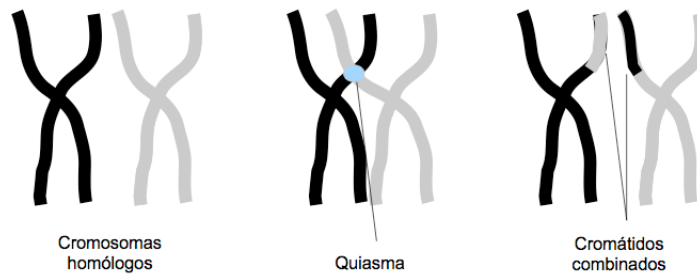


Figura 3.3: Representación del proceso de cruce. Dos cromosomas homólogos producen otro con nuevas asociaciones de genes y alelo

3.3.3. Mutación

La mutación es el oxígeno de la evolución, es el proceso encargado de introducir nuevos elementos a las existentes fórmulas genéticas. A finales del 1800 y principios del 1900, Hugo de Vries uno de los descubridores de los (hasta entonces desconocidos) trabajos de Mendel, notó algunos cambios abruptos, súbitos y ocasionales en los patrones de herencia de algunas plantas. Estos cambios los catalogó como una

3.3. OPERADORES DE EVOLUCIÓN GENÉTICA

“gran fuerza” de la evolución y les acuñó el término de “mutaciones” [8] en la edición alemana de su trabajo “Teoría de la Mutación”.

Una mutación es básicamente un cambio heredable en la secuencia del *nucleótido* (compuesto orgánico constituido por una base nitrogenada, un azúcar y ácido fosfórico) en el ADN o en el número, forma o estructura del cromosoma [18, 56]. Pueden ser clasificadas en muchos tipos de acuerdo a la naturaleza del cambio o ruido introducido. Por ejemplo, una mutación que cambia un solo nucleótido es conocida como una *mutación puntual*, mientras que si la cantidad de ADN o el número de genes es cambiado se clasifican como *duplicaciones* o *poliploidización* [56]. A continuación se describen algunos de los tipos de mutaciones más conocidas:

Puntuales: son substituciones de una base en el código del ADN. Se cree que son causadas por errores aleatorios en la síntesis del ADN o en la reparación de sitios dañados por agentes químicos, alta radiación, etc. El ADN o ácido desoxirribonucleico está compuesto por cuatro bases: Citosina (C), Timina (T), Adenina (A) y Guanina (G). Estas bases forman 2 grupos, las *Pirimidinas* y las *Purinas*. Por razones específicas, solo se forman lazos cuando A y T o cuando G y C se alinean en lados opuestos de la “cinta” en el modelo de doble hélice del ADN de Watson y Crick.

Transiciones: Cuando la *polimerasa* (enzima que cataliza la formación de algunos biopolímeros, como los ácidos nucleicos) del ADN substituye un elemento del mismo grupo (pirimidina-pirimidina o purina-purina) ocurre una mutación puntual llamada *transición*. Este tipo de mutaciones puntuales no son muy corruptivas, es decir, usualmente no tienen efecto en la adaptabilidad del individuo.

Transversiones: Cuando el cambio de la polimerasa envuelve grupos cruzados (una purina es substituida por una pirimidina o viceversa), se dice que la mutación puntual es de tipo *transversión*. Estos cambios son un poco más corruptivos, es decir, el ruido que introducen tiene efectos en la adaptabilidad del individuo [18, 56].

Otras: Un cambio en un aminoácido ocasiona otro tipo de mutaciones. Estas son llamadas *substituciones de reemplazo* o si no hay cambio aparente *substituciones silenciosas* [18].

3.3. OPERADORES DE EVOLUCIÓN GENÉTICA

Poliploidización: Es una mutación donde el número de genes es cambiado. Se cree que este tipo de mutaciones son las responsables por los cambios en el *cariotipo* (características cromosómicas de la célula) entre las especies.

Duplicaciones: Estas mutaciones aumentan el total de ADN en el genoma. Es muy probable que la evolución de algunas nuevas funcionalidades hayan sido favorecidas por este tipo de mutación [56].

En las mutaciones puntuales, las transiciones o los cambios no tan corruptivos, ocurren al menos con el doble de la frecuencia que las transversiones (las más corruptivas). Esto es posiblemente porque el ruido en la hélice del ADN causado por una transición es menos probable que sea reconocido como un error y por tanto menos probable de ser inmediatamente corregido.

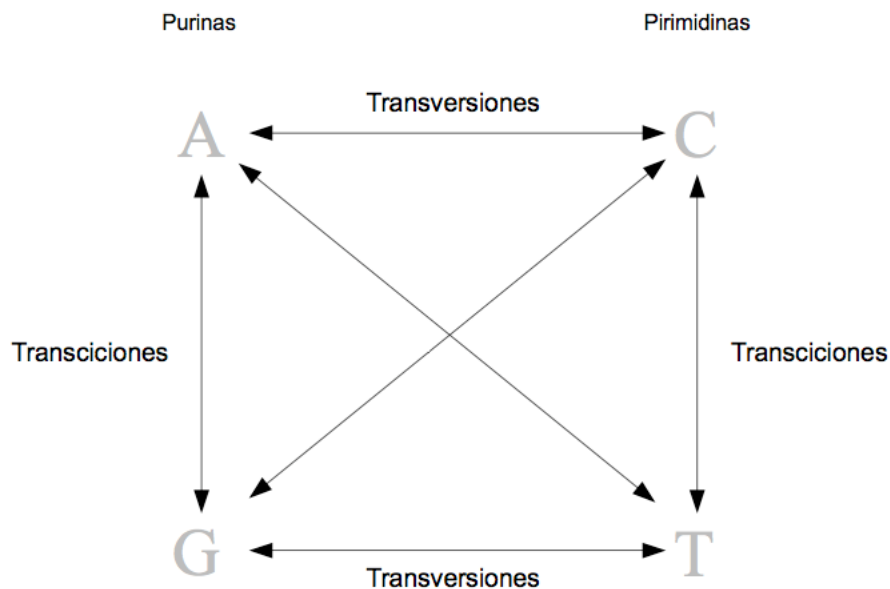


Figura 3.4: Mutaciones puntuales en la formación de pares complementarios. Substituciones en el mismo grupo son transiciones, en grupos cruzados son transversiones. Diagrama basado en un cuadro de [18], página 80

Si la vida te da un limón, haz limonada.

Anónimo

4

Búsqueda y Optimización

El manejo, gestión y administración de recursos para la resolución de problemas son actividades fundamentales no solamente de la ingeniería sino de cualquier área del saber. En la cita de apertura de este capítulo se lee una sutil referencia a este hecho y al aprovechamiento eficaz de los recursos disponibles. Las técnicas, métodos y/o algoritmos para la resolución de problemas, son generalmente llamados técnicas o métodos de búsqueda (con algún descriptor de la naturaleza de cómo se busca como por ejemplo métodos de búsqueda estocástica), pero ¿Búsqueda de qué?, ¿En dónde? y ¿Cómo hacerlo? Estas preguntas con sus diferentes enfoques son parte fundamental del quehacer académico puesto que son intrínsecas al proceso de resolución de problemas, que en definitiva, es la esencia misma del descubrimiento y generación de conocimiento.

Este capítulo brinda una muy breve introducción al campo de la optimización e intenta definir, aclarar y describir algunos conceptos básicos muy comunes en el argot de los Algoritmos Genéticos y de la Inteligencia Artificial en general. Se espera que el lector que comienza su incursión en alguno de estos campos, posea una mejor base para evitar el ruido producto de mal interpretaciones y/o el mal uso de algunos términos en la literatura especializada disponible.

4.1 Búsqueda

El verbo *buscar* es definido como *hacer algo para hallar a alguien o algo*¹, y en efecto, un algoritmo que resuelve un problema de modo general no hace más que *buscar* una solución a ese problema dentro de un universo de soluciones posibles.

En [49], los autores definen una *meta y el conjunto de recursos para alcanzarla* como el *problema*, y al proceso de explorar lo que esos recursos pueden hacer le denominan *búsqueda*. Gran diversidad de técnicas de búsqueda, de diferentes naturaleza, han sido propuestas (ver Figura 4.1) y cuando esa búsqueda está orientada a encontrar la mejor solución dentro de un grupo de soluciones factibles, entonces estamos en presencia de un método de *optimización*.

4.1.1. Espacios de búsqueda

El espacio de búsqueda es como su nombre lo indica, el área (física o virtual) a ser explorada con el objetivo de conseguir una instancia (o conjunto de instancias) que satisfagan una meta propuesta (el problema). Este espacio está directamente relacionado con el *dominio* del problema y se le suele llamar de diferentes formas según diferentes escenarios que condicionan la búsqueda.

Espacio solución

En un problema donde no existan restricciones de ningún tipo sobre la búsqueda, al dominio se le llama el *espacio de soluciones posibles*, o simplemente el *espacio solución*.

Espacio factible

Si una o más restricciones sobre el problema afectan posibles soluciones haciéndolas no válidas, es decir, transformándolas en soluciones no factibles, entonces el espacio solución se divide en soluciones factibles y soluciones no factibles. Al conjunto de las soluciones que no entran en conflicto con las restricciones del problema se le llama el *espacio de soluciones factibles*.

¹Diccionario de la Real Academia de la Lengua Española

4.1. BÚSQUEDA

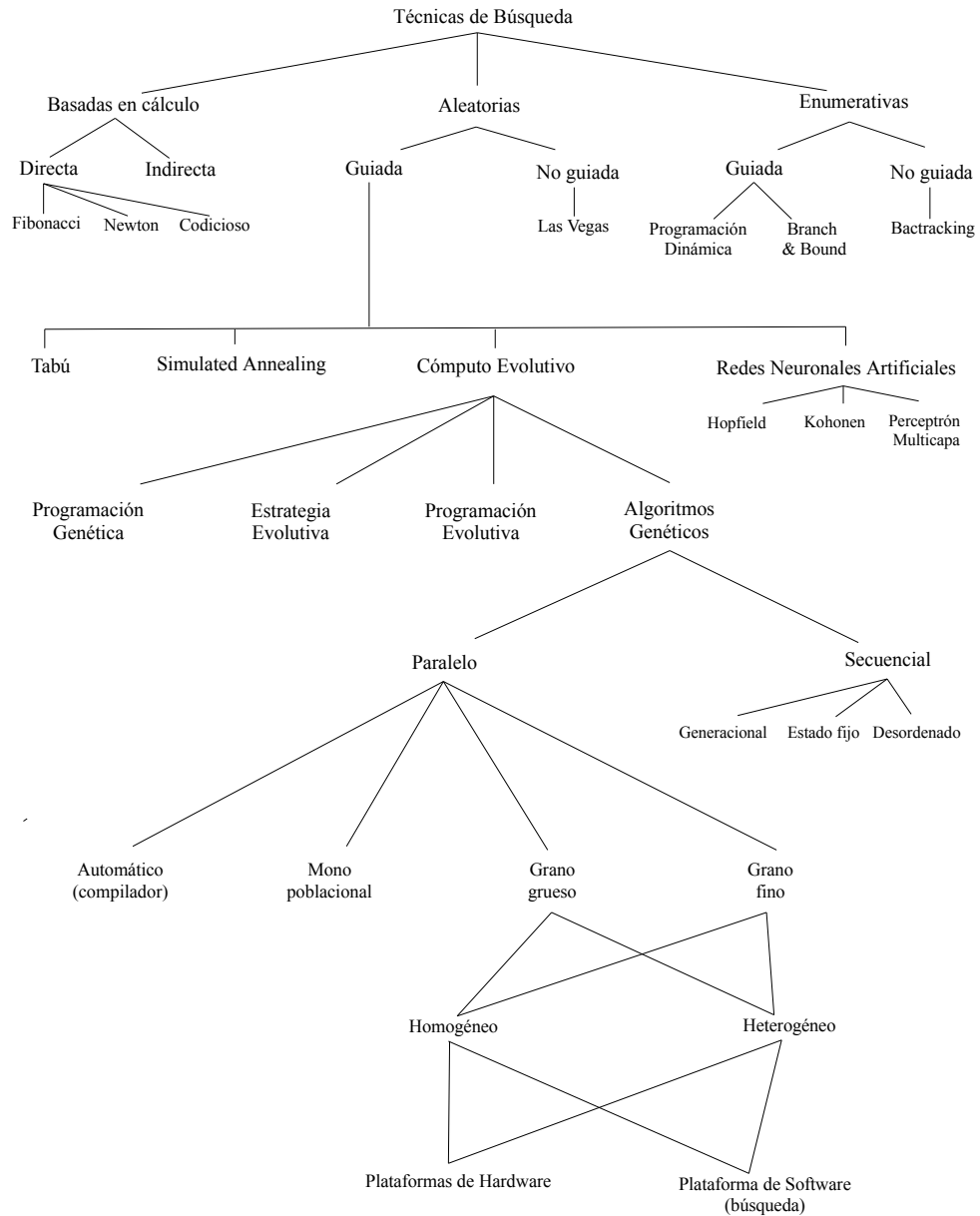


Figura 4.1: Esquema general de algunas de las técnicas de búsqueda mas comunes. El diagrama está basado en una descripción de [54], pág. vi

4.2 Heurística

La palabra *heurística* viene del griego *heuristicos* que significa “encontrar” o “descubrir”. Otras definiciones la describen como lo *concerniente a técnicas basadas en la experiencia para la solución de problemas, aprendizaje y descubrimiento* [57]. Este término es comúnmente confundido con “regla”, o “algoritmo” pero su concepto es un poco mas profundo.

Según [49], el significado técnico de la palabra “heurística” ha ido evolucionando desde que Arquímedes corrió desnudo por las calles de Siracusa gritando *eureka!*, *eureka!* (lo encontré!, lo encontré!) refiriéndose al principio de flotación que descubrió mientras tomaba su baño.

Los citados autores comentan que en 1957 el término fue usado como el estudio de métodos para el descubrimiento e invento de técnicas de resolución de problemas, en particular para el problema de demostraciones matemáticas. Igualmente mencionan que en 1963 el término fue usado como lo opuesto a algoritmo; un proceso que puede resolver un problema específico pero que no ofrece garantía alguna de hacerlo. Finalmente, señalan que en la actualidad la palabra heurística es usada mas comúnmente como un adjetivo para referirse a una técnica que mejora el caso promedio de rendimiento de una tarea de resolución de problemas pero que no necesariamente mejora el rendimiento del peor caso.

En resumen, el término heurística es usado para referirse a métodos que aceleran el proceso de búsqueda mediante la aplicación de reglas básicas (rules of thumb), estimaciones orientadas, juicios con fundamento o sentido común [57].

4.2.1. Meta-heurística

Una meta-heurística es una aplicación de la heurística con un enfoque mas genérico, esto es, la meta-heurística es la aplicación de uno o mas métodos de heurística para la resolución de una familia de problemas; para resolver una clase general de problemas computacionales (en el contexto de ciencias de la computación) combinando varios procesos de *caja negra*. Los métodos de optimización modernos de la Inteligencia Artificial son en su mayoría métodos de meta-heurística, éstos combinan varias heurísticas diferentes en un modelo de búsqueda general.

4.3 Optimización

En este capítulo he separado los términos de búsqueda y optimización para efectos didácticos, aunque como se menciona en la sección 4.1, el proceso de optimización es un proceso de búsqueda del mejor de una serie de elementos según algún criterio, es decir, es la búsqueda de la mejor solución posible a un problema dado.

Matemáticamente, el concepto de optimización puede ser expresado de la siguiente manera: Dada una función sobre un conjunto X (en este ejemplo, de números reales) $f : X \rightarrow \mathbb{R}$, se pretende buscar un elemento $x^* \in X$ tal que:

$$f(x^*) \leq f(x) \quad \forall x \in X$$

en este caso x^* es un elemento mínimo (un punto óptimo en un problema de minimización). Igualmente, si

$$f(x^*) \geq f(x) \quad \forall x \in X$$

entonces x^* es un elemento máximo (un punto óptimo en un problema de maximización).

Para profundizar un poco más, considere y como una función de una variable x , es decir, $y = f(x)$, si la primera derivada de esta función en un punto x^* se hace 0, este punto es un óptimo ó un *punto de inflexión*, numéricamente, si $f'(x^*) = 0$, entonces x^* es un máximo, un mínimo o un punto de inflexión. Determinando la primera derivada de mayor orden diferente a 0, podemos concluir si x^* es un óptimo o un punto de inflexión [44].

4.3.1. Optimización combinatoria

En el campo de las matemáticas aplicadas y ciencias de la computación, generalmente se le llama optimización combinatoria al proceso de encontrar una solución óptima de entre un conjunto finito de soluciones posibles. Siendo el espacio solución finito, es razonable pensar que una búsqueda exhaustiva llevará a encontrar la solución óptima con una inversión de energía “razonable” (en términos de costos

computacionales y tiempo de búsqueda). No obstante, en muchos de estos problemas el espacio solución aunque finito, es para efectos prácticos, infinito, pues la inversión de energía para navegarlo de manera exhaustiva sobrepasa la capacidad real de hacerlo. Tome por ejemplo el conocido caso del Vendedor Viajero o TSP como lo encontrará con frecuencia en la literatura (del inglés *Traveller Salesman Problem*). Este problema consiste en realizar un ciclo Hamiltoniano sobre un grafo fuertemente conexo buscando minimizar la distancia recorrida o costo total incurrido, representado por el peso de la arista de enlace entre los nodos.

Para una configuración menor a 12 nodos, el costo computacional de exploración de todas las alternativas posibles para recorrer el Hamiltoniano, puede ser tomado por un computador moderno, no obstante, por cada nodo que se le agregue al circuito, las posibilidades de rutas aumentan de manera exponencial y por ejemplo, para calcular y comparar todas las posibles rutas para un grafo de tan solo 20 nodos, a la capacidad de cómputo moderno necesitaríamos mas años de los que la tierra lleva en existencia según el registro fósil.

4.3.2. Optimización Multiobjetivo

En un problema de optimización simple, se busca uno o varios óptimos (locales o globales) a una única función objetivo. La diferencia con un problema de optimización multiobjetivo es que ésta considera simultáneamente varias funciones objetivo a la vez, pudiendo estas tener conflictos entre sí. Formalmente, el problema multiobjetivo se puede describir como [13]:

$$\begin{array}{ll}
 \text{Maximizar/Minimizar} & f_m(X), \quad m = 1, 2, \dots, M \\
 \text{sujeto a} & g_j(X) \geq 0 \quad j = 1, 2, \dots, J \\
 & h_k(X) = 0 \quad k = 1, 2, \dots, K \\
 & x_i^{(I)} \leq x_i \leq x_i^{(S)} \quad i = 1, 2, \dots, n
 \end{array}$$

donde una solución X es un vector de n variables de decisión acotadas por lo límites inferior $x_i^{(I)}$ y superior $x_i^{(S)}$.

Una particularidad de los problemas de optimización multiobjetivo, es que debido a que generalmente las funciones objetivo tienen conflictos entre sí, la solución no es un punto único en el espacio solución sino un conjunto de soluciones que representan una especie de “equilibrio” de solución, donde ninguna es mejor que otra para todos las funciones objetivo del problema. A este conjunto de soluciones se le llama conjunto óptimo de Pareto y el diseñador del algoritmo multiobjetivo debe escoger desde su criterio, qué solución tomar, pues desde el punto de vista teórico, todas son igualmente buenas, es decir, todas son *no dominadas*.

Las soluciones del conjunto óptimo de Pareto, forman en el espacio de funciones objetivo lo que se llama el *frente Pareto*. Las métricas de estimación de rendimiento para algoritmos que resuelven problemas de optimización multiobjetivo, generalmente miden características de este frente, como proporción de individuos en él, separación entre ellos, distancia a este frente, dispersión máxima de los individuos en el frente, entre otras. Debido a que el diseñador en la mayoría de los casos debe escoger una solución de entre varias posibles, en la medida que el algoritmo proporcione más soluciones no dominadas, representando una mayor y mejor muestra del frente Pareto, será considerado de mejor rendimiento.

Tipos de problemas en optimización

Diferentes clasificaciones han sido dadas a los problemas de optimización dependiendo de su naturaleza, complejidad, etc. Entre las diferentes categorías podemos encontrar problemas de optimización lineal, no lineal, sin restricciones, con restricciones, de programación entera, programación de valores reales y otros. La Figura 4.2 muestra una copia basada en un esquema general en [44] con algunos de los métodos convencionales de optimización.

4.3. OPTIMIZACIÓN

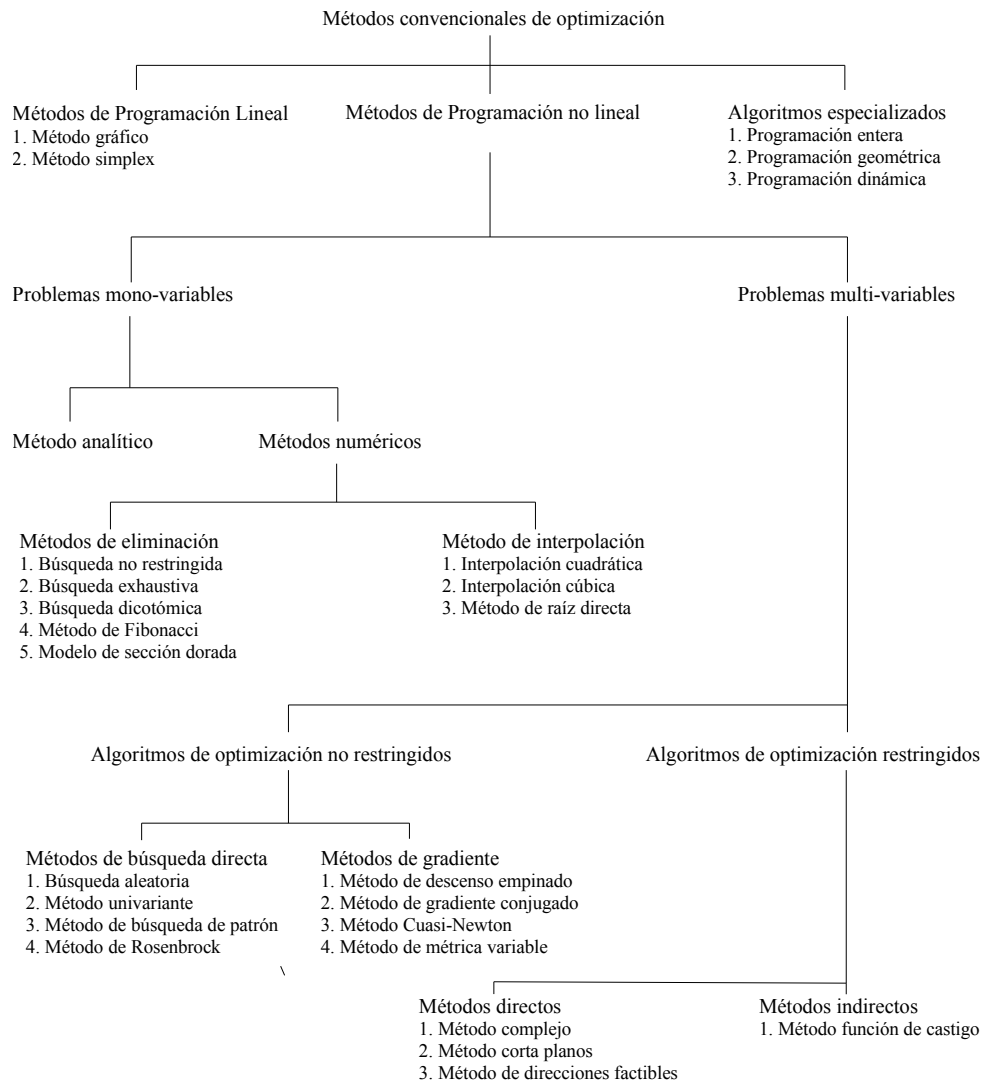


Figura 4.2: Esquema general de algunos de los métodos de optimización mas usados. El diagrama está basado en una descripción de [44], pág. 15

Parte II

Modelos Seminuméricos

*DILBERT: ¿Esta seguro que eso es aleatorio?
CONTADOR: Ese es el problema con la aleatoriedad;
nunca se puede estar seguro!*

Scott Adams

5

Números Aleatorios

5.1 Introducción

¿El número 2 es aleatorio? Con esta jocosa pregunta Donald Knuth comenta de números aleatorios en uno de sus trabajos sobre algoritmos seminuméricos [31]. El comentario trata el punto que cuando hablamos de números aleatorios estamos de hecho hablando de secuencias numéricas y no de un elemento aislado, es decir, la aleatoriedad no está en el número en sí, sino en el conjunto de números observados. Esto pareciera ser lógico, obvio, y auto-explicativo, sin necesidad alguna de aclaratoria, después de todo - (¿Cuán aleatorio es el 2?) -, no obstante, otro conocido autor en el área de programación numérica relata el incidente de uno de sus colegas trabajando con un conocido generador de números aleatorios de IBM y llamando al consultor del centro de programación después de observar un comportamiento extraño en las secuencias obtenidas. Según el autor, la respuesta del consultor fue que había hecho mal uso de la herramienta:

*“Nosotros le garantizamos que cada número es aleatorio individualmente,
pero no garantizamos que mas de uno lo sea. . .”*

Esta respuesta fue considerada por el autor en mención para expresar la misma idea de la pregunta jocosa de Knuth (ver[15] página 281).

La *aleatoriedad* es un controversial punto que yace en los fundamentos teóricos de los algoritmos genéticos así como en el de cualquier otra máquina estocástica. Conocer los principios elementales de un dispositivo generador de secuencias *aleatorias* es de vital importancia para el sustento de un buen análisis del trabajo experimental y para la apropiada formulación, argumentación y discusión de las conclusiones derivadas. Un algoritmo genético basado en secuencias aleatorias con una marcada *aberración lineal*, o una fuerte correlación intrínseca entre secuencias de dígitos, de alta predictibilidad, o de naturaleza no uniforme, no posee los requisitos mínimos para hacer un análisis teórico bien sustentado. Una estimación citada en [15] (véase página 280) sugiere que si todos los artículos científicos cuyos resultados son dudosos (por causa de un “mal” generador de números aleatorios) desaparecieran de las bibliotecas especializadas, quedaría un espacio en cada estante de aproximadamente el tamaño de un puño puesto en horizontal. Si esta estimación se acerca a la realidad entonces habla por sí sola de la importancia de los generadores de números aleatorios, y aún si no fuere el caso, el sentido común elemental sugiere que una edificación se sustenta en sus bases, de la misma manera que una máquina estocástica se sustenta en gran parte de los generadores aleatorios a su disposición.

Estos generadores de números aleatorios, o generadores de números *pseudo aleatorios* como generalmente se les denomina entre filósofos, físicos, matemáticos, computistas entre otros (y que mas adelante se hará claro el por qué), merecen entonces una especial atención en la elaboración de un compendio de aperos de algoritmos genéticos, y en especial de elementos de programación seminumérica. Desde su concepto hasta la discusión de criterios de evaluación y calidad, este capítulo discute los elementos principales de una familia de tales generadores; su naturaleza, su mecánica, sus componentes, sus limitaciones y fortalezas con el objetivo de brindar al investigador las herramientas necesarias para el análisis crítico de estos agentes, de manera que pueda proponer y desarrollar modelos experimentales en algoritmos genéticos y/u otras máquinas estocásticas, mas robustos y mejor preparados para la derivación de conclusiones y elementos de discusión.

Para ciertas aplicaciones, la calidad de un generador aleatorio puede no ser de mucha importancia, no obstante, para otro tipo de sistemas como el software de seguridad, la calidad es vital. Similarmente, para el análisis teórico de máquinas estocásticas por ejemplo, un buen generador de números aleatorios es una parte esencial para el desarrollo de conclusiones bien fundamentadas.

5.2 Aleatoriedad

En la sección 5.1 me referí a la aleatoriedad como un punto controversial, debido a la dificultad de adaptar el significado de lo que generalmente entendemos por aleatorio al proceso de generar series de números al azar en un computador.

El concepto de *aleatoriedad* es, desde el punto de vista conceptual, una contradicción a la naturaleza determinista de las computadoras. Literalmente, aleatoriedad implica *dependencia de algún suceso fortuito*¹, es decir, sin orden o planeamiento; al azar. Pero ¿Cómo puede la máquina mas determinista creada por el hombre generar una serie de números que a nuestro modo de ver sean de naturaleza aleatoria? Esto plantea una situación de conflicto, pues si la máquina opera bajo reglas deterministas, entonces cualquier programa usado generará una salida que es totalmente predecible, y esto, al menos en términos de lo que generalmente entendemos como *aleatorio*, es una contradicción, es decir, ¿Si sé qué número seguirá en la serie, cómo puede entonces ser esto un evento aleatorio? Mas aún, considere la paradoja de predictibilidad planteada por Lagarias: ([33] citado en [29]).

Si una función determinista es impredecible, entonces será difícil probar cualquier cosa sobre ella; en particular, será difícil probar que es impredecible.

Esta incompatibilidad de conceptos; por un lado la aleatoriedad demandando cierta *impredecibilidad* y por el otro, el computador siendo incapaz de generar rutinas no deterministas, hacen que la definición de estos generadores sea algo peculiar, y mas aún, plantea un escenario difícil a la hora de proponer criterios absolutos para su evaluación.

No obstante, hoy día la mayoría de las herramientas de programación contienen utilitarios para la generación de series *pseudo aleatorias*, y mas aún, se han creado dispositivos físicos llamados *Generadores Verdaderamente Aleatorios* o TRNG por sus siglas en inglés (True Random Number Generators), que generan secuencias numéricas basándose en eventos físicos de naturaleza aleatoria como por ejemplo el ruido atmosférico o el tiempo de decaimiento de una fuente radioactiva, reduciendo así el grado de predictibilidad en la secuencia a niveles prácticamente nulos. En los sitios [47, 17] se pueden obtener en línea, secuencias numéricas verdaderamente aleatorias basadas en estos dos métodos respectivamente. En la experimentación con

¹Diccionario de la Real Academia Española.

máquinas estocásticas, la calidad con la que estos generadores son programados o desarrollados dictaminará en gran medida la calidad de los argumentos que sustentan las conclusiones de trabajos empíricos.

5.2.1. Desviadas Uniformes

Las desviadas uniformes son números aleatorios en un rango (generalmente entre 0 y 1), distribuidos de forma que todos tienen la misma probabilidad de selección. Lo que generalmente conocemos como generadores de números aleatorios en las utilidades de los sistemas operativos o herramientas de programación, son usualmente desviadas uniformes. Tome cualquiera de estos utilitarios y al generar varios números de forma consecutiva, observará que cubren el rango de selección de manera pseudo-uniforme, es decir, que cualquier número tiene la misma probabilidad de ser seleccionado que otro.

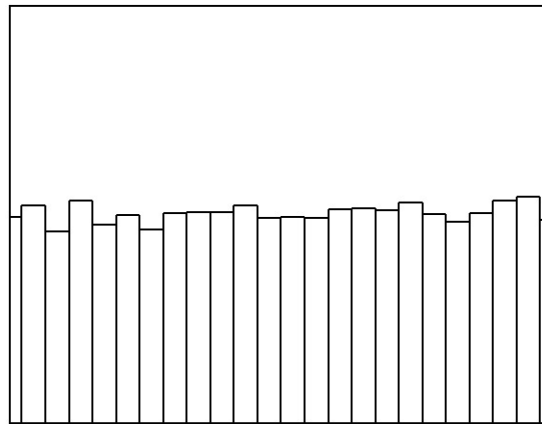


Figura 5.1: Distribución de frecuencia de una desviada uniforme)

La figura 5.1 ilustra la frecuencia de una secuencia hipotética de números aleatorios con una distribución uniforme. Observe que independientemente de los valores de la distribución, la frecuencia (eje vertical) es relativamente similar para todos los números. Este es el caso más común pero no el único. Algunos generadores de números pseudo-aleatorios se basan en distribuciones diferentes a la uniforme. Un ejemplo

son las secuencias aleatorias obtenidas por alguna distribución como la Gaussiana con media μ y desviación estándar σ específicas. La figura 5.2 ilustra la forma de una distribución de Gauss con media 0 y desviación unitaria $G(1,0)$.

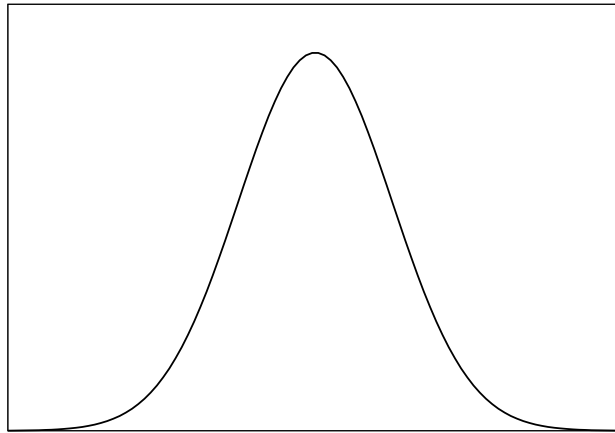


Figura 5.2: Forma de una distribución de Gauss; $G(\mu, \sigma)$

Hoy en día, los principales sistemas operativos del mercado como Unix, Windows, GNU/Linux, Mac OS X y otros, ofrecen un conjunto de utilitarios básicos para la generación de secuencias pseudo-aleatorias. Estos generadores son en la mayoría de los casos (al menos en los sistemas operativos mencionados), implementados a través de métodos de congruencia lineal donde un valor inicial llamado *semilla*, es la base para generar la secuencia siguiente.

Las desviadas uniformes no son la única fuente de generación de secuencias pseudo-aleatorias. Como se mencionó anteriormente, otras distribuciones pueden ser usadas, como la de Gauss, la de Poisson, la binomial, etc., no obstante, estas son generalmente implementadas realizando ciertas operaciones sobre una o mas desviadas uniformes (ver [15]), y por su uso, utilidad, y naturaleza de aplicación, son menos convencionales y por tanto se ofrecen en sistemas o soluciones de software mas especializadas.

Generadores de congruencia lineal

Las desviadas uniformes convencionales, se implementan usando métodos de congruencia lineal. En el mas común, una semilla o valor inicial es argumento de arranque para la generación de un valor siguiente, y este nuevo valor es el arranque para uno siguiente y así se genera una secuencia cíclica de valores desde uno inicial.

$$I_{j+1} = (a \cdot I_j + c) \text{ módulo } m \quad (5.1)$$

En la ecuación 5.1, el valor inicial I_j es la base para el cálculo del siguiente entero I_{j+1} mediante la multiplicación de un entero positivo a llamado el *multiplicador*, sumado a otro entero positivo c llamado el *incremento*, y tomando el resto de la división (módulo) entre un entero positivo m llamado el *módulo*.

La secuencia obtenida de la ecuación 5.1 es cíclica y si los parámetros a , c y m son escogidos acertadamente, no solo habrán ocurrido todos los valores en el rango especificado con aproximadamente la misma frecuencia, sino que el ciclo puede ser de máximo período, es decir, de tamaño m . El siguiente cuadro muestra un fragmento del código de implementación.

Generador de congruencia lineal (tomado de [15])

```
unsigned long proximo = 1;

unsigned int rand(void) //Generador pseudo-aleatorio
{
    proximo = proximo * 987654321 + 12345;
    return (unsigned int)(proximo/65536) % 32768;
}

void srand(unsigned int semilla)
{
    proximo = semilla;
}
```

5.2. ALEATORIEDAD

El generador descrito arriba tiene como parámetros $a = 987654321$, $c = 12345$ y $m = 32768$. Del código de Lenguaje C se desprende que el rango de la función $rand()$ está entre $[0..32767]$, además si la semilla es 0, el algoritmo no puede generar secuencias, es decir, todos los valores que arroja son iguales a cero.

En [40], los autores luego de evaluar varios generadores de números aleatorios proponen uno de la forma:

$$I_{j+1} = (a \cdot I_j) \text{ módulo } m \quad (5.2)$$

y además sugieren que con los valores de $a = 7^5 = 16807$ y $m = 2^{31} - 1 = 2147483647$ los resultados obtenidos con su Generador Estándar Mínimo son al menos iguales al de cualquier otro con $c \neq 0$ [15]. La implementación del GEM se ilustra abajo, note que el algoritmo devuelve un número real entre (0.0 y 1.0). El XOR (ó exclusivo) con la MASCARA es para evitar que el algoritmo caiga en ciclos infinitos y genere el valor de 0 todo el tiempo.

