

## C A P Í T U L O 3

### ALGORITMOS GENÉTICOS.

Los algoritmos genéticos forman parte del área de investigación de la computación evolutiva, esta a su vez al conjunto de métodos usados para definir procedimientos heurísticos que pueden ser aplicados a un amplio conjunto de diferentes problemas. Estos métodos son conocidos como meta-heurística. Dicho de otra manera la meta-heurística puede ser vista en general como algoritmos de sistemas que pueden ser aplicados para diferentes problemas de optimización con modificaciones pequeñas y relativas para adaptarse a problemas específicos [26]. Teniendo en cuenta la definición de meta-heurística, podemos empezar a desglosar una de sus áreas de interés que es la Computación Evolutiva, para después poder a través de los conceptos de ésta filosofía, llegar a la definición de algoritmo genético y las eficientes soluciones que esta técnica proporciona.

La computación evolutiva toma al modelo natural de evolución biológica originalmente propuesto por Charles Darwin en su *Teoría de la evolución*. La que explica el cambio adaptativo de las especies por el principio de selección natural, que favorece a las mismas para la supervivencia y para evoluciones futuras que les ayuden a adaptarse mejor a sus condiciones ambientales (concepto comúnmente conocido como "*la supervivencia del más apto*", sin embargo, fue acuñado por H. Spencer, seguidor de Darwin que quería explicar la teoría de Darwin aparentemente tautológica). Además de la selección, otro factor al que Darwin otorga importancia es a la ocurrencia de pequeñas variaciones supuestamente unidireccionales y aleatorias entre los fenotipos. Estas mutaciones predominan a través de la selección si prueban que son importantes para la adaptación de la especie al medio actual, de otra forma perecen. La primordial fuerza conductora de la selección se da en el fenómeno natural de generación de descendencia. Bajo condiciones ambientales ventajosas, el tamaño de la población crece exponencialmente, este proceso es limitado generalmente por la limitación de recursos. Cuando los recursos dejan de ser suficientes para sostener a todos los individuos que conforman la población, los organismos con mutaciones contarán con una ventaja selectiva para explotar los recursos con mayor eficiencia.

La bioquímica y genética modernas han ampliado la teoría darwiniana mediante hallazgos microscópicos concernientes a mecanismos hereditarios. La teoría resultante se conoce como la Teoría sintética de la evolución o *neodarwinismo*.

Teniendo en consideración ambas teorías, existen conceptos importantes como son: gen, cromosoma, genotipo, fenotipo, adaptación, transferencia y mutación.

Partiendo de las teorías y conceptos anteriormente expuestos, es posible decir que existe un marco biológico que sustenta el hecho de que los procesos naturales tienen una orientación hacia la sucesión de improvisaciones sobre las características genéticas de sus individuos con el fin de obtener cada vez mejores respuestas de éstos a sus condiciones ambientales.

La Computación Evolutiva puede dividirse en tres campos de estudio: Estrategias Evolutivas, Algoritmos Genéticos (AG), Programas Evolutivos y Programación Genética.

Sin embargo, solo uno de estos tres nos interesa y es Algoritmo Genético que se define como un algoritmo de búsqueda basado en mecanismos de selección natural y genética natural [29].

Los algoritmos genéticos han demostrado ser una estrategia enormemente poderosa y exitosa para resolver problemas, demostrando de manera espectacular el poder de los principios evolutivos. Se han utilizado algoritmos genéticos en una amplia variedad de campos para desarrollar soluciones a problemas difíciles o más difíciles que los abordados por los diseñadores humanos. Además, las soluciones que consiguen son a menudo más eficientes, más elegantes o más complejas que nada que un ingeniero humano produciría [30]. Un ejemplo de esto es la aplicación para predecir la estructura tridimensional de una proteína, basándose en la secuencia de aminoácidos que la componen. Este fue desarrollado por Schulze-Kremer. Utilizaba números reales para representar los famosos "ángulos de torsión" entre los enlaces peptídicos que conectan a los aminoácidos.

### **3.1. Terminología ocupada para AG**

Como se menciona en la introducción existen conceptos importantes que son necesarios para conceptualizar el funcionamiento de un algoritmo genético por lo que se detallan a continuación:

Gen. Es la característica, el símbolo o indicador (Figura 3.1).

