

2 Marco Teórico

En este capítulo se hace una breve descripción de los antecedentes al problema de maximización de la disponibilidad con relación al costo, así como definiciones, investigaciones y formulas utilizadas para el desarrollo de esta tesis.

2.1 Antecedentes

El problema de maximización de la disponibilidad consiste en la obtención de un conjunto de soluciones factibles que maximicen la disponibilidad con relación al costo, dando alternativas en la toma de decisiones. Para obtener tales soluciones se hace uso de Programación Multiobjetivo, que ofrece una serie de soluciones factibles. El método de programación multiobjetivo y la reducción del espacio de soluciones usando algoritmos evolutivos son métodos no explorados hasta el momento para este tipo de problema de maximización de la disponibilidad con relación al costo de acuerdo con investigaciones de los autores Helio Fiori de Castro y Katia Lucchesi Cavalca [2]. Los algoritmos evolutivos están basados en conceptos biológicos de la evolución de la especie. Este método utiliza la mecánica de la selección natural y genética, combina la supervivencia del más apto entre estructuras de secuencias con un intercambio de información estructurado [3].

2.2 Disponibilidad

La disponibilidad es un criterio de desempeño para sistemas reparables que toma en cuenta factores como la confiabilidad y capacidad del mantenimiento de un componente o sistema. La disponibilidad es definida como la probabilidad de que un componente opere apropiadamente en un determinado periodo de tiempo, es decir, la probabilidad de que el componente no falle o experimente una acción de reparación cuando este necesita ser usado [4]. La disponibilidad es calculada mediante la ecuación (2.1).

$$A = \frac{\text{Tiempo de vida}}{\text{Tiempo de vida} + \text{Tiempo de reparación}} \quad (2.1)$$

2.3 La disponibilidad de un sistema

Teniendo en cuenta que la disponibilidad es una probabilidad, las reglas de la teoría de la probabilidad pueden ser aplicadas para conocer la disponibilidad de los sistemas. Para n componentes independientes en serie, teniendo para cada componente su disponibilidad $[A_i(t)]$, la disponibilidad del sistema $[A_s(t)]$ es dada por la ecuación (2.2);

$$A_s(t) = \prod_{i=1}^n A_i(t) \quad (2.2)$$

Para n componentes independientes en paralelo, la disponibilidad del sistema es dada por;

$$A_s(t) = 1 - \prod_{i=1}^n [1 - A_i(t)] \quad (2.3)$$

Para configuraciones complejas, el sistema puede ser analizado a través del uso del diagrama de bloques de confiabilidad [4].

El incremento de la disponibilidad se puede obtener de las siguientes dos formas:

1. Obtener niveles altos de disponibilidad en cada componente.
2. Aplicar componentes redundantes.

La aplicación de componentes redundantes consiste, en este caso, en incrementar el número de máquinas disponibles para realizar una misma tarea.

Ambas maneras de obtener niveles altos de disponibilidad traen consigo costos al sistema [2].

2.4 Confiabilidad

Confiabilidad es la probabilidad que un componente no haya fallado antes del tiempo t . La Confiabilidad puede obtenerse mediante un análisis de la tasa media de fallas de los componentes.

El elemento fundamental de la teoría de Confiabilidad es definir la variable aleatoria T como el tiempo transcurrido hasta la falla. Así, la variable aleatoria T tiene:

$$Fd : f_T(t), t \in 0 \quad (2.4)$$

$$fda : F_T(t), t \in 0 \quad (2.5)$$

Función de Confiabilidad

A los modelos que describen el tiempo de falla (t_f) se les llama distribuciones de vida. También a $F(t)$ se le llama función de distribución, $F(t)$ puede interpretarse como:

- a) Probabilidad o seguridad de que una unidad de la población falle antes de t unidades de tiempo.
- b) Fracción de la población que fallan antes de t unidades de tiempo.