Generador Estándar Mínimo de [40]

```
double GEM( int idum)
{
  const int IA=16807, IM=2147483647, IQ=127773;
  const int IR=2836, MASCARA=123459876;
  const double AM = 1.0/(double)(IM);
  int k;
  double ans;

  idum ^= MASCARA;           //XOR con MASCARA
  k = idum/IQ;               //permite semilla 0
  idum = IA*(idum-k*IQ)-IR*k;
  if (idum < 0) idum += IM;
  ans = AM*idum;
  idum ^= MASCARA;         //queda como estaba
  return ans;
}
```

Esta implementación lleva cálculos adicionales puesto que las operaciones de multiplicación con los parámetros a y m desbordan la capacidad de las variables enteras. El algoritmo usa una aproximación de factorización de m propuesta por [53] para descomponer las operaciones que puedan causar saturación al ancho de banda de las variables enteras y hacer mas “portátil” el algoritmo.

Randú

Este generador de números aleatorios de IBM usaba los parámetros $a = 65539$ y $m = 2^{31}$. Esta selección de valores para los parámetros multiplicador y módulo respectivamente, hizo muy famoso al algoritmo por la terrible correlación lineal que presentaba. La sección 5.2.2 describe en mas detalle esta falla del generador y presenta un gráfico que el lector puede fácilmente reproducir en cualquier utilidad para realizar gráficos científicos como el gnuplot.

5.2.2. Evaluación de Generadores Aleatorios

En el campo de la inteligencia artificial, los debates filosóficos sobre el concepto de inteligencia, los medios para medirla o estimarla, y mas controversial aún, los principios para determinarla o rechazarla son aún abundantes. Cuando la supercomputadora Deep Blue venció dos veces a Gary Kasparov en ajedrez, y en otras tres oportunidades el gran maestro no pudo ganarle teniendo que aceptar “tablas”, se planteó la pregunta ¿Deep Blue es inteligente?. Muchos se respaldaban en la *prueba de Turing* para evitar el difícil escrutinio de los principios a medir. La prueba propuesta por Alan Turing en 1950 pone a un juez humano en comunicación directa con 2 agentes a quienes no puede ver ni ellos se pueden ver entre sí. Uno de estos agentes es un humano y el otro una máquina. Si el juez o árbitro no es capaz de determinar confiablemente cuál agente es la máquina y cuál el humano, entonces la máquina pasa la prueba de inteligencia. Esta prueba le deja la subjetiva y compleja tarea de abstracción de lo que es ser “inteligente” al mismo ente que en principio dudó de la inteligencia de la máquina; el humano, es decir, ocurre una especie de vicio o aberración cíclica filosófica.

En conversaciones informales con académicos de las principales universidades del país, les preguntaba ¿Cómo evaluaría a un generador de números aleatorios para saber si es bueno o malo? La respuesta más común era que si el generador logra-

ba “engañarlo” en cuanto a las secuencias que generaba, entonces era bueno. Este enfoque esta basado muy posiblemente en la prueba de inteligencia de Turing, no obstante, para la ingeniería y las ciencias computacionales en general, este criterio de evaluación es en el mejor de los casos, incompleto.

En las ciencias computacionales la palabra *random*, tiene un significado mas elaborado. Por ejemplo la RAM (Random Access Memory) o memoria de acceso aleatorio de un computador, permite que se acceda a valores en todo su registro en cualquier momento, sin restricciones de acceso, al igual que en una desviada uniforme todos los números son igualmente probables de ser seleccionados. Por esta razón, para un computista un buen generador de números aleatorios se refiere principalmente a la capacidad general del algoritmo en cuanto a:

- Mantener la muestra (las secuencias que se vayan obteniendo en el rango determinado) a lo largo de todo su período, distribuida según la naturaleza del tipo de generador (uniforme, gauss, gamma, poisson, etc.).
- No presentar en la muestra patrones que permitan predecir el siguiente número de la secuencia (correlaciones).

El primer punto se refiere a la habilidad de acceder a todo el recurso de manera “sesgada” o “dirigida”, mas no predecible en el orden del muestreo como plantea el segundo punto (la capacidad del algoritmo de “engañar” al humano en cuanto a la aleatoriedad de la secuencia).

Con esta idea central en mente podemos discutir un paquete de evaluación para generadores de números aleatorios desde el enfoque computista. Si una secuencia aleatoria es lo suficientemente impredecible como para “engañar” a un humano en cuanto a los patrones o correlaciones que presenta, pero no realiza un muestreo del recurso o fuente de generación según la distribución esperada, entonces podemos clasificarla con un nivel de calidad inferior a otra que si lo cumpla. De igual manera, una secuencia aleatoria en perfecta armonía con la distribución que la rige pero que posee fuertes o marcados patrones que permiten que sean predecible los elementos futuros de la muestra, tampoco es deseable. En las desviadas uniformes de congruencia lineal, una debilidad común es la correlación en k -dimensiones, mas pronunciada a mayores planos generados [15] [45].

Existen muchas *suites* o paquetes estadísticos para evaluar generadores de números aleatorios, todos evalúan diferentes características del generador para medir sus debilidades y fortalezas frente a indicadores deseados o ideales. Ninguna prueba per se es determinante de manera absoluta, la estimación de la calidad del generador se hace en base a la consideración de varias pruebas en conjunto.

Para la comparación de algunas desviadas uniformes en este documento, se describen las siguientes pruebas de calidad:

Visual: Esta prueba se basa en la capacidad del cerebro humano de reconocer información visual y encontrar patrones de ruido o tergiversaciones de los datos. Consiste en graficar en el plano o espacio, puntos generados aleatoriamente y observar la distribución de los mismos. Esta prueba tiene el inconveniente que esta limitada representaciones en el plano o espacio (dimensiones ≤ 3) y que además la posición del observador puede “esconder” ciertos artificios en la distribución.

Entropía y máxima compresión: Los algoritmos de compresión de datos calculan la entropía de un archivo para estimar cuánto mas puede ser comprimido. Si el archivo presenta abundantes patrones o formas deterministas, entonces se pueden extraer estas formas y “empaquetar” el resto de los datos. Al desempaquetar, se reconstruye la información a partir de las formas archivadas. Por otra parte, si el algoritmo es poco predecible (muy aleatorio) entonces tendrá poca holgura para ser comprimido. En esta prueba se calcula la entropía de una secuencia de números aleatorios generados y la máxima compresión estimada. En esta prueba. una alta entropía y un bajo porcentaje de compresión posible son mejores.

Chi-cuadrado: Se compara la distribución de la secuencia de números aleatorios generada con una distribución verdaderamente aleatoria (ver sección 5.2), y se estima la probabilidad (presentada en %) que la secuencia difiera del valor estimado. Para esta prueba una probabilidad menor es mejor.

Media aritmética: Es el simple cálculo de la media aritmética de la secuencia de números para ser comparada con la media aritmética esperada según la distribución. Para N números generados dentro del rango $(lim_{inf} \dots lim_{sup})$, la media esperada de la desviada uniforme es $(lim_{sup} - lim_{inf})/2$.

Cálculo de π por Monte Carlo: Imagine un caso de tiro al blanco donde el blanco es un círculo de radio r dentro de un cuadrado de lado $2 \cdot r$, es decir, el círculo está circunscrito en el cuadrado. Los disparos se simulan mediante la generación de secuencias aleatorias interpretadas en pares (x_i, y_i) (coordenadas), de manera que los dos primeros números forman el primer par, los siguientes dos números el segundo par, etc. Al final de la sesión de disparos (fin de la secuencia aleatoria), los impactos dentro del círculo, o sea, aquellos puntos cuyas coordenadas $x_i^2 + y_i^2$ fueron menores a r^2 (por la ecuación $x^2 + y^2 = r^2$), se consideran aciertos, mientras que los impactos fuera del círculo son desaciertos. La relación entre aciertos y desaciertos se usa para el cálculo de π , es decir, dado que el área del cuadrado es $(2r)^2$ y el área del círculo es πr^2 , se calcula π como $(4 \cdot M)/N$ donde M es el número total de aciertos y N el número total de puntos de coordenadas, o aciertos + desaciertos. El valor esperado para π debe ser aproximado a 3,14159265.

Correlación lineal: Al igual que con el cálculo de π por Monte Carlo, se toman los números en la secuencia aleatoria como pares o puntos de coordenadas (x_i, y_i) . El coeficiente de correlación de Pearson r que estima la asociación entre números sucesivos, se calcula mediante la ecuación:

$$r = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2} \sqrt{\sum_i (y_i - \bar{y})^2}} \quad (5.3)$$

donde \bar{x} y \bar{y} son las medias aritméticas de los x_i 's y y_i 's respectivamente [15]. El coeficiente de correlación r toma valores entre -1 y 1, con -1 denotando total correlación negativa, es decir, la asociación entre los puntos describe una línea recta con pendiente negativa, donde a medida que y decrece, x crece. Si $r = 1$, la asociación es de total correlación positiva, es decir, los puntos están en una recta con inclinación positiva y tanto x como y crecen simultáneamente. Si no hay correlación entre los puntos, $r = 0$. Para esta prueba un valor más cercano a 0 es mejor.

Para aplicar estas pruebas se seleccionaron 3 algoritmos, todos basados en métodos de congruencia lineal. Estos fueron:

- rand16

- rand32
- randú

Donde la diferencia entre los 2 primeros radica en el ancho de banda (16 contra 32 bits) del generador, afectando así la amplitud del módulo o período de la secuencia, y el tercero; randú de IBM. Este último fue implementado de acuerdo a los parámetros reportados en [40] y [15]. Fue incluido para incluir elementos de discusión de las pruebas de secuencias aleatorias.

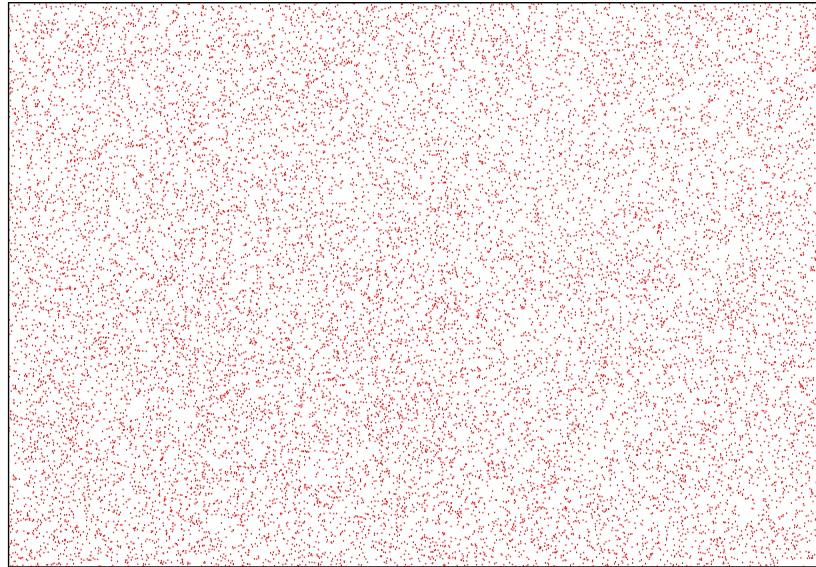


Figura 5.3: 10^4 puntos distribuidos al azar por ran16

La figura 5.3 muestra 10000 puntos distribuidos aleatoriamente en el plano 2D por la desviada uniforme ran16. No se observan patrones claros o determinantes en la figura, es decir, los puntos se aprecian “aceptablemente” bien distribuidos en el recuadro o al menos no se observan fuertes patrones de arreglos de los puntos en la

cuadrícula. Esta prueba visual solo muestra la capacidad de distribución de puntos de ran16 en el plano bidimensional. Para $k > 3$ dimensiones, esta prueba se hace más compleja o imposible de evaluar.

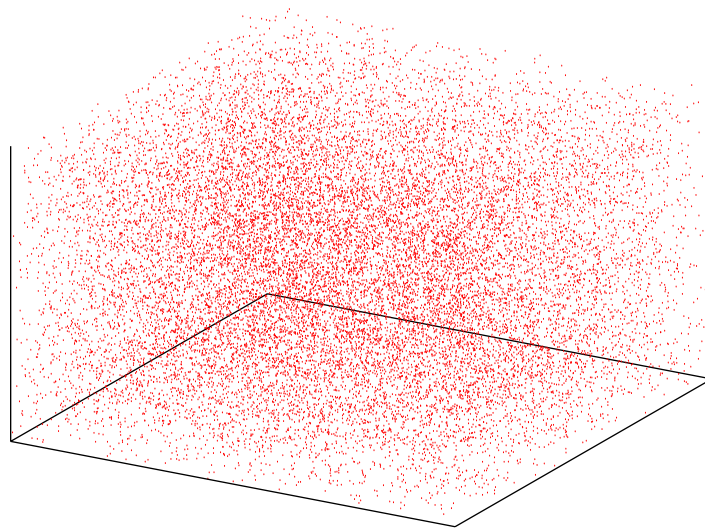


Figura 5.4: 10^4 puntos distribuidos al azar por ran16 en el espacio

Observe que los puntos de ran16 en el espacio de 3 dimensiones de la figura 5.4 se aprecian distribuidos de manera pseudo-uniforme. Hay pequeñas áreas sin cubrir a lo largo de las aristas del cubo (observe la distribución de los puntos en la arista alineada con el eje vertical de la gráfica). La mayor densidad de puntos observada hacia el centro de la figura se debe la posición del observador y no a la distribución real de éstos.

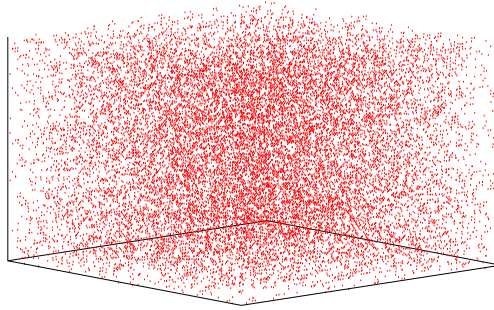


Figura 5.5: 10^4 puntos distribuidos al azar por randú en el espacio

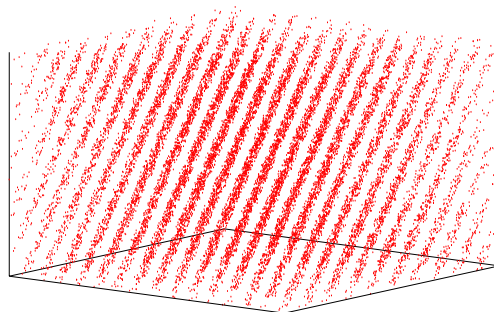


Figura 5.6: Randú. El punto de observación es desplazado

5.2. ALEATORIEDAD

Las figuras 5.5 y 5.6 muestran 10^4 puntos distribuidos en el espacio por el generador randú. En la figura inferior el punto de vista del observador es desplazado aproximadamente 15 grados con respecto al plano horizontal formado por los ejes x y y . Note como una distribución sin aparentes formas o patrones en la dispersión de puntos que se aprecia en la figura superior (5.5), se descubre a puntos agrupados formando planos paralelos al cambiar el punto de observación.

Un examen visual no es suficiente por sí solo para estimar la calidad de un generador. El cuadro 5.1 muestra los resultados de los generadores aleatorios en las pruebas estadísticas descritas.

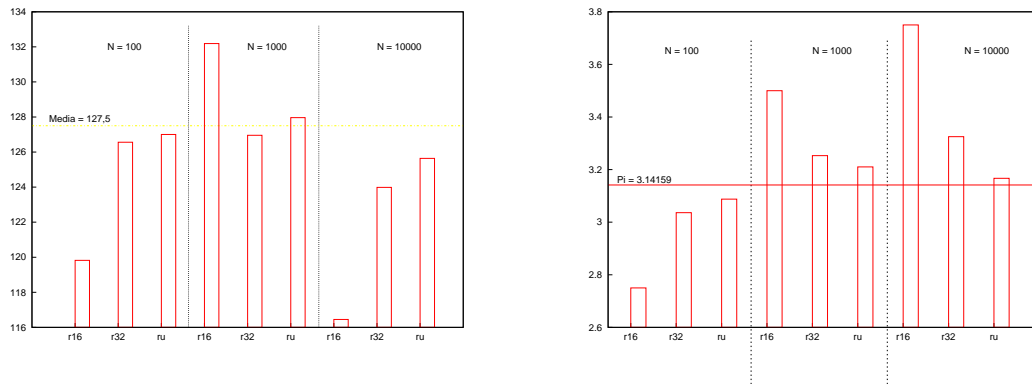


Figura 5.7: Media aritmética (izquierda) y valor de π por Monte Carlo

La figura 5.7 compara los resultados de la prueba de media aritmética y cálculo del valor de π por Monte Carlo para secuencias de 100, 1000 y 10000 números aleatorios generados por cada uno de los generadores rand16, rand32 y randú. Los resultados están tabulados en el cuadro 5.1 y se resaltan los mejores valores de cada experimento. Las secuencias generadas están entre 0 y 256, de manera que la media aritmética esperada para una secuencia de números sacados de una distribución uniforme es 127,5. El valor que mas se aproxima es el generado por rand32 con $N = 10^4$. Para el cálculo del valor de π mediante Monte Carlo, el generador con el valor mas cercano es randú para $N = 10^4$, sin embargo, observe que en la figura 5.6 se muestra una fuerte correlación en $n = 3$ dimensiones.

Cuadro 5.1: Cuadro comparativo de desviadas uniformes

	N	Entropía	%	χ^2	%	Media aritmética	Monte Carlo π	Correlación
ran16	10^2	6.356307	20	232.0	83.73	119.82	2.75	-0.068753
ran16	10^3	7.795738	2	268.22	27.25	126.56	3.0361	-0.001278
ran16	10^4	7.980218	0	270.46	24.17	127.0035	3.0876	-0.003503
ran32	10^2	6.213661	22	278.88	14.56	132.19	3.5	0.023764
ran32	10^3	7.775441	2	284.1	10.17	126.955	3.2530	0.036316
ran32	10^4	7.980375	0	271.08	23.36	127.9644	3.2101	-0.000487
randú	10^2	6.288758	21	263.52	34.36	116.45	3.75	0.07099
randú	10^3	7.801778	2	257.98	43.6	123.983	3.3253	0.054656
randú	10^4	7.981723	0	251.57	54.89	125.641	3.1668	0.014617

Valor de entropía esperado	mayor valor - menor %
Chi-cuadrado (χ^2) esperado	mayor valor - menor %
Media aritmética esperada	127.5
Valor de π esperado	3.14159265
Correlación esperado	0

5.2. ALEATORIEDAD

Finalmente, la evaluación de generadores aleatorios depende principalmente del tipo de aplicaciones al que se orientan. Para realizar análisis teóricos sobre algoritmos genéticos es necesario usar desviadas uniformes de un nivel “aceptable” de calidad e implementadas en estructuras de datos apropiadas para la aplicación. Imagine por ejemplo una simulación de un problema de física de partículas mediante Monte Carlo. Suponga que desea evaluar 10^6 puntos diferentes usando un generador aleatorio de 16 bits (módulo ≤ 32767 y por tanto el período o tamaño del ciclo del generador es a lo sumo esta cantidad), esto significa que evaluará cada punto aproximadamente 30 veces, en vez de evaluar 10^6 puntos diferentes. Las conclusiones producto de esa simulación no tendrán el fundamento adecuado. Por otro lado, muchas otras aplicaciones posiblemente no requieran de generadores aleatorios de alta calidad, no obstante, incluso en este caso, conocer o tener las herramientas para evaluar la calidad de las desviadas uniformes usadas puede sustentar mejor cualquier estudio teórico de los resultados.

Una prueba de evaluación de números aleatorios por sí sola no es determinante de su calidad. Un buen generador puede no rendir bien en algunas pruebas. La situación que si debe llamar la atención es que un generador no rinda apropiadamente en todas las pruebas a las que se somete (por ejemplo uniformidad, correlación, independencia, etc.). Estos generadores están sujetos a presentar vicios en sus secuencias que pudieran introducir ruido muy dañino en cierto tipo de aplicaciones. Tome el caso de una aplicación para la generación de claves de seguridad de sistemas operativos, en este tipo de sistemas los generadores de secuencias aleatorias deben ser cuidadosamente analizados y escogidos.

Parte III

El Algoritmo Genético

Creatividad es la habilidad de introducir orden a la aleatoriedad de la naturaleza.

Eric Hoffer (1902 - 1983)

6

Algoritmos Genéticos

6.1 Introducción

Los Algoritmos Genéticos (AG), nacieron formalmente en la década de los 60 cuando John Holland y un grupo de sus estudiantes simulaban sistemas adaptativos naturales en medios artificiales [27]. Son métodos estocásticos de búsqueda donde los principios básicos de la genética son llevados a un modelo computacional, esto es; los operadores de cambio genético que impulsan el proceso evolutivo en los seres vivos, son simulados en algoritmos computacionales.

En general, los componentes básicos de un AG son una población finita de soluciones posibles (al problema que se intenta resolver), un conjunto de operadores genéticos, y algún método o medio por el cual se pueda “medir” o “estimar” la calidad de un individuo relativa a otros, es decir, una forma de ordenar en términos de algún criterio de calidad, los individuos de una población.

Los AG poseen una serie de características que los diferencian de otros métodos de búsqueda u optimización y les brindan una serie de ventajas para su uso en algunos tipos de problemas. Algunas de estas características son:

Paralelismo: Los AGs son métodos basados en población, por lo que las operaciones y evaluaciones son hechas sobre un conjunto de soluciones factibles y en cada paso generacional emerge una nueva población de soluciones. Por supuesto, en una máquina de un solo procesador, estrictamente hablando no es posible el paralelismo, sin embargo, los AGs son también propicios para aprovechar el potencial de hardware multi-procesador puesto que pueden evolucionar poblaciones diferentes en los distintos CPUs (unidades de procesamiento central por sus siglas en inglés) como el caso de los AG de Isla. No obstante, sin importar el número de procesadores de un equipo, desde el punto de vista del número de soluciones analizadas, evolucionadas y evaluadas por generación (iteración del algoritmo), los AGs llevan un tipo de paralelismo implícito [20].

Búsqueda ciega: Los AGs buscan sobre un espacio solución de manera ciega, es decir, no van específicamente orientados hacia un objetivo conocido o manejando información del dominio para su planificación. El operador de selección es el guía de la búsqueda a través de la función de calidad dada por el diseñador, sin embargo, esta evaluación no es usada para deliberadamente ubicar una solución en un espacio o punto óptimo específico.

Independencia del problema: Una vez que una representación adecuada ha sido escogida, la dinámica del AG es independiente del problema que busca resolver. Sus operadores de variación actúan de manera independiente del problema o contexto. En [20] (página 2), el autor acota:

Estos algoritmos son computacionalmente simples pero poderosos en su búsqueda de mejores soluciones. Mas aún, no están fundamentalmente limitados por presunciones restrictivas sobre el espacio de búsqueda (continuidad, existencia de derivadas, unimodalidad, multimodalidad, etc).

La separación funcional del algoritmo con la estructura del problema le da al AG una gran versatilidad al permitir aplicar la misma dinámica evolutiva a la resolución de distintos problemas de diferente índole.

El AG estándar busca dentro de una población de individuos (posibles soluciones), uno que satisfaga apropiadamente los requerimientos del problema, en otras palabras, busca la mejor solución de entre un conjunto de soluciones posibles, no obstante, para algunos problemas es no solo deseable encontrar una única y muy

6.2. EL ALGORITMO GENÉTICO SIMPLE

buena solución, sino que conviene encontrar varias soluciones con cierto nivel de calidad.

Algunos autores han clasificado los AGs en cuatro grandes categorías:

- Cliente-servidor
- Isla
- Celulares
- Jerárquicos

Los de tipo cliente-servidor tienen una sola población y la evaluación de la adaptabilidad es repartida entre varios procesadores. Los de tipo Isla incluyen varias sub-poblaciones que evolucionan de manera independiente de las otras, con migraciones ocasionales de individuos entre los grupos o *islas*. Usualmente se escogen diferentes parámetros para configurar los operadores genéticos para cada grupo. En los AGs Celulares, la población es estructurada en el espacio de manera que los operadores genéticos están limitados a una pequeña vecindad que ocasionalmente se solapan y un AG jerárquico es un modelo híbrido entre un modelo de isla y un enfoque de cliente-servidor (o celular) [22, 6].

Los AGs han sido exitosamente usados en muchas áreas del conocimiento, en especial en las ciencias de la ingeniería y la tecnología. Han probado ser una herramienta útil en problemas de optimización, diseño y otros tipos con un amplio registro de aplicaciones exitosas. Su simplicidad en términos de implementación y su sobresaliente rendimiento, los han hecho una técnica muy popular.

6.2 El Algoritmo Genético Simple

Como se mencionó en la sección 6.1, los Algoritmos Genéticos son métodos estocásticos de búsqueda que emulan la evolución genética para navegar un espacio de soluciones en busca de un elemento que mejor satisfaga un criterio específico de calidad de entre varios elementos factibles (paralelismo implícito). En la actualidad existen muchas versiones, extensiones, refinaciones, modificaciones, etc. de algoritmos genéticos, todas con diferentes características orientadas a tratar con distintas

6.2. EL ALGORITMO GENÉTICO SIMPLE

familias de problemas, por ejemplo, unos *exploran* mas ampliamente el espacio solución mientras que otros son diseñados para *explotar* mas eficazmente una región específica de este espacio. A la versión original de Holland [27] usualmente se le llama el algoritmo genético simple o algoritmo genético canónico y sin importar las modificaciones de los AG en la actualidad, la mecánica básica sigue siendo la misma.

La dinámica evolutiva del algoritmo genético que permite que las soluciones generadas vayan paulatinamente adaptándose a su entorno o ambiente (el problema que resuelven), se basa en tres elementos principales:

Una Población finita de soluciones posibles. Al igual que una población de individuos biológicos buscando sobrevivir a las imposiciones del ambiente (obtención de alimento, agua, territorio, etc. para reproducirse), el AGS simula una población de individuos artificiales donde cada uno representa una posible solución al problema que se intenta resolver. Este problema es entonces el ambiente de esos individuos. La población va paulatinamente evolucionando de manera que con el paso de las generaciones (iteraciones del algoritmo), los individuos se van volviendo cada vez mas aptos, es decir, las soluciones propuestas por el algoritmo van constantemente mejorando.

Una forma de evaluación y tasación de individuos. En la supervivencia del mas apto del paradigma Darwiniano, el ambiente determina qué individuos son mas aptos que otros, por ejemplo, una presa incapaz de desplazarse para escapar del asedio de su depredador, es funcionalmente menos apta que otra capaz de moverse velozmente, en términos evolutivos la primera es menos apta que la segunda (para el ambiente que enfrentan). Similarmente, el problema que el AGS intenta resolver, planteado en forma de una función de adaptabilidad, ordena los individuos de la población según la calidad de la solución que representan.

Un mecanismo de cambio de los individuos. Para que una población evolucione, se necesitan cambios que mejoren la adaptabilidad del individuo a su entorno. Intuitivamente, la prole de una generación debe estar en promedio, mejor adaptada que sus padres puesto que los genes beneficiosos (genes que permiten alguna ventaja competitiva) tienen mayor probabilidad de ser heredados por causa de la selección natural. De esta manera, cambios que resultan en un incremento en la adaptabilidad del individuo, tienden a permanecer en la población por mas tiempo y en consecuencia, paulatinamente el promedio

6.2. EL ALGORITMO GENÉTICO SIMPLE

poblacional va convergiendo a un mejor nivel de adaptación. Es importante destacar que en sistemas biológicos el ambiente no es estático, es decir, constantemente presenta variaciones que de algún modo pueden ocasionar que un rasgo considerado ventajoso, deje de serlo o incluso pase a ser desventajoso de un momento a otro. En el AGS, aunque también se simulan ambientes dinámicos, generalmente se trabaja en un contexto estático y por ello es usual tomar un intervalo de n generaciones para simular el proceso evolutivo.

El algoritmo 1 muestra los pasos generales del AGS. En el paso 1 se genera una población inicial (generalmente de manera aleatoria), y comienza el ciclo evolutivo medido en generaciones (variable i) donde los individuos son evaluados en términos de su adaptabilidad, son sometidos a operadores de cambio y se genera una nueva población. Algunos operadores son aplicados según el resultado de un evento de Bernoulli. Por ejemplo, en el paso 6, un evento de Bernoulli es aplicado y de resultar positivo se ejecuta el cruce. Un evento de este tipo es similar a lanzar una moneda, si cae cara se considera el resultado como positivo, de lo contrario se considera negativo.

Algoritmo 1 El Algoritmo Genético Simple

```
1: generar población inicial  $P_0$ 
2:  $i = 0$ 
3: repetir
4:   evaluar individuos en  $P_i$ 
5:   seleccionar parejas de individuos en  $P_i$ 
6:   si bernoulli(cruce)  $\rightarrow$  positivo entonces
7:     cruzar pareja y llevar prole a  $P_{i+1}$ 
8:   si no
9:     llevar pareja a  $P_{i+1}$ 
10:  fin si
11:  mutar individuos en  $P_{i+1}$ 
12:   $i = i + 1$ 
13: hasta que condición de parada se cumpla
```

Hemos discutido que uno de los principales elementos del AGS es una población de individuos que representan soluciones potenciales al problema planteado, pero ¿Cómo representar estas soluciones? ¿Cómo construir estos individuos? Esta es una de las principales tareas en el diseño del AGS y es también la clave que el problema

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

sea independiente de la mecánica del algoritmo pues los operadores evolutivos de cambio serán aplicados a la representación de la solución y no a la variable de diseño directamente o a otros elementos del algoritmo.

6.3 Elementos de inicio del Algoritmo Genético

Para que el AG pueda aplicar la dinámica evolutiva sobre un problema de información, se deben diseñar una serie de elementos y operadores que lo permitan. Es en este diseño donde yace el éxito o fracaso del algoritmo, su convergencia o estancamiento, su versatilidad o ineficacia, etc. No existe una regla para la escogencia y parametrización de los operadores, su planificación o forma de aplicación. El investigador debe “graduar” su modelo a medida que recoge experiencia sobre el problema al que aplica el Algoritmo. A continuación se plantean los elementos mas importantes del AG.

6.3.1. Representación

La representación de una posible solución al problema se hace manifiesta a través de una estructura de datos llamada el *cromosoma*. En el AGS esta estructura es una cadena, vector o arreglo de longitud l , de un alfabeto de k símbolos, usualmente binarios ($k = 2$). El *ancho de banda* del cromosoma se calcula usando la expresión $A_{banda} = k^l$ y el mayor número representable es $A_{banda} - 1$. La idea de la representación de una potencial solución en una estructura de datos es hacer posible la aplicación de los operadores de cambio al cromosoma para ir evolucionando (en términos prácticos: mejorando) las soluciones. Para esto necesitamos un método de codificación (traducir la solución a la estructura de dato apropiada) y un método de decodificación, es decir, el paso inverso.

Binaria

El tipo de representación mas común para el AGS es la representación binaria, es decir, las potenciales soluciones al problema que se abarca son codificadas a un cromosoma binario, generalmente en un arreglo de longitud l . Tomemos por ejemplo el problema de minimizar $f(x) = x^2$ en un dominio real de $[-10 \dots 10]$. El problema que busca resolver el AGS es encontrar un x^* en el espacio de búsqueda de los números reales entre -10 y 10, que haga que $f(x^*) \leq f(x) \quad \forall x \in [-10 \dots 10]$. Vemos

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

entonces que las potenciales soluciones al problema están en los números reales entre -10 y 10, en consecuencia, lo que debemos representar en el cromosoma binario son números reales en ese rango.

Para el ejemplo planteado, debemos diseñar una forma de representar números reales en una estructura discreta (un arreglo de binarios por ejemplo). Esto se puede hacer de muchas formas, una de ellas es usar la notación de *coma flotante* de la IEEE, que puede representar números reales en el rango de $3,4 \cdot 10^{+/-38}$ con 7 dígitos de precisión en un ancho de banda de 4 bytes. Otra forma mucho mas común, es hacer la proyección lineal directa del rango de reales a representar, a la estructura de ancho de banda determinada. Para un arreglo de 8 posiciones de números binarios, tenemos un ancho de banda de $A_{banda} = 2^8 = 256$ y contando el 0, el mayor número representable es $256 - 1 = 255$. Esto significa que tenemos un margen de 255 unidades para representar el conjunto infinito de números reales entre -10 y 10 como se muestra en la figura 6.1.

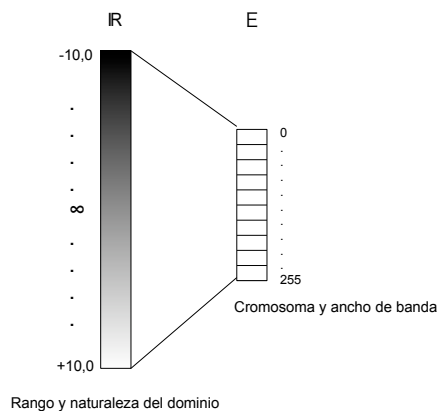


Figura 6.1: Codificación del cromosoma. Los números reales entre -10 y 10 (de naturaleza continua e infinita), deben ser representados en una estructura entera con ancho de banda 256.

Para el caso planteado, el proceso de codificación es simplemente llenar el arreglo con 1's o 0's, el problema viene en cómo interpretar una tira binaria del arreglo como un número real en el rango permitido. Este es el proceso de descodificación. Una forma de hacerlo es hacer equivaler los extremos del rango a representar con los

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

extremos del cromosoma, es decir, tomamos el -10 como el número 0, y el +10 como el número 255 (ver figura 6.1) y el resto se asignan según una simple proyección lineal o una regla de 3. Ejemplo, debemos expresar los números entre -10 y 10 con 255 números enteros, es decir, tenemos $(10 - (-10))/255 \approx 0,07843$ unidades de separación entre un número y otros (precisión), de manera que si el 0 corresponde a -10,0, entonces el 1 corresponde a $-10,0 + 0,07843 = -9,92157$, el 2 corresponde a $-10,0 + (0,07843 * 2) = -9,84314$ y así sucesivamente. De manera genérica, la decodificación de un número entero a su equivalente número real de un rango acotado por $R_{inf} \dots R_{sup}$ con un ancho de banda de representación de A_{banda} es:

$$Valor_{real} = R_{inf} + \frac{R_{sup} - R_{inf}}{A_{banda} - 1} * Valor_{entero} \quad (6.1)$$

Y para el caso específico del ejemplo, la decodificación de un valor entero $Valor_{entero}$ al equivalente $Valor_{real}$ sería:

$$Valor_{real} = -10 + \frac{10 - (-10)}{2^8 - 1} * Valor_{entero} = -10 + 0,07843 * Valor_{entero}$$

Observe que a medida que el ancho de banda es mayor, la *granularidad* con la que podemos representar los números reales es menor y por lo tanto aumenta la precisión. Por ejemplo, para este caso, $A_{banda} = 8$ (suponga que son bits), la precisión es de 0,07843 aproximadamente. Si usáramos 16 bits, esta precisión sería aproximada a 0,00035 (un número mas pequeño significa que hay menos holgura entre dos números representados, por lo tanto hay mayor precisión).

El ejemplo discutido es un escenario del caso mas sencillo, esto es porque se necesita representar una sola variable (x). Suponga que el problema a resolver es de múltiples variables $f(x, y, z)$ por ejemplo. Necesitamos representar todas las variables de diseño dentro del mismo cromosoma, una alternativa es asignarle a cada variable de diseño una porción del cromosoma y usando la ecuación 6.1 para decodificar de manera independiente cada segmento.

La figura 6.2 muestra 3 variables codificadas en segmentos de 6 bits c/u en un cromosoma de 18 bits. Para la decodificación se usa la ecuación 6.1 sustituyendo los parámetros según el caso.