Cromosoma. Es la cadena generada (Figura 3.1).

Alelo. Valor que toma de la cadena.

Genotipo. La estructura; la configuración del cromosoma.

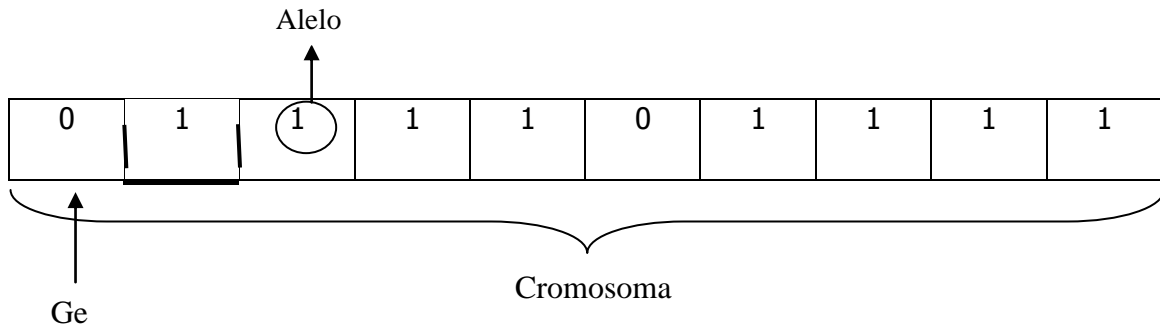
Fenotipo. Las soluciones actuales

Individuo. Genotipo + Fenotipo.

Adaptación. Es la estrecha relación (y en algunos casos idéntica) del individuo con el valor de la función objetivo de la solución representada por éste [30].

Población. Conjunto de individuos.

Función Objetivo. Es la función a la que es sometido cada individuo de una población para saber cual es cuantitativamente el valor o medida de su aptitud, utilidad o lo bueno que es para maximizar la función.



**Figura 3.1.** Representación de un cromosoma en algoritmos genéticos

## 3.2. Operadores genéticos

Son aquellas operaciones que se realizan sobre la población inicial para crear nuevas generaciones [1]. Se dividen en simples y de diversidad.

### 3.2.1. Simples.

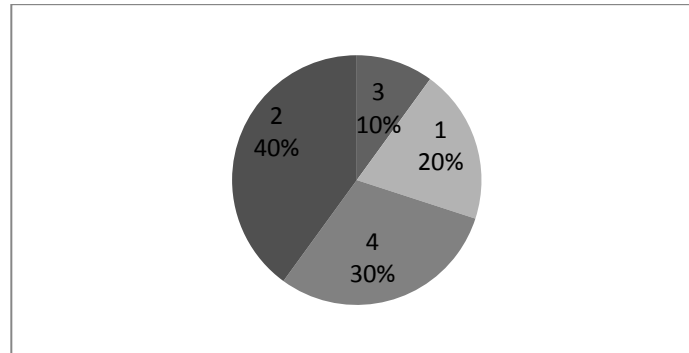
a. Selección. Es el proceso en el que cada cadena individual es copiada en concordancia a los valores de la función objetivo. Es decir solo aquellas cadenas que tenga el más alto valor de aptitud serán parte de la siguiente generación.

Ésta puede ser implementada de muchas formas, entre ellas están:

Roulette Wheel. En este procedimiento cada cadena actual de la población tiene un pedazo de la ruleta en proporción de su aptitud, la suma de las aptitudes nos da un total que permite crear para cada cadena un porcentaje de su aptitud [28].

No.	Cadena	Aptitud	% de Total
1	01101	169	14.4
2	11000	576	49.2
3	01000	64	5.5
4	10011	361	30.9
<b>Total</b>		1170	100.0

Tabla 3.1 basada en el ejemplo proporcionado por Goldberg [28]



“Roulette Wheel” basado en la Tabla 3.1 del ejemplo dado por Goldberg [28].