En los estudios de Confiabilidad la atención se centra en las unidades que no fallan o que sobreviven un tiempo t , por lo que se define la función de Confiabilidad o sobrevivencia ($R(t)$), denotada por; [5]

$R(t) = 1 - F(t)$, la cuál puede interpretarse como:

- a) Probabilidad de que una unidad de la población no haya fallado antes del tiempo t .
- b) Fracción de la población que sobrevive al tiempo t .

2.5 Capacidad de mantenimiento

La capacidad de mantenimiento es la probabilidad que un artículo sea reparado exitosamente después de una falla. También definida como la probabilidad de realizar una acción acertada de la reparación dentro de un tiempo dado. La capacidad de mantenimiento mide la facilidad y velocidad con las cuales un sistema se puede restaurar al estado operacional después de que ocurra una falta.

La tabla 2.1 muestra la relación existente entre las dos variables que involucran la disponibilidad; confiabilidad y capacidad de mantenimiento, ilustrando de qué manera decrece o incrementa la disponibilidad.

TABLA 2.1 RELACIÓN CONFIABILIDAD Y CAPACIDAD DE MANTENIMIENTO

Confiabilidad	Capacidad de mantenimiento	Disponibilidad
Constante	Disminuye	Disminuye
Constante	Incrementa	Incrementa
Incrementa	Constante	Incrementa
Disminuye	Constante	Disminuye

2.6 Algoritmos y Heurísticos

Una de las herramientas utilizadas dentro de la investigación de operaciones son los algoritmos y métodos heurísticos. Un algoritmo es la secuencia de pasos que conducen a la realización de una tarea, es un conjunto ordenado finito y bien definido de etapas que conducen a la obtención de un resultado. Un algoritmo puede ser representado de las siguientes formas: Detallada (escrita en un determinado lenguaje de programación), Simbólica (descrito con lenguaje próximo al natural) y Gráfica (por medio de diagramas de flujo).

Un heurístico se puede definir como cualquier regla o estrategia por la cual se limita la búsqueda de soluciones en problemas cuyas regiones de solución son demasiado grandes. Los heurísticos no garantizan soluciones

óptimas, pero aquellas que presentan son muy buenas y son obtenidas en tiempos medios menores que los generados por los algoritmos exactos.

2.7 Algoritmos Genéticos

Los algoritmos genéticos son una heurística basada en métodos denominados evolutivos por emular la evolución natural, particularmente la de los seres vivos. John Holland estudió la habilidad de una población de cromosomas para explorar el espacio de búsqueda siendo esta una investigación realizada mediante el mecanismo de recombinación [3].

Los Algoritmos desarrollados inicialmente por Holland eran simples, pero dieron soluciones satisfactorias a problemas que eran considerados como “difíciles” en aquel tiempo. El campo de los algoritmos genéticos ha evolucionado desde entonces, principalmente debido a las innovaciones introducidas en la década de los 80's.

El propio Holland publicó en el año de 1992 un artículo de divulgación sobre algoritmos genéticos, diciendo que éstos han ido incorporando mecanismos cada vez más elaborados, bajo la motivación de la necesidad de resolución aproximada, de problemas prácticos de una amplia variedad [3].

Las características de la evolución que hay que tener en cuenta están descritas brevemente por Davis (1991) de la siguiente manera:

- La Evolución es un proceso que opera en los cromosomas en lugar de los seres vivos que ellos codifican.

- Los procesos de selección natural provocan que aquellos cromosomas que codifican estructuras con mayor rendimiento se reproduzcan más frecuentemente que aquellos que no lo hacen.
- Las mutaciones pueden causar que los cromosomas de los hijos sean diferentes a los de los padres y los procesos de recombinación pueden crear cromosomas bastante diferentes en los hijos por la combinación de material genético de los cromosomas de los dos padres.
- La Evolución biológica no tiene memoria.