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

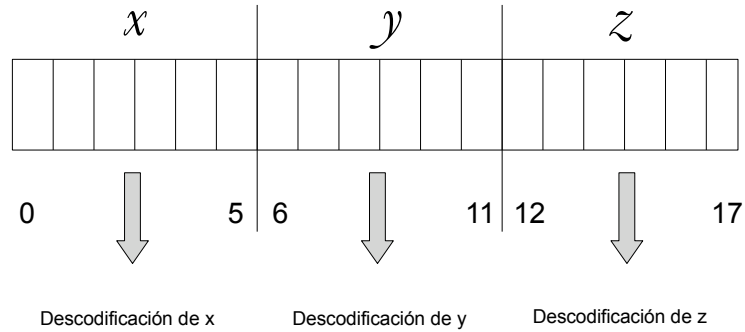


Figura 6.2: Cromosoma con múltiple variables

La figura 6.3 muestra un cromosoma de dos variables de diseño (también variables objetivo) con representación binaria. Cada variable tiene 8 *genes* que componen su *genotipo*. Aplicando la ecuación 6.1 se obtiene el *fenotipo* de cada variable.

Gray

Al igual que la codificación binaria, la codificación Gray, llamada así por Frank Gray de Laboratorios Bell, quien introdujo una solicitud de patente por este desarrollo en 1947 [59], utiliza un alfabeto en base 2 (1's y 0's). La diferencia entre la codificación Gray y la binaria normal es que entre dos números sucesivos cualesquiera representados en Gray, hay solo 1 bit de diferencia, es decir la distancia de Hamming entre dos números sucesivos en Gray es 1.

La codificación de una variable de diseño en Gray, requiere que ésta sea llevada primero a su expresión en binario para luego pasarla a Gray [13]. Una tira binaria (b_1, b_2, \dots, b_l) con $b_i \in \{0, 1\}$ es convertida a una tira Gray (g_1, g_2, \dots, g_l) usando una función $\gamma^{-1} : \mathbb{B}^l \rightarrow \mathbb{B}^l$ como se describe a continuación: ([1] citado en [13])

$$g_i = \begin{cases} b_i & \text{si } i = 1 \\ b_{i-1} \oplus b_i & \text{si } i \neq 1 \end{cases} \quad (6.2)$$

Donde \oplus es suma en módulo 2, es decir, una suma binaria sin contemplar aca-

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

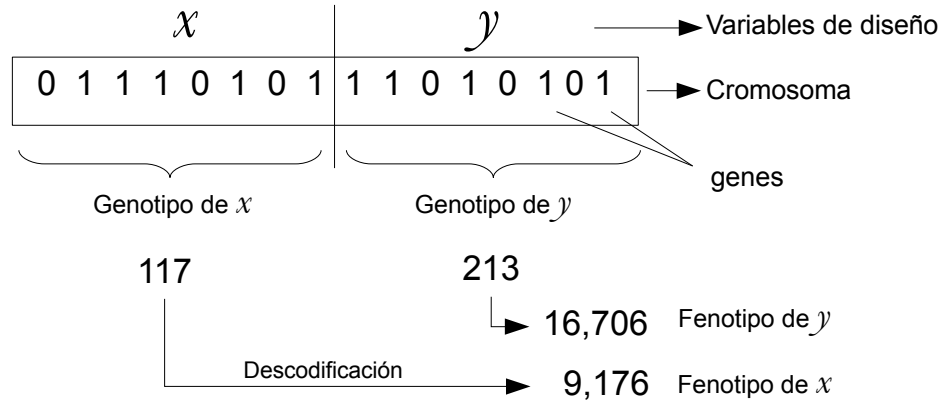


Figura 6.3: Cromosoma con representación binaria

reco, o lo que es lo mismo un XOR (operador ó exclusivo). El cuadro 6.1 muestra los diferentes valores para usar en las conversiones de Gray a binario y viceversa (ver algoritmos 2 y 3).

A	B	\oplus
1	1	0
1	0	1
0	1	1
0	0	0

Cuadro 6.1: Tabla de verdad del XOR

Para descodificar una tira en Gray a su correspondiente valor según la variable de diseño, se hace el mismo procedimiento a la inversa, es decir, se lleva de Gray a binario a través de la ecuación 6.3 (tomada de [13]) y luego se lleva a entero o a lo que corresponda.

$$b_i = \bigoplus_{j=1}^i g_j \quad \text{para } i = \{1, \dots, l\} \quad (6.3)$$

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

El algoritmo 2 muestra la secuencia de pasos para la conversión de una tira binaria a su equivalente en Gray. Teniendo la cadena binaria b_i de longitud l como entrada, se copia el bit mas significativo a la posición mas significativa de la cadena Gray a formar. Luego se van sumando en módulo 2 los bits de menor orden (de manera secuencial) formando los bits de la cadena Gray.

Algoritmo 2 Conversión de binario a Gray

Entrada: b_i $i = \{0, 1, \dots, l\}$ {cadena binaria}

Salida: g_i $i = \{0, 1, \dots, l\}$ {cadena Gray}

1: $i = 1$

2: $g_1 = b_1$

3: **repetir**

4: $i = i + 1$

5: $g_i = b_i \text{ XOR } b_{i-1}$

6: **hasta que** $i > l$

El proceso de conversión de Gray a binario es similar pero la suma en módulo 2 (o exclusivo), se hace entre los bits de la cadena Gray y los bits que se vayan formando en la cadena binaria (ver algoritmo 3). Se comienza copiando el bit mas significativo de la cadena Gray a la posición mas significativa de la cadena binaria y se continua según lo ya descrito.

Algoritmo 3 Conversión de Gray a binario

Entrada: g_i $i = \{0, 1, \dots, l\}$ {cadena Gray}

Salida: b_i $i = \{0, 1, \dots, l\}$ {cadena binaria}

1: $i = 1$

2: $b_1 = g_1$

3: **repetir**

4: $i = i + 1$

5: $b_i = b_{i-1} \text{ XOR } g_i$

6: **hasta que** $i > l$

La codificación en Gray hace que dos números enteros sucesivos cualesquiera tengan representaciones que difieren en 1 solo bit (distancia Hamming = 1). Esta

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

característica evita que pequeños cambios en la estructura del cromosoma (el *genotipo*), produzcan cambios abruptos en el valor de la variable de diseño (el *fenotipo*). Estos abruptos cambios pueden ocasionar lo que se conoce como *riscos de Hamming*, esto es, diferencias significantes en el espacio de representación, de porciones del espacio de solución que no son tan diferentes.

Binaria desordenada

En la representación binaria normal, los genes se codifican en un locus fijo, es decir, su posición en el cromosoma es fija y lo que cambia es su valor (1 ó 0). Suponga que un esquema solución, por ejemplo, una configuración óptima de variables de decisión en el genotipo necesita que ciertos genes alejados entre sí en el cromosoma, tengan determinados valores. El operador de cruce utilizado puede reducir las probabilidades que los genes con el mismo valor continúen en el mismo locus, es decir, el cruce pudiera afectar la configuración ya obtenida puesto que un hijo hereda un cromátido de genes generalmente cercanos de un padre.

La representación desordenada, propuesta en[19], busca incluir en el cromosoma no solo el valor del gen sino también su posición. De esta manera, dos genes de locus distantes pueden ser representados de manera adyacente en la estructura de dato del cromosoma. Básicamente, esta representación usa dos elementos para representar un gen, $g = (x \ y)$ donde x es el locus o ubicación del gen, y su valor es y . Ejemplo:

$$[(3 \ 0) (1 \ 1) (8 \ 1) (5 \ 1) (2 \ 0) (4 \ 0) (7 \ 0) (6 \ 1)]$$

significa (leyendo de izquierda a derecha) que los genes de la tercera, segunda, cuarta y séptima posición en el cromosoma, tienen el valor de 0, mientras que el primero, octavo, quinto y sexto gen tienen un valor de 1.

Este tipo de representación es capaz de mantener información que en el genotipo esta separada, de manera adyacente. Tome por ejemplo el segundo y tercer gen, en el cromosoma mostrado están adyacentes, no obstante, ambos están en los extremos del cromosoma (posiciones 1 y 8) respectivamente.

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

Real desordenada

Un esquema similar de representación desordenada es propuesto por Deb en [12], pero para su uso en variables de tipo real. La idea de este tipo de representación es usar tres elementos para representar un gen, y varios genes pueden tener información sobre una misma variable de decisión. Por ejemplo, un gen es compuesto por los elementos $g = (x, y, z)$ donde x es un número de identificación de la variable decisión, y es la representación de la mantisa o el exponente y z el valor de éste. Para descodificar el genotipo a una variable real se usa la ecuación:

$$x_i = \text{mantisa}_i \cdot \text{base}^{\text{exponente}_i} \quad (6.4)$$

donde base es una constante (usualmente 10) y $x_i \in \mathbb{R}$. Para calcular el exponente se suman algebraicamente todos los genes con el componente x igual (misma variable decisión), $y = E$ (exponente) y el signo representado en z indica el sentido de la suma (+ ó -). Considere el siguiente cromosoma:

$$[(1 \text{ E } -) (2 \text{ E } -) (1 \text{ M } 0) (1 \text{ M } 1) (2 \text{ E } +) (1 \text{ M } +) (2 \text{ M } 0) (1 \text{ E } -)]$$

Vemos que la terna (1 E -) aparece dos veces, esto significa que la variable decisión 1 tiene un exponente de -2 mientras que el exponente de la segunda variables es calculado con (2 E -) (2 E +), es decir, el exponente es 0.

El cálculo de la mantisa es un poco mas complejo, primero se agrupan unos y ceros en las ternas con la misma variable decisión y con el identificador de la mantisa $y = M$, separados por los símbolos “-” o “+”. El número de grupos resultantes determinará la granularidad de la variable. El rastreado del cromosoma se hace de izquierda a derecha, por ejemplo, para la variable decisión 1, las ternas de mantisa son: (1 M 0) (1 M 1) (1 M +) El número de 1's y 0's separados por símbolos \pm es 01+ de manera que hay un cero y un uno en el primer grupo y nada en el segundo (después del +). El número posible de combinaciones unitarias (solo un símbolo) en dos bits es de 3 (no unos, un uno, dos unos). Se divide el intervalo de búsqueda de la mantisa (0,1) entre 3 y quedan los siguientes intervalos: (0,0.333), (0.333, 0.667), (0.667,1). El grupo 01 tiene un solo uno por lo tanto es del segundo grupo (un uno),

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

y el símbolo “+” indica que el resultado del cálculo del siguiente grupo se suma al límite inferior del presente, en este caso 0.333. Como no hay mas ceros o unos después del +, la mantisa es el promedio entre los dos límites del intervalo seleccionado, es decir $(0,333 + 0,667)/2 = 0,5$ y el resultado final es: variable decisión 1 = $0,5 \cdot 10^0$.

Real simple

Algunos algoritmos genéticos, aunque no el AGS, usan directamente una representación de tipo real. En este tipo de representación el valor de la variable de diseño es llevado al cromosoma en un número real. Para el ejemplo planteado al comienzo de la sección 6.3.1, la variable de diseño x ($x \in \mathbb{R}$) es representada directamente en el cromosoma sin necesidad de codificación binaria y posterior descodificación como se muestra en la figura 6.4.

Al usar este tipo de representación, los operadores evolutivos deben ser modificados puesto que en el AGS estos operadores actúan sobre los *genes* del cromosoma, que para el caso de representación binaria del ejemplo, es cada bit o cada celda del arreglo. Para el caso de un número real, debe diseñarse una estrategia de cambio en el genotipo que permita evaluar cómo se afecta el fenotipo en términos de adaptabilidad, por ejemplo, ¿Cómo se realizaría una mutación dentro de un genotipo donde cada gen es un número real? ¿Cómo se realizaría un cruce entre dos cromosomas que representan una sola variable decisión?

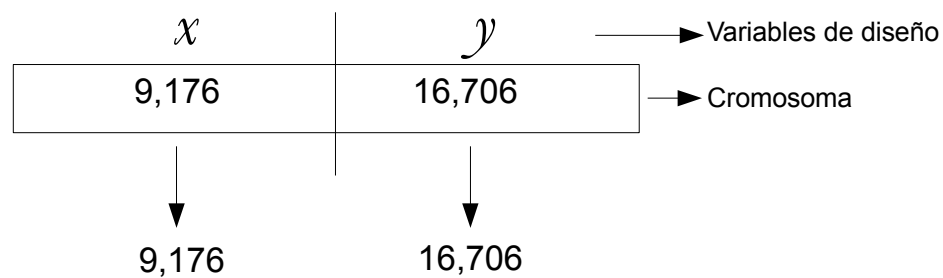


Figura 6.4: Cromosoma con representación real simple. Para este caso, la descodificación para la obtención del fenotipo de la variable objetivo no es necesaria

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

Otras representaciones

La representación de una variable objetivo en una estructura de dato de cromosoma no necesariamente debe ser de forma numérica. El cromosoma es el contenedor de la estructura de la variable objetivo, de su ADN. Cualquier representación que permita luego de un proceso de descodificación llegar a la variable objetivo original y que además permita operadores de evaluación actuar sobre sí para la tasación de los individuos según algún criterio de calidad (adaptabilidad), puede ser usada para el cromosoma.

Un ejemplo de posibles representaciones no numéricas es el problema del **vendedor viajero**. Este es un problema de optimización combinatoria considerado “np-completo” por la dimensión del espacio solución que puede llegar a tener. Se trata de un agente viajero que debe completar un circuito compuesto por varias ciudades, sin repetir la visita a ciudad alguna y en la menor distancia posible, es decir, el circuito recorrido debe ser mínimo.

Las posibles soluciones a este problema son entonces los diferentes circuitos posibles que el agente puede hacer. Suponga que se tienen cinco ciudades (A,B,C,D,E) y una tabla con las distancias entre ellas. Una posible solución es que el agente parta de la ciudad C, luego visite la B, siga a la E, pase por la A, luego D y finalmente regrese a C. Este circuito pudiera ser representado como una lista con las ciudades que se van visitando en el orden de visita; $C \rightarrow B \rightarrow E \rightarrow A \rightarrow D \rightarrow C$. Para evaluar qué tan bueno es este circuito, se suman las distancias recorridas de acuerdo a la tabla de distancias y la ruta mas corta es la mejor.

La figura 6.5 muestra un cromosoma con una representación no numérica, en este caso, alfabética. El genotipo del individuo x (compuesto por todos sus genes) es CBEADC mientras que su fenotipo es $C \rightarrow B \rightarrow E \rightarrow A \rightarrow D \rightarrow C$ que representa un circuito de distancia d .

Con los tipos de representaciones descritas, el lector debe concluir que cualquier estructura de datos; numérica, semi-numérica, simbólica, etc. puede ser usada como una representación del cromosoma para el AG, siempre que esta estructura permita:

- Codificar y descodificar la representación usada a una única solución (individuo), es decir, sin ambigüedad.

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

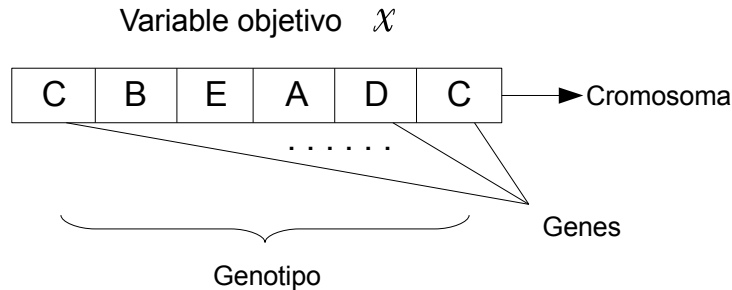


Figura 6.5: Cromosoma con representación alfabética

- La posibilidad de aplicar una función de calidad para determinar y ordenar los individuos en base a su adaptabilidad (qué tan buena es la solución).

6.3.2. Función de adaptabilidad

La función de adaptabilidad es el mecanismo por el que el AG mide la calidad de los individuos en la población, por esta razón es también llamada la función de calidad. En el AGS, las soluciones se representan en forma binaria, por lo tanto, las variables objetivo deben ser codificadas para llevarlas al cromosoma y luego estas tiras binarias deben ser sometidas a una función de descodificación para poder evaluar la solución que representan mediante la función objetivo y de adaptabilidad. Si tomamos la descodificación como una función $\lambda : \mathbb{B} \rightarrow \mathbb{S}$ donde \mathbb{S} representa un espacio específico del problema (espacio solución), y la función objetivo $f : \mathbb{S} \rightarrow \mathbb{R}$, entonces la función de adaptabilidad es una transformación $\phi = f \circ \lambda$ [13].

La adaptabilidad de un individuo se refiere al grado de éxito que éste, o los cambios de comportamiento, estructurales, etc. a los que ha sido sometido, han manifestado en la supervivencia, reproducción, o capacidad de realizar alguna tarea. Para el caso de un AG, el problema representa el ambiente de los individuos en la población (potenciales soluciones), generalmente (pero no siempre) es estático, es decir, no varía en el tiempo y esto permite hacer una evaluación más estándar de la calidad de las soluciones obtenidas por el algoritmo en todas sus iteraciones (generaciones).

La función de adaptabilidad está basada en la función objetivo del problema,

6.3. ELEMENTOS DE INICIO DEL ALGORITMO GENÉTICO

incluso en muchos AG son la misma función, no obstante, dependiendo del problema a tratar, algunas veces es necesario realizar una transformación sobre la función objetivo f para asignar un nivel de calidad al individuo.

En el ejemplo de optimización planteado en la sección 6.3.1, se quiere encontrar un x^* que haga $f(x^*)$ mínima en el dominio $(-10 \dots 10)$, con $f(x) = x^2$. Las potenciales soluciones (individuos), son números reales en el rango permitido y la función de calidad, debe entonces ser capaz de tasar los individuos y decidir cuáles son mejores adaptados al problema que se resuelve. Fácilmente se puede observar que entre dos individuos x_1, x_2 con $x_i \in (-10 \dots 10)$ y $f(x_1) < f(x_2)$, x_1 es “mejor” o mas apto que x_2 porque el problema trata de minimizar f y por tanto su adaptabilidad es mayor.

Genotipo x	Fenotipo x	$f(x)$	Adaptabilidad
00000000	-10,0	100	0,04
00111011	-5,37	28,84	0,12
10010111	1,84	3,39	1,0
11011011	7,18	51,55	0,07
11111111	10,0	100	0,04

Cuadro 6.2: Adaptabilidad de x para $f(x)$

El cuadro 6.2 muestra la tasación de 5 diferentes individuos según su adaptación. Note que el individuo con mayor adaptabilidad (1,0) es el de menor valor en f . Esto es porque el problema es de minimización. La decodificación de los fenotipos fue hecha usando la ecuación 6.1, observe que para todos los individuos, incluso para los dos genotipos extremos (máxima y mínima expresión del cromosoma), se tiene que $-10 \leq \text{fenotipo} \leq 10$.

La escala de adaptabilidad en el cuadro 6.2 se obtuvo *normalizando* los valores de $f(x)$ relativo a la población, es decir, $\text{adaptabilidad}_{x_i} = \frac{f(x^*)}{f(x_i)}$ donde $f(x^*)$ es el mínimo de las soluciones de la población; $f(x^*) \leq f(x_i)$ con $i = (1, 2, \dots, p)$ para una población de p individuos. De esta manera, todos los valores de adaptabilidad están en una escala de $(0 \dots 1)$ siendo 1 el individuo mas o mejor adaptado. Note que la normalización es por población/generación, es decir, es local.

6.4 Operadores de variación

Los operadores de variación, operadores de cambio, operadores evolutivos u operadores genéticos son algunos de los términos por los que se llama al conjunto de funciones y herramientas que permiten la evolución del cromosoma.

6.4.1. Selección

Selección es el proceso de escoger uno de una serie de elementos dado algún criterio. En sistemas biológicos, la recompensa de la reproducción es la continuación del código genético de los individuos, sin embargo, para merecer esta recompensa, los sujetos deben ser seleccionados primero (generalmente en términos de su adaptabilidad), de ahí la gran importancia que tiene este operador en el proceso evolutivo.

En un AG, si la *presión selectiva* es muy alta, puede resultar en un estancamiento del algoritmo en un óptimo local, esto es porque el individuo más fuerte (más apto), se apodera del banco genético y pronto la falta de diversidad hará difícil que el proceso “salte” fuera del remolino de atracción del óptimo local. Por otra parte, si la presión selectiva es muy baja, el algoritmo perderá efectividad puesto que no es seguro que la adaptabilidad promedio de la población converja a un valor apropiado.

Muchos esquemas de selección han sido propuestos, en la mayoría de ellos la adaptabilidad se mantiene como la variable principal de criterio selectivo [28]. Entre los principales esquemas de selección están el de adaptabilidad proporcional, truncado, rango y torneo.

Adaptabilidad proporcional

En el esquema de selección de adaptabilidad proporcional, el número de copias esperadas de una solución específica es proporcional a su adaptabilidad. La forma más común de selección de adaptabilidad proporcional es la llamada *rueda de ruleta*. En este esquema, la solución es asignada a un área de la ruleta proporcional a su nivel de adaptación; mientras mayor sea su adaptabilidad, más grande es el área asignada.

6.4. OPERADORES DE VARIACIÓN

Este esquema de selección es el usado por el algoritmo genético simple y simula una rueda de ruleta donde el tamaño de cada casilla es proporcional al nivel de adaptabilidad del individuo. Al igual que con el juego, mientras mas grande sea una casilla, mayor es la probabilidad de ser seleccionada.

El cuadro 6.3 muestra la adaptabilidad de una población compuesta por 4 individuos para un algoritmo genético en la solución del problema de minimizar $f(x) = x^2$, $x \in (-10 \dots 10)$. La sumatoria de todas las adaptabilidades en la población es la referencia para el cálculo del peso relativo de la adaptabilidad de cada individuo.

Genotipo x	Fenotipo x	$f(x)$	Adaptabilidad de x	Peso de x
10001101	1,06	1,12	0,31	15 %
10001001	0,75	0,56	0,63	28 %
10000111	0,59	0,35	1,0	45 %
10001110	1,13	1,28	0,27	12 %
Total			2,21	100 %

Cuadro 6.3: Tabla de adaptabilidad para la selección de ruleta

Para la programación de la selección de rueda de ruleta en el AG, se necesita una ponderación de la adaptabilidad de cada individuo donde el mejor adaptado tenga un mayor valor. En el ejemplo del cuadro 6.3, el mejor individuo según la función objetivo es el de fenotipo 0,59, cuyo valor objetivo es 0,35. Para ser sometido a la selección de rueda de ruleta debemos representar la adaptabilidad de manera que los menores valores produzcan mayor ponderación (puesto que son “mejores” en el contexto del problema). Una forma es tomar la adaptabilidad como el inverso o recíproco de $f(x)$, es decir, $\Lambda_i = 1/f(x_i)$, otra forma es normalizar todos los valores de manera que las ponderaciones queden en un rango de $(0 \dots 1)$, donde 1 corresponde al individuo mejor adaptado.

La probabilidad que un individuo x_i con adaptabilidad Λ_i sea seleccionado con este esquema en una población de N individuos es:

$$p_{sel} = \frac{\Lambda_i}{\sum_{j=0}^N \Lambda_j}$$

6.4. OPERADORES DE VARIACIÓN

La figura 6.6 muestra la analogía del algoritmo 4 con la rueda de ruleta. Se comienza con simular la potencia del giro de la rueda (ver paso 2). La ruleta sigue girando (pasos 5 al 8) hasta que el indicador de la ruleta cae en una casilla específica (paso 8). El individuo cuya adaptabilidad corresponde a esa casilla, es el seleccionado. Note que la probabilidad de un individuo a ser seleccionado es directamente proporcional a su adaptabilidad.

Una de las principales ventajas de este esquema de selección es su simplicidad, lo que lo hace fácil de ver, entender e implementar. Es el operador selectivo usado por el algoritmo genético simple y ofrece una alternativa sencilla de representación de la selección Darwiniana del más apto. Entre sus desventajas está el “ruido” al que son vulnerables pues la tasa de evolución depende de la varianza de la adaptabilidad de la población [54].

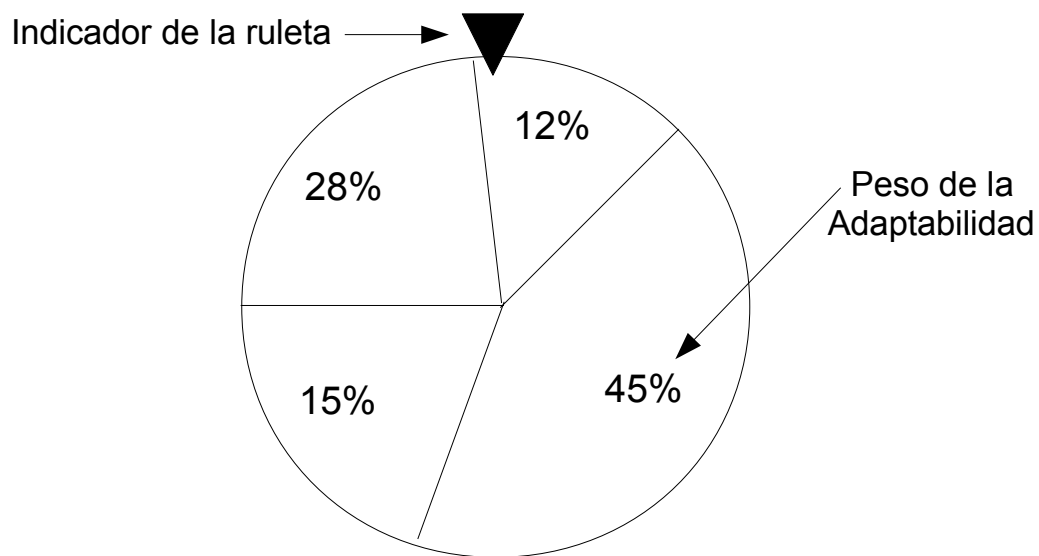


Figura 6.6: Selección de rueda de ruleta

Algoritmo 4 Selección de rueda de ruleta

Entrada: Población de N individuos

Salida: Individuo ganador de la ruleta (x_j)

- 1: $\Lambda_{global} = \sum_{i=1}^N \Lambda_i$
 - 2: Giro_ruleta = $random(0, 1) * \Lambda_{global}$
 - 3: $j = 0$
 - 4: Indicador_ruleta = 0
 - 5: **repetir**
 - 6: $j = j + 1$
 - 7: Indicador_ruleta = Indicador_ruleta * Λ_j
 - 8: **hasta que** (Indicador_ruleta \geq Giro_ruleta) o ($j \geq N$)
-

Selección por Truncado

El esquema de selección por truncado propuesto en [39], es elitista y en comparación a otros métodos de selección, es mas artificial [58]. En la actualidad su uso no es muy común, encontrándose principalmente en AGs con poblaciones muy grandes y criterios de selección muy específicos.

El esquema consiste en ordenar a la población en términos de la adaptabilidad, establecer un umbral de truncado T y seleccionar los mejores individuos de acuerdo a este umbral. De acuerdo al análisis realizado en [3], la tasa de reproducción (la relación entre el número de individuos con adaptabilidad Λ antes y después de la selección) en este esquema es:

$$R_r(\Lambda) = \begin{cases} \frac{1}{T} & : S(\Lambda) > (1 - T)N \\ 0 & : \text{en otro caso} \end{cases} \quad (6.5)$$

Donde $S(\Lambda)$ es la distribución o número de individuos en la población con adaptabilidad Λ , y T el umbral de truncado. Como el esquema selecciona individuos en una fracción T de la población, la pérdida de diversidad genética estimada es:

$$p_{d,\gamma}(T) = 1 - T \quad (6.6)$$

Y la *intensidad de selección* respecto al umbral T , definida como la adaptabilidad promedio de la población luego del truncado (asumiendo una distribución inicial de Gauss con $G(0,1)$), es:

6.4. OPERADORES DE VARIACIÓN

$$I_\gamma(T) = \frac{1}{T} \frac{1}{\sqrt{2\pi}} \exp^{-\frac{\Lambda_c^2}{2}} \quad (6.7)$$

donde Λ_c se determina de acuerdo a $T = \int_{f_c}^{\infty} \frac{1}{\sqrt{2\pi}} \exp^{-\frac{\Lambda^2}{2}} df$. Para ver en detalle el análisis de donde derivan las ecuaciones 6.5, 6.6 y 6.7 de este esquema, ver [3].

Puede ver en el algoritmo 5 que todos los individuos de la población con adaptabilidad por encima de la proporción establecida por el umbral de truncado T , tienen la misma probabilidad de ser seleccionados. Observe que dado que la población está ordenada respecto a adaptabilidad, los individuos a seleccionar son solo aquellos que están en el rango de $((1 - T)N \dots N)$.

Algoritmo 5 Selección por truncado

Entrada: Población P de N individuos, Umbral de truncado $T \in [0 \dots 1]$

Salida: Población de individuos seleccionados P'

- 1: Ordenar(P por adaptabilidad) {Peor individuo en la primera posición}
 - 2: **para** $i = 0$ a N **hacer**
 - 3: individuo_a_seleccionar = *random*{ $[(1 - T)N], \dots, N$ }
 - 4: agregar individuo_a_seleccionar a P'
 - 5: **fin para**
-

Selección por rango

En este tipo de selección los individuos se ordenan por adaptabilidad y luego, como en el caso de la selección por adaptabilidad proporcional, el resultado de la selección es una función lineal del rango [24].

Tome una población con una gran varianza entre la adaptabilidad de los individuos, por ejemplo, considere un individuo con 95 % de peso en la distribución de la adaptabilidad. En la selección por rueda de ruleta, este individuo prácticamente anularía la probabilidad del resto de la población a ser escogidos puesto que solo 5 % del área de la rueda de ruleta es asignada entre el resto de la población. Al cabo de unas cuantas iteraciones, es muy probable que la población este compuesta por varias copias del mismo individuo, es decir, el genotipo mas fuerte se impone disminuyendo así la variedad genética considerablemente. Esto puede llevar a una

6.4. OPERADORES DE VARIACIÓN

convergencia “anormalmente” rápida lo que puede hacer caer al algoritmo en una vasija de atracción local o punto de estancamiento.

En la selección por rango, la adaptabilidad se asigna según el rango del individuo en la población. El individuo menos apto tendrá un rango de 1 mientras que el mas apto tendrá un rango de N . La probabilidad de selección de cada individuo se asigna en función lineal a su rango, de manera que las fuertes diferencias entre individuos que estarían presentes en una asignación de peso o proporción directa se pueden “suavizar” mediante el diseño de esta función. La figura 6.7 muestra una distribución del área de selección de la rueda de ruleta (arriba), y una “suavización” de las diferencias a través de la asignación por rango (abajo).

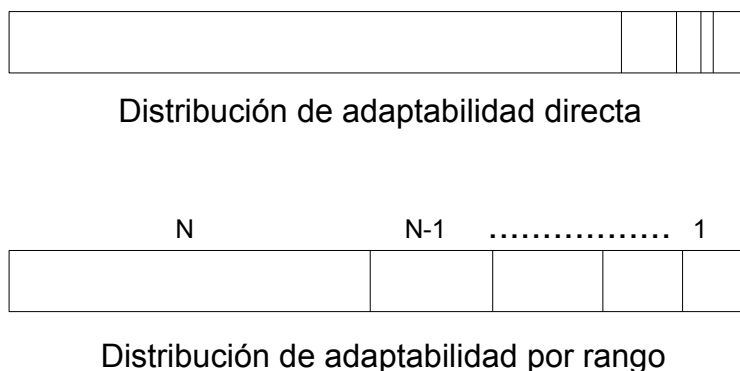


Figura 6.7: Probabilidad de selección en ruleta y rango

Selección por torneo

En la selección por torneo, k individuos son escogidos aleatoriamente de la población y se enfrentan entre sí simulando una competencia o torneo. El individuo con la mejor adaptabilidad gana. Esto se repite tantas veces como el tamaño de la población N . La forma mas usual de este esquema es el torneo binario ($k = 2$), pero puede darse en cualquier expresión con la obvia restricción de $k \leq N$. Si para un esquema de selección por torneo tomamos $k = 1$ (torneo unario), éste se hace similar a una selección aleatoria pues no habrá criterio selectivo alguno mientras que

6.4. OPERADORES DE VARIACIÓN

si $k = N$ el esquema se convierte en una selección radical del mas apto pues solo el individuo con mejor adaptabilidad en la población puede ser escogido.

Muchos estudios analíticos de la selección por torneo han sido propuestos, en ellos se pueden encontrar teoremas y argumentaciones sobre el comportamiento del algoritmo y el tamaño del torneo, la pérdida esperada de diversidad genética, las probabilidades de selección, etc. (ver [2, 7, 38, 21, 4]). Para dar una idea general de las probabilidades de selección en este esquema, considere el caso de torneo binario con reemplazo (posiblemente el más común). Como primero deben seleccionarse aleatoriamente $k = 2$ individuos para seleccionar de allí el mejor, la probabilidad de escoger al peor individuo puede ser calculada considerando los siguientes casos:

Caso 1: El peor individuo es escogido dos veces (reemplazo).

Caso 2: El peor individuo no es escogido.

Caso 3: El peor individuo es escogido primero y el segundo no es el peor.

Caso 4: El peor individuo no es escogido primero, el segundo es el peor

y en una población de tamaño N , las respectivas probabilidades son:

- $Prob(caso_1) = \left(\frac{1}{N}\right) \cdot \left(\frac{1}{N}\right) = \left(\frac{1}{N}\right)^2$
- $Prob(caso_2) = \left(1 - \frac{1}{N}\right) \cdot \left(1 - \frac{1}{N}\right) = \left(1 - \frac{1}{N}\right)^2$
- $Prob(caso_3) = \left(\frac{1}{N}\right) \cdot \left(1 - \frac{1}{N}\right)$
- $Prob(caso_4) = \left(1 - \frac{1}{N}\right) \cdot \left(\frac{1}{N}\right)$

De esta manera, si denotamos P_{mejor} como la probabilidad de seleccionar al mejor individuo en la población, y P_{peor} la probabilidad de seleccionar al peor, tenemos que para cualquier torneo binario con reemplazo:

$$P_{peor} = \left(\frac{1}{N}\right)^2 + 2 \cdot \frac{1}{N} \left(1 - \frac{1}{N}\right) \cdot (1 - P_{mejor}) \quad (6.8)$$

La ecuación 6.8 muestra la probabilidad de seleccionar al peor individuo en la población con selección por torneo binario con reemplazo. Para ver el análisis teórico de otros casos, ver [4] y [21].

6.4. OPERADORES DE VARIACIÓN

Algoritmo 6 Selección por torneo

Entrada: Población P de N individuos, tamaño de torneo k

Salida: Población de individuos seleccionados P'

- 1: **para** $i = 1$ a N **hacer**
 - 2: seleccionar aleatoriamente k individuos de P
 - 3: comparar adaptabilidad
 - 4: agregar ganador a P'
 - 5: **fin para**
-

La figura 6.8 muestra una instancia de torneo binario ($k = 2$). Dos individuos son seleccionados aleatoriamente de la población y se enfrentan entre sí, el de mejor adaptabilidad resulta vencedor.

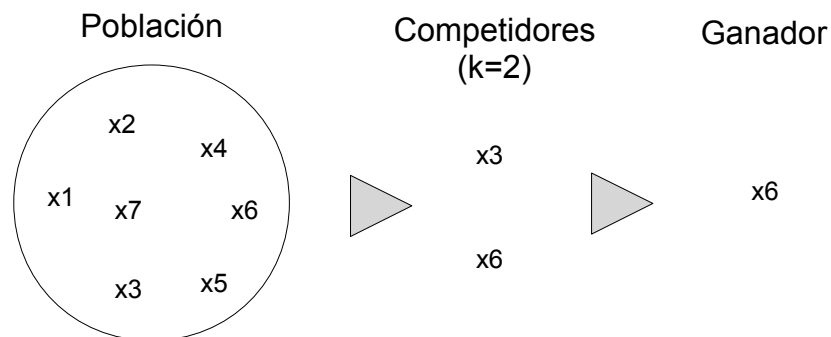


Figura 6.8: Selección por torneo binario

6.4.2. Cruce

El operador de cruce es el responsable de la combinación del material genético de uno o mas padres (según su forma de reproducción; sexual o asexual) en un nuevo individuo. En el AGS, aunque no hay simulación de géneros, la forma de reproducción es sexual, es decir, se generan 2 individuos de la combinación de los genotipos de 2 padres.