Para la reproducción, en este ejemplo, simplemente se gira la ruleta cuatro veces, la cadena donde quede después del giro será copiada y depositada en un conjunto que será tentativamente la nueva población. Por ejemplo, cuando se gira la ruleta ponderada la cadena1 tiene un 0.1440 posibilidades de salir mientras la cadena2 tiene un 0.492 de posibilidades de salir escogida, en consecuencia las cadenas selectas serán aquellas con mayor índice de probabilidad, es decir mayor aptitud.

Ranking. Éste se basa en el método de Selección, la probabilidad de selección de un individuo no depende tan solo de su valor aptitud como en el caso de la selección proporcional, pero sí de la aptitud relativa en comparación con otro miembro de la población: es decir, este es comparado en un rango de la población total cuando todos los individuos son ordenados en incremento o decremento basados en el valor de su aptitud [30].

Torneo. Parte del principio de que cierto número de individuos tenga participación en el torneo y de este se selecciona el mejor de los participantes para sobrevivir en la próxima generación. Los miembros del torneo son escogidos al azar y el tamaño de participantes en el torneo es pequeño, que va desde 2 a 10 individuos típicamente. El proceso del torneo es repetido  $\mu$  veces para seleccionar una población de  $\mu$  miembros.

## b. Crossover (Recombinación).

Este método se puede dividir en tres:

Cruza de un solo punto. Este se da en dos pasos; primero, se escogen pares de forma aleatoria, segundo, de forma aleatoria se escoge un número del 1 al  $l - 1$  (donde  $l$  es la longitud de la cadena) al que se llamará  $k$ , posteriormente dos cadenas nuevas se crean al intercambiar los caracteres contenidos entre las posición  $k+1$  y la longitud  $l$ . Ejemplo.

$$A_1=0110|1$$

$$A_2=1100|0$$

Supongamos que aleatoriamente se escogió que  $k=4$  (punto de cruce |). El resultado del crossover da dos nuevas cadenas donde la prima significa las cadenas que forman una nueva generación.

$$A'_1=0110|0$$

$$A'_2=1100|1$$

Cruza multi-punto. Este operador crossover utiliza un número predefinido de puntos distribuidos uniformemente en el cromosoma y la cruce se realiza intercambiando la cadena comprendida entre el par de puntos de cruces tanto del padre como del hijo.

Cruza uniforme. Este operador crossover fue originalmente definido para trabajar con cadenas binarias. El operador intercambia cada bit con una probabilidad entre los dos padres. La probabilidad de intercambio tiene un valor de la mitad, sin embargo los ajustes son posibles.

## c. Mutación.

La mutación como operador genético puede hacer que exista una alteración aleatoria del valor en cualquier posición de la cadena.

### 3.2.2. Diversidad.

Los métodos de diversidad extienden a los algoritmos genéticos para el dominio que requiere el mantenimiento y la localización de soluciones múltiples. Mientras tradicionalmente GAs se dedica principalmente a realizar la optimización, los métodos de diversidad son más adecuados a los problemas de clasificación y máquina de aprendizaje, etc.

Los métodos de diversidad se dividen de acuerdo a su estructura y comportamiento. En la actualidad, existen dos muy exitosos que son el de "Fitness sharing" y "Crowding", ambos métodos son capaces de localizar y mantener múltiples soluciones dentro de una población, si esas soluciones tienen idénticas o diferentes aptitudes.

a. Fitness Sharing (Sharing). Es un mecanismo de escalamiento de aptitud que modifica únicamente la fase de asignación de aptitud de un GA. Sharing puede ser usado en combinación con otros mecanismos de escalamiento, pero debería ser el último en aplicarse, justo antes de la selección [31].

Para una función multimodal la perspectiva de maximización, parte de la siguiente idea que es como sigue: Si se requieren individuos similares para compartir la ganancia o la adaptabilidad, entonces el número de individuos que pueden residir en cualquier porción grafica de la función de adaptabilidad está limitado por adaptabilidad de dicha porción de la gráfica. Los resultados del método de Sharing se dan cuando los individuos son colocados en regiones óptimas de la representación grafica de la función de adaptabilidad. El número de individuos que reside cerca de cualquier pico teóricamente será proporcional a la altura del pico [31].