Como ocurre en la Evolución biológica, la Evolución simulada estará diseñada para encontrar cada vez mejores cromosomas mediante una manipulación “ciega” de sus contenidos. El termino “ciega” se refiere al hecho de que el proceso no tiene ninguna información acerca del problema que está tratando de resolver, exceptuando el valor de la función objetivo.

En la concepción original de los algoritmos genéticos, la función objetivo es la única información por la que se evalúa el “valor” de un cromosoma. Por lo tanto, estas metaheurísticas están basadas en integrar e implementar eficientemente dos ideas fundamentales: la habilidad de representaciones simples para codificar configuraciones de nuestro problema de optimización y poder hacer transformaciones simples para modificar y mejorar estas configuraciones. Las mejoras son guiadas mediante un mecanismo de control que permite a una población de cromosomas evolucionar como las poblaciones

de seres vivos lo hacen. Un algoritmo genético puede ser visto como una estructura de control que organiza o dirige un conjunto de transformaciones y operaciones diseñadas para simular estos procesos de Evolución [3].

En la Evolución de los seres vivos, el problema al que cada individuo se enfrenta cotidianamente es el de la supervivencia. Para ello cuenta con las habilidades innatas provistas por su material genético. A nivel de los genes, el problema es el de buscar aquellas adaptaciones beneficiosas en un medio ambiente hostil y cambiante. Debido a la selección natural, cada especie gana una cierta cantidad de “conocimiento”, el cuál es codificado e incorporado en la nueva conformación de sus cromosomas. Esta conformación se ve alterada por las operaciones de reproducción. Algunas de ellas son las mutaciones aleatorias, inversión de partes del cromosoma y la recombinación, que es el intercambio de material genético proveniente de dos cromosomas padres.

Los algoritmos genéticos tienen diversas aplicaciones en problemas de Optimización entre las cuales se pueden mencionar las siguientes; Numérica, Combinatoria, Aprendizaje de maquinas, Clasificación, Redes neuronales, Robots, Sensores.

2.7.1 Composición de un Algoritmo Genético

En el contexto de encontrar la solución óptima a problemas de optimización combinatoria de gran escala, un algoritmo genético básico funciona de la siguiente manera. Una población de cromosomas (los padres potenciales de una nueva población) se mantiene a lo largo de todo el proceso evolutivo. A cada uno de ellos se le adjudica un valor de rendimiento que está relacionado con el valor de la función objetivo por optimizar. Cada cromosoma está, pues, representando un punto del espacio de configuraciones (el espacio de búsqueda) del problema.

Dos cromosomas (originalmente sólo restringidos a una fila de números binarios), son seleccionados para ser los padres de una nueva solución mediante el mecanismo de recombinación. En los algoritmos originales de Holland, uno de ellos era elegido de acuerdo a su valor de rendimiento (a mayor valor de rendimiento mayor probabilidad de ser elegido), mientras que el otro padre era elegido aleatoriamente. La operación de Recombinación daba dos configuraciones binarias “hijos” y se calculaban sus valores de rendimiento. Estas soluciones reemplazaban dos soluciones de la población que eran elegidas al azar. Este proceso se repetía tantas veces como se necesitaban (es decir, hasta que se alcanzaba algún criterio más o menos lógico para detener la simulación) [3].

Desde sus inicios, se han dedicado grandes esfuerzos a estudiar las múltiples variaciones de este esquema básico de los algoritmos genéticos. Independientemente de lo sofisticado que pueda ser el diseño de estos algoritmos genéticos, existen cinco componentes que deben ser incluidos:

- Una representación, en términos de “cromosomas”, de las configuraciones del problema.
- Una manera de crear las configuraciones de la población inicial.
- Una función de evaluación que permite ordenar los cromosomas de acuerdo con la función objetivo.
- Operadores “genéticos” que permiten alterar la composición de los nuevos cromosomas generados por los padres durante la reproducción.
- Valores de los parámetros que el algoritmo genético usa (tamaño de la población, probabilidades asociadas con la aplicación de los operadores genéticos).