6.4. OPERADORES DE VARIACIÓN

Los operadores de cruce mas comúnmente usados por los AGs, son el cruce de 1 punto, el cruce de n puntos y el cruce uniforme. Estos operadores se describen a continuación.

Cruce de 1 punto: Se selecciona aleatoriamente un punto de corte (quiasma) y se generan las crías combinando los segmentos de cromosoma de los padres que separa el quiasma. Se hace obvio que el punto de corte q debe ser aleatoriamente escogido como $1 \leq q \leq N$.

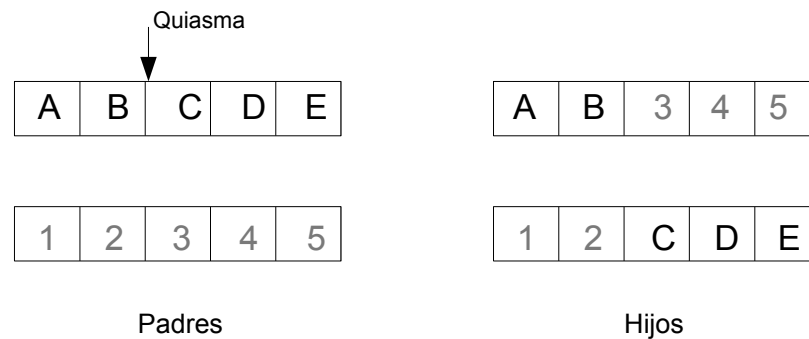


Figura 6.9: Cruce de un punto

La figura 6.9 muestra el cruce de 1 punto para combinar 2 cromosomas en 2 hijos. Los segmentos complementarios de cromosoma separados por el quiasma, forman los nuevos individuos.

Cruce de n puntos: Es similar al cruce de 1 punto pero con n quiasmas. Estos puntos de corte se eligen aleatoriamente y los segmentos complementarios forman los nuevos individuos de acuerdo al criterio del diseñador.

La figura 6.10 da un ejemplo de una posible combinación de cromátidos o segmentos complementarios del cromosoma, para la creación de 2 individuos mediante un operador de cruce de n puntos. Note que otras combinaciones son posibles, la cría resultante será producto del criterio del diseñador.

Cruce uniforme: En el cruce uniforme, una máscara o filtro de probabilidad es colocado sobre uno de los padres y la cría resultante obtiene sus genes en

6.4. OPERADORES DE VARIACIÓN

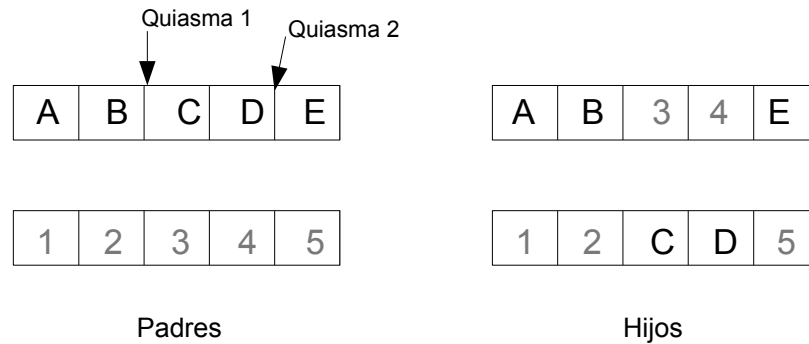


Figura 6.10: Cruce de n puntos

función a este filtro. Por ejemplo, si este filtro fuera de la forma $F_i = 0,5$ para $i = (1, 2, \dots, N)$, entonces cada gen de la cría sería igual al padre 1 con una probabilidad de 50 %.

6.4.3. Mutación

El operador de mutación es el principal operador de variación en algunos algoritmos evolutivos como la *estrategia evolutiva*. En el algoritmo genético la mutación es el principal medio para salir de un estancamiento en un óptimo local. Típicamente, el evento de mutación está regido por un experimento de Bernoulli donde un número aleatorio es disparado y determinará si un locus en el cromosoma de un individuo en específico, mutará o no dependiendo de una probabilidad determinada P_{mut} .

En el AGS, todos los locus de todos los individuos son sometidos a este escenario de Bernoulli, es decir, gen a gen se saca un número aleatorio de una distribución uniforme en el intervalo $(0 \dots 1)$ y se compara con la probabilidad de mutación P_{mut} . Si el número es menor o igual se considera el ensayo positivo y se procede a mutar al gen, de lo contrario se evalúa el siguiente. En un AGS con representación binaria, con una población P de N individuos, en cualquier momento t representado como P_n^t , cada individuo O_i con $i = (1, 2, \dots, N)$ es una cadena de un alfabeto binario y:

$$P_n^t = \{O_1^t, O_2^t, \dots, O_n^t\}$$

6.4. OPERADORES DE VARIACIÓN

si j representa un locus específico en O_i , entonces el esquema de mutación puede ser descrito como:

$$O_{ij}^t = \begin{cases} 1 - O_{ij}^t & : \text{ con probabilidad } P_{mut} \\ O_{ij}^t & : \text{ con probabilidad } 1 - P_{mut} \end{cases}$$

Para cromosomas con representación no binaria el esquema de mutación debe ser diseñado de acuerdo a la naturaleza de la representación. Por ejemplo, en el caso del vendedor viajero (ver sección 6.5 en la página 65), una forma de mutación pudiera ser intercambiar dos letras en el cromosoma, cambiando así la ruta del circuito.

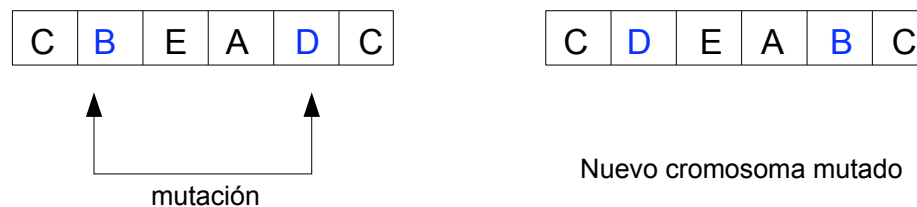


Figura 6.11: Mutación simple

La figura 6.11 muestra una forma de mutación simple en un cromosoma no binario. A la izquierda la solución representada en el cromosoma es un circuito $C \rightarrow B \rightarrow E \rightarrow A \rightarrow D \rightarrow C$, comparando un número aleatorio $a \in (0 \dots 1)$ con P_{mut} , si $a \leq P_{mut}$ se considera un evento de Bernoulli positivo y hay mutación. Si el cromosoma es binario se invierte el valor del gen a mutar, si como en este caso, el cromosoma no es binario y el orden de los símbolos tiene efecto sobre el fenotipo (se altera la distancia recorrida), una forma de mutación es simplemente elegir otra posición del cromosoma al azar e intercambiar los valores. A la izquierda de la figura se ve el cromosoma mutado con un intercambio de nodos en el circuito representado.

*Lo escucho y lo olvidaré. Lo veo y lo recordaré.
Lo hago y lo entenderé.*

Kong-fu-tse (Confucio)

7

Experimentación

7.1 Introducción

En este capítulo se presentan algunas experiencias de laboratorio en el uso de los algoritmos genéticos con la intención de guiar o señalar directrices en la planificación y diseño de experimentos, así como en el análisis de los resultados que permitan concluir o al menos provean bases para discutir un planteamiento específico, por ejemplo, una hipótesis sobre rendimiento, una conclusión sobre cómo se comparan dos observaciones, etc.

Una interrogante muy común en el trabajo experimental es si alguna modificación o extensión a un AGS, sus parámetros, o elementos funcionales, produce “mejores” resultados que una observación previa, en otras palabras, hemos resuelto un problema con un AGS pero deseamos saber si un cambio en la estructura del algoritmo o en alguno de sus operadores puede llevar a una mejoría en el rendimiento. Otras interrogantes comunes en la experimentación son aquellas basadas en la necesidad de entender mejor el principio funcional de los componentes del objeto de estudio. Por ejemplo, en los AGS, una práctica común es diseñar experimentos que permitan observar el comportamiento de sus principales operadores por separado, es decir,

tomar estos operadores y analizarlos de manera independiente buscando entender mejor su comportamiento. Es muy importante tener en cuenta a la hora de escribir conclusiones, que por la naturaleza estocástica del AG, se hace impropio la generalización de lo concluido al universo de los AG. En la medida que la batería de experimentos este “mejor” diseñada, se podrán construir conclusiones mas precisas, pero siempre recordando que no podemos generalizar al universo de AG's. Estas experiencias ayudan a descubrir algunas “reglas rápidas” de manera informal.

7.1.1. Consideraciones Importantes

A la hora de diseñar y planear la batería de experimentos en busca de argumentos y conclusiones sobre algún evento, es importante tener en cuenta lo que se persigue, de manera de poder reportar los resultados con base científica. Para lograr esto hay 2 elementos principales que deben tenerse en cuenta.

En el diseño de toda fase experimental, se debe tener en cuenta que esta sea:

Completa: Esto significa que todos los constructores principales de los experimentos deben ser informados y aclarados. El juego de parámetros usados, los criterios de decisión, etc. Por ejemplo, en una experimentación con algoritmos genéticos es muy importante informar el tamaño de población usada (si éste fue fijo, si fue variable debe informarse en detalle el criterio de variación), al igual que el resto de los parámetros, pero la semilla usada para el generador aleatorio es irrelevante, porque los experimentos deben estar diseñados de manera de medir tendencias estadísticas generales independientes de sus semillas aleatorias. El criterio fundamental a tener en cuenta en el reporte de la experimentación es que éste es completo si otro investigador es capaz de reproducir el mismo experimento solo con la información dada.

Precisa: Esto se refiere a que los experimentos diseñados deben ser coherentes con las inquietudes o hipótesis que se estudian, que el diseño de los experimentos es razonablemente acertado como para leer resultados de tendencias estadísticas generales, y que lo leído y reportado del experimento esté de acuerdo con los resultados obtenidos. Si la tendencia estadística de un mismo evento varía significativamente al usar semillas aleatorias diferentes, lo mas probable es que la planeación y diseño de la experimentación no haya sido precisa.

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

El resultado de una investigación puede despertar el interés de otros investigadores en el área a repetir los experimentos, cosa que no lograrán si la información dada no es *completa*, y al repetir los experimentos las posibles refutaciones o críticas generalmente serán de los controles experimentales usados, criterios de medición, u otros elementos que afectan la *precisión* de lo reportado.

7.2 Comparando métodos de selección

El operador de selección en un AG es el encargado de “guiar” el proceso evolutivo según las exigencias del ambiente (medidas por la función de evaluación). Este operador es fundamental en el tiempo de convergencia del algoritmo y la diversidad genética poblacional.

7.2.1. Diseño experimental:

El primer paso es escoger los diferentes métodos de selección a ser estudiados y comparados. Esto puede obedecer a diferentes necesidades del investigador y la justificación de la escogencia debe ser discutida en términos de estas necesidades, esta discusión forma parte de la precisión del diseño. Para nuestro caso, usaremos el operador de selección usado en el Algoritmo Genético Simple original propuesto por Holland: la selección por rueda de ruleta, y uno de los más usados en la actualidad por su velocidad de convergencia y su capacidad de influir en la *presión selectiva* como lo es la selección por torneo[36]. Igualmente, se toman métodos de selección absolutos para resaltar las comparaciones. En resumen, los métodos escogidos para formar la base de los experimentos son:

- Rueda de Ruleta
- Torneo binario (k=2)
- Torneo de 3 (k=3)
- Torneo de 4 (k=4)
- Selección aleatoria (un individuo al azar)

Luego, como parte de la precisión del diseño, debemos discutir los criterios de estudio que formarán la base de comparación. Como se planteó anteriormente, la

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

principal tarea del operador de selección es “guiar” el proceso evolutivo. Esta guía debe establecer un balance entre la *presión selectiva* que informalmente se refiere a la relación entre la máxima adaptabilidad a la adaptabilidad promedio de la población, o en otras palabras, a la tendencia del método a seleccionar al individuo mejor adaptado para las siguientes generaciones, y la diversidad genética o varianza en el banco genético poblacional.

Para estimar, medir o cuantificar estos criterios, se implementan dos métricas de rendimiento como se describen a continuación:

M1: Diversidad genética. El promedio de individuos diferentes por generación, en la población.

M2: Tiempo de convergencia. El número de generaciones en promedio que le toma al individuo mejor adaptado, apoderarse del banco genético de la población.

7.2.2. Descripción de los experimentos

Para cada uno de los métodos de selección mencionados, se genera una población inicial aleatoria P_1 (con cromosoma binario) y se somete repetidas veces al método de selección. Ningún otro operador es usado de manera de observar solo los cambios introducidos por la selección. La población P_2 es obtenida de aplicar el operador selección tantas veces como individuos haya en la población, y así sucesivamente:

$$P_{i+1} = Sel_{j=1}^{|P|}(P_i)$$

donde $i = (1, 2, \dots, N - 1)$, N es el número de generaciones, $j = (1, 2, \dots, |P|)$ y $|P|$ el tamaño de la población medido en número de individuos.

El tamaño de la población o $|P|$ varía de 50 a 100 individuos y el tamaño del cromosoma binario varía de 100 a 200 genes. Todos los experimentos se totalizan sobre un total de 50 corridas experimentales.

Para determinar la adaptabilidad de cada individuo se genera un esquema binario aleatorio que se supone el óptimo global del problema. Para evitar posibles ambigüedades en el uso de la distancia de Hamming, se convierte la representación

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

binaria de cada individuo a un número decimal y su adaptabilidad es el inverso de la distancia a la representación decimal del esquema binario dado.

$$Adapt_i = \frac{1}{|Optimo - Valor_i|}$$

donde *Optimo* es el valor en decimal del cromosoma binario dado como solución y *Valor_i* es el valor en decimal del *i*-ésimo individuo.

7.2.3. Resultados experimentales

Una forma común de presentar los resultados experimentales es comentar someramente las pruebas por aislado y luego haciendo los análisis comparativos respectivos. El uso de imágenes es muy útil para mostrar tendencias o comportamientos genéricos, no obstante, es aconsejable complementar las imágenes con información mas precisa y detallada, que aunque pueda parecer innecesaria, permita al lector interesado en hacer un escrutinio minucioso de los experimentos, realizar sus propias pruebas a partir de los datos brindados.

Para el este ensayo experimental, cada uno de los métodos de selección escogidos (Rueda de ruleta, Selección aleatoria, Torneo binario, Torneo de 3 y Torneo de 4) se implementaron con las combinaciones de parámetros del cuadro 7.1.

La figura 7.1 muestra las pruebas experimentales sobre el método de selección de rueda de ruleta para 50 y 100 individuos con longitud de cromosoma de 100 y 200 genes. Las imágenes en la columna izquierda muestran el número de individuos diferentes por generación promediados en 50 corridas experimentales. Las imágenes de la columna derecha muestran el número de generaciones necesarias por corrida experimental, para la convergencia del banco genético de la población a un único genotipo. Para ambas columnas, de arriba a abajo se tiene: tamaño de población $P = |50|$ y longitud de cromosoma $L_C = 100$, segunda fila: $P = |50|$, $L_C = 200$, tercera fila: $P = |100|$, $L_C = 100$ y última fila: $P = |100|$, $L_C = 200$. Las líneas horizontales de las imágenes de la columna derecha, representan el promedio de generaciones necesarias para la convergencia de la población a un solo genotipo, es

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

decir, el tiempo generacional promedio de toma del banco genético por el individuo mejor adaptado. En la columna derecha, de arriba a abajo, esta la media poblacional: $\mu = 10$, desviación típica de la población: $\sigma = 9,25$, segunda fila: $\mu = 11,54$, $\sigma = 10,95$, tercera fila: $\mu = 10,6$, $\sigma = 10,81$, última fila: $\mu = 12,06$, $\sigma = 11,08$. Observe los ocasionales “picos” en las imágenes de la columna derecha, esto es debido a la naturaleza estocástica del algoritmo, note que en las dos imágenes inferiores (donde $|P| = 100$) se observa un mayor número de estos picos, a diferencia de las dos imágenes superiores donde $|P| = 50$.

La figura 7.2 muestra los resultados experimentales con el método de selección de torneo binario ($k = 2$). Al igual que en la figura 7.1, en la columna izquierda la abscisa representa las generaciones o épocas evolutivas, y la ordenada el número promedio de individuos diferentes por generación. En la columna derecha el eje x representa las corridas experimentales y el eje y las generaciones necesarias para la convergencia de la población a un genotipo único. Los valores de la media poblacional y desviación típica poblacional, de arriba a abajo son: $\mu = 8,3$, $\sigma = 1,4$, segunda fila: $\mu = 8,52$, $\sigma = 1,71$, tercera fila: $\mu = 9,46$, $\sigma = 1,54$, última fila: $\mu = 9,32$, $\sigma = 1,1$. Observe que a diferencia del comportamiento del método de selección de Rueda de ruleta, la curva de las imágenes en la columna izquierda tienen una caída menos empinada pero que alcanza el nivel 0 (promedio de individuos diferentes igual a cero), en menos generaciones. Igualmente en las imágenes de la columna derecha se aprecia cómo la convergencia es más “estable” en el sentido que la desviación típica de la población es más pequeña.

Las figuras 7.3 y 7.4 muestran las corridas experimentales de los métodos de selección de Torneo de tamaños 3 y 4 respectivamente. Se puede apreciar como la curva en las imágenes de la columna izquierda de las figuras va acentuando su caída, es decir, se va haciendo más empinada y tanto el promedio como la desviación típica poblacional en las imágenes de la columna derecha (métrica M2) van haciéndose más pequeñas. Comparando este comportamiento con el método de selección aleatoria mostrado en la figura 7.5, vemos que en este último, la curva es mucho menos empinada y no llega a hacerse 0, es decir, para el número de generaciones o épocas evolutivas usadas, ningún individuo se apodera del banco genético poblacional. Esto pareciera ser muy ventajoso puesto que mantiene la diversidad genética alta, no obstante, esto también significa que el método no converge, y la razón por la que no se muestran las gráficas de M2 es porque en ningún momento del proceso evolutivo se hizo homogéneo el banco genético.

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

Las figuras 7.6 y 7.7 muestran lado a lado el comportamiento de los cinco métodos de selección usados bajo las estimaciones de las métricas M1 y M2. Observe como a medida que el tamaño de torneo aumenta en la figura 7.6, el número promedio de generaciones para convergencia a un genotipo, disminuye. Similarmente, en la figura 7.7, mientras mas grande el tamaño del torneo, mas empinada es la curva de M1, es decir, mas rápidamente se va perdiendo la diversidad genética.

Tamaño Polbación ($ P $)	Tamaño Cromosoma (L_C)
50	100
50	200
100	100
100	200
Generaciones : 50	Experimentos: 50

Cuadro 7.1: Parámetros experimentales

Análisis de resultados

En el análisis de resultados se evalúan las métricas propuestas y se aplican los estadísticos pertinentes que permitan la formulación de conclusiones coherentes con lo analizado y con rigurosidad científica. Independientemente de la forma o estructura en la que se presente el trabajo experimental, es necesaria una presentación de resultados y sus estimaciones según las métricas propuestas.

Para los experimentos realizados se propusieron dos métricas de evaluación del rendimiento, M1 y M2. Los cuadros 7.5 y 7.6 muestran los valores obtenidos para estas métricas respectivamente, para todos los escenarios experimentales planteados. Observe que las diferencias en las medias de las observaciones para todos los métodos de selección cuando varía la longitud del cromosoma L_C , parecieran no ser muy marcadas. Esta relación es consistente con todos los experimentos hechos.

Para evaluar la validez de esta relación es necesario aplicar alguna prueba estadística que permita concluir con algún nivel de confianza sobre las diferencias relativas. Es importante destacar que todos estos procedimientos no demuestran nada de manera absoluta y rigurosa, es decir, no se puede asegurar (científicamente

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

hablando) que lo encontrado es una verdad irrefutable, es tan solo el respaldo fundamentado de que esa es la tendencia estadística observada, descartando posibles errores de observación o ruido fortuito bajo algún nivel de confianza.

Para determinar si hay diferencias significativas en el rendimiento de los métodos de selección causadas por la longitud de cromosoma (L_C) según los criterios evaluados en las métricas **M1** y **M2**, aplicamos para cada una, una prueba t de Student apareada, donde se evaluará el resultado obtenido en cada método de selección con cromosomas de 100 y 200 genes. Solo la longitud del cromosoma variará para las dos observaciones de manera de medir con mayor precisión los efectos de L_C sobre mas dos métricas y corroborar o corregir el comportamiento observado en las gráficas.

Los cuadros 7.7 y 7.8 muestran los resultados del estadístico t de Student para las métricas M1 y M2 respectivamente. El nivel de confianza usado para todas las pruebas es de 95 %, este nivel de confianza es ampliamente usado en las ciencias sociales y goza también de amplia popularidad en las ciencias tecnológicas. El resultado del estadístico aplicado sobre M1 indica que la diferencia entre las medias observadas no es significativa mientras que para M2 si lo es.

Finalmente, el cuadro 7.9 muestra en resumen la comparación en ambas métricas de los diferentes métodos de selección comparados dos a dos. Las pruebas t realizadas fueron apareadas, buscando medir la diferencia en medias tomando en cuenta lo que un algoritmo obtuvo solo al variar la variable de medición de la métrica respectiva. Para todos los casos, la diferencia fue significativa.

M1	$ P = 50$		$ P = 100$	
	$L_C = 100$	$L_C = 200$	$L_C = 100$	$L_C = 200$
Ruleta	1.68	1.71	3.10	3.23
Aleatorio	6.08	6.08	12.9	12.9
Torneo 2	2.35	2.35	4.85	4.78
Torneo 3	1.84	1.84	3.76	3.80
Torneo 4	1.62	1.62	3.35	3.34

Cuadro 7.5: M1 para los diferentes métodos de selección

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

M2	P = 50		P = 100	
	$L_C = 100$	$L_C = 200$	$L_C = 100$	$L_C = 200$
Ruleta	10	11.54	10.6	12.06
Aleatorio	48.54	48.54	50	50
Torneo 2	8.3	8.52	9.46	9.32
Torneo 3	5.5	5.28	6.38	6.42
Torneo 4	4.46	4.48	5.2	5.08

Cuadro 7.6: M2 para los diferentes métodos de selección

M1	
Prueba :	t -Student (apareada)
Nivel de confianza :	95 %
Grados de libertad :	9
Error estándar :	0.016
t :	0.7513
P(2 colas) :	0.4717
Observación: Diferencia no significativa	

Cuadro 7.7: Prueba t -Student para diferencias en L_C (M1).

M2	
Prueba :	t -Student (apareada)
Nivel de confianza :	95 %
Grados de libertad :	9
Error estándar :	0.250
t :	2.2875
P(2 colas) :	0.048
Observación: Diferencia significativa	

Cuadro 7.8: Prueba t -Student para diferencias en L_C (M2).

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

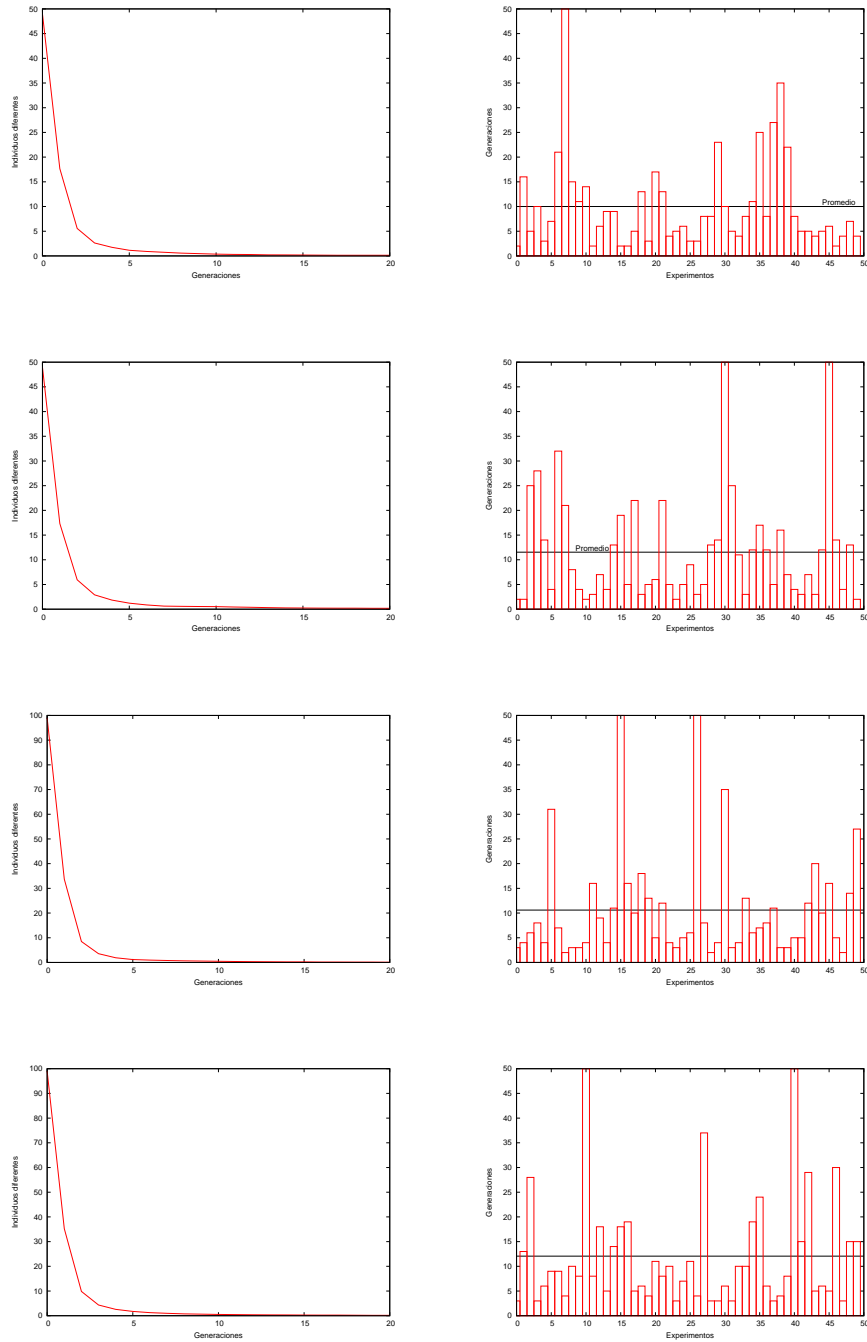


Figura 7.1: M1 (izquierda) y M2 (derecha) para selección por Rueda de Ruleta

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

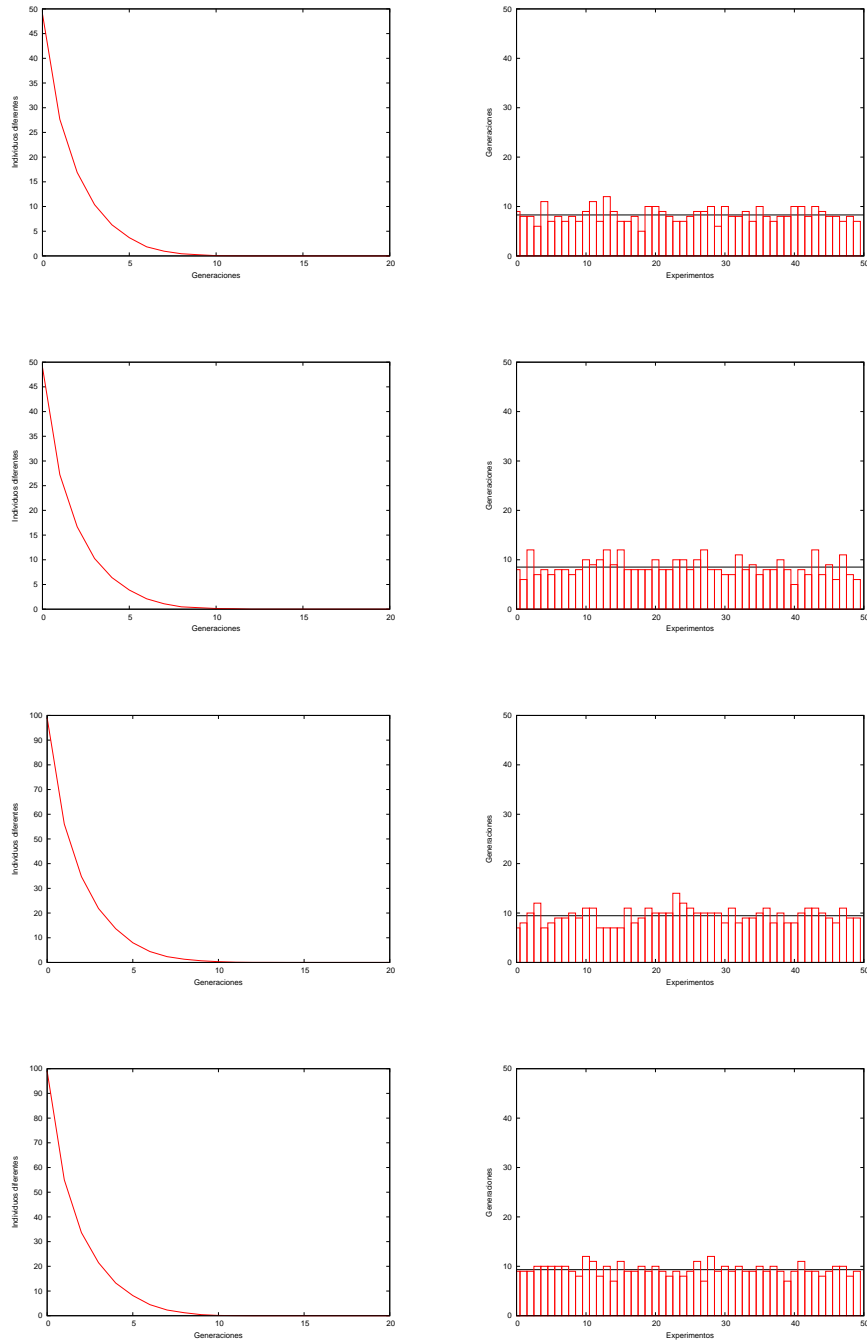


Figura 7.2: Pruebas experimentales con Torneo binario. M1 (izquierda) M2 (derecha)

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

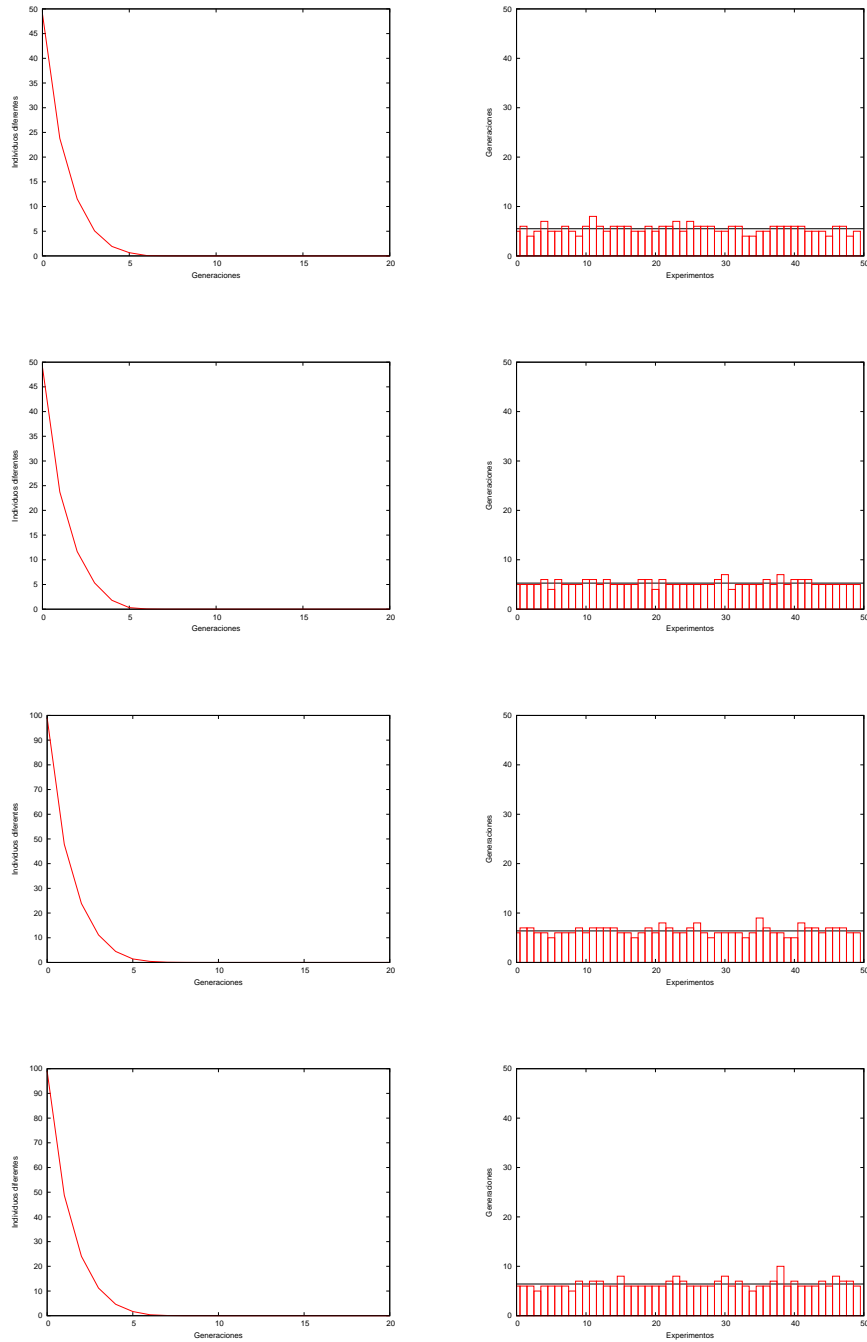


Figura 7.3: Pruebas experimentales con Torneo de 3. M1(izquierda), M2(derecha).