El método de Sharing funciona al disminuir la adaptabilidad de cada elemento de la población por la cantidad relacionada al número de individuos similares. Específicamente, un elemento del método shared fitness,  $F'$ , es igual a su anterior aptitud  $F$  dividido por el tamaño del nicho. El número de individuos del nicho es la suma de la función de sharing ( $sh$ ), es decir:

$$F^{(i)} = \frac{F(i)}{\sum_{j=1}^{\mu} sh(d(i, j))}$$

La función sharing es una función de distancias  $d$  entre dos elementos de poblaciones; este regresa a '1' si los elementos son idénticos, o '0' si ellos cruzan algún umbral de desigualdad, y un valor intermedio para niveles intermedios de desigualdad. El umbral de no similaridad es especificado por la constante  $\sigma_{share}$ ; si la distancia entre los elementos de las dos poblaciones es grande que o igual para  $\sigma_{share}$  esto hace que no afecte a otros de la función. Una función sharing común es:

$$sh(d) = \begin{cases} 1 - (d/\sigma_{share})^\alpha \\ 0 \end{cases}$$

Donde  $\alpha$  es una constante para regular la forma de la función sharing.

Mientras se distinga la naturaleza de los nichos basándonos en el fenotipo, los nichos GAs pueden emplear cualquier medida de distancia para genotipo o fenotipo. Escoger el apropiado depende de cómo se resuelva el problema [31].

b. Crowding. En la técnica de Crowding, se inserta un nuevo elemento a la población para reemplazar elementos similares. Para determinar la similitud, como en el método de sharing se utiliza una medida de distancia, ya sea fenotípica o genotípica. El método de Crowding tiende a esparcir a sus individuos entre los picos más prominentes de espacio de búsqueda. A diferencia del método de sharing, crowding no asigna elementos proporcionalmente en los picos de aptitud, en cambio, el número de individuos reunidos alrededor de un pico está mayormente determinado por la influencia que el operador crossover muestra sobre los individuos para hacerlos permanecer en el pico [31].

Para reemplazar elementos similares, crowding se esfuerza para mantener la diversidad preexistente de la población. Aun así, los errores de reemplazamiento pueden ser prevenidos algunas veces manteniendo individuos en picos de vecindarios deseados. El algoritmo determinístico de crowding es diseñado para minimizar el número de reemplazamientos erróneos y de esta forma permitir diversidad efectiva.

El crowding determinístico de trabaja como sigue. Primero agrupa la población en  $\mu/2$  pares. Entonces se cruzan estos pares y mutan para una nueva generación. Cada descendiente compite con cada uno de los padres que lo produjo. Para cada par de la descendencia, 2 torneos de padre-hijo son posibles. El Crowding determinístico fuerza a que los elementos más similares compitan (tanto como padre e hijos) [31].

### **3.3. Algoritmo.**

#### 3.3.1. Pseudocódigo.

A continuación se presenta el pseudocódigo basado en el artículo de Abdelmalik Moujahid, Iñaki Inza y Pedro Larrañaga [40]:

BEGIN

Generar una población inicial.

Calcular la función de evaluación de cada individuo.

WHILE NOT Terminado DO

BEGIN /\* Producir nueva generación \*/

FOR Tamaño población/2 DO

BEGIN /\*Ciclo Reproductivo \*/

Seleccionar dos individuos de la anterior generación,  
para el cruce. Cruzar con cierta probabilidad los dos  
individuos obteniendo dos descendientes.

Mutar los dos descendientes con cierta probabilidad.

Calcular la función de evaluación de los dos  
descendientes mutados.

Insertar los dos descendientes mutados en la nueva generación.

END

IF la población ha convergido THEN

Terminado := TRUE

END

END



### 3.3.2. Parámetros.

La extracción del conjunto de parámetros estándar para algoritmos genéticos no es trabajo trivial. Sin embargo, Bäck, apoyándose en distintos autores, propone los siguientes parámetros [35]:

<b>Parámetros</b>	<b>Valores</b>
<b>Probabilidad de mutación</b>	$p_m=0.001$
<b>Operador de recombinación</b>	$r_{\{0.6,2\}}$
<b>Longitud por objeto variable</b>	$l_x=30$
<b>Tamaño de la población</b>	$\mu=50$

Probabilidad de mutación. La tasa de mutación ( $p_m$ ) es la probabilidad de que dos individuos seleccionados sean recombinados para crear nuevos individuos. Esta probabilidad es variable, sin embargo, comúnmente se toma la De Jong [35] que es  $p_m=0.001$ .