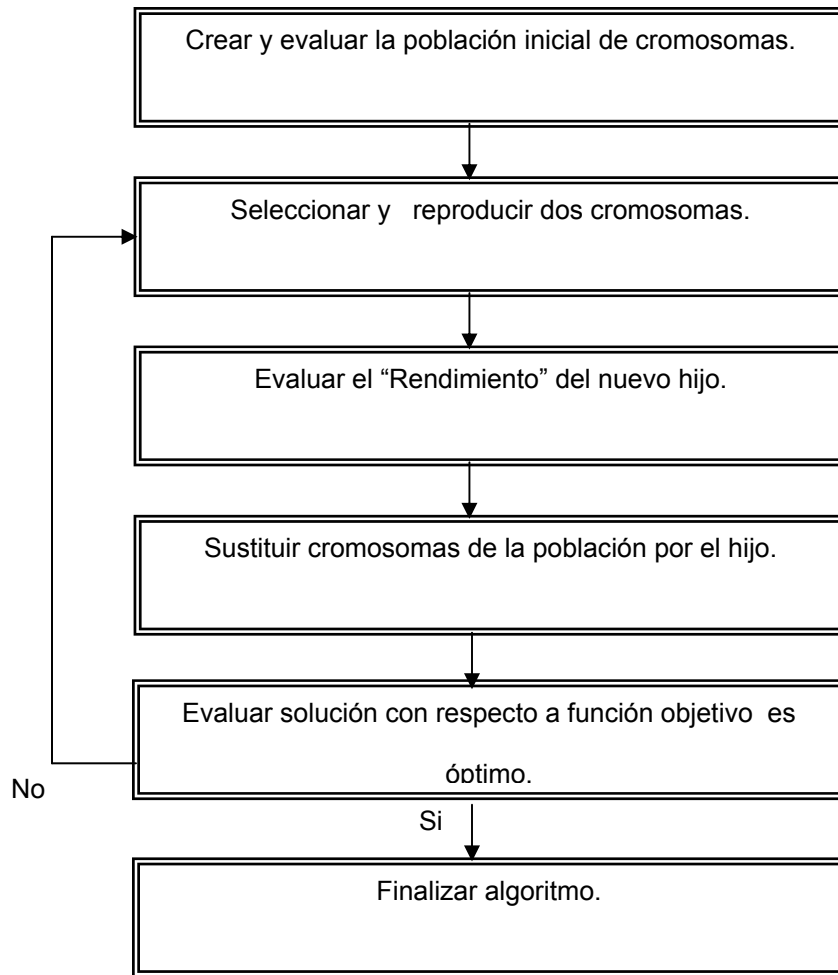


FIGURA 2.1 ESQUEMA BÁSICO DE LOS ALGORITMOS GENÉTICOS

2.7.2 Elementos Básicos

En los trabajos de Holland y en los realizados por varios de sus seguidores, los cromosomas eran representados por listas binarias. Este tipo de codificación ha sido utilizada en muchos campos distintos, incluso en algunos en los que no es particularmente obvio. Una de las ventajas es que este tipo de representación ha sido analizada con mayor detalle. También, una

representación binaria es una codificación natural para un gran número de problemas de optimización como por ejemplo el problema de la mochila. Debido a que las representaciones binarias algunas veces no son efectivas en un gran número de problemas prácticos, algunos investigadores han comenzado a explorar el uso de otras representaciones, en general en conexión con aplicaciones industriales de los algoritmos genéticos.

La codificación más común de las respuestas es a través de listas binarias, aunque se han utilizado también números reales y letras. Ejemplos de otras representaciones incluyen las ordered lists en bin-packing, embedded lists en production scheduling y variable-element lists en semiconductor layout. De tal manera la selección de una representación está fuertemente asociada con el diseño de operadores al utilizar algoritmos genéticos [3].

2.7.2.1 Población Inicial

La Población Inicial de un AG puede ser creada de diferentes maneras.