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

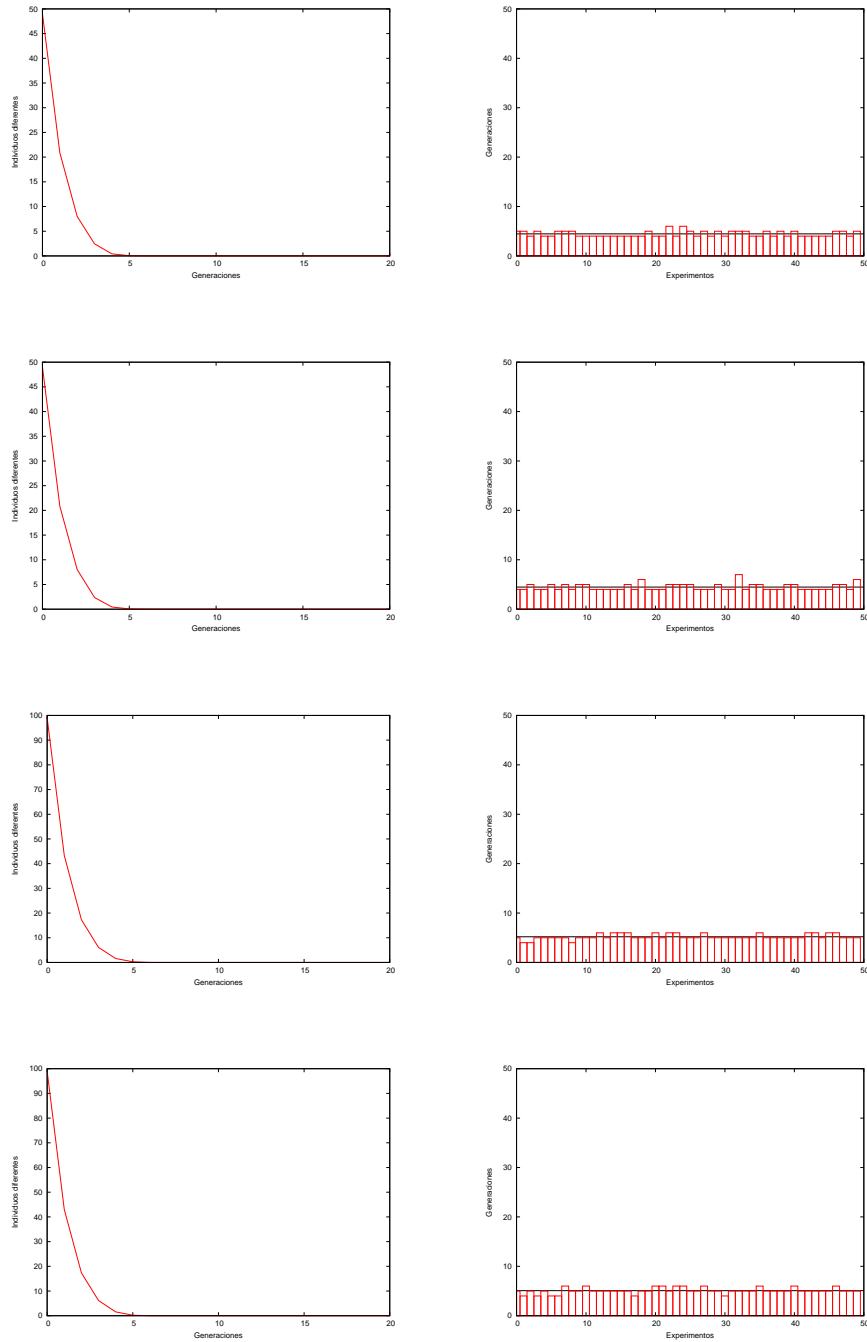


Figura 7.4: Pruebas experimentales con Torneo de 4. M1(izquierda), M2(derecha)

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

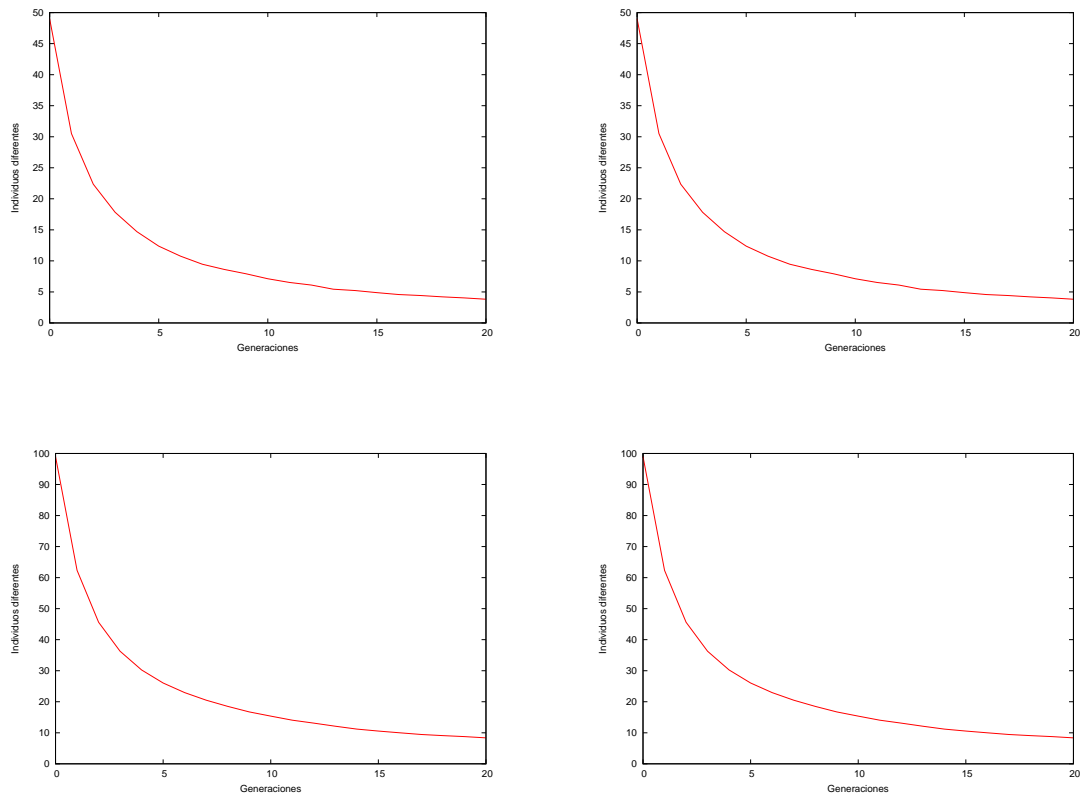


Figura 7.5: Pruebas experimentales con el método de selección aleatoria

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

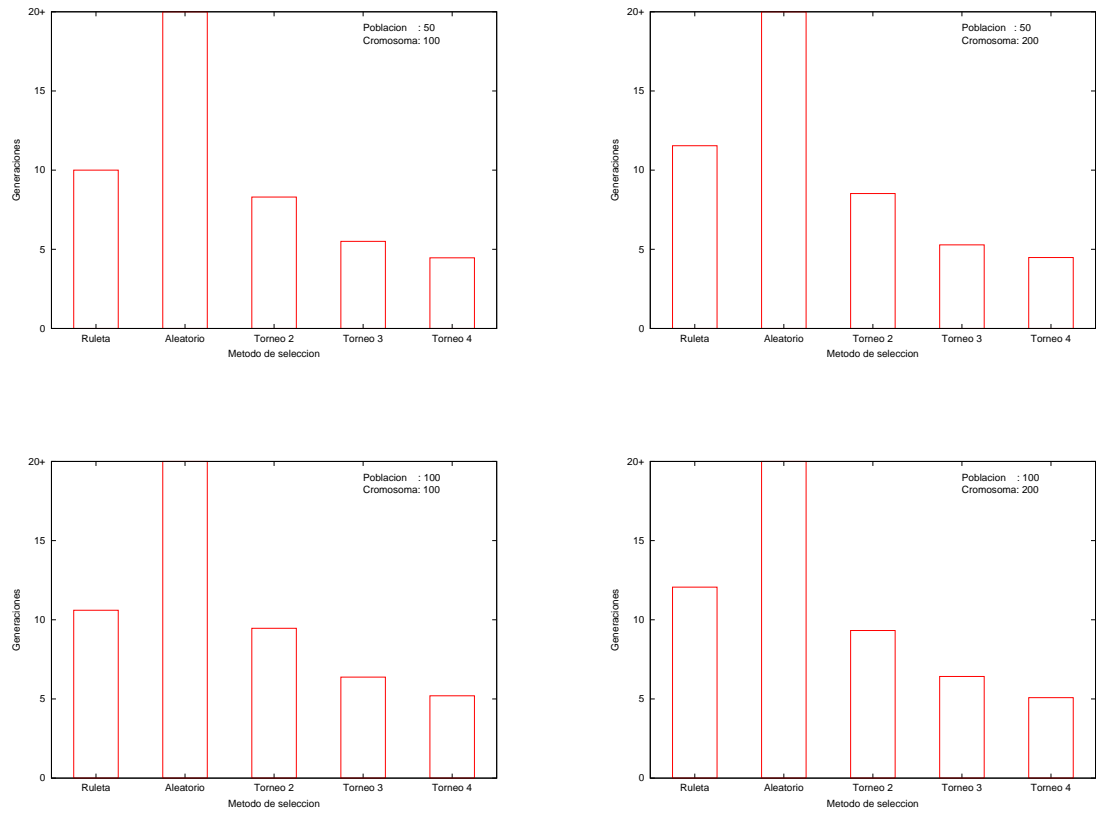


Figura 7.6: Cuadro comparativo métodos de selección para M2

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

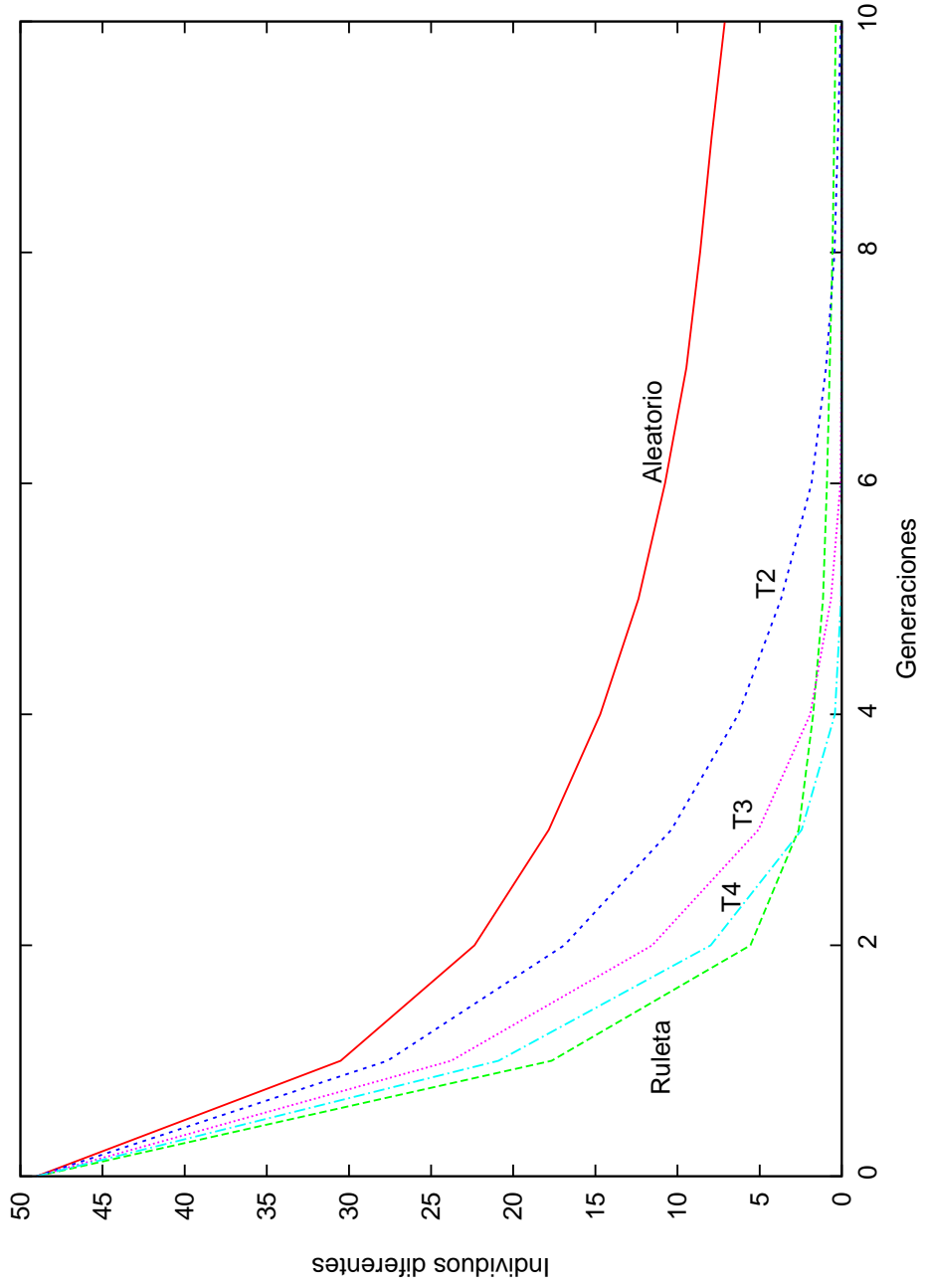


Figura 7.7: Promedio de individuos diferentes por generación por método de selección (M1)

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

$ P = 50, L_C = 100$		$ P = 50, L_C = 100$	
M1	$\mu = 1,68$ $\sigma = 7,31$	M1	$\mu = 6,08$ $\sigma = 8,33$
M2	$\mu = 10$ $\sigma = 9,34$	M2	$\mu = 48,54$ $\sigma = 3,76$
$ P = 50, L_C = 200$		$ P = 50, L_C = 200$	
M1	$\mu = 1,71$ $\sigma = 7,29$	M1	$\mu = 6,08$ $\sigma = 8,33$
M2	$\mu = 11,54$ $\sigma = 11,06$	M2	$\mu = 48,54$ $\sigma = 3,76$
$ P = 100, L_C = 100$		$ P = 100, L_C = 100$	
M1	$\mu = 3,1$ $\sigma = 14,67$	M1	$\mu = 12,9$ $\sigma = 16,8$
M2	$\mu = 10,6$ $\sigma = 10,92$	M2	$\mu = 50$ $\sigma = 0$
$ P = 100, L_C = 200$		$ P = 100, L_C = 200$	
M1	$\mu = 3,23$ $\sigma = 14,74$	M1	$\mu = 12,9$ $\sigma = 16,8$
M2	$\mu = 12,06$ $\sigma = 11,19$	M2	$\mu = 50$ $\sigma = 0$

(a) Rueda de ruleta

(b) Selección aleatoria

Cuadro 7.2: Resultados de M1y M2 para Rueda de ruleta y selección aleatoria.

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

$ P = 50, L_C = 100$	$ P = 50, L_C = 100$								
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">M1</td> <td style="padding-left: 10px;">$\mu = 2,35$ $\sigma = 8,17$</td> </tr> <tr> <td style="text-align: center;">M2</td> <td style="padding-left: 10px;">$\mu = 8,3$ $\sigma = 1,4$</td> </tr> </table>	M1	$\mu = 2,35$ $\sigma = 8,17$	M2	$\mu = 8,3$ $\sigma = 1,4$	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">M1</td> <td style="padding-left: 10px;">$\mu = 1,84$ $\sigma = 7,69$</td> </tr> <tr> <td style="text-align: center;">M2</td> <td style="padding-left: 10px;">$\mu = 5,5$ $\sigma = 0,85$</td> </tr> </table>	M1	$\mu = 1,84$ $\sigma = 7,69$	M2	$\mu = 5,5$ $\sigma = 0,85$
M1	$\mu = 2,35$ $\sigma = 8,17$								
M2	$\mu = 8,3$ $\sigma = 1,4$								
M1	$\mu = 1,84$ $\sigma = 7,69$								
M2	$\mu = 5,5$ $\sigma = 0,85$								
$ P = 50, L_C = 200$	$ P = 50, L_C = 200$								
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">M1</td> <td style="padding-left: 10px;">$\mu = 2,35$ $\sigma = 8,14$</td> </tr> <tr> <td style="text-align: center;">M2</td> <td style="padding-left: 10px;">$\mu = 8,52$ $\sigma = 1,71$</td> </tr> </table>	M1	$\mu = 2,35$ $\sigma = 8,14$	M2	$\mu = 8,52$ $\sigma = 1,71$	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">M1</td> <td style="padding-left: 10px;">$\mu = 1,84$ $\sigma = 7,7$</td> </tr> <tr> <td style="text-align: center;">M2</td> <td style="padding-left: 10px;">$\mu = 5,28$ $\sigma = 0,63$</td> </tr> </table>	M1	$\mu = 1,84$ $\sigma = 7,7$	M2	$\mu = 5,28$ $\sigma = 0,63$
M1	$\mu = 2,35$ $\sigma = 8,14$								
M2	$\mu = 8,52$ $\sigma = 1,71$								
M1	$\mu = 1,84$ $\sigma = 7,7$								
M2	$\mu = 5,28$ $\sigma = 0,63$								
$ P = 100, L_C = 100$	$ P = 100, L_C = 100$								
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">M1</td> <td style="padding-left: 10px;">$\mu = 4,85$ $\sigma = 16,57$</td> </tr> <tr> <td style="text-align: center;">M2</td> <td style="padding-left: 10px;">$\mu = 9,46$ $\sigma = 1,54$</td> </tr> </table>	M1	$\mu = 4,85$ $\sigma = 16,57$	M2	$\mu = 9,46$ $\sigma = 1,54$	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">M1</td> <td style="padding-left: 10px;">$\mu = 3,76$ $\sigma = 15,55$</td> </tr> <tr> <td style="text-align: center;">M2</td> <td style="padding-left: 10px;">$\mu = 6,38$ $\sigma = 0,85$</td> </tr> </table>	M1	$\mu = 3,76$ $\sigma = 15,55$	M2	$\mu = 6,38$ $\sigma = 0,85$
M1	$\mu = 4,85$ $\sigma = 16,57$								
M2	$\mu = 9,46$ $\sigma = 1,54$								
M1	$\mu = 3,76$ $\sigma = 15,55$								
M2	$\mu = 6,38$ $\sigma = 0,85$								
$ P = 100, L_C = 200$	$ P = 100, L_C = 200$								
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">M1</td> <td style="padding-left: 10px;">$\mu = 4,78$ $\sigma = 16,46$</td> </tr> <tr> <td style="text-align: center;">M2</td> <td style="padding-left: 10px;">$\mu = 9,32$ $\sigma = 1,1$</td> </tr> </table>	M1	$\mu = 4,78$ $\sigma = 16,46$	M2	$\mu = 9,32$ $\sigma = 1,1$	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">M1</td> <td style="padding-left: 10px;">$\mu = 3,8$ $\sigma = 15,61$</td> </tr> <tr> <td style="text-align: center;">M2</td> <td style="padding-left: 10px;">$\mu = 6,42$ $\sigma = 0,87$</td> </tr> </table>	M1	$\mu = 3,8$ $\sigma = 15,61$	M2	$\mu = 6,42$ $\sigma = 0,87$
M1	$\mu = 4,78$ $\sigma = 16,46$								
M2	$\mu = 9,32$ $\sigma = 1,1$								
M1	$\mu = 3,8$ $\sigma = 15,61$								
M2	$\mu = 6,42$ $\sigma = 0,87$								
(a) Torneo con $k = 2$	(b) Torneo con $k = 3$								

Cuadro 7.3: M1y M2 para Selección de Torneo $k = 2, k = 3$.

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

$ P = 50, L_C = 100$		
M1	$\mu = 1,68$	$\sigma = 7,31$
M2	$\mu = 10$	$\sigma = 9,34$

$ P = 50, L_C = 200$		
M1	$\mu = 1,71$	$\sigma = 7,29$
M2	$\mu = 11,54$	$\sigma = 11,06$

$ P = 100, L_C = 100$		
M1	$\mu = 3,1$	$\sigma = 14,67$
M2	$\mu = 10,6$	$\sigma = 10,92$

$ P = 100, L_C = 200$		
M1	$\mu = 3,23$	$\sigma = 14,74$
M2	$\mu = 12,06$	$\sigma = 11,19$

Cuadro 7.4: M1 y M2 para Selección de Torneo con $k = 4$

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

Métrica	Métodos de selección	t	Error estándar	P(2 colas)	Observación
M1	Ruleta-Torneo2	$t = 3,9716$	$\varepsilon = 0,291$	$P = 0,0285$	Δ significativa
	Torneo2 -Torneo3	$t = 5,0421$	$\varepsilon = 0,153$	$P = 0,015$	Δ significativa
	Torneo3 - Torneo4	$t = 5,2068$	$\varepsilon = 0,063$	$P = 0,0138$	Δ significativa
M2	Ruleta-Torneo2	$t = 4,8817$	$\varepsilon = 0,44$	$P = 0,0164$	Δ significativa
	Torneo2 -Torneo3	$t = 30,844$	$\varepsilon = 0,097$	$P = 0,0001$	Δ significativa
	Torneo3-Torneo4	$t = 9,5234$	$\varepsilon = 0,114$	$P = 0,0025$	Δ significativa
Nivel de confianza: 95 % - Grados de Libertad: 4					

Cuadro 7.9: Cuadro comparativo de métodos de selección bajo t -Student sobre las métricas M1 y M2

7.2. COMPARANDO MÉTODOS DE SELECCIÓN

Conclusiones

Las pruebas realizadas a los experimentos nos permiten concluir bajo un cierto nivel de confianza, algunas relaciones de rendimiento desde el enfoque de los criterios de las métricas establecidas. Es importante reportar los resultados obtenidos separando lo observado, de las inferencias y sesgos propios, esto último puede incluirse en una sección de discusión o recomendaciones donde el uso de argumentos especulativos no afecta los hallazgos hechos y pueden promover un debate que enriquezca la actividad académica.

De lo experimentado podemos concluir que en términos del número promedio de individuos diferentes por generación (**M1**), no es determinante la longitud del cromosoma L_C en los resultados promedios de los diferentes métodos. Las pruebas estadísticas señalan que la diferencia entre las mediciones de M1 para todos los experimentos, no fue estadísticamente significativa. Esto sugiere que para los escenarios experimentales planteados, la longitud del cromosoma no es un factor crucial en la variedad genética en la población.

Otra relación sobre la que las pruebas nos permiten concluir, es que tanto la variedad genética como la velocidad de convergencia son sensibles al tipo de selección usada y para el caso del método de selección por torneo, el tamaño de éste influye de manera significativa sobre estas variables. Mientras mas grande es el tamaño del torneo, menos probabilidad tiene un individuo no muy apto a ser escogido de la población para formar parte de las siguientes generaciones, en otras palabras, el tamaño de torneo k influye sobre la presión selectiva que el algoritmo aplica, y esto a su vez influye tanto en la variedad genética de la población como en la velocidad de convergencia a un genotipo único.

7.3 Comparando métodos de cruce

Los experimentos de esta sección están orientados a analizar algunos métodos de cruce en cuanto a su capacidad de producir soluciones dispersas en el espacio representado por los individuos cruzados. Algunos métodos de cruce se pueden configurar para sesgar la herencia del material genético recombinado hacia uno de los padres. Otros métodos presentan un comportamiento estadísticamente más balanceado al llevar a la descendencia un relativo equilibrio de genes heredados de los padres. El fin de la siguiente experimentación es observar cuán uniforme un método de cruce distribuye la descendencia en el espacio acotado por los padres.

7.3.1. Diseño experimental

Para estos experimentos se toman dos individuos con representaciones binarias opuestas, es decir, dos individuos con genotipos contrarios y se cruzan repetidamente entre sí para observar la distribución de las soluciones obtenidas del cruce. Los métodos analizados, tipo de representación y longitud de cromosoma se muestran en el cuadro 7.10.

Método de cruce	Tipo de representación	Longitud de cromosoma
Un punto	Binaria simple	32
Dos puntos	Binaria simple	32
Uniforme	Binaria simple	32

Cuadro 7.10: Métodos de cruce a ser analizados

Los individuos son representados mediante dos cromosomas opuestos de 32 genes cada uno, es decir, el padre 1 tiene el genotipo [1111111111...11111] y el padre 2 tiene el genotipo [0000000000...00000]. El fenotipo es un número real en el intervalo [0,1] y se calcula llevando el cromosoma a un número entero como indica la ecuación 7.2 y este valor a su equivalente real según la ecuación 7.3. Ambos individuos se cruzan repetidas veces y se analiza la distribución de los valores obtenidos en distancia Hamming a cualquiera de los padres ya que:

$$\text{distancia_Hamming}_{(x_1, \text{padre}_2)} = L_C - \text{distancia_Hamming}_{(x_1, \text{padre}_1)}$$

7.3. COMPARANDO MÉTODOS DE CRUCE

Si los valores fueron uniformemente distribuidos, las distancias de Hamming de los individuos obtenidos a cualquier padre, ejemplo, el padre₁ siguen una distribución normal con media $\mu \approx L_C/2$.

Método de cruce	Un punto
Tipo de representación	Binaria simple
Número de cruces	32
Distancia Media (\mathbb{R})	0.5
Desviación típica	0.491827
Distancia Hamming promedio	16
Desviación típica	9.87420
Longitud de cromosoma (L_C)	32

Cuadro 7.11: Datos experimentales para cruce de 1 punto

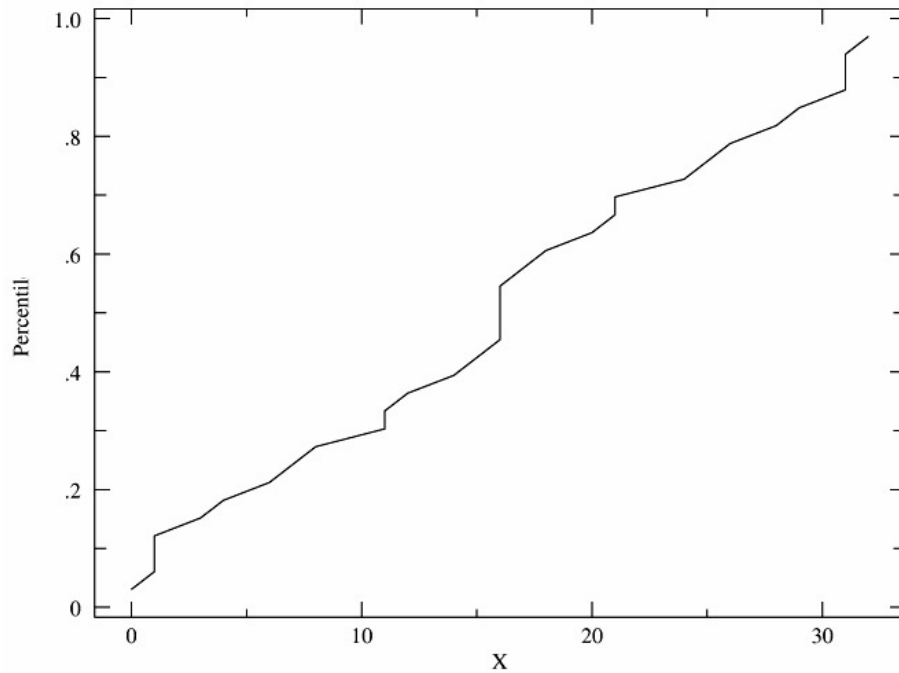


Figura 7.8: Percentiles para cruce de 1 punto

7.3. COMPARANDO MÉTODOS DE CRUCE

Prueba Kolmogorov-Smirnov para cruce de un punto	
Mediana	16.0
Valor tope superior	32.0
Valor tope inferior	0.0
(K-S) P	0.99
Confianza	95 %
Prueba K-S consistente con distribución normal	

Cuadro 7.12: Prueba Kolmogorov-Smirnov para cruce de 1 punto

La figura 7.8 muestra un gráfico de percentiles para el cruce de un punto. Note que la mediana (16) se encuentra en el percentil .50. Note también el comportamiento de pseudo línea recta ascendente, como debería comportarse una observación extraída de una distribución normal. El cuadro 7.12 muestra los resultados de la prueba Kolmogorov-Smirnov donde el estadístico calculado muestra una fuerte consistencia con la distribución normal ($P = 0.99$).

En la figura 7.9 se muestra a la izquierda la distribución de frecuencia acumulada (distancias de Hamming) para 32 individuos y a la derecha el gráfico de percentiles para el método de cruce de 2 puntos.

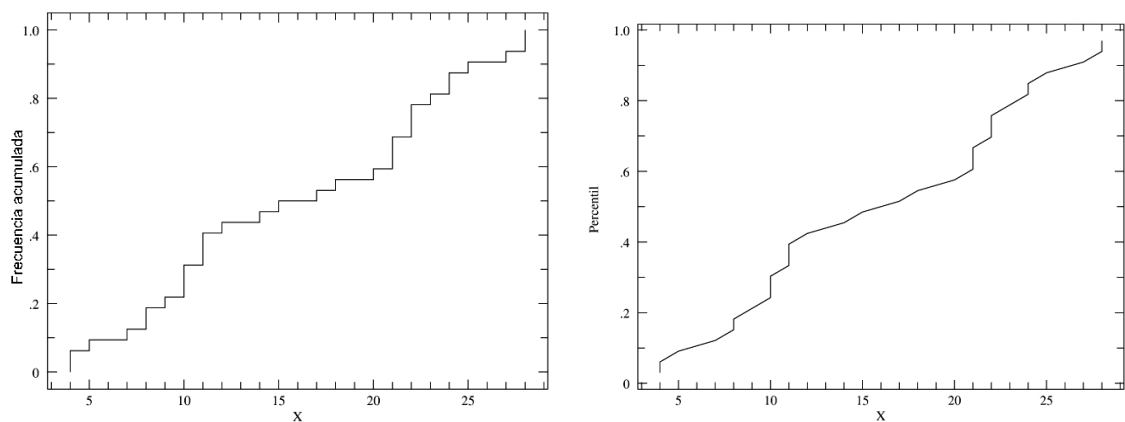


Figura 7.9: Frecuencia acumulada (izquierda) y percentil (derecha) Cruce de 2 puntos

7.3. COMPARANDO MÉTODOS DE CRUCE

Prueba Kolmogorov-Smirnov para cruce de dos puntos					
Media	16	Mediana	16	Valor superior	28
Desv. típica	7.5	Desv.Absoluta Mediana	6.69	Valor inferior	4
(K-S) P	0.38	Confianza	95 %	media: 13.3 a 18.7	
Prueba K-S consistente con distribución normal					

Cuadro 7.13: Prueba Kolmogorov-Smirnov para cruce de 2 puntos

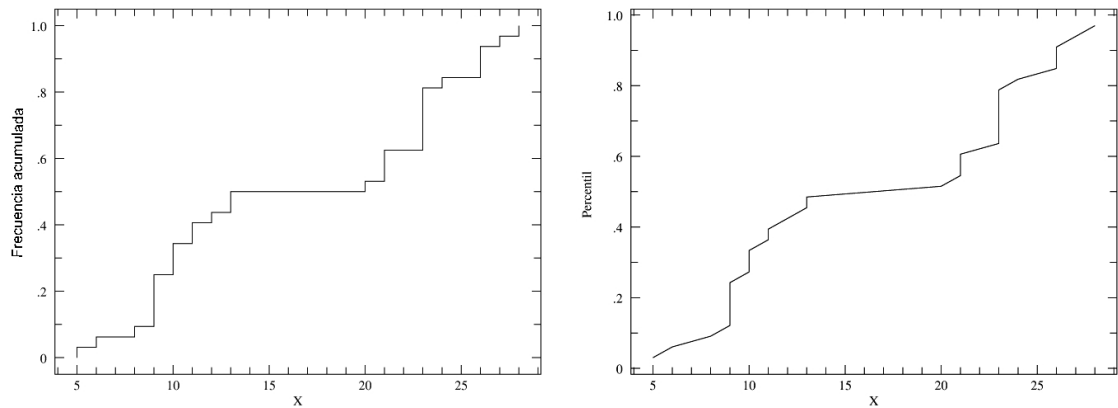


Figura 7.10: Frecuencia acumulada (izquierda) y percentil (derecha) Cruce uniforme

Prueba Kolmogorov-Smirnov para cruce uniforme					
Media	16.625	Mediana	16.5	Valor superior	28
Desv. típica	7.46	Desv.Absoluta Mediana	7.0	Valor inferior	5
(K-S) P	0.14	Confianza	95 %	media: 13.9 a 19.3	
Prueba K-S no consistente con distribución normal					

Cuadro 7.14: Prueba Kolmogorov-Smirnov para cruce uniforme

Análisis de resultados

Si los puntos obtenidos por el operador de cruce representaron una muestra relativamente equitativa del espacio entre los individuos cruzados, entonces los puntos estuvieron uniformemente distribuidos y por tanto al calcular su distribución de

7.3. COMPARANDO MÉTODOS DE CRUCE

frecuencia, ésta tendría un comportamiento normal o pseudo-normal (recuerde una de las formas de verificar la uniformidad de las secuencias aleatorias generadas por algunas desviadas uniformes de congruencia lineal discutidas en el capítulo 5).

Al aplicar la prueba de Kolmogorov-Smirnov para comprobar qué tanto se acercan estas distancias de Hamming promedio entre los individuos obtenidos por los métodos de cruce del cuadro 7.10 y cualquiera de los padres, a una distribución normal, vemos que para el caso de los métodos de cruce de 1 punto y de dos puntos, el resultado de la prueba es consistente con el comportamiento de normalidad. Para el caso del cruce uniforme con máscara = 0.7 (sesgo del 70 % hacia un padre), el resultado de (K-S) no es consistente con una distribución normal como muestran los gráficos de la figura 7.10 (compare con la línea mas recta de la figura 7.8 por ejemplo) y el cuadro 7.14.

La prueba χ^2 (chi cuadrada) es otra opción para comparar la naturaleza de estas distribuciones, no obstante, debido a que algunas de las observaciones en las distancias de Hamming eran menores a 1, y para el caso de las distancias en números reales éstas estaban normalizadas ($0 \leq \text{distancia} \leq 1$), es mas confiable la prueba Kolmogorov-Smirnov [15].

Con las anteriores pruebas experimentales, podemos concluir que para el caso de representaciones binarias simples, que descodifican fenotipos uni-dimensionales, enteros, de manera directa (la descodificación es tan solo una conversión entre sistemas numéricos), los operadores de cruce de 1 punto y de dos puntos, generan individuos relativamente distribuidos de manera uniforme sobre el espacio solución cuando los padres están en los extremos opuestos o polos de este espacio.

7.4 Comparando tipos de representación

Otra actividad experimental común con algoritmos genéticos es la comparación de rendimiento que cambios o extensiones al algoritmo en sí pueden tener sobre el mismo problema. Por ejemplo, el algoritmo genético simple usa una representación de tipo binario. Una representación binaria es simplemente un código que usa un alfabeto de 2 símbolos para indexar o denotar información. La forma más conocida es quizás la representación binaria simple, donde un número decimal cualquiera es transformado a una secuencia binaria descomponiéndolo en una suma de potencias de 2. Este tipo de codificación tiene una característica que números muy cercanos pueden tener representación muy diferente, por ejemplo, el número 255 tiene la representación binaria simple: 11111111, mientras que el 256 (una unidad de diferencia con 255) tiene representación: 10000000, en otras palabras, casi antípoda al primero.

Otros tipos de codificación binaria como la Gray, buscan mantener la relación de distancia de la información en su dominio original en el dominio binario, de manera que la representación binaria Gray de los números decimales 255 y 256 por ejemplo, sería diferente en tan solo 1 símbolo, en este caso 1 bit. Una inquietud lógica que surge en el campo experimental de algoritmos genéticos con representación binaria, es si esta característica puede influir en el rendimiento del algoritmo. Para tal fin, se diseñan una serie de experimentos e instrumentos de medición, que permitan abrir una discusión en este aspecto.

7.4.1. Diseño experimental

Una de las variables sobre la que queremos tener elementos de estudio es el efecto del tipo de representación sobre el rendimiento del algoritmo genético. El primer paso para un estudio de este tipo es la escogencia, implementación y/o diseño de las funciones objetivo que formarán la base experimental.

Para que las conclusiones puedan ser más genéricas, la idea es que las funciones objetivo sean de diferente naturaleza (continuas, no continuas, diferenciables, reales, discretas, multimodales, unimodales, multivariable, decepcionantes, etc.), para poder construir una base experimental lo suficientemente general y completa que permita ampliar el alcance de las conclusiones derivadas. De nuevo, es importante mencionar que las conclusiones que se formulen no pueden ser absolutas o consideradas inmutables e irrefutables, es decir, no podemos asegurar algo como *.^{el} tipo*

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

de representación X es la 'mejor' representación posible para los algoritmos genéticos en optimización simple"; una conclusión así es un Tao fuera del alcance de la práctica puesto que habría que evaluar toda función objetivo posible en ese campo, bajo todas las variantes de representación y relaciones de ésta con el resto de los elementos del algoritmo.

No obstante lo anterior, es importante sin embargo la actividad experimental para conocer mejor el comportamiento del algoritmo genético en relación a las variables de estudio. Esto si bien no nos proporciona una verdad absoluta, si nos da muy valiosa información para el estudio del comportamiento del algoritmo y el refinamiento de las *reglas rápidas* (del inglés *rules of thumb*).

Por motivo de simplicidad y comodidad práctica, ya que el objetivo de esta monografía es orientar el proceso experimental con algoritmos genéticos, seleccionamos solo una función objetivo de manera de hacer mas simple la formulación de métricas de rendimiento y discutir el proceso de evaluación. La función seis jorobas de camello, es una función muy usada en la literatura para el análisis de rendimiento de los AG por su simplicidad y naturaleza multimodal. Su descripción formal es:

$$f(x, y) = 4x^2 - 2,1x^4 + \frac{1}{3}x^6 + x \cdot y - 4y^2 + 4y^4 \quad -5 \leq x, y \leq 5$$

Esta función tiene varios mínimos locales y dos mínimos globales como se muestra a continuación:

$$x_{min}, y_{min} = (0,08983, -0,7126), (-0,08983, 0,7126) \quad \text{mín } f(x, y) = -1,0316285$$

La figura 7.11 muestra la función seis jorobas de camello en 3D, observe el despliegue de varios mínimos locales y dos globales que hacen que el espacio solución sea de naturaleza multimodal. Los mínimos locales pueden atraer a un algoritmo genético a converger hacia ellos, a esto se le denomina *vasija de atracción* o *remo-*

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

lino y si la estructura del algoritmo no estimula la variedad genética, es muy fácil que éste se quede “atrapado” en este mínimo local.