Operador de recombinación. El operador de recombinación nos indica el número de puntos de cruce y la probabilidad que debe ser aplicada, en este caso Bäck sugiere dos puntos de cruce y la probabilidad de 0.6.

Longitud de la cadena por variable objeto (variable de decisión). En algoritmos genéticos se trabaja con cadenas binarias de longitud fija, estas son divididas en  $n$  segmentos de igual tamaño solo cuando las variables están en el mismo dominio  $l_x$ , cada segmento codificado corresponde a una variable de decisión. El valor de  $l_x$ , depende de la resolución que se quiera para el rango o dominio de cada variable. Sin embargo Bäck por razones de comparación con estrategias evolutivas y programación evolutiva donde esta variable debe ser de alta precisión le da un valor de 30bits.

Tamaño de la población. Es el número de individuos con los que cuenta una población, este parámetro es dado por Bäck en referencia a trabajos realizados por De Jong y J.J. Grefenstette.

### 3.4. Extensiones y Variantes.

Con el afán de robustecer a los Algoritmos Genéticos Simples (AGs), varios investigadores proponen operadores y métodos de desarrollo naturales, que son capaces de lograr éste objetivo; mediante el uso de multi-criterios o de trabajar poblaciones de forma paralela (diferentes procesadores). Por lo que a continuación se describen algunos de ellos:

### 3.4.1. Coevolución.

Se define coevolución ...“Al cambio en la composición genética de una especie (o grupo de especies) como respuesta a un cambio genético en otra. De esta manera más general y estricta, la coevolución es un cambio recíproco en especies que interactúan”... [36].

Dentro de la computación evolutiva se han desarrollado algunos enfoques coevolutivos en los que generalmente se usan dos o más especies que se relacionan en alguna de estas modalidades que son: simbiosis, competencia, depredador, presa, parasitismo y que evolucionan de manera independiente a través de un algoritmo genético (propio), en la mayoría de los casos [36].

### 3.4.2. Paralelos.

Cuando se toma un algoritmo genético simple y queremos realizar paralelos con éste, podemos seguir tres enfoques distintos, esto son:

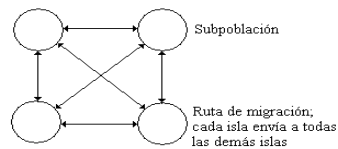
- Cada procesador opera independientemente en una población aislada de individuos, compartiendo los “mejores” con los otros procesadores mediante el operador **migración**. Se suelen llamar algoritmos de grano grueso.
- Cada procesador hace una parte de cada paso del algoritmo (selección, cruce y mutación) sobre la población común. Llamados algoritmos de grano fino.
- Utilizar una implementación híbrida que sea combinación de las dos anteriores.

Cada procesador genera de forma independiente su propia subpoblación inicial de individuos. Después cada procesador lleva a cabo  $k$  generaciones de individuos, para lo que debe evaluar la capacidad, seleccionar los padres de la siguiente generación y realizar los cálculos de cruce y mutación en su subpoblación.

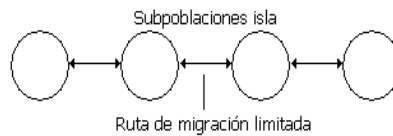
Para después de  $k$  generaciones los procesadores comparten sus mejores individuos con los otros procesadores y debido a la incorporación de la migración, ahora los cambios en una población no sólo proceden del cruce y la mutación, sino también de la introducción de nuevas especies. El operador de migración es responsable de varias tareas necesarias para realizar el intercambio de individuos; entre ellas seleccionar los emigrantes, enviar los emigrantes, recibir los inmigrantes e integrar los inmigrantes.

Los dos enfoques de migración más populares son el modelo de la isla y el modelo de la ‘pasarela’ o ‘trampolín’ (stepping stone).