- Aleatoria
- Acorde al problema
- Híbridos

Al enfrentarse con un problema nuevo se puede aprender mucho inicializando esta población de manera aleatoria. Hacer evolucionar una población que ha sido generada aleatoriamente hasta llegar a tener una bien

adaptada (es decir, con configuraciones satisfactorias que sean soluciones aproximadas del problema de optimización).

Es una buena prueba para saber cómo está funcionando la implementación del AG debido a que características esenciales y críticas de la solución final deben ser consecuencia de este proceso evolutivo y no de las características de los métodos usados para generar la población inicial.

2.7.2.2 Selección

Existen diferentes tipos de estrategias de selección:

- Directa
- Aleatoria

Sabiendo la aptitud de cada cromosoma, se procede a la selección de elementos que se cruzarán en la siguiente generación (presumiblemente, se escogerá a los "mejores").

Selección directa: toma elementos de acuerdo a un criterio objetivo, como cuales son "los x mejores" o "los x peores", son empleados cuando se quieren seleccionar dos individuos distintos y se selecciona el primero por un método aleatorio o estocástico [6].

Selección aleatoria

- Selección equiprobable
- Selección estocástica

Selección equiprobable: todos tienen la misma probabilidad de ser escogidos.

Selección estocástica: la probabilidad de que un individuo sea escogido depende de una heurística, los distintos procedimientos estocásticos son:

- Selección por sorteo
- Selección por escaños
- Selección por restos estocásticos
- Selección por ruleta
- Selección por torneo

Selección por sorteo: cada individuo de la población tiene asignado un rango proporcional o inversamente proporcional a su adaptación. Se escoge un número aleatorio dentro del rango global y el escogido es aquel que tenga dicho número dentro de su rango. La probabilidad de ser escogido es proporcional e inversamente proporcional al grado de adaptación del individuo.

Selección por escaños: se divide el rango del número aleatorio en un número predeterminado de escaños. Es más probable escoger un elemento de baja probabilidad por este método que en el de selección por sorteo.

Selección por restos estocásticos: igual que el método de selección de escaños, solo que los escaños no son asignados directamente, es decir, se hacen de manera aleatoria, la probabilidad de escoger un elemento de muy baja probabilidad es mas alta que en la selección por escaños.

Selección por ruleta: se define un rango con las características de la selección por sorteo, el número al azar será un número aleatorio forzosamente

menor que el tamaño del rango. El elemento escogido será aquel en cuyo rango este el número resultante de sumar el número aleatorio con el resultado total que sirvió para escoger al elemento anterior. El comportamiento es similar al de una ruleta, donde se define un avance cada tirada a partir de la posición actual. Tiene la ventaja de que no es posible escoger dos veces consecutivas el mismo elemento.

Selección por Torneo: la idea de este método es muy simple, se baraja la población y después se hace competir a los cromosomas que la integran en grupos de tamaño predefinido (normalmente compiten en parejas) en un torneo del que resultarán ganadores aquéllos que tengan valores de aptitud más altos. Si se efectúa un torneo binario (ejemplo, competencia por parejas), entonces la población se debe barajar 2 veces. Nótese que esta técnica garantiza la obtención de múltiples copias del mejor individuo entre los progenitores de la siguiente generación (si se efectúa un torneo binario, el mejor individuo será seleccionado 2 veces). Una vez realizada la selección, se procede a la reproducción sexual, cruce o recombinación de los individuos seleccionados [6].

2.7.2.3 Recombinación

Una vez que los padres han sido elegidos, se utiliza un operador genético llamado recombinación. A veces, la elección de un mecanismo de selección adecuado permite un tratamiento matemático más riguroso como el de Prugel-Bennett, Shapiro en el año 1994. La recombinación es el procedimiento por el cual dos seres vivos intercambian parte de su material genético para crear un nuevo organismo [3].