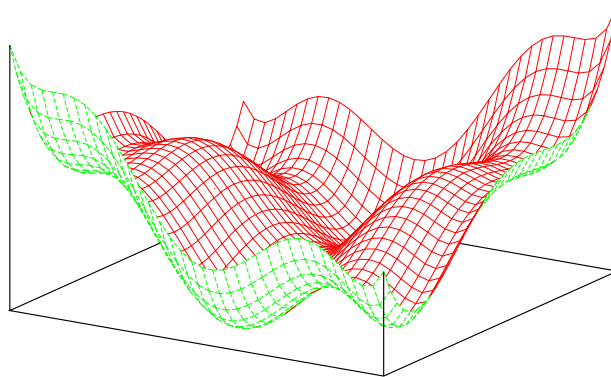


Figura 7.11: Función 6 Jorobas de camello

Para medir o estimar el desempeño del algoritmo, implementamos las siguientes métricas de evaluación, todas promediadas sobre el número total de corridas experimentales:

- M1:** Promedio de la adaptabilidad del mejor individuo por generación, en todas las corridas experimentales.
- M2:** Promedio de la adaptabilidad de la población por generación, en todas las corridas experimentales.
- M3:** Promedio de la adaptabilidad de la población en la última generación, en todas las corridas experimentales.
- M4:** Promedio de la mejor solución encontrada por el algoritmo en cada corrida experimental.

7.4.2. Descripción de los experimentos

Los experimentos incluirán la implementación de la función mencionada bajo la representación binaria simple y binaria Gray. Algunas variantes como el método de selección (la sección 7.2 concluye algunas diferencias significativas entre métodos de selección en términos de dos métricas propuestas) son agregadas para efectos de la comparación.

Debido a que la función tiene su dominio en los números reales ($x, y \in \mathbb{R}[-5 : 5]$) y nuestra representación debe ser binaria como la usada en el AGS, debemos aplicar métodos de transformación que permitan llevar un número real a un equivalente entero para luego a un equivalente binario y viceversa. Esto implica que elementos de un conjunto infinito (números reales), deben ser representados en un arreglo binario de longitud finita, y por tanto habrá una pérdida de información proporcional al tamaño del arreglo (longitud del cromosoma).

Para llevar una cadena binaria a su equivalente número entero:

$$\text{Número_entero} = \sum_{i=0}^{L_C-1} (2^i \cdot \vec{c}_i) \quad (7.1)$$

Donde L_C es la longitud del cromosoma \vec{c} y \vec{c}_i su i -ésima componente mas significativa. Para llevar un número real entre L_{inf} y L_{sup} a su equivalente número entero:

$$\text{Número_entero} = \lceil \frac{\text{Número_real} - L_{inf}}{L_{sup} - L_{inf}} \cdot (2^{L_C} - 1) \rceil \quad (7.2)$$

donde $\lceil \dots \rceil$ es una función de aproximación al entero inmediato superior, y

$$\text{Número_real} = L_{inf} + \frac{\text{Número_entero} \cdot (L_{sup} - L_{inf})}{2^m - 1} \quad (7.3)$$

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

es la expresión para llevar un número entero a su equivalente real en el rango de acotación dado ($L_{inf} \dots L_{sup}$).

Con 7.1, 7.2 y 7.3 podemos fácilmente llevar un cromosoma binario a un único equivalente entero que luego llevamos a un único equivalente en real para su evaluación en la función de adaptación. La precisión de las variables x y y viene dada por el ancho de banda o longitud de cromosoma, de manera que la *granularidad* de estas variables (lo que define el tamaño o magnitud de la diferencia entre dos valores consecutivos en la escala real representada) viene dada por la expresión:

$$G_{x_i} \approx \frac{L_{sup} - L_{inf}}{2^{(L_C/N)}}. \quad (7.4)$$

Donde N es el número de variables representadas en el cromosoma. Para L_C constante, mientras menor sea el número de variables representadas, mas fina la granularidad de la conversión, es decir, mayor precisión. La ecuación 7.4 asume que todas las variables tienen cromátidos de la misma longitud.

Los experimentos se implementaron usando representaciones de codificación binaria simple y codificación binaria Gray, manteniendo los demás parámetros iguales, esto con la intención de evaluar el rendimiento en igualdad de condiciones para los dos tipos de codificación. En todos los casos los parámetros usados fueron como a continuación se detallan:

Tamaño de la población :	50
Número de generaciones :	50
Corridas experimentales :	50
Longitud de cromosoma L_C :	24
Número de variables :	2
Granularidad de variables :	0.002441
Probabilidad de cruce :	75 %
Probabilidad de mutación :	0.1 %

Cuadro 7.15: Parámetros experimentales para tipos de representación binaria

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

Para la escogencia de estos parámetros se hizo experimentación previa refinando valores iniciales obtenidos mediante *reglas rápidas* reportadas en la literatura [2, 3, 51] como “buenos valores” de comienzo para refinar a mejores valores. La misma configuración es usada para ambos algoritmos con el fin de analizar mejor las diferencias de rendimiento al variar la representación usada.

7.4.3. Resultados experimentales

La figura 7.12 muestra la adaptabilidad promedio del mejor individuo por generación (M1) y la adaptabilidad promedio de la población (M2), para las pruebas del algoritmo genético con representación binaria simple y selección de rueda de ruleta. La mejor solución obtenida por corrida experimental (M3) se muestra en la figura 7.13. La línea punteada representa el mínimo global del problema.

El cuadro 7.16 muestra el resumen de las métricas de rendimiento aplicadas al algoritmo con representación binaria simple y métodos de selección de rueda de ruleta y torneo binario. Los promedios son sobre todas las corridas experimentales.

El mínimo global para la función de adaptación usada es -1.0316285, observe que solo el algoritmo con método de selección de torneo binario obtuvo en promedio soluciones muy cercanas a este punto.

Métrica	Representación binaria simple	
	Sel. Rueda de ruleta	Sel. Torneo binario
M1	$\mu = -0,916218$ $\sigma = 0,0629592$	$\mu = -0,994921$ $\sigma = 0,0493805$
M2	$\mu = -0,657133$ $\sigma = 1,36808$	$\mu = -0,264412$ $\sigma = 1,79237$
M3	$\mu = -0,976255$ $\sigma = 0,08956$	$\mu = -1,02226$ $\sigma = 0,0126631$
M4	$\mu = -0,942851$ $\sigma = 2,09465$	$\mu = -0,976976$ $\sigma = 0,888901$

Cuadro 7.16: Métricas de rendimiento para representación binaria simple

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

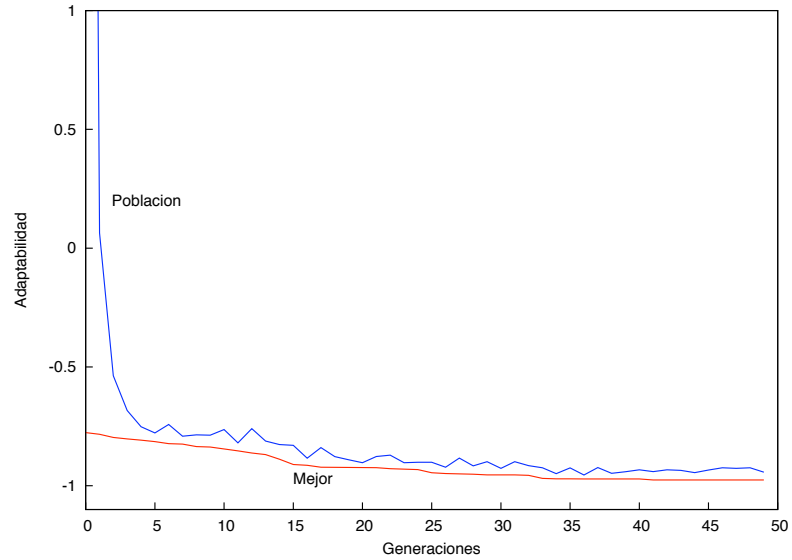


Figura 7.12: Representación binaria con rueda de ruleta. Métricas M1 y M2.

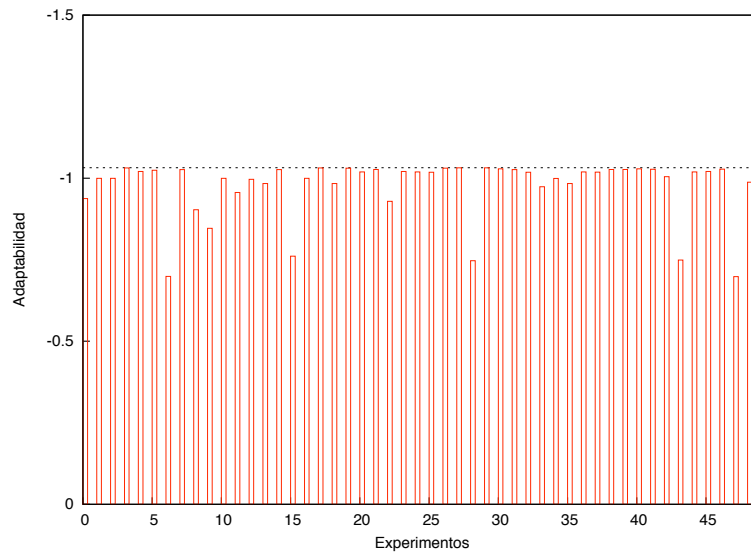


Figura 7.13: Representación binaria con rueda de ruleta. Métrica M3.

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

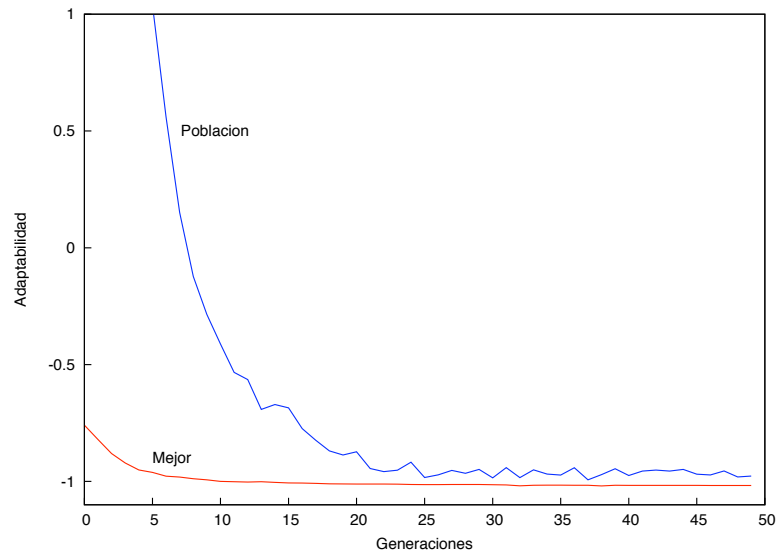


Figura 7.14: Representación binaria con torneo binario. Métricas M1 y M2.

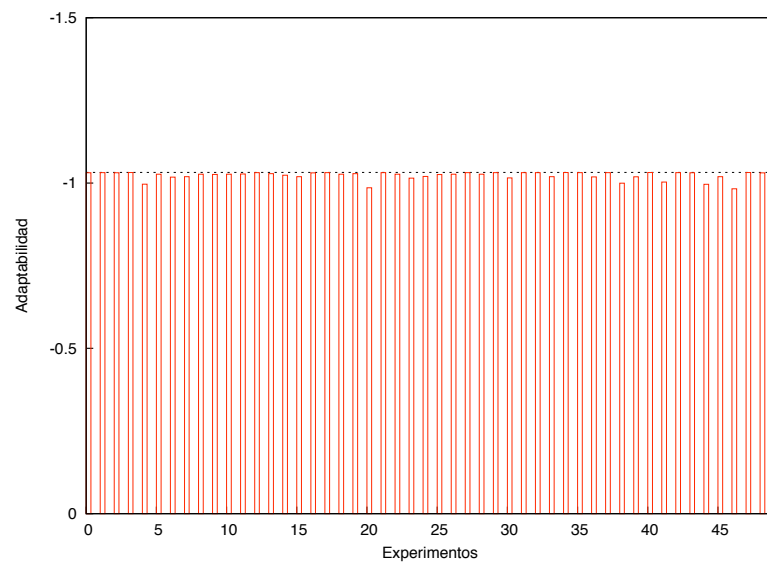


Figura 7.15: Representación binaria con torneo binario. Métrica M3.

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

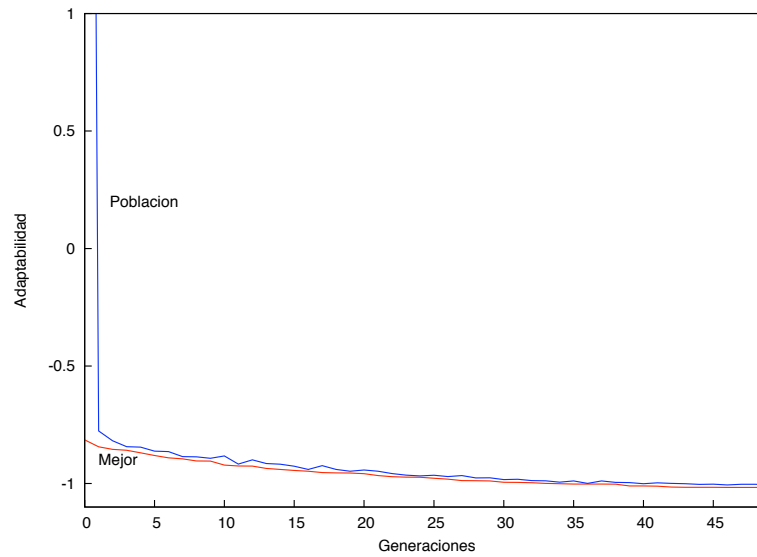


Figura 7.16: Representación binaria Gray con rueda de ruleta. Métricas M1 y M2.

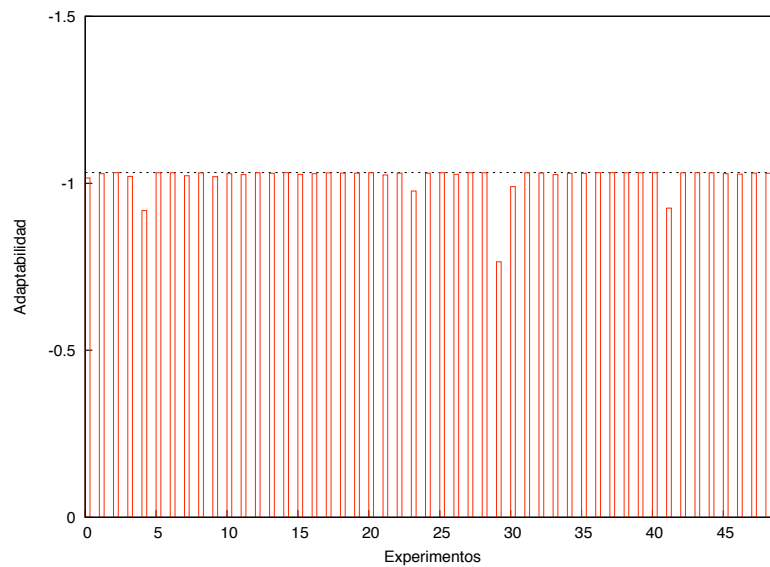


Figura 7.17: Representación binaria Gray con rueda de ruleta. Métrica M3.

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

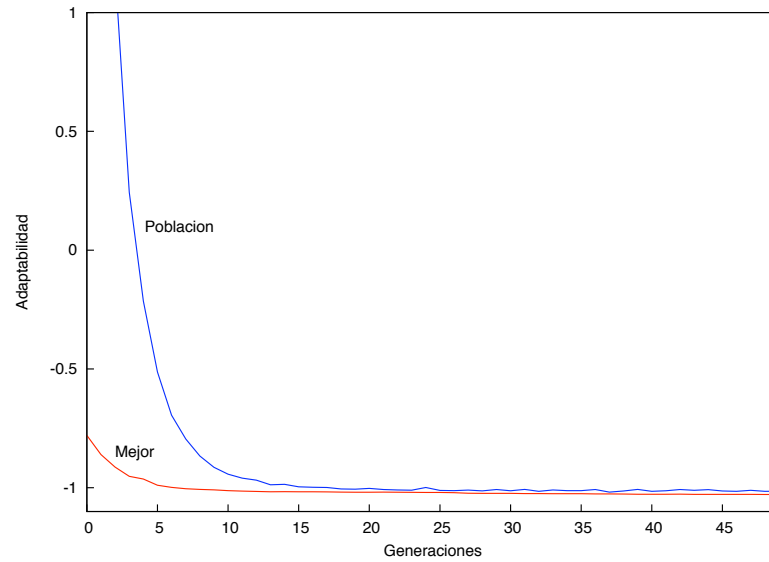


Figura 7.18: Representación binaria Gray con torneo binario. Métricas M1 y M2.

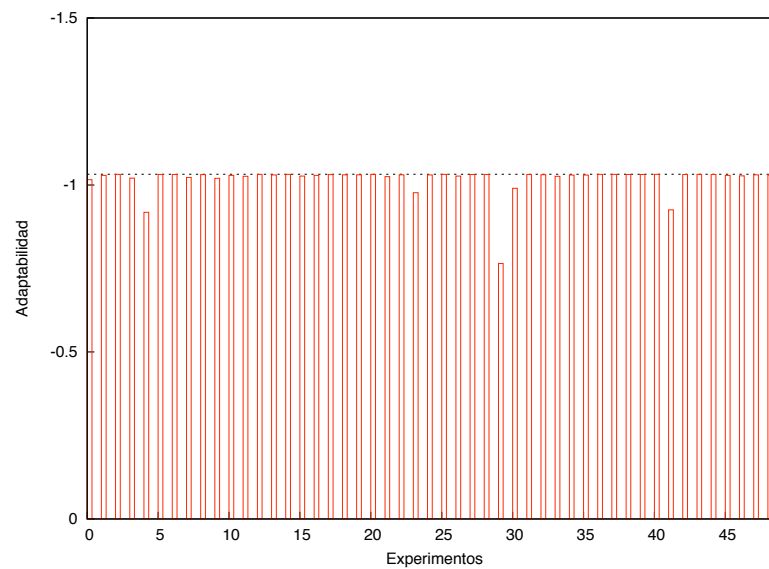


Figura 7.19: Representación binaria Gray con torneo binario. Métrica M3.

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

Métrica	Representación binaria Gray	
	Sel. Rueda de ruleta	Sel. Torneo binario
M1	$\mu = -0,96075$ $\sigma = 0,0532744$	$\mu = -1,00754$ $\sigma = 0,0442621$
M2	$\mu = -0,754378$ $\sigma = 1,35483$	$\mu = -0,612066$ $\sigma = 1,5252$
M3	$\mu = -1,01783$ $\sigma = 0,0427403$	$\mu = -1,02909$ $\sigma = 0,00768696$
M4	$\mu = -1,0059$ $\sigma = 0,665272$	$\mu = -1,01454$ $\sigma = 0,266309$

Cuadro 7.17: Métricas de rendimiento para representación binaria Gray

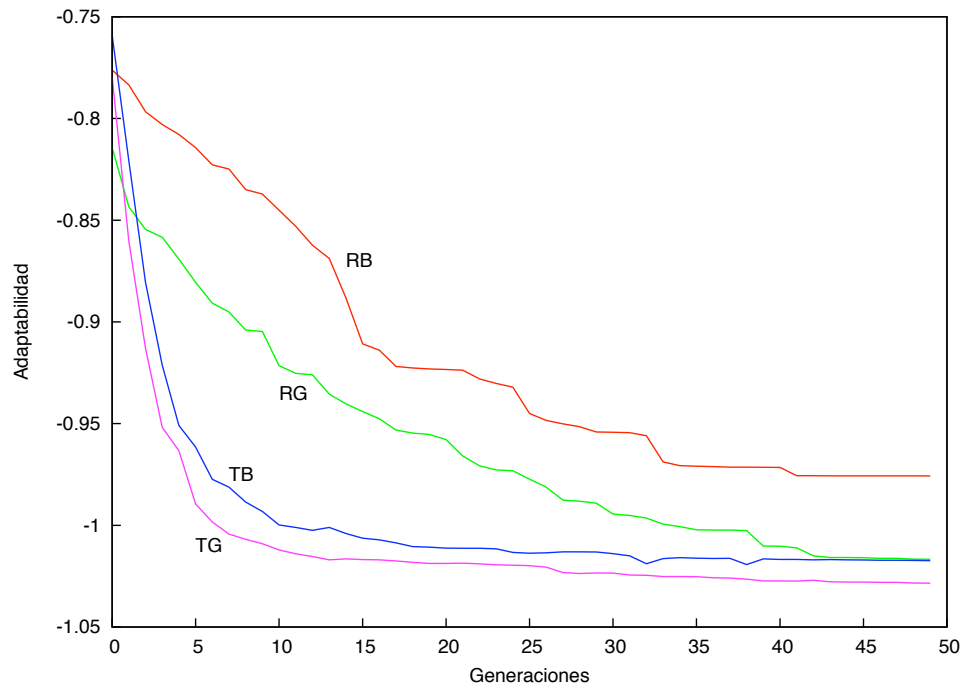


Figura 7.20: Comparación de M1 para tipos de representación y selección.

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

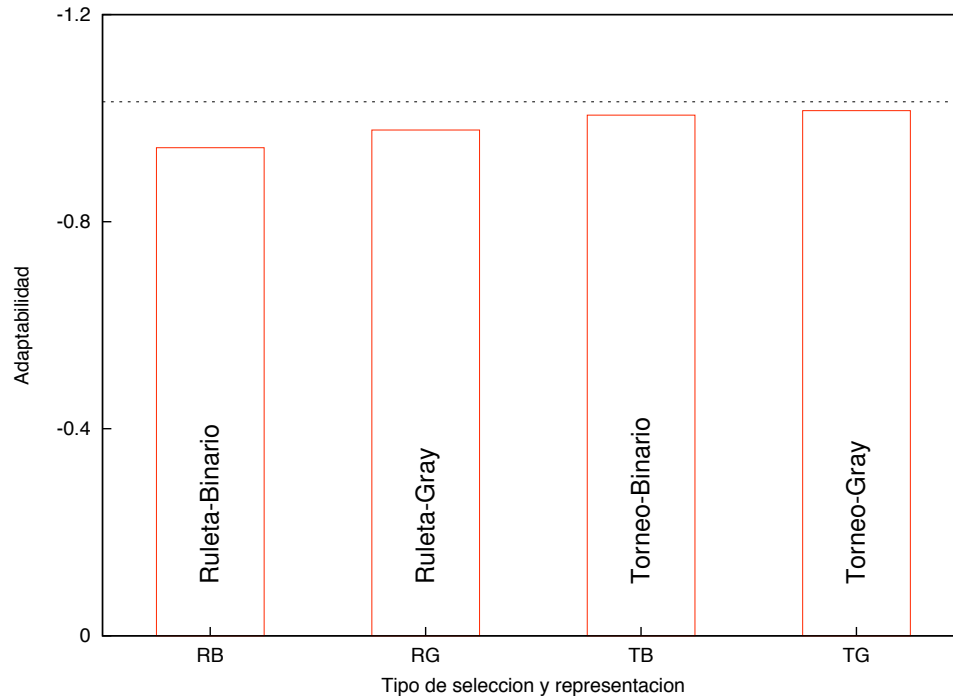


Figura 7.21: Comparación de M4 por tipo de selección y representación.

Análisis de resultados

Para todos los escenarios experimentales se observó un aparente mejor rendimiento para los casos de representación binaria Gray que representación binaria simple, no obstante, como muestra el cuadro 7.18, aunque estas diferencias fueron constantes en todas las observaciones, algunas de ellas no fueron estadísticamente significativas.

Observe la figura 7.20 que muestra la relación entre representaciones binaria simple y Gray para la métrica de evaluación M1. En cada caso, los algoritmos con representaciones en binario Gray muestran mejor rendimiento que sus contrapartes en binario simple. Solo el algoritmo implementado con el método de selección de torneo consiguió en promedio rendimiento cercano al mínimo global, los algoritmos con método de selección de rueda de ruleta tendieron a converger con mas lentitud a esta marca, lo que sugiere que necesitan mayor número de generaciones, o diferente

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

configuración de parámetros iniciales para lograr la convergencia. Las etiquetas en cada curva muestran el tipo de selección y tipo de representación usada. RB es rueda de ruleta con codificación binaria, RG rueda de ruleta con codificación Gray, TB significa selección de torneo con representación binaria y finalmente TG es torneo con representación Gray.

Al igual que en la sección 7.2, el desempeño de los algoritmos con método de selección de rueda de ruleta fue relativamente inferior a los implementados con torneo para las métricas establecidas, la función objetivo evaluada y los parámetros iniciales usados. En esta sección, observamos que para todos los casos, la representación Gray obtuvo un mejor rendimiento que la representación binaria simple. Para poder derivar conclusiones más robustas y precisas sobre esta relación de rendimiento, es necesario implementar una mayor cantidad de experimentos que incluyan más funciones de evaluación de diferente naturaleza, diferentes métodos de selección, mutación y cruce, y diferentes parámetros de configuración inicial. Los resultados deben ser sometidos a estimadores estadísticos paramétricos o no paramétricos según amerite el caso y en lo posible es aconsejable usar algún estimador de rango como la prueba Wilcoxon o Mann-Whitney, ambas muy documentadas y con facilidades de pruebas en línea a través de internet.

Conclusiones y Discusión

El diseño experimental planteado y los resultados obtenidos sugieren que la codificación en binario Gray puede ser más provechosa para su uso en problemas de optimización simple de funciones objetivo similares en naturaleza a la función usada. Un mayor número de experimentos es necesario para poder concluir relaciones de rendimiento más concretas y específicas.

El mejor desempeño en promedio, según las métricas establecidas y al menos para M3, de la codificación Gray, puede ser en parte debido a la característica mencionada de conservar la distancia entre puntos de muestreo consecutivos de un espacio solución a un espacio representación. Intuitivamente, esto pudiera sugerir que la búsqueda con cromosomas Gray es más detallada o refinada que la búsqueda con cromosomas binarios simples, pues se evita el ruido de la representación cuando el algoritmo va convergiendo a un óptimo. Estos saltos abruptos son conocidos como “riscos de Hamming” y pueden retrasar la búsqueda fina del algoritmo. La figura

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

7.22 es un ejemplo clásico donde ruido introducido en la búsqueda fina del algoritmo puede ser perjudicial para la convergencia, pues puede alejar al algoritmo de la vasija de atracción, que para el caso de la función representada, es el único mínimo global. Mayor experimentación es necesaria para concluir al respecto.

Las figuras 7.22 y 7.23 muestran dos ejemplos de funciones de naturaleza diferente. Observe el tejido de espacio solución de cada una y plantéese posibles inconvenientes y ventajas que el algoritmo pueda encontrar al navegar dicho espacio en busca del mínimo, eso le ayudará a diseñar mejores experimentos y variables de control. Los escenarios experimentales deben en lo posible incluir funciones de evaluación que representen diferentes retos para el algoritmo, esto permite considerar mejor los controles de los experimentos y por ende influirán a formular conclusiones mas concretas y precisas.

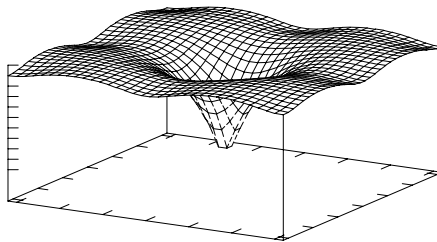


Figura 7.22: Función de Ackley.

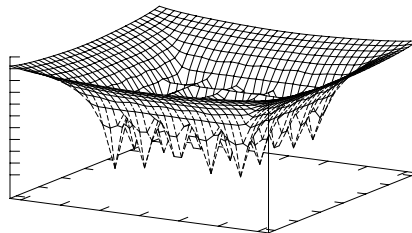


Figura 7.23: Función Madriguera de zorros de Shekel.

7.4. COMPARANDO TIPOS DE REPRESENTACIÓN

Selección: Rueda de ruleta

	Binaria simple			Binaria Gray			Estadísticos			
	μ	σ		μ	σ		t	Confianza	G.L	Observación
M	-0.9162	0.0629		-0.9608	0.0533		3.8582	95 %	98	Δ significativa
M1	-0.6571	1.3681		-0.7544	1.3548		0.3562	95 %	98	Δ no significativa
M2	-0.9763	0.0896		-1.0178	0.0427		2.9927	95 %	98	Δ significativa
M3	-0.9429	2.0947		-1.0059	0.6653		0.2027	95 %	98	Δ no significativa
M4										

Selección: Torneo binario

	Binaria simple			Binaria Gray			Estadísticos			
	μ	σ		μ	σ		t	Confianza	G.L.	Observación
M	-0.9162	0.0629		-0.9608	0.0533		1.3862	95 %	98	Δ no significativa
M1	-0.6571	1.3681		-0.7544	1.3548		1.0456	95 %	98	Δ no significativa
M2	-0.9763	0.0896		-1.0178	0.0427		3.3413	95 %	98	Δ significativa
M3	-0.9429	2.0947		-1.0059	0.6653		0.2896	95 %	98	Δ no significativa
M4										

Cuadro 7.18: Comparación de representación binaria y Gray bajo la prueba t -Student.

7.5 El Vendedor Viajero

El Vendedor viajero es un conocido problema de optimización combinatoria donde se busca completar un ciclo de un grafo Hamiltoniano con la menor distancia recorrida posible (ver sección 4.3.1). El espacio solución crece exponencialmente con cada nodo incrementado en el grafo. Suponga que un agente viajero debe recorrer un circuito de ciudades separadas entre sí por una distancia d , y que no debe pasar por la misma ciudad mas de una vez. El problema del vendedor es planificar una ruta que le permita recorrer todas las ciudades del circuito con la menor distancia posible. Obviamente, la magnitud del espacio solución es proporcional al número de total de nodos L del circuito.

En esta sección compararemos el rendimiento de dos algoritmos genéticos en la resolución del problema del vendedor viajero. Se tomó un algoritmo genético simple y se le introdujeron cambios al operador de selección para crear una extensión llamada **GGA** por sus siglas en inglés Gendered Genetic Algorithm. En este laboratorio se comparará el GGA con la forma original del algoritmo genético que llamaremos **SGA**, para mantener sus siglas del inglés de Standard Genetic Algorithm.

7.5.1. Extensión al Algoritmo Genético

El GGA introduce solo una extensión al operador de selección del algoritmo genético simple, dividiendo la población en dos subgrupos; grupo (x) y grupo (y). El operador de cruce de 1 punto se aplica solo entre miembros de diferentes grupos y la selección de los $x \in X$ es mediante algún método de adaptabilidad proporcional (rueda de ruleta en este caso), mientras que la selección de los $y \in Y$ se hace de acuerdo a la siguiente expresión:

$$\frac{w_1 \cdot g(\text{Edad}(y)) + w_2 \cdot f(y) + w_3 \cdot \Delta f(y)}{w_1 + w_2 + w_3} \quad (7.5)$$

todos los elementos de 7.5 son valores normalizados, de manera que:

$$0 \leq f(y), \Delta f(y), w_i, g(Edad(y)) \leq 1$$

y además

$$\sum_{i=1}^3 w_i = 1 \tag{7.6}$$

Los w_i son factores de peso que ponderan la preferencia de selección de individuos en Y según el criterio del diseñador. $Edad(y)$ es el número de épocas o generaciones que el individuo ha sobrevivido, y g es una función triangular definida como:

$$g(Edad(y)) = \begin{cases} 1 - \frac{|Edad(y) - \mu|}{\sigma} & : \quad Edad(y) < \mu + \sigma \\ 0 & : \quad Edad(y) \geq \mu + \sigma \end{cases} \tag{7.7}$$

$\Delta f(y)$ es una medida de diferencia entre la adaptabilidad del individuo y en comparación con la de su(s) cría(s), definida de la siguiente manera:

$$\Delta f(y) = \Delta f(y) + \left(0,5 + \frac{f(y_{hijo}) - f(y)}{2}\right) \tag{7.8}$$

En la ecuación 7.5 puede apreciar como cambiando los parámetros w_1 , w_2 y w_3 se puede “guiar” el esquema de selección hacia cualquiera de los tres factores que ellos ponderan. Esto ayuda a incrementar el control del algoritmo en cuanto diversidad genética y velocidad de convergencia, no obstante, también aumenta la complejidad del mismo.

7.5.2. Diseño experimental y resultados

Para la implementación del Vendedor viajero, la representación usada fue una lista de ciudades (nodos del grafo) en el orden en que son visitadas. Un cromosoma con el genotipo: [1425] representa el circuito (fenotipo) $1 \rightarrow 4 \rightarrow 2 \rightarrow 5 \rightarrow 1$. El

7.5. EL VENDEDOR VIAJERO

operador de cruce fue el de 1 punto y el esquema de mutación es simplemente el intercambio de dos ciudades en el cromosoma. Para tener una idea del óptimo global y dar una idea intuitiva de la adaptabilidad, las ciudades del circuito fueron puestas sobre la circunferencia de un círculo. Comenzando en cualquier nodo, el óptimo global será el tour de distancia D , correspondiente al ciclo entre ciudades consecutivas alrededor del círculo. Con $C_{m,n}$ representando los costos de enlace entre los nodos m y n (distancia entre ellos), la función de adaptabilidad para todo individuo a en la población P , es definida como:

$$f(a) = \frac{D}{\sum_{i=1}^{L-1} C_{m_i, m_{i+1}} + C_{m_L, m_1}} \quad (7.9)$$

donde L es el número total de ciudades o nodos del circuito. Los resultados son promediados sobre un total de 100 corridas experimentales.

Las métricas de evaluación usadas son M1 y M2 definidas en la experimentación de las secciones 7.2 y 7.4 como:

M1: Adaptabilidad promedio del mejor individuo por generación en todas las corridas experimentales:

$$M1 = \frac{1}{E} \sum_{e=1}^E \left(\frac{1}{T} \sum_{t=1}^T f_t^*(A) \right) \quad (7.10)$$

donde $f_t^*(A) = mejor\{f(a_1), f(a_2), f(a_3), \dots, f(a_{|P|})\}$, E es el número total de corridas experimentales y T el número total de generaciones.

M2: Adaptabilidad promedio de la población por generación en todas las corridas experimentales.

$$M2 = \frac{1}{E} \sum_{e=1}^E \left(\frac{1}{T} \sum_{t=1}^T \left(\frac{1}{|P|} \sum_{j=1}^{|P|} f_t(a_j) \right) \right) \quad (7.11)$$

7.5. EL VENDEDOR VIAJERO

Los resultados experimentales son comparaciones entre el SGA y el GGA de manera que la configuración de parámetros es la misma para ambos casos. En experimentos previos, se obtuvo que un juego de parámetros adecuados fue $|P| = 200$, probabilidad de cruce 75 % y probabilidad de mutación 0.1 %. Para los parámetros específicos del GGA se usaron los siguientes valores: $w_1 = 0,5$, $w_2 = 0,38$, $w_3 = 0,12$, $\mu = 2$ y $\sigma = 4$ (ver ecuaciones 7.5 y 7.7).