En el modelo isla, se permite enviar los individuos a cualquier subpoblación:



En el modelo de la 'pasarela' (stepping stone), la migración está limitada, ya que sólo se permite que los emigrantes se desplacen a las subpoblaciones vecinas



Se implementan algunos operadores genéticos (selección, cruce y mutación) en paralelo en la población global.

Existen cuatro estrategias de paralelización:

En el método centralizado, el maestro envía algunos individuos a los procesadores esclavos, los cuales calculan un número conocido de generaciones y envían sus resultados de vuelta al maestro, que ejecuta el algoritmo de cruce para la población completa.

El método semi-distribuido consiste en clusters de procesadores trabajando con el método centralizado e intercambiando soluciones entre ellos.

El método distribuido es la estrategia de grano grueso tradicional, donde cada procesador tiene su propia subpoblación e intercambia los mejores individuos cada cierto tiempo.

La implementación totalmente distribuida consiste en el método distribuido sin ningún intercambio de individuos, cada procesador ejecutando su propio algoritmo secuencial sobre una subpoblación sin comunicación.

### 3.4.3. Locales.

Para Lozano y García [38], los algoritmos genéticos locales (AGL) son procedimientos que iterativamente refinan soluciones dadas y que la diferencia con procedimientos de mejora iterativa clásicos reside en el uso de operadores genéticos para realizar el refinamiento.

Por lo que ellos proponen que un AGL esté basado en un AG estacionario que emplea un método de Crowding para favorecer la formación de nichos en la población. El AGL cruza iterativamente la solución actual, inicialmente dada, con individuos de la población pertenecientes a nichos cercanos a éste. Después, si el nuevo descendiente es mejor que la solución actual, ésta se inserta en la población por medio del método por agrupamiento llamado selección con torneo restringido y el descendiente se acepta como nueva solución actual.

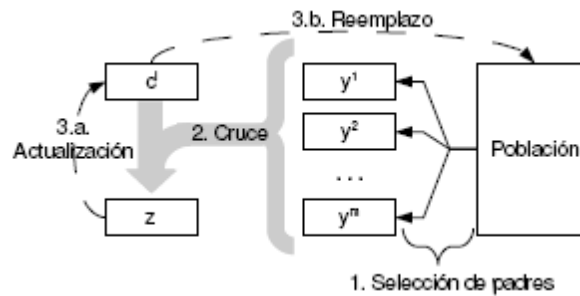


Figura 3.2. Esquema de un AG, propuesto por Lozano y García [38].

#### 3.4.4. Multiobjetivo.

Se le llama un problema Multiobjetivo ó Multicriterio cuando en éste existen varios criterios que se deben cumplir para encontrar una solución simultáneamente. Por ello, la cualidad de un individuo es mejor descrita no por un escalar si no por una medida vectorial.

En un problema con objetivos no conmensurables en conflicto la mejora a cualquier combinación de estos objetivos representará una mejora en la solución total siempre y cuando no ocurra una degradación de los objetivos restantes. Por ser esto posible, entonces la solución actual se considera como optima en "Sentido Pareto", "Pareto Optimo" o "No Dominada". El conjunto de todas las soluciones no dominadas es conocido como Frontera Pareto.

Los enfoques actuales para optimización multiobjetivo con algoritmos evolutivos se dividen en tres grupos [31]:

1. Función de Agregación. Los objetivos son numéricamente combinados dentro de una sencilla función objetivo para ser optimizada.

2. Basados en la población. Diferentes objetivos afectan la selección o des-selección de diferentes partes de la población en turno. Las aproximaciones basadas en categorías separadas de la población de acuerdo a cada objetivo también entran en esta categoría.
3. Basado en Pareto. La población es jerarquizada haciendo uso directo de la definición de dominancia de Pareto.

Existen varios métodos, sin embargo; en este apartado solo discutiremos éstos:

- Acercamiento por la suma de los pesos. Los  $n$  objetivos  $f_1, \dots, f_n$  son ponderados y sus pesos son llamados  $w_1, \dots, w_n$ . La suma de todos los factores es una medida escalar de costo para cada individuo. Esta medida puede ser usada como la base para la selección. Este método es ampliamente conocido, es intuitivo y simple de implementarse y puede ser usado prácticamente con todos los optimizadores.