En los algoritmos genéticos, la recombinación intercambia el material de dos cromosomas para generar nuevos individuos. El operador original de los algoritmos genéticos, es punto único de recombinación. Este operador elige aleatoriamente un punto de ruptura de tal manera que el material genético más allá de ese punto es intercambiado entre dos padres para crear dos hijos.

Al irse desarrollando el campo de los algoritmos genéticos, se fueron haciendo notorias ciertas limitaciones del punto único de recombinación, principalmente al no poder combinar características presentes en los dos cromosomas. Una solución ha sido el uso de la reproducción de dos puntos de recombinación. En este caso, son elegidos dos puntos de ruptura al azar y es intercambiado el material genético entre ambos. Esta Recombinación también presenta sus restricciones, lo cuál llevó a Ackley en 1987 a desarrollar la “Recombinación uniforme”, comenzando por el primer elemento, es elegido al azar un padre para que contribuya con su primer elemento al primer hijo, mientras que el segundo hijo recibe el elemento del segundo padre, este proceso continúa hasta que todos los elementos han sido asignados [3].

Normalmente la recombinación se maneja dentro de la implementación del algoritmo genético como un porcentaje que indica con qué frecuencia se efectuará. Esto significa que no todas las parejas de cromosomas se cruzarán, sino que habrá algunas que pasarán intactas a la siguiente generación. De

hecho existe una técnica desarrollada hace algunos años en la que el individuo más apto a lo largo de las distintas generaciones no se cruza con nadie y se mantiene intacta hasta que surge otro individuo mejor que él, que lo desplazará, dicha técnica es llamada elitismo [6].

2.7.2.4 Mutación

Además de la selección y la cruce, existe otro operador llamado mutación, el cual realiza un cambio a uno de los genes de un cromosoma elegido aleatoriamente. Cuando se usa una representación binaria, el gene seleccionado se sustituye por su complemento (un cero cambia en uno y viceversa). Este operador permite la introducción del nuevo material del cromosoma en la población, tal y como sucede con sus equivalentes biológicos, éste operador esta concebido como un operador cuyo papel es secundario. Existen básicamente dos maneras de implementarlo. La primera variante cambia el elemento que la prueba de probabilidad permite (es decir, si el i -ésimo elemento vale 1 y pasa la prueba de probabilidad, la nueva lista contendrá un 0 en la i -ésima posición). En la segunda variante, se genera al azar un nuevo elemento para sustituir al elemento que pasó la prueba de probabilidad [3].

Al igual que la cruce, la mutación se maneja como un porcentaje que indica con qué frecuencia se efectuará, aunque se distingue de la primera por ocurrir mucho más esporádicamente [6].

Si supiéramos la respuesta a la que debemos llegar de antemano, entonces detener el algoritmo genético sería algo trivial. Sin embargo, eso casi nunca es posible, por lo que normalmente se usan 2 criterios principales de detención: correr el algoritmo genético durante un número máximo de generaciones o detenerlo cuando la población se haya estabilizado (por ejemplo cuando todos o la mayoría de los individuos tengan la misma aptitud).

2.7.3 Evaluación del nivel de Rendimiento

La función utilizada para asignar los valores de Rendimiento a los miembros de la población debe ser tal que dé como resultado una buena discriminación. Es muy importante, el diseñar una función de evaluación, tener en cuenta las restricciones del problema. La satisfacción de estas restricciones puede ser llevada a cabo al imponer penalidades en individuos que las violan o creando decodificadores de la representación que eviten generar individuos no factibles.

Una vez evaluado el rendimiento de un individuo, se lleva a cabo una prueba para comparar ese valor con el rendimiento asociado al mejor individuo creado durante todo el proceso evolutivo. Si el rendimiento del mejor individuo es superior al de mejor encontrado hasta ese momento, es reemplazado garantizando así que al término de la simulación el mejor individuo encontrado estará en la memoria a pesar de no estar presente necesariamente en la población final. La convergencia del proceso evolutivo en un individuo particular es generalmente usada como un criterio de finalización del algoritmo genético.