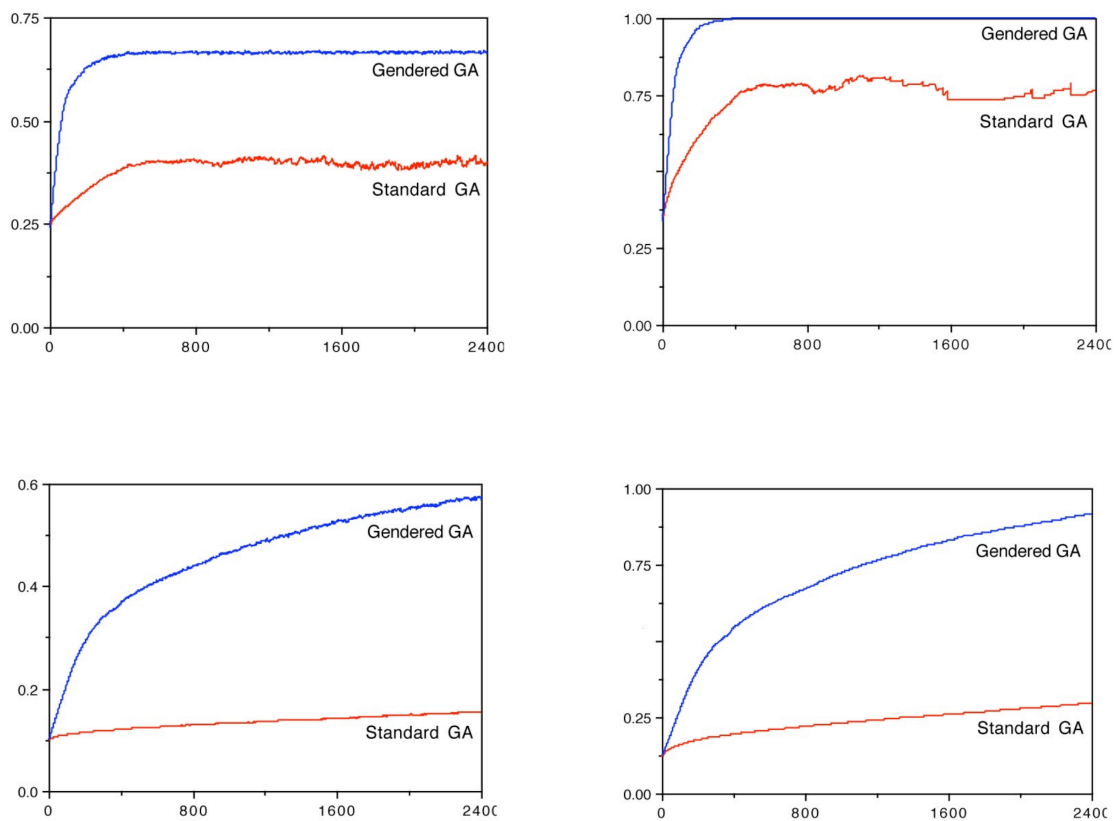


Figura 7.24: Resultados de M1 y M2 del Vendedor viajero para el GGA y SGA con 50 ciudades. La adaptabilidad promedio de la población M1 (izquierda), y del mejor individuo M2 (derecha). El eje horizontal (x) muestra las iteraciones o generaciones del algoritmo y el vertical (y) la adaptabilidad.

En la figura ??, para 20 ciudades de circuito, podemos ver que la adaptabilidad promedio de la población en el GGA tiene una evolución mas marcada hacia estados

7.5. EL VENDEDOR VIAJERO

asintótico. Igualmente muestra una aparente ventaja en la velocidad de convergencia del promedio de adaptabilidad del mejor individuo, logrando el máximo global de 1.0 (la adaptabilidad está normalizada). En el SGA esta medida se acerca a 0.75.

La figura 7.24 muestra como los resultados difieren para 50 ciudades que obviamente representa un espacio solución mucho mas grande. Para este caso, el GGA de nuevo resulta en ascenso mas pronunciado que el SGA, logrando mejores soluciones en menos iteraciones (generaciones) del algoritmo.

El cuadro 7.19 muestra el número de rutas óptimas logradas de 100 corridas experimentales para los dos algoritmos (GGA y SGA), y para diferente número de ciudades.

Nodos	Rutas	Standard GA		Gendered GA	
		M2	Optima	M2	Optima
20	10^{18}	0.7389	96	0.9826	100
50	10^{64}	0.2379	0	0.7180	44
200	10^{374}	0.0356	0	0.1006	0
500	10^{1133}	0.0123	0	0.0264	0

Cuadro 7.19: Comparación de rendimiento del GGA y SGA en la métrica M1.

GA	Mutación 10 %	Mutación 30 %	Mutación 50 %
Standard	0.180	0.162	0.155
Gendered	0.513	0.458	0.458

Cuadro 7.20: Tolerancia al ruido en la métrica M1 del GGA y SGA.

El cuadro 7.20 muestra la estimación de M2 para diferentes niveles de mutación (ruido) en ambos algoritmos. El resultado de las 100 corridas experimentales sugieren que el enfoque selectivo del GGA pudiera ayudar a tolerar el ruido introducido al cromosoma.

7.6 Una prueba en Optimización multiobjetivo

Para implementar experimentos de algoritmos genéticos sobre problemas de optimización multiobjetivo (ver sección 4.3.2 para su definición), se implementó el algoritmo VEGA [52] (Vector Evaluated Genetic Algorithm) ó algoritmo genético de evaluación vectorial, y se comparó contra una modificación del GGA para resolver problemas multiobjetivo.

VEGA es uno de las propuestas mas simples de optimización mediante cómputo evolutivo y es una extensión directa y simple de un algoritmo genético, de hecho, como su nombre lo indica, es un AG con un método diferente de evaluación de la adaptabilidad. VEGA evalúa un vector con al menos 2 funciones objetivo.

El algoritmo VEGA original como lo describe [13] es como sigue:

Paso 1 Iniciar un contador de función objetivo en $i = 1$ y definir $q = N/M$, donde N es el tamaño total de la población y M el número de funciones objetivo.

Paso 2 Para todas las soluciones $j = 1 + (i + 1) \cdot q$, asignar la adaptabilidad como:

$$F(x^{(j)}) = f_i(x^{(j)})$$

Paso 3 Realice selección de proporción de adaptabilidad en todas las soluciones q para crear un pool de apareamiento P_i

Paso 4 Combine todos los pool de apareamiento: $P = \bigcup_{i=1}^M P_i$. Realice el cruce y mutación en P para crear una nueva población.

A la adaptación del GGA para trabajar como VEGA en la evaluación de vectores con múltiples funciones objetivo se le denominará **GVEGA**. Al igual que con al algoritmos genético, el cambio de GGA a GVEGA es simple y directo, tan solo se conserva el operador de selección descrito en la sección 7.5.1 y se procede como se indica en el procedimiento anterior.

En el GVEGA, el elemento de diversidad genética controlado por w_1 , se tomó como el número de selecciones que el individuo tuvo en cada vuelta. Aquellos con muchas selecciones consecutivas fueron menos favorecidos comparados a aquellos mas cercanos a un nivel específico. El factor w_3 fue relacionado al nivel de dominancia

7.6. UNA PRUEBA EN OPTIMIZACIÓN MULTIOBJETIVO

del individuo. Tres niveles de dominancia fueron definidos, dando mayor prioridad a aquellos en el grupo donde todos los individuos eran soluciones no dominadas.

Para ambos algoritmos (VEGA y GVEGA) el tamaño de la población fue $|P| = 60$ y total de 5 corridas experimentales fueron hechas. Todas las soluciones no dominadas de la última generación de cada corrida, fueron incluidas en un conjunto Q , de manera que $|Q| \leq 300$.

El problema experimental (funciones objetivo) así como los parámetros iniciales fueron tomados de [13]. Es un problema de maximización con dos objetivos en conflicto definido como sigue:

$$\begin{aligned} \text{Max} - \text{Ex} \quad & \left\{ \begin{array}{l} \text{Maximizar } f_1(x) = 1,1 - x_1, \\ \text{Maximizar } f_2(x) = 60 - \frac{1+x_2}{x_1}, \end{array} \right. \\ \text{sujeto a} \quad & \left\{ \begin{array}{l} 0,1 \leq x_1 \leq 1, \\ 0 \leq x_2 \leq 5. \end{array} \right. \end{aligned}$$

Para este problema, el frente óptimo de Pareto se encuentra en:

$$x_2^* = 0 \quad \text{y} \quad 0,1 \leq x_1 \leq 1$$

Las métricas de rendimiento implementadas para la comparación de VEGA y GVEGA han sido ampliamente usadas en optimización multiobjetivo [14] y se describen a continuación:

Número de soluciones no dominadas: El número de soluciones no dominadas $|Q|$ en la última generación luego de todas las corridas experimentales. Mientras mayor sea esta métrica al evaluar el algoritmo, se considera mejor su rendimiento.

Rata de error: El número de soluciones de $|Q|$ que no son miembros del conjunto óptimo de Paret. Una rata de error $RE = 0$, significa que todas las soluciones

7.6. UNA PRUEBA EN OPTIMIZACIÓN MULTIOBJETIVO

no dominadas en Q son del conjunto óptimo Pareto, por lo tanto, una menor rata de error se considera un mejor rendimiento.

$$RE = \frac{\sum_{i=1}^{|Q|} e_i}{|Q|}$$

Distancia generacional: Define la distancia promedio de las soluciones en Q al conjunto óptimo Pareto. Una distancia generacional DG pequeña, intuitivamente es tomado como un mejor rendimiento.

$$DG = \frac{(\sum_{i=1}^{|Q|} d_i^p)^{1/p}}{|Q|}$$

Separación: Es una medida relativa calculada entre soluciones consecutivas en Q . Una separación SP mas pequeña es considerada mejor rendimiento.

$$S = \sqrt{\frac{1}{|Q|} \sum_{i=1}^{|Q|} (d_i - \bar{d})^2}$$

donde $d_i = \min_{k \in Q \wedge k \neq i} \sum_{m=1}^M |f_m^i - f_m^k|$ y \bar{d} es la media de esta distancia.

Maxima dispersión: Mide la diagonal formada por los valores extremos de la función en el conjunto no dominado. Para los resultados de estos experimentos fue normalizada de manera que la Máxima dispersión MD =1 significa que Q tiene soluciones en extremos opuestos del frente de Pareto.

$$D = \sqrt{\frac{1}{M} \sum_{m=1}^M \left(\frac{\max_{i=1}^{|Q|} f_m^i - \min_{i=1}^{|Q|} f_m^i}{F_m^{\max} - F_m^{\min}} \right)^2}$$

7.6.1. Resultados preliminares

Un total de 5 corridas experimentales fueron realizadas. El tamaño de la población fue $|P| = 60$ por lo tanto el número máximo posible de soluciones no dominadas es $|Q| \leq 300$. Para el GVEGA, la proporción del tamaño del subgrupo y fue

7.6. UNA PRUEBA EN OPTIMIZACIÓN MULTIOBJETIVO

El GVEGA	
Paso	Descripción
1	Establecer los parámetros: Tamaño de población, número de individuos para cada subgrupo x e y , probabilidad de cruce y mutación, parámetros de control selectivo w_1, w_2, w_3, μ y σ .
2	Dividir la población P en M subpoblaciones de individuos con M igual al número de funciones objetivo y cada subpoblación es de tamaño $\frac{ P }{M}$. Iniciar poblaciones de manera aleatoria.
3	De acuerdo a una proporción establecida en el paso 1, dividir cada subpoblación en dos grupos x e y .
4	Para cada subpoblación P_j , asignar adaptabilidad como: <div style="text-align: center; margin: 10px 0;"> $F(y_i^j) = f^{(j)}(y_i^j) \quad \text{y} \quad F(x_i^j) = (\text{ver ecuación 7.5})$ </div> donde $f^{(j)}$ es una medida específica de adaptabilidad para la función objetivo j y ($j = 1, 2, \dots, M$)
5	Barajar la población para evitar selección sesgada a posición.
6	Realizar selección de rueda de ruleta continuamente para crear un pool de apareo de tamaño $ P $.
7	Realizar cruce de acuerdo a la probabilidad establecida en el paso 1. Crear dos individuos uno de cada grupo (x, y).
8	Si la probabilidad de cruce $p_c < 1$, llenar el remanente en la nueva generación con individuos del pool de apareo e incrementar en uno su factor <i>Edad</i> descrito en la ecuación 7.7 si el individuo es del grupo y .
9	Realizar mutación de acuerdo a la probabilidad establecida.
10	Determinar los niveles de dominancia y actualizar la información en individuos en y para el próximo proceso de selección (a ser ponderado por w_3).

Cuadro 7.21: Descripción del GVEGA

7.6. UNA PRUEBA EN OPTIMIZACIÓN MULTIOBJETIVO

de 50 % con mutaciones por el orden de 10^{-3} (0.1 %). Los valores para w_i fueron $w_1 = 0,12$, $w_2 = 0,08$, $w_3 = 0,8$ y la probabilidad de cruce fue de 85 % para ambos algoritmos.

Los cuadros 7.22, 7.23 y 7.24 muestran el resultado de las corridas para 20, 100 y 250 iteraciones respectivamente en ambos algoritmos. La rata de error (RE) fue calculada tomando distancias menores a 0.01 de un individuo al frente de Pareto, como 0, es decir, considerando al individuo como miembro del conjunto óptimo.

VEGA tuvo un mejor desempeño relativo a GVEGA en la métrica de máxima dispersión (MD) para los experimentos con 20 iteraciones y en la métrica de distancia generacional (DG) para los experimentos con 100. Todas las soluciones encontradas por VEGA con los parámetros dados, en todos los experimentos, estuvieron a mas de 0,01 unidades de distancia del frente de Pareto, por lo tanto, la rata de error computada (RE) fue máxima para cada caso. VEGA también pareció perder habilidad de encontrar soluciones no dominadas al incrementar el número de iteraciones. La figura 7.26, muestra las soluciones encontradas por ambos algoritmos en cada diferente corrida y para tener una idea de cómo afectan los pesos w_i en el desempeño de GVEGA, la figura 7.27 muestra su rendimiento para cada valor w_i puesto en 1 (y por tanto descartando los otros dos, ver ecuación 7.6).

Algoritmo	NoD	%NoD	RE	S	DG	DG(Var)	MD
VEGA	19	6.33	1	1.2979	0.301158	10.06	0.904365
GVEGA	34	11.33	0.0882	0.1325	0.012432	0.000133	0.64247

Cuadro 7.22: Cuadro comparativo para 20 iteraciones de VEGA y GVEGA. Las métricas mostradas son NoD: número de soluciones no dominadas, RE: rata de error, S: separación, DG: distancia generacional y su varianza, MD: máxima dispersión.

7.6. UNA PRUEBA EN OPTIMIZACIÓN MULTIOBJETIVO

Algoritmo	NoD	%NoD	RE	S	DG	DG(Var)	MD
VEGA	18	6	1	3.13531	0.730552	784.231	0.96352
GVEGA	22	7.33	0	0.943789	0.969962	3730.18	0.964743

Cuadro 7.23: Cuadro comparativo para 100 iteraciones de VEGA y GVEGA. Las métricas mostradas son NoD: número de soluciones no dominadas, RE: rata de error, S: separación, DG: distancia generacional y su varianza, MD: máxima dispersión.

Algoritmo	NoD	%NoD	RE	S	DG	DG(Var)	MD
VEGA	9	3	1	20.8143	1.59941	1896.5	0.626
GVEGA	120	40	0	0.05808	0.003350	8.142 E-05	0.847186

Cuadro 7.24: Cuadro comparativo para 250 iteraciones de VEGA y GVEGA. Las métricas mostradas son NoD: número de soluciones no dominadas, RE: rata de error, S: separación, DG: distancia generacional y su varianza, MD: máxima dispersión.

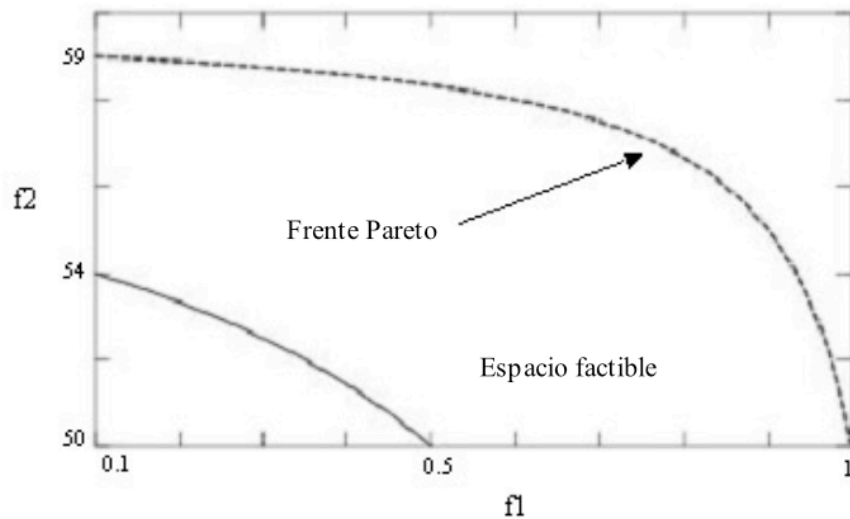


Figura 7.25: Función de prueba Optimización Multiobjetivo

7.6. UNA PRUEBA EN OPTIMIZACIÓN MULTI OBJETIVO

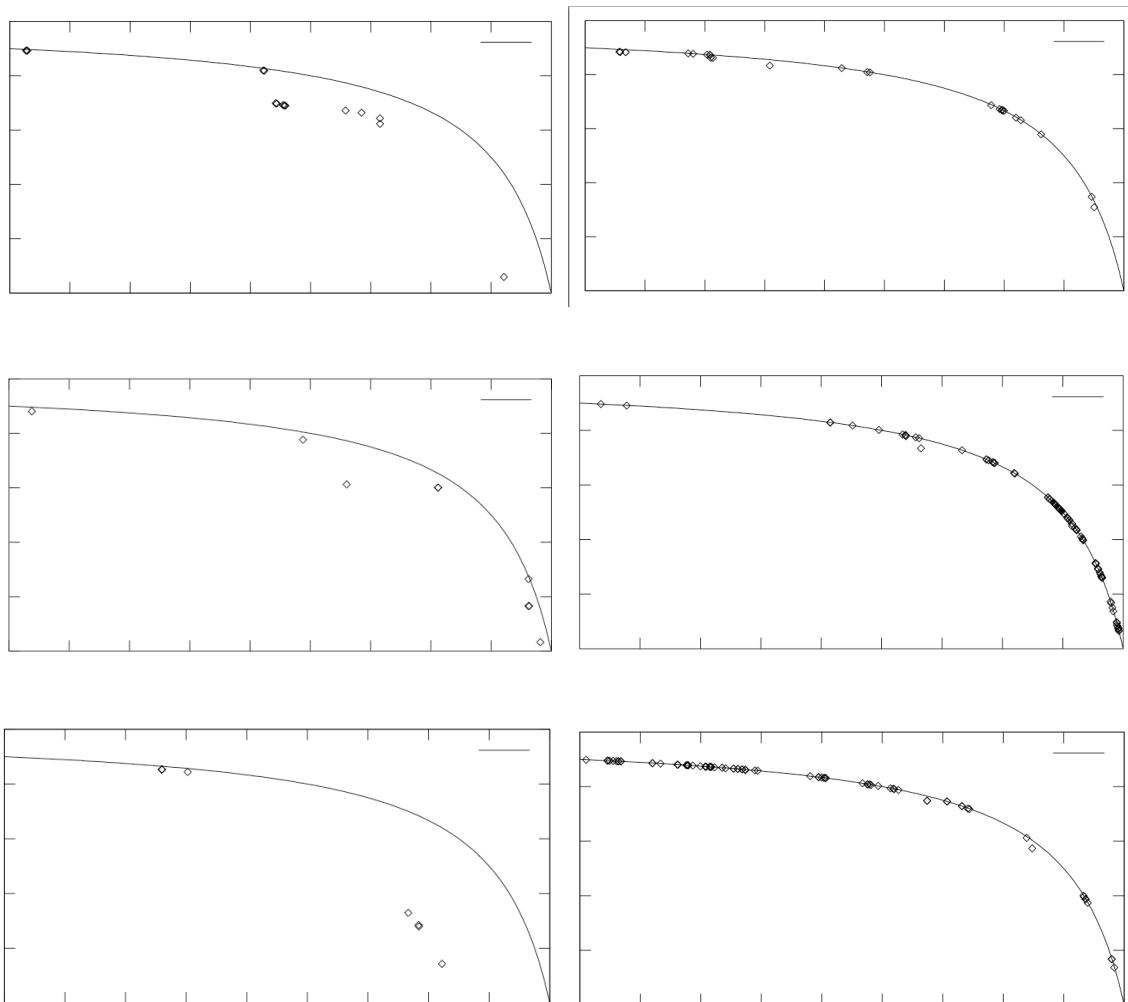


Figura 7.26: Comparación de 20 iteraciones (arriba), 100 iteraciones (al medio) y 250 iteraciones (abajo), para VEGA (izquierda) y GVEGA (derecha). Los ejes vertical (y) y horizontal (x) son f_1 y f_2 respectivamente.

7.6. UNA PRUEBA EN OPTIMIZACIÓN MULTIOBJETIVO

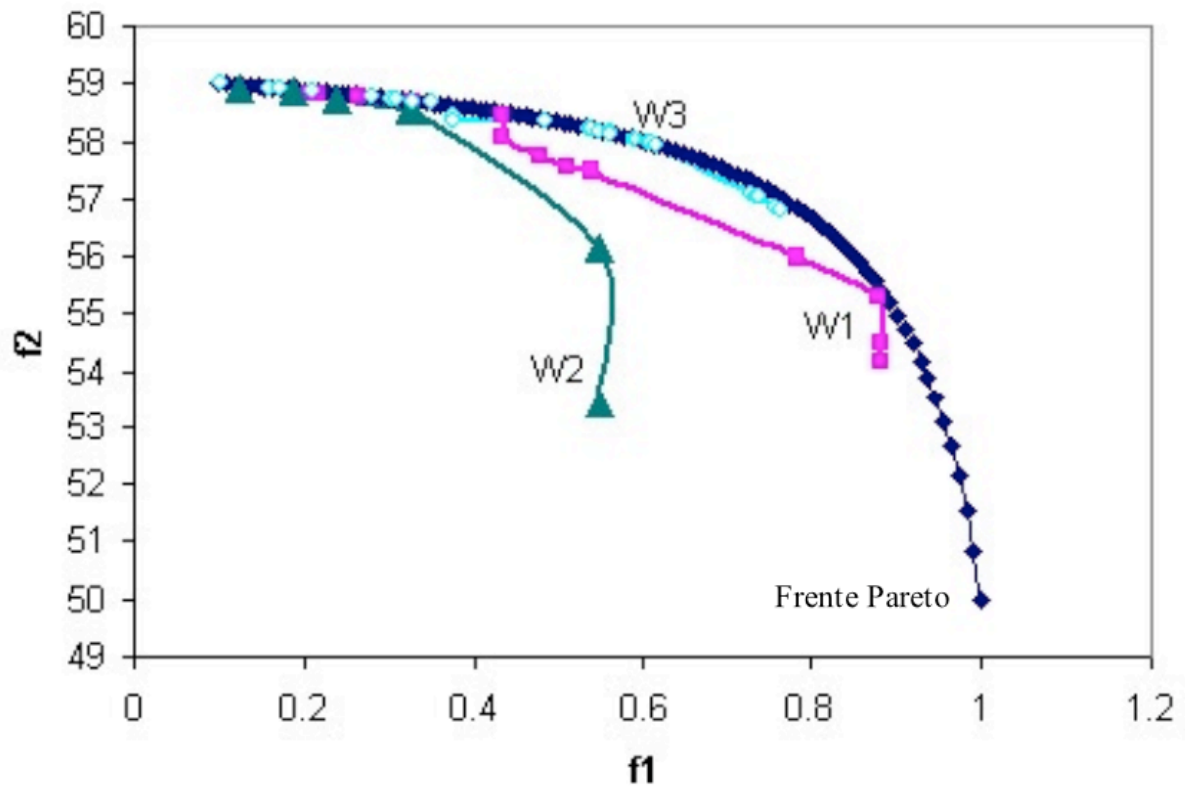


Figura 7.27: Comparación de corridas de GVEGA con factores de ponderación w_i variantes. El W_i indica el factor con valor 1.

Parte IV
Métodos Numéricos

*Puedo probar cualquier cosa con la estadística,
excepto la verdad.*

George Canning

8

Estimaciones Estadísticas

8.1 Introducción

La descripción del comportamiento, tendencia y naturaleza en general de los datos es un punto de vital importancia para la sana argumentación y el apropiado fundamento de las conclusiones de un trabajo científico experimental. Igualmente, para el desarrollo de un debate o discusión académica, e incluso para el caso de una investigación teórico-especulativa, el sustento estadístico apropiado de los datos en estudio, es de indiscutible importancia. Es cierto que la estadística no ofrece herramientas para probar de manera determinante una hipótesis, de hecho, como comenta la cita de Canning que abre este capítulo, lo que ofrece es un medio para apoyar o respaldar una hipótesis descartando una lista de otras hipótesis competidoras [15], pero nos da una serie de herramientas para “describir” tendencias, naturaleza o comportamientos de los datos, que nos colocan en terreno más firme para desarrollar conclusiones sobre los resultados de la investigación de campo.

Este capítulo describe algunas de las diferentes pruebas y métodos estadísticos de descripción de los datos, usados más frecuentemente para analizar los resultados experimentales de los AG.

8.2 Distribuciones y descriptores

Una distribución puede ser vista como el conjunto o el arreglo de valores de una variable donde se aprecia la frecuencia de ocurrencia de estos valores y su esparcimiento sobre el área lógica de distribución. En la sección 6.4.1 se mencionó la distribución de la adaptabilidad de los individuos en una población de un AG, esto se refiere a cómo están repartidos los valores de la variable adaptabilidad en la población, la frecuencia y esparcimiento de estos valores. Observe que hay una sutil diferencia entre los términos población y distribución. El primero refiere al conjunto de elementos o individuos del conjunto y el segundo se refiere a la distribución de los valores de alguna característica variable de estos elementos. Algunas distribuciones presentan ciertas características especiales que las hacen muy útiles al estudiar patrones de comportamiento de los datos en distribuciones similares o parecidas en naturaleza. Estas son ampliamente conocidas así como el conjunto de ecuaciones que describen su comportamiento, algunas de las más comunes son la distribución normal, binomial y de Poisson entre otras.

Un estimador o descriptor estadístico es una herramienta que permite calcular un estimado en algún patrón de comportamiento en los datos de una distribución como por ejemplo, las tendencias centrales. A continuación se mencionan algunos de los más importantes.

La media aritmética: Este es uno de los descriptores más sencillos y populares.

La media aritmética estima el valor sobre el que el *cluster* central se encuentra.

$$\bar{x} = \frac{1}{N} \sum_{j=0}^{N-1} x_j \quad (8.1)$$

La ecuación 8.1 da un estimado del valor central sobre el que se agrupa el resto, no obstante, cuando los valores en una distribución están muy separados, la media aritmética converge pobremente, es decir, no nos da una buena estimación de cuánto varían estos valores. Por ejemplo, si en una población de 4 individuos, con adaptabilidad de 2, 2, 8 y 8 respectivamente, según la

8.2. DISTRIBUCIONES Y DESCRIPTORES

ecuación 8.1, $\bar{x} = \frac{20}{4} = 5$, decir que en términos relativos la adaptabilidad de los individuos es alrededor de 5, puede ser válido pero no muy preciso.

La varianza: Con un valor central de la distribución ya estimado, se puede calcular la varianza alrededor de este valor, es decir, qué tanto se reparten los valores sobre este eje central. Una forma de calcular esto es a través de la varianza del conjunto de valores:

$$Var(A) = \frac{1}{N-1} \sum_{j=0}^{N-1} (x_j - \bar{x})^2 \quad (8.2)$$

$A = (x_0, x_1, \dots, x_{N-1})$, observe el caso del ejemplo anterior con $\bar{x} = 5$, y vea cuánto es la varianza acumulada del conjunto de valores en la distribución a la media aritmética. La ecuación 8.2 realiza este cálculo.

La desviación estándar: La desviación típica, o desviación estándar, se define como la raíz cuadrada de la varianza:

$$\sigma(A) = \sqrt{Var(A)} \quad (8.3)$$

$$DAM(A) = \frac{1}{N} \sum_{j=0}^{N-1} |x_j - \bar{x}| \quad (8.4)$$

$A = (x_0, x_1, \dots, x_{N-1})$, la ecuación 8.3 es la desviación estándar de un conjunto de datos A , la ecuación 8.4 es la desviación absoluta media que aunque no muy común e incluso impopular para muchos, es al igual que la estándar, un descriptor de la amplitud de la distribución [15].

La mediana: En una función de distribución de probabilidad $p(x)$, la mediana es el valor x_{med} para el que valores menores y mayores a x son igualmente probables:

$$\int_{-\infty}^{x_{med}} p(x) dx = \frac{1}{2} = \int_{x_{med}}^{\infty} p(x) dx \quad (8.5)$$

8.2. DISTRIBUCIONES Y DESCRIPTORES

La mediana de un conjunto de valores $A = (x_0, x_1, \dots, x_{N-1})$ se estima encontrando el valor x_i con igual número de valores sobre y bajo el.

$$x_{med} = \begin{cases} x_{(N-1)/2} & : \text{ si } N \text{ es impar} \\ \frac{1}{2}(x_{(N/2)-1} + x_{N/2}) & : \text{ si } N \text{ es par} \end{cases} \quad (8.6)$$

Observe en la ecuación 8.6 que cuando el número de valores en el conjunto A , o sea, $|A|$ es par, la mediana se toma como la media entre los dos valores centrales [15].

Inclinación: La inclinación o torsión (del inglés *skewness*), describe el grado de asimetría que una distribución presenta alrededor de su media.

$$Incl(A) = \frac{1}{N} \sum_{j=0}^{N-1} \left[\frac{x_j - \bar{x}}{\sigma} \right]^3 \quad (8.7)$$

En teoría, esta medida no-dimensional (no posee la misma unidad de los valores medidos, es decir, es solo una cifra), da un estimado de la forma de la distribución en cuanto a su simetría. Un valor positivo significa que la distribución es asimétrica pesada a la derecha, o sea, valores positivos mayores en x . Un valor negativo significa que es asimétrica con mayor extensión de su cola izquierda, o cola con negativos menores en x . Un valor cercano a cero, significa que la distribución es relativamente simétrica (no tiene una pronunciada inclinación).

Algunos autores se refieren a los estimadores descritos arriba clasificándolos según el *momento* de los datos, o la suma de las potencias enteras de los valores [15]. Con esta consideración en mente es fácil observar entonces que la media aritmética depende del primer momento de los datos, la varianza y la desviación estándar del segundo momento y la inclinación o grado de asimetría, depende del tercer momento de los datos.

Esta clasificación es una analogía y no debe confundirse con el *momento de orden* k de una variable aleatoria general respecto al origen [48].

8.3 Comparando distribuciones

En el trabajo experimental se hace imperativo la comparación de eventos, situaciones, poblaciones, etc. cuya variable de interés a nuestro estudio este representada en una distribución de probabilidad/frecuencia. Una de las primeras acciones a realizar cuando tenemos 2 distribuciones sobre las que queremos concluir alguna relación, es si tienen la misma media. Una medida para estimar cuán diferentes son dos medias, es el número de *errores estándar* entre ellas. El error estándar usualmente se calcula $\frac{\sigma}{\sqrt{N}}$, es decir, la desviación estándar entre la raíz cuadrada del número de puntos en la muestra.

8.3.1. La prueba $t - Student$

Con este concepto de error estándar, la estimación convencional para estimar si la diferencia entre medias es significativa o no, es la prueba $t - Student$. Cuando dos distribuciones tienen la misma varianza y se quiere estimar lo significativo o no de la diferencia de medias, la t de Student se calcula en tres pasos; primero se calcula el error estándar:

$$S_D = \sqrt{\frac{\sum_{i \in A} (x_i - \bar{x}_A)^2 + \sum_{i \in B} (x_i - \bar{x}_B)^2}{N_A + N_B - 2} \left(\frac{1}{N_A} + \frac{1}{N_B} \right)} \quad (8.8)$$

Donde A y B son las dos muestras a comparar, N_A y N_B son el número de puntos en cada muestra. Con S_D se procede a calcular t de la siguiente manera:

$$t = \frac{\bar{x}_A - \bar{x}_B}{S_D} \quad (8.9)$$

8.3. COMPARANDO DISTRIBUCIONES

y por último se evalúa lo significativo de t de acuerdo a la distribución *Student* con $N_A + N_B - 2$ grados de libertad. Cuadros con estos valores se pueden encontrar tabulados en la literatura, de manera que solo queda comparar, no obstante, si el lector desea saber de dónde provienen, la definición matemática de la función es:

$$A(t|v) = \frac{1}{v^{1/2}B(\frac{1}{2}, \frac{v}{2})} \int_{-t}^t \left(1 + \frac{x^2}{v}\right)^{-\frac{v+1}{2}} dx \quad (8.10)$$

$$A(0|v) = 0 \quad \text{y} \quad A(\infty|v) = 1$$

El número que se obtiene de esta evaluación esta en $(0 \dots 1)$ y es la probabilidad que $|t|$ sea al menos del tamaño observado por causa fortuita, para distribuciones con medias iguales. Esto significa que un valor pequeño para este número implica que la diferencia observada es “significativa”, en otras palabras, el valor obtenido de la evaluación de t de acuerdo a la distribución de Student, es la probabilidad que para v grados de libertad, el valor t que mide las diferencias observadas de las medias sea mas pequeño que el valor observado si las medias fueran las mismas, de manera que si $A(t|v) > 0,99$ por ejemplo, entonces $1 - A(t|v)$ es el nivel de significación por el que la hipótesis que las medias son iguales es descartada [15].

Observe que el caso planteado arriba es para distribuciones con igual varianza o al menos con diferencias no significativas. Otro caso posible es cuando esta diferencia si es significativa y aún así nos interesa saber si las medias son iguales o diferentes. En esta situación t se calcula de la siguiente manera:

$$t = \frac{\bar{x}_A - \bar{x}_B}{[Var(x_A)/N_A + Var(x_B)/N_B]^{1/2}} \quad (8.11)$$

Esta estimación estadística no es común para el caso del análisis en algoritmos genéticos y por tanto no se ampliará en este documento, no obstante, el lector puede referirse a [15] para un estudio completo del mismo. Intuitivamente, observe que si dos distribuciones tienen varianza significativamente diferente, entonces deben tener igualmente formas diferentes en cuyo caso la diferencia de medias no será muy útil

y una estimación somera pudiera incluso llevar a falacias. Este estimado estadístico debe aplicarse con suma cautela y tras análisis apropiado de las distribuciones.

8.3.2. Correlación lineal

Una manera muy común de medir o estimar la correlación lineal entre pares de cantidades es a través del coeficiente de correlación r de Pearson. Para el cálculo de este coeficiente se evalúan pares $(x_i, y_i), i = (0, 1, \dots, N - 1)$ para estimar si es posible observar una “relación” lineal entre los datos. La ecuación 8.12 muestra el cálculo de r [15].

$$r = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2} \sqrt{\sum_i (y_i - \bar{y})^2}} \quad (8.12)$$

las barras sobre las variables significan la media de las observaciones de esa variable (por ejemplo, \bar{x} es la media de los x_i). El coeficiente r puede tomar valores desde -1 a 1, ($-1 \leq r \leq 1$) y generalmente se denomina a un valor -1 como una correlación *completamente negativa* o +1 como *completamente positiva* que significa que los datos (x_i, y_i) están sobre una recta con pendiente positiva, de manera que si x se incrementa, y también se incrementa. Un coeficiente $r \approx 0$ significa que las variables x e y no están correlacionadas.

En el capítulo 5 se habló de algunos vicios del generador aleatorio de IBM llamado RANDÚ. Una de las pruebas que este generador no pudo pasar fue la de correlación en serie lineal, pues al generar puntos en el espacio (x, y, z) se observa un bien definido patrón en la formación de sub-planos, apreciados en la figura 5.6. Esto muestra una fuerte correlación entre ternas de datos de este generador.

La correlación lineal en serie es una prueba muy usada para generadores de números pseudo-aleatorios, no obstante, no es trivial su uso puesto que puede no apreciarse para todos los escenarios. Por ejemplo, para el caso de randú, si se generan puntos aleatorios en el plano y no en el espacio, el patrón no es visible. La correlación se observa fácilmente para el caso de tres planos.