Descripción Formal.

$$f(a_i) = \sum_{k=1}^n \omega_k f_k(a_i)$$

- Método de  $\epsilon$ -perturbación. Se da cuando los objetivos son tratados como restricciones y solo uno es tratado como función

$$\text{Minimizar } f_k(\vec{x}),$$

$$\text{Sujeto a } f_j(\vec{x}) \leq \epsilon_j \quad \forall j \neq k$$

$$\vec{x} \in F$$

Para encontrar una solución Pareto-óptima, en un conjunto de valores de  $\epsilon_j$  se escoge una función objetivo  $j$ -ésima (donde  $j \neq k$ ). Posteriormente, el objetivo simple es limitado para resolver el problema de optimización, encontrando una solución  $A$ . Este procedimiento se continua con diferentes valores de  $\epsilon_j$  para encontrar diferentes soluciones Pareto-óptimo.

- Clasificación por Pareto. Otro método muy simple para asignar la aptitud es utilizar los valores de aptitud que se reflejen directamente en la relación de prevalencia [34]. Las Figura 3.3 y la Tabla 3.2 ilustran las relaciones de Pareto en una población de 15 individuos y sus correspondientes valores objetivo  $f_1$  y  $f_2$ , ambos objeto de minimización. Para lograrlo, podemos imaginarnos dos diferentes aproximaciones:

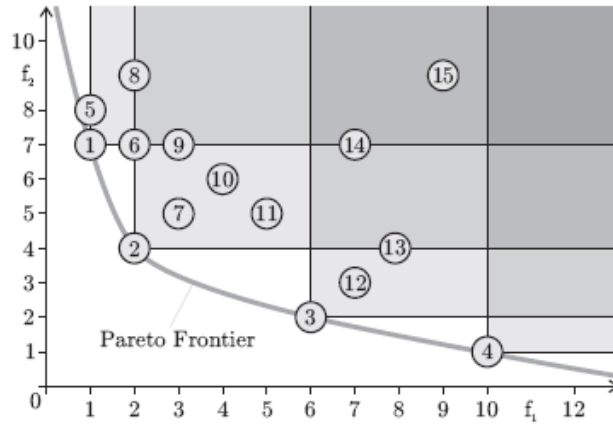


Figura 3.3. Ejemplo de escenario para una clasificación por Pareto [34].

Para cada individuo, se asigna un valor inversamente proporcional al número de individuos que se impone, como  $v(p_1 \cdot x) = \frac{1}{|\{p_2 \in \text{Pop} : p_1 \cdot x > p_2 \cdot x\}| + 1}$ . Se ha llenado la columna "Ap. 1" de la Tabla 3.2 con tal tipo de adaptación para la optimización por Pareto, por ejemplo, el caso especial donde la relación de la dominancia de Pareto se utiliza para definir la prevalencia. Los individuos que dominan a muchos otros obtendrán un valor de adaptabilidad menor que aquellos a los que se imponen muchos. Al revisar estos valores, la desventaja de esta aproximación es evidente: promueve a los individuos que residen en las regiones más saturadas de espacio del problema y desestima aquellos de las áreas de exploración escasa.

X	prevalece a	es prevalectido por	Ap. 1	Ap. 2
1	{5,6,8,9,14,15}	∅	1/7	0
2	{6,7,8,9,10,11,13,14,15}	∅	1/10	0
3	{12,13,14,15}	∅	1/5	0
4	∅	∅	1	0
5	{8,15}	{1}	1/3	1
6	{8,9,14,15}	{1,2}	1/5	2
7	{9,10,11,14,15}	{2}	1/6	1
8	{15}	{1,2,5,6}	1/2	4
9	{14,15}	{1,2,6,7}	1/3	4
10	{14,15}	{2,7}	1/3	2
11	{14,15}	{2,7}	1/3	2
12	{13,14,15}	{3}	1/4	1
13	{15}	{2,3,12}	1/2	3
14	{15}	{1,2,3,6,7,9,10,11,12}	1/2	9
15	∅	{1,2,3,5,6,7,8,9,10,11,12,13,14}	1	13

Tabla 3.2. Tabla de la relación de dominancia de Pareto para la población que se ilustran en la Figura 3.3. [34]

Al hacer esto, logra exactamente lo contrario a lo que deseamos. En vez de explorar el espacio del problema y entregar una exploración amplia de la frontera de los candidatos a la mejor solución posible, centra todo esfuerzo en una trayectoria única. Obteniéndose únicamente un subconjunto de las soluciones mejores, incluso es posible que este método de asignación de la adaptabilidad lleve a una convergencia prematura a un óptimo local. Un buen ejemplo para este problema son los 4 individuos no-prevalentes {1, 2, 3, 4} de la frontera de Pareto. Donde la mejor adaptabilidad es asignada al elemento 2, seguido por el elemento 1. Aunque el individuo 7 es dominado (por 1), su adaptabilidad es mayor a la adaptabilidad del elemento no-dominado 3.

El candidato a solución 4 obtiene la adaptabilidad 1, la peor posible, ya que no prevalece a ningún otro elemento. Sus oportunidades de reproducción son igualmente bajas que aquellas del individuo 15 que es dominado por todos los demás elementos, excepto 4. Por lo tanto, ambos candidatos de solución muy probablemente no serán seleccionados y se desvanecerán en la siguiente generación. La pérdida del candidato 4 reducirá en gran medida la diversidad e incluso incrementará la concentración en el área cercana a 1 y 2 ya de por sí saturada.

Un acercamiento mucho mejor para la asignación de adaptabilidad basado directamente en la prevalencia se propuso por primera vez por Goldberg. En éste, la idea es asignar el número de individuos que prevalecen para cada candidato a solución. De esta forma, no ocurren los efectos negativos previamente expuestos. Como se muestra en la columna "Ap2" de la misma tabla, todos los 4 individuos no-prevalentes ahora tienen la mejor adaptabilidad posible 0. Por esto, la presión de exploración se aplica a un área mucho más amplia de la frontera de Pareto. Esta otra clasificación por Pareto se lleva a cabo, primero: removiendo todos los individuos no-prevalentes de la población y asignándoles el valor 0. Después, se realiza lo mismo con el resto de los individuos dominados únicamente por aquellos con clasificación 0 (ahora no-dominados) se removerán y obtendrán la categoría 1. Este procedimiento se repite hasta que a todos los candidatos de solución se les ha asignado una adaptabilidad propia. El algoritmo propone una forma simple de realizar la Clasificación de Pareto. Debido a que seguimos la idea de comparadores de más libre prevalencia en vez de relaciones de dominancia de Pareto, nos referiremos en forma simultánea a este acercamiento como "Prevalence ranking" (Comparación por prevalencia).

El algoritmo NSGA-II propuesto por Kalyanmoy Deb [42] se basa en el cálculo de este "prevalence ranking" al que se denomina pareto ranking. Deb mejora el cálculo de las

fronteras Pareto de  $O(mN^3)$  descrito en el párrafo anterior a  $O(mN^2)$  además de incorporar métodos de diversidad basados en distancias dentro de cada frontera.

### **3.5. Utilización en sistemas biométricos.**

Los algoritmos genéticos han sido ocupados en una gran diversidad de problemas, por eso, nos es raro pensar que también puede ayudarnos a resolver algunos aspectos de los sistemas biométricos, por ejemplo un sistema de reconocimiento de locutor [39], a través de un algoritmo genético simple nos ayudan a realizar la búsqueda del vector característico en la población total, haciendo un proceso de verificación.

Sin embargo, para identificación o reconocimiento de patrones de firma no se encontró ningún documento, lo que no significa que no lo podamos utilizar en la parte de búsqueda o como una herramienta que apoye a otras para encontrar la mejor solución o estructura en el caso de modelos ocultos de Markov [41].

Por lo anterior se propone incluir a los algoritmos genéticos como una herramienta más para poder mejorar un sistema biométrico, en este caso en especial el de reconocimiento de una firma autógrafa.

El desarrollo de nuestro proyecto se describe en el capítulo siguiente.