2.8 Programación Multiobjetivo

La optimización con objetivos múltiples extiende la teoría de la optimización permitiendo que múltiples objetivos sean optimizados simultáneamente. Ha sido usada durante años en economía por Takayama en el año de 1974, utilizada también en ciencias gerenciales por Evans en el año de 1984 y gradualmente se ha introducido en la ingeniería en el año de 1995 [9].

Es conocida por diversos nombres tales como Optimización Pareto, Optimización Vectorial, Optimización Eficiente, Optimización Multicriterio y otros más. Las soluciones suministradas son denominadas Optima Pareto, Vector Máximo, Puntos Eficientes y Soluciones NoDominadas [7].

Casi todos los problemas reales involucran la optimización de varios objetivos que a menudo son conflictivos y competitivos entre sí. Mientras que en la optimización con un simple objetivo usualmente la solución óptima está claramente bien definida, no sucede lo mismo en problemas con objetivos múltiples. En lugar de un óptimo simple, existen un conjunto de alternativas conocidas como soluciones óptimo Pareto. Éstas soluciones son óptimas en el sentido de que ningunas otras soluciones son superiores a éstas cuando todos los objetivos son considerados. Los problemas de optimización multiobjetivo son muy comunes en el área de la ingeniería y las técnicas para resolverlos son muy diferentes a las de la optimización mono objetivo [7].

El uso de la optimización multiobjetivo es un punto de estudio importante para el desarrollo de este trabajo de investigación en la búsqueda de niveles altos de disponibilidad, el cual involucra factores relacionados como: confiabilidad y capacidad del mantenimiento. Por medio de esta optimización de objetivos se plantea un enfoque de cuáles son los factores prioritarios para la aplicación de multiobjetivo y así minimizar costos e incrementar el punto principal el cuál es la disponibilidad [2].

Los solución de algoritmos evolutivos multiobjetivo es un conjunto de soluciones Pareto que contiene a todas las soluciones de compromiso, obtenidas al considerar simultáneamente todas las funciones objetivos.

A diferencia de la solución mono-objetivo, que hace que la toma de decisiones obtenga un abanico de soluciones factibles, en el sentido Pareto, para elegir la solución que mejor se adecue a sus necesidades [8].

2.8.1 Modelo de Programación Multiobjetivo

Encontrar el vector $X^* = [X^*_1, X^*_2, \dots, X^*_n]^T$ los cuales satisfacen las m restricciones.

$$g_i(X) \geq 0 \quad i = 1, 2, \dots, m \quad (2.6)$$

Las p restricciones iguales;

$$h_i(X) = 0 \quad i = 1, 2, \dots, p \quad (2.7)$$

Y optimizaran el vector de funciones;

$$f(X) = [f_1(X), f_2(X), \dots, f_k(X)]^T \quad (2.8)$$

2.8.2 Óptimo Pareto

La mayor parte de los problemas de optimización del mundo real tienen varias funciones objetivo (normalmente en conflicto entre sí) que deben satisfacerse simultáneamente. En muchos casos, a fin de simplificar el problema, se suelen transformar todas estas funciones objetivo, menos una, en restricciones, de manera que puedan usarse técnicas convencionales de optimización [9].

En investigación de operaciones este problema de multiobjetivo datan del concepto de solución compromiso definido en 1896 por el economista Vilfredo Pareto ("óptimo de Pareto"), no ha sido sino hasta hace unos cuantos años que la comunidad de computación evolutiva se percató de las enormes ventajas que presentan las heurísticas denominadas algoritmos evolutivos (por ejemplo, los algoritmos genéticos) para optimización multiobjetivo (por ejemplo, su capacidad de generar todo el frente de Pareto en una sola corrida) [10].

2.9 Planteamiento de Formulas utilizadas

La disponibilidad (A), probabilidad de que un sistema opere exitosamente en un determinado periodo de tiempo, puede ser calculada como una razón de tiempo de