8.3.3. Diferencia en distribuciones

En el análisis de resultados en algoritmos genéticos, entre los estimadores estadísticos más útiles están aquellos que nos permiten concluir si dos distribuciones cualesquiera son diferentes o si los datos que las componen son consistentes con alguna función de distribución, es decir, si no podemos asegurar con algún grado de confianza que son diferentes.

Muchos estimadores estadísticos han sido diseñados para responder diferentes preguntas acerca de las diferencias relativas en las distribuciones, no obstante, ellos dependen de la naturaleza de la distribución en sí. Por ejemplo, analizar relaciones entre dos distribuciones con datos continuos se basa en diferentes estimadores que con datos agrupados. Obviamente, pagando el precio de una pequeña pérdida de información, los datos continuos pueden agruparse en lotes representativos aunque para algunos esta pérdida de información es mejor evitarla y trabajar con los estimadores diseñados naturalmente para cada situación.

La prueba Chi-Cuadrada

Esta prueba es quizás la de más amplia aceptación y uso para datos agrupados. Suponga que tiene un número de eventos N_i que fueron observados en el i -ésimo grupo de un total de G grupos, y que n_i es el número esperado de estos eventos según alguna distribución conocida. El estimador chi-cuadrado se define como:

$$\chi^2 = \sum_i^G \frac{(N_i - n_i)^2}{n_i} \quad (8.13)$$

Analizando la ecuación 8.13, podemos ver que para un valor muy grande de χ^2 , la suposición que el conjunto de los N_i es obtenido de una distribución representada por el conjunto de los n_i (hipótesis nula), es muy improbable. Igualmente, si para un valor $j \leq G$, tenemos que $N_j \neq 0$ y $n_j = 0$ entonces χ^2 se hace infinito y esto es lógico porque en este caso los N_i no pudieron haber sido obtenidos de los n_i .

8.3. COMPARANDO DISTRIBUCIONES

Kolmogorov-Smirnov

Esta prueba de similitud de distribuciones es también una de las mas ampliamente aceptadas para datos continuos como función de una sola variable. La lista de puntos de datos puede ser convertida a un estimador $S_N(x)$ de la función de distribución acumulativa de la distribución de probabilidad de donde fueron obtenidos. Si N eventos están en x_i , con $i = 0, \dots, N - 1$, entonces $S_N(x)$ es la *función indicadora* que clasifica los datos a la izquierda de x . El estimador Kolmogorov-Smirnov D se define como el valor máximo de la diferencia absoluta entre dos funciones de distribución acumulativas. Para comparar la $S_N(x)$ de un conjunto de datos con la de una función de distribución acumulativa conocida $P(x)$, el estimador es:

$$D = \max_{-\infty < x < \infty} |S_N(x) - P(x)| \quad (8.14)$$

nivel significativo,

$$Q_{KS} = 2 \sum_{j=1}^{\infty} (-1)^{j-1} \exp^{-2j^2\lambda^2} \quad (8.15)$$

con

$$Q_{KS}(0) = 1, Q_{KS}(\infty) = 0$$

finalmente,

$$Prob(D > \text{valor obtenido}) = Q_{KS} \left(\left[\sqrt{N_e} + 0,12 + \frac{0,11}{\sqrt{N_e}} \right] D \right) \quad (8.16)$$

que nos da la probabilidad aproximada que D sea mayor que el valor obtenido (para descartar la hipótesis nula que las distribuciones son iguales) con el rango significativo calculado en la ecuación 8.15. N_e es el número efectivo de puntos de datos.

8.4 Evaluación del AG

En la sección 6.3.2 se diferenció entre la función objetivo del AG $f(x)$, y la función de adaptabilidad $\Lambda(x)$ con la finalidad de clarificar el hecho que en algunos casos los datos deben ser transformados para poder ser usados por algún operador. Por ejemplo, el método de selección por rueda de ruleta necesita que la adaptabilidad de los individuos sean cantidades positivas. Si la función objetivo arroja valores negativos, estos valores deben ser transformados a cantidades positivas manteniendo su rango o jerarquía relativa de calidad dentro de la población. Los operadores de selección usualmente trabajan bajo un criterio selectivo basado en el nivel de adaptabilidad, esto implica que para problemas de optimización donde se persigue un máximo, se busque un valor mayor en la adaptabilidad, pero si se persigue un valor mínimo, entonces la meta es lo contrario. Esta situación se puede resolver estandarizando la función objetivo mediante su transformación a un problema de maximización, o aplicando transformaciones a los valores de la evaluación de la función objetivo, es decir, usar una función de calidad Λ .

Para mayor simplicidad, en el resto de este capítulo se usará la función objetivo como el indicador de calidad de la adaptabilidad del individuo, es decir, quedando claro lo expuesto en el párrafo anterior, en lo adelante se tomará $\Lambda(x) = f(x)$ para la discusión de los estimadores estadísticos en el análisis de resultados experimentales del algoritmo genético.

El primer paso para la evaluación de un AG, es definir qué se quiere evaluar, es decir, reunir las variables, interrogantes, ideas y criterios por evaluar, y luego diseñar un mecanismo para “medir” las observaciones experimentales con respecto a estos elementos. A estos mecanismos de medición se les llama *métricas de rendimiento* y su función es la tabulación de un esquema comparativo de calidad con respecto a una variable observada.

Muchas métricas de rendimiento de un AG han sido propuestas. En [30], De Jong propone dos esquemas de medición que llamó *métrica caliente* y *métrica fría* (del inglés *on-line* y *off-line* respectivamente). Los nombres se refieren a la diferencia en la naturaleza de aplicaciones hipotéticas de un AG. En aplicaciones en frío, muchas evaluaciones de la función objetivo pueden ser realizadas y un registro de la mejor solución encontrada es llevado para su posterior uso. Estas aplicaciones no tienen

restricciones en cuanto al tiempo de respuesta. Por otro lado, cuando la aplicación necesita de soluciones buenas en espacios cortos de tiempo (problemas de control de procesos por ejemplo), el algoritmo debe responder *en caliente* a esta situación generando buenas soluciones en tiempos de respuesta aceptables.

Otras métricas de rendimiento han sido propuestas en la literatura, no obstante, el diseño de estas métricas dependerá de los criterios sobre los que busca concluir el investigador en la evaluación de un aspecto específico del AG. A continuación se describen algunas métricas de rendimiento muy comunes en la literatura:

M1: Métrica de rendimiento en frío. Se refiere a la media de la mejor adaptabilidad por generación calculada sobre todas las pruebas (corridas) experimentales. Se define:

$$M1 = \frac{1}{E} \sum_{e=1}^E \left(\frac{1}{T} \sum_{t=1}^T f_t^*(A) \right) \quad (8.17)$$

donde $f_t^*(A) = \text{mejor}\{f(a_1), f(a_2), \dots, f(a_{|P|})\}$ o el mejor valor de la evaluación de la función de todos los individuos a de la población P en la generación t de un total de T generaciones corridas en E experimentos.

M2: Métrica de rendimiento en caliente. Se refiere a la media de la adaptabilidad de la población por generación calculada sobre todas las pruebas experimentales. Se define:

$$M2 = \frac{1}{E} \sum_{e=1}^E \left(\frac{1}{T} \sum_{i=1}^T \left(\frac{1}{|P|} \sum_{j=1}^{|P|} f_t(a_j) \right) \right) \quad (8.18)$$

E es el número total de corridas experimentales y T el número total de generaciones del algoritmo.

M3: Métrica de rendimiento final. Se refiere a la media de la mejor adaptabilidad en la última generación del algoritmo. Se define:

$$M3 = \frac{1}{N} \sum_{e=1}^E f_T^*(A) \quad (8.19)$$

donde $f_T^*(A) = mejor\{f(a_1), f(a_2), \dots, f(a_{|P|})\}$ o el mejor valor de la evaluación de la función de todos los individuos a en la última generación T sobre un total de E corridas experimentales.

M4: Métrica de rendimiento global. Se refiere a la media de la mejor solución obtenida por el algoritmo promediada por todas las corridas experimentales. Se define:

$$M4 = \frac{1}{E} \sum_{e=1}^E mejor\{F_1^*(A), F_2^*(A), \dots, F_e^*(A)\} \quad (8.20)$$

$F_e^*(A) = mejor\{f_1(A), f_2(A), \dots, f_r(A)\}$ o la mejor solución encontrada en todas las generaciones en la corrida experimental e .

Estas métricas permiten evaluar el desempeño del algoritmo en términos de la convergencia de la adaptabilidad y con respecto al tiempo (generaciones), pero ¿Qué pasa si se desea comparar el desempeño de dos algoritmos diferentes (basados en poblaciones de la misma naturaleza), o de operadores diferentes aplicados a una misma población?, o si se desea saber ¿Cómo se afectó una variable al hacer un cambio cualquiera?, etc. En primer lugar, aunque en la práctica suele a veces obviarse o darse por sentado, debemos comprobar si las distribuciones de la variable que queremos estudiar en las poblaciones son de la misma naturaleza (ambas son normales o pseudo-normales por ejemplo), de manera de no comparar peras con manzanas. Esto lo podemos hacer con una prueba chi-cuadrado o kolmogorov-smirnov según la forma de los datos (ver sección 8.3.3). Si la prueba sugiere que ambas distribuciones son de similar naturaleza, entonces podemos aplicar un estimador de tendencia central a ambas (como la media aritmética) y comparar usando una prueba t – *Student*. El resultado de esta prueba, junto con su nivel o grado de confianza, nos da una base mas completa y robusta para argumentar nuestras conclusiones sobre las diferencias

observadas. Debe recordar (como dice la cita de apertura de este capítulo), que estos estimadores no le permiten al investigador demostrar o probar una conclusión sobre una observación medida, tan solo le permiten afirmar con cierto grado de certeza o confianza, que lo que se observa y mide, *sugiere* lo que se concluye.

8.4.1. Experimentación con el AG

Observe que las métricas descritas en la sección 8.4 son estimadores de promedios sobre un número total de experimentos E . Esto significa que para hacer más robustas las conclusiones sobre los resultados experimentales, no podemos basarnos en una sola corrida del algoritmo. La razón es que estos son métodos estocásticos de búsqueda y por lo tanto debemos construir una sólida base experimental que abarque diferentes configuraciones de parámetros. Por ejemplo, hacer una corrida de un AG y concluir sobre su desempeño basándonos en los estimadores estadísticos aplicados a las poblaciones en el tiempo (en las diferentes generaciones), no es científicamente apropiado puesto que existe la posibilidad que la configuración de parámetros aleatorios del algoritmo haya beneficiado el comportamiento de alguna variable de estudio. Esto puede llevarnos a concluir argumentos imprecisos y en el peor caso, significativamente incorrectos. El escenario es aún peor si la población es relativamente pequeña y el número de iteraciones o generaciones no es grande.

Para la aplicación de las métricas descritas, es necesario no solo diseñar el AG con parámetros de corridas aceptables (por ejemplo número de generaciones “suficientemente” grande como para apreciar la convergencia de la población), sino también hacerlo repetidamente con configuraciones distintas. Esto es imperativo cuando se desea realizar un análisis teórico del comportamiento. Obviamente, no hay una regla determinista que permita asignar valores a los diferentes parámetros, esto depende en primer lugar del sentido común del investigador y de su experiencia en el campo.

El diseño experimental con un AG, debe incluir una serie de corridas con los mismos parámetros de control pero variando los argumentos aleatorios, dicho de otra forma, debe correrse varias veces el algoritmo con los mismos parámetros de control como tamaño de la población, número de generaciones, probabilidad de cruce, mutación y otros de configuración de operadores, pero debe cambiarse la *semilla aleatoria* que inicia las desviadas uniformes para los eventos de Bernoulli.

8.5 Importancia para el trabajo científico

En el trabajo científico experimental, una de las etapas mas importantes, quizás tan importante como el método mismo y el control en sí, es la rigurosidad en la comunicación de resultados, hallazgos, conclusiones, etc. Una clara, precisa, bien fundamentada, completa, objetiva y amplia conclusión, dará paso a una rica y posiblemente provechosa discusión donde se puedan nutrir aspectos de la investigación y abrir, encontrar, proponer; nuevos enfoques, aspectos, campos de aplicación, etc. Una conclusión también puede ser sesgada, especulativa, subjetiva, confusa e imprecisa, perjudicando así el trabajo operativo que resume, pues es posible que muchos investigadores no le den la seriedad debida al trabajo experimental por los vicios vistos en la conclusión.

La figura 8.1 representa un ciclo típico en el quehacer de la experimentación con los algoritmos genéticos y cualquier otro método basado en poblaciones. Una vez que bajo el diseño experimental se han obtenido distribuciones observadas de alguna variable de interés sobre las que necesitamos concluir alguna relación; como por ejemplo, ver cómo afecta el incremento del parámetro k en la selección por torneo, aseguramos que ambas distribuciones son comparables, es decir, de naturaleza semejante, mediante la aplicación de la(s) prueba(s) estadística apropiada(s) y de ser procedente, se estima alguna tendencia central en las distribuciones para luego evaluar si la diferencia entre ellas es significativa o no y tener fundamentos apropiados para la elaboración de conclusiones. Generalmente, estas conclusiones abren paso a una discusión abierta y razonablemente especulativa con la intención es estimular la emergencia de nuevos enfoques, extensiones, modificaciones, críticas, etc. de lo experimentado.

La conclusión se refiere al conjunto de argumentos y recopilaciones de probanzas de donde se infiere y se hace examen crítico objetivo de lo experimentado, no al punto monográfico de la forma del documento. Esto quiere decir que la importancia que se le da a las conclusiones en este trabajo no tiene que ver con el punto en sí, de hecho, un documento es igualmente completo y científicamente válido, si carece de un punto llamado “conclusiones” (aunque algunos metodólogos puedan disentir), pero que reporta de manera clara y amigable al lector los elementos discutidos, bien sea en un análisis y discusión de resultados, una crítica observacional o cualquier sección que discuta la experimentación bajo los términos arriba descritos.

8.5. IMPORTANCIA PARA EL TRABAJO CIENTÍFICO

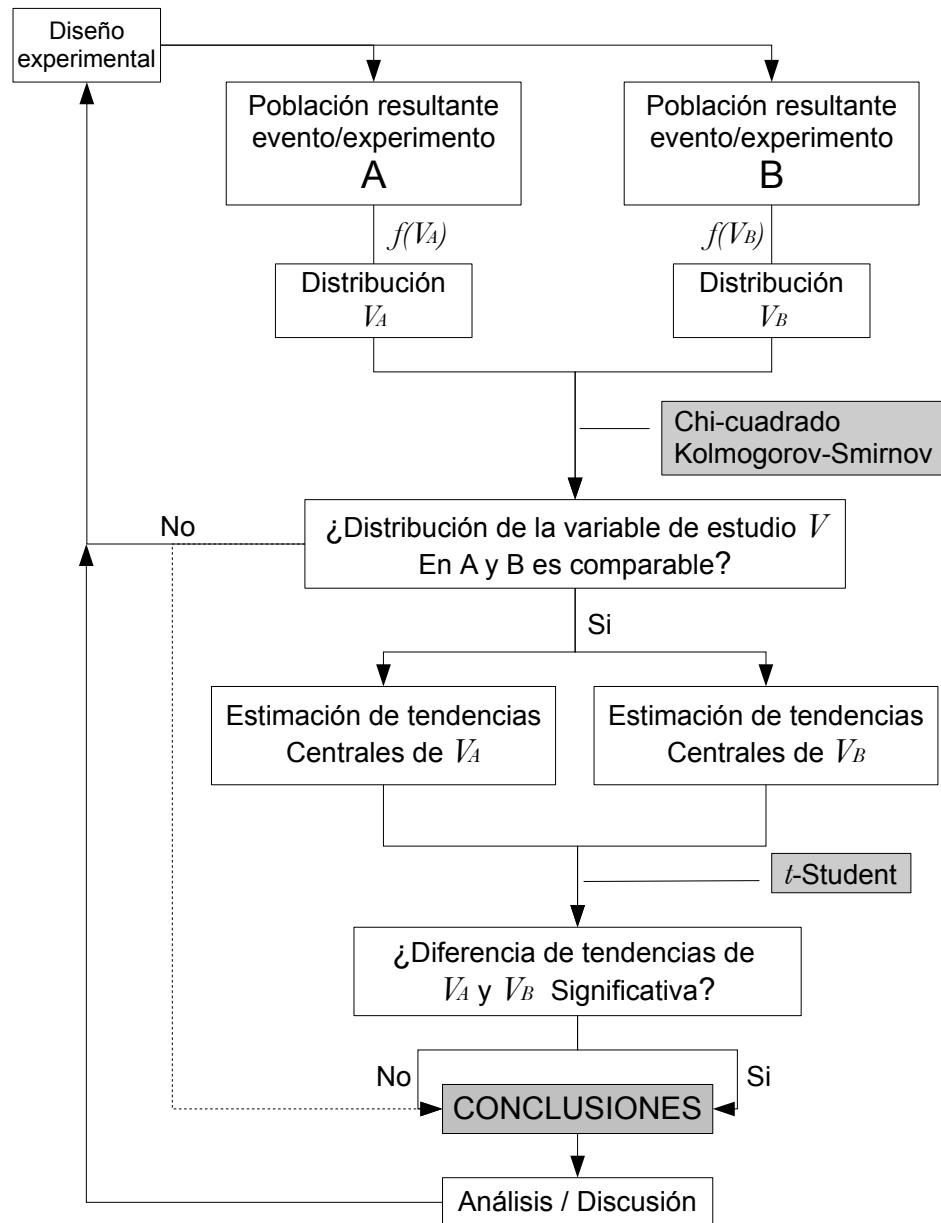


Figura 8.1: Un ciclo experimental típico en un AG

En la búsqueda de la verdad hay preguntas que no son importantes. ¿De qué está hecho el universo?, ¿Es eterno, infinito? Si un hombre detuviera su búsqueda para responderlas, moriría antes de hallar el camino a la iluminación.

Buda

9

Theorice

En esta discusión filosófica especulativa sobre elementos del trabajo experimental científico, discuto algunos tópicos *no escritos* o *no oficiales* pero que considero que deben tener en cuenta los investigadores que se embarcan en una investigación de campo, como por ejemplo, los estudiantes que comienzan sus trabajos de grado en la Maestría en Ciencias de la computación.

La primera reflexión es sobre los resultados de la investigación. ¿Qué se ha conseguido? ¿A qué se deben esos resultados? ¿Cómo pueden mejorarse o qué nuevas posibilidades plantean? A lo largo de este documento se discutió sobre la importancia de las conclusiones en el trabajo científico en general y experimental en particular. La necesidad de argumentar de manera correcta, precisa y fundamentada, los hallazgos. Reportarlos sin asunción o sesgos infundados y observar extrema cautela en las inferencias basadas en correlaciones descubiertas. Estos son elementos de sentido común en la investigación que le añaden rigurosidad científica al proceso, no obstante, el uso de argumentos especulativos; basados en observaciones informales, creencias o incluso en la intuición, no está del todo vetado en el proceso de investigación, al contrario, son una parte interesante que pueden enriquecer el debate académico y pueden plantear nuevos enfoques al trabajo realizado. Esto no debe confundirse con el reporte de resultados ni con el proceso experimental. En muchos

trabajos de ingeniería, e incluso de las ciencias básicas y sociales, los investigadores incluyen comentarios, ideas e hipótesis buscando dar explicación o respuestas a las interrogantes que quedan abiertas. Estos comentarios generalmente enriquecen y orientan el debate científico a discusiones con el potencial de rendir productivos frutos.

Una reflexión directa de esta situación, es que el tesista de postgrado puede buscar orientar la discusión de la evaluación de su trabajo, al terreno donde él piensa que ha dado un mejor aporte, a donde él siente que su investigación tiene mas que ofrecer. Incluyendo una discusión razonablemente especulativa de los resultados obtenidos, puede despertar la curiosidad académica y la sana crítica científica de los investigadores que le evalúan, y puede enriquecer significativamente los resultados obtenidos. Claro está, esta discusión debe estar separada del proceso de control y método de la investigación.

Una pregunta que planteada de diferentes maneras, le he escuchado a investigadores miembros de un jurado evaluador de tesis doctorales y de maestría, es ¿Con qué aspecto de tu trabajo te sientes mas satisfecho?, ¿Qué has logrado, cuál fue tu verdadero aporte?, ¿Cómo crees que esto puede aplicarse a tal situación? Todas estas preguntas parecieran buscar “romper el hielo” de la defensa e invitar al tesista a discutir argumentos de fondo de su investigación. Una forma de tomar el punzón y orientar la discusión, es plantear de antemano (de manera apropiada y académica), un elemento de debate. Para esto será inevitable incluir elementos especulativos pero debe recordar que el trabajo experimental ya se expuso, ahora valen ciertas libertades.

Un buen ejercicio para el tesista de postgrado, es plantearse todas estas y otras interrogantes, y discutir las con su tutor antes de presentar su trabajo de grado al jurado evaluador. Esto le ayudará a hacer reflexiones de fondo de lo alcanzado, lo que indiscutiblemente lo preparará mejor para una defensa.

Índice alfabético

- t* – *Student*, 137, 144
- ácido desoxirribonucleico, 17
- ápeiron, 7
- óvulo, 16

- De Jong, Kenneth, 142

- aberración lineal, 33
- adaptabilidad, 65
- adenina, 21
- ADN, **21**, 64
- aleatoriedad, 4, 33, 34
 - números aleatorios, 32
- alelo, 17, 19, 20
- algoritmo genético, 2, 3, 16, **50**
 - búsqueda ciega, 51
 - clases de, 52
 - celular, 52
 - cliente servidor, 52
 - isla, 52
 - jerárquico, 52
 - independencia del problema, 51
 - paralelismo, 51
- algoritmo genético simple, **52**, 53
- algoritmos seminuméricos, 32, 33
- altruismo animal, 14
- aminoácido, 21
- ancho de banda, 55
- arjé, 7
- Arquímedes, 26

- búsqueda, 23, 24
 - espacio de, 24
 - espacio factible, 24
 - espacio solución, 24
- Bernoulli, 54, 76, 145
- biología evolutiva, 7, 8

- célula, 17
- cariotipo, 22
- chi-cuadrada, 140, 144
- cigoto, 16, 17
- citosina, 21
- cluster, 134
- coma flotante, 56
- competencia, 11, 13
- computación evolutiva, 1, 2, 15, 16
- congruencia lineal, 37
- conjunto óptimo Pareto, 29
- creación especial , **6**
- cromátido, 20, 75
- cromátidos, 108
- cromosoma, 16, 18, 19, 63, 64, 67
 - codificación, 56
 - descodificación, 56
 - homólogo, 16
 - representación de, 55
 - textbf, 16
- cruce, 19, 20

- Darwin

- Charles, 7, 8, 11
- Erasmus, 7
- de Mileto
 - Anaximandro, 7, 8
 - Tales, 7
- Demócrito, 7
- desviación absoluta media, 135
- desviación estándar, 135
- desviada uniforme, 4, 35, 36, 145
- diferencia de distribuciones, 140
- diploide, 16, 18, 19
- distancia Hamming, 58, 81
- distribuciones
 - binomial, 134
 - normal, 134
 - Poisson, 134
 - tipos de, 134
- división de reducción, 16
- dominancia, 18, 29
- dominancia genética, 10

- error estándar, 137
- espacio de soluciones factibles, 51
- espacio solución, 51, 64, 65
- espermatozoide, 16
- estadística, 133
- estrategia evolutiva, 76
- eureka, 26
- evolución, 6, 8, **9**, 11

- fenotipo, 12, 17, 18, 58, 61, 64, 66
- fentopio, 77
- frente Pareto, 29
- función de adaptabilidad, 53, 65
- función de calidad, 65
- función indicadora, 141

- gameto, **16**, 17

- gametogénesis, 16, 17, 19
- gen, 17, 58, 63
- genética, 15
- generador estándar mínimo, 38
- genotipo, 12, 17, 20, 61, 64, 66
- GGA, 119, 124
- gnuplot, 39
- grafo Hamiltoniano, 119
- grafos Hamiltonianos, 28
- Gray, Frank, 58
- guanina, 21
- GVEGA, 124

- haploide, 16, 19
- Heráclito de Éfeso, 13
- heterocigoto, 18
- heurística, **26**
- hipótesis nula, 140
- Holland, John, 50, 80
- homólogo, 18
- homocigoto, 18

- IEEE, 56
- inclinación, 136
- Inteligencia Artificial, 1, 26

- knuth Donald, 32
- Kolmogorov-Smirnov, 141, 144
- kosmoi, 7

- Leucipo, 7
- locus, 17, 19, 76

- máquina estocástica, 33
- métricas de rendimiento, 142
 - caliente, 142, 143
 - final, 144
 - fría, 142, 143

- global, 144
- Mann-Whitney, 116
- media aritmética, 134
- mediana, 135
- meiosis, 16, 17, 19
- Mendel Gregor, 9, 10, 17, 20
- meta-heurística, 26
- Minsky, Marvin, 1
- modelos seminuméricos , 4
- momento, 136
- Morgan Thomas, 9
- mutación, 8, 10, 11, 13, 19–21
 - duplicación, 21, 22
 - poliploidización, 21, 22
 - puntual, 21
 - substitución de reemplazo, 21
 - substitución silenciosa, 21
 - transición, 21, 22
 - transversión, 21
- normalización, 66, 68
- nucleótido, 21
- om, 7
- operadores de cambio, 50
 - selección, 51
- operadores de variación, 67
 - cruce, 74
 - mutación, 76
 - selección, 67
- optimización, 23, 24
 - combinatoria, 27, 119
 - multiobjetivo, 28, 124
 - textbf, 27
- paradigma neo-Darwiniano, 11
 - características, 14
 - procesos básicos, 11
- paradoja de predictibilidad, 34
- pirimidina, 21
- ploide, 19
- polimerasa, 21
- postulados de Darwin, 10
- presión selectiva, 67, 80, 81, 98
- problema
 - definición de, 24
- procesos estocásticos, 2
- prole, 16
- prueba de Turing, 39
- punto de inflexión, 27
- purina, 21
- quiasma, 19, 20, 75
- recesividad genética, 10
- recesivo, 18
- representación
 - desordenada, 62
- representación binaria, 55, 104
- representación binaria desordenada, 61
- representación Gray, 58, 104
- representación real
 - simple, 63
- reproducción, 11, 13, 14
 - sexual, 14
- riscos de Hamming, 61, 116
- rueda de ruleta, 142
- selección, 11, 13, 19
 - adaptabilidad proporcional, 67
 - intensidad de, 70
 - rango, 71
 - rueda de ruleta, 67, 80
 - torneo, 72, 80, 83
 - truncado, 70
- selección familiar, 19

selección natural, 10, 19, 53
selección sexual, 19
semilla aleatoria, 145

t-Student, 137
teoría de cuerdas, 7
teoría de la evolución, 4
timina, 21
topografía de adaptación, 12
TRNG, 34
TSP, 28
Turing, Alan, 39

Ussher James, 6

variables de diseño, 58
variables objetivo, 58
varianza, 135, 137
vedas, 7
VEGA, 124
vendedor viajero, 28, 64, 77, 120
Vries, Hugo de, 20

Watson y Crick, 21
Wilcoxon, 116
Wright Sewall, 12

Bibliografía

- [1] Thomas Bäck. Optimal mutation rates in genetic search. In S. Forrest, editor, *Proceedings of the 5th International Conference on Genetic Algorithms*, pages 2–8. Morgan Kaufmann, 1993.
- [2] Thomas Bäck. Selective pressure in evolutionary algorithms: A characterization of selection mechanisms. In *Proceedings of the First IEEE Conference on Computational Intelligence*, volume 1, pages 57–62. IEEE, 1994.
- [3] Tobias Blickle and Lothar Thiele. A comparison of selection schemes used in genetic algorithms. Technical Report Nr. 11, Swiss Federal Institute of Technology, Gloriastrasse 35, 8092 Zurich, Suiza, Diciembre 1995.
- [4] Tobias Blickle and Lothar Thiele. A mathematical analysis of tournament selection. In Larry J. Eshelman, editor, *Proceedings of the 6th International Conference on Genetic Algorithms*, pages 9–16. Morgan Kaufmann, 1995.
- [5] R.I. Bowman. Morphological differentiation and adaptation in the galápagos finches. *University of California Publications in Zoology*, (58):1–302, 1961.
- [6] D.J. Cavicchio. *Adaptive search using simulated evolution*. PhD thesis, University of Michigan, Ann Arbor, 1970.
- [7] Mandira Chakraborty and Uday Chakraborty. An analysis of linear ranking and binary tournament selection in genetic algorithms. In *Proceedings of the International Conference on Information, Communications and Signal Processing ICICS97*, Septiembre 1997.
- [8] Richard Creath and Jane Maienschein, editors. *Biology and Epistemology*. Cambridge University Press, 2000.
- [9] Biology Daily. <http://www.biologydaily.com>, Marzo 2011.

- [10] Charles Robert Darwin. *The Origin of Species*. Oxford University Press, 1859 original edition, 1998.
- [11] Richard Dawkins. *The Selfish Gene*. Oxford University Press, 3rd edition, May 2006.
- [12] Kalyanmoy Deb. *Binary and floating-point function optimization using messy genetic algorithms*. PhD thesis, University of Alabama, 1991.
- [13] Kalyanmoy Deb. Encoding and decoding functions. In T. Bäck, D.B. Fogel, and Z. Michalewicz, editors, *Evolutionary Computation 2: Advanced Algorithms and Operators*, chapter 2, pages 4–11. Institute of Physics Publishing, 2000.
- [14] Kalyanmoy Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley and Sons, 2001.
- [15] William H. Press et al. *Numerical Recipes in C++*. Cambridge University Press, second edition, 2002.
- [16] David B. Fogel. Principles of evolutionary processes. In Thomas Back, David B. Fogel, and Zbigniew Michalewicz, editors, *Handbook of Evolutionary Computation*, chapter A2.1, pages 1–3. IOP Publishing Ltd and Oxford University Press, 1997.
- [17] Fourmilab. <http://www.fourmilab.ch/hotbits/>. mayo 2010.
- [18] Scott Freeman and Jon C. Herron. *Evolutionary Analysis*. Prentice Hall, second edition, 2001.
- [19] D. E. Goldberg. Messy genetic algorithms: Motivation, analysis, and first results. *Complex Systems*, (3):493–530, 1989.
- [20] David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, 1989.
- [21] David E. Goldberg and Kalyanmoy Deb. A comparison of selection schemes used in genetic algorithms. In G.J.E. Rawlings, editor, *Foundations of Genetic Algorithms*, pages 69–93. Morgan Kauffmann, 1991.
- [22] V.S. Gordon and D. Whitley. Serial and parallel genetic algorithms as function optimizers. In *5th IGCA*, pages 177–183, 1993.

- [23] P.R. Grant. Natural selection and darwin's finches. *Scientific American*, 4(265):60–65, October 1991.
- [24] John Grefenstette and James Baker. How genetic algorithms work: A critical look at implicit parallelism. In David Schagger, editor, *Proceedings of the 3^d International Conference on Genetic Algorithms*, pages 20–27. Morgan Kaufmann, 1989.
- [25] William D. Hamilton. The genetical evolution of social behaviour. *Journal of Theoretical Biology*, 1(7):1–16, 1964.
- [26] Antoni Hoffman. *Arguments on Evolution: a Paleontologist's Perspective*. Oxford University Press, 1989.
- [27] John H. Holland. Outline for a logical theory of adaptive systems. *Journal of the ACM*, (3):297–314, 1962.
- [28] Marcus Hutter. Fitness uniform selection to preserve genetic diversity. In *Proceedings of the 2002 IEEE Congress on Evolutionary Computation*, volume 1, pages 783–788. IEEE, 2002.
- [29] Robert E. Johnson and Hui Liu. Pseudo-random numbers: Out of uniform. In *SAS Users Group International Conference, SUGI*, 2000.
- [30] Kenneth De Jong. *An analysis of the behavior of a class of genetic adaptive systems*. PhD thesis, Department of Computer Science, University of Michigan, 1975.
- [31] Donald E. Knuth. *The Art of Computer Programming*, volume 2: Seminumerical Algorithms. Addison-Wesley, third edition, November 1997.
- [32] Snyder L. and Champness W. Molecular genetics of bacteria. *American Society of Microbiology*, 1997.
- [33] J.C. Lagarias. Pseudorandom numbers. *Statistical Science*, 8(1):31–39, 1993.
- [34] Ernst Mayr. *Toward a New Philosophy of Biology: Observations of an Evolutionist*. Harvard University Press, September 1989.
- [35] Gregor Johann Mendel. Versuche über pflanzenhybriden. *Verhandlungen des naturforschenden Vereines in Brünn*, pages 3–47, 1865.

- [36] B. Miller and David E. Goldberg. Genetic algorithms, tournament selection, and the effects of noise. Technical Report 95006, University of Urbana Champaign Illinois, Genetic Algorithm Laboratory, 1995.
- [37] Marvin Minsky. *The Society of Mind*. Simon & Schuster, 1985.
- [38] Tatsuya Motoki. Calculating the expected loss of diversity of selection schemes. *Evolutionary Computation*, 10(4):397–422, 2002.
- [39] Heinz Mühlenbein and Dirk Schlierkamp-Voosen. Predictive models for the breeder genetic algorithm -i. continuous parameter optimization. *Evolutionary Computation*, 1:25–49, 1993.
- [40] Stephen K. Park and Keith W. Miller. Random number generators: Good ones are hard to find. *Communications of the ACM*, 31(10):1192–1201, Octubre 1988.
- [41] Raymond C. Paton. Principles of genetics. In Thomas Back, David B. Fogel, and Zbigniew Michalewicz, editors, *Handbook of Evolutionary Computation*, chapter Evolutionary Computation: The Background, pages A2.2:1 – A2.2:9. IOP Publishing Ltd and Oxford University Press, 1997.
- [42] Judy Pearsall, editor. *Concise Oxford Dictionary*. Oxford University Press, 2001.
- [43] John Philoponus and J.O. Urmson. *Philoponus On Aristotle's Physics 5-8 with Simplicius on Aristotle on the Void*. Ancient Commentators on Aristotle Series. Cornell University Press, February 1994.
- [44] D.K. Pratihari. *Soft Computing*. Alpha Science, 2008.
- [45] Federal Information Processing Standards Publication. Data encryption standard. Technical Report 46, U.S. Department of Commerce, National Bureau of Standards, 1977.
- [46] Gadagkar Raghavendra. *Survival Strategies, Cooperation and Conflict in Animal Societies*. Harvard University Press, 1997.
- [47] Random.org. <http://www.random.org/integers/>. mayo 2010.
- [48] Sixto Ríos. *Métodos Estadísticos*. Ediciones del Castillo S.A., 1977.

- [49] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Artificial Intelligence Series. Prentice Hall, 1995.
- [50] José Sánchez-Velazco. *Gendered Selection Strategies for Genetic Algorithms*. PhD thesis, The University of Birmingham, School of Computer Science, 2005.
- [51] José Sánchez-Velazco and John Bullinaria. Sexual selection with competitive/co-operative operators for genetic algorithms. In O. Castillo, editor, *IASTED Intenational Conference on Neural Networks and Computational Intelligence*, pages 191–196. ACTA, May 2003.
- [52] J.D. Schaffer. Multiple objective optimization with vector evaluated genetic algorithms and their applications. In *Proceedings of the First International Conference on Genetic Algorithms*, pages 93–100, 1985.
- [53] Linus Schrage. A more portable fortran random number generator. *ACM Transactions on Mathematical Software*, 5(2):132–138, 1979.
- [54] S.N. Sivanandam and S.N. Deepa. *Introduction to Genetic Algorithms*. Springer-Verlag, 2008.
- [55] J. Maynard Smith. *The Evolution of Sex*. Cambridge University Press, 1978.
- [56] Stephen C. Stearns and Rolf F. Hoekstra. *evoltuion: an introduction*. Oxford University Press, 2000.
- [57] Wikipedia. <http://en.wikipedia.org/wiki/heuristic>, Marzo 2011.
- [58] Wikipedia. http://en.wikipedia.org/wiki/truncation_selection, Marzo 2011.
- [59] Wikipedia. http://en.wikipedia.org/wiki/gray_code, Marzo 2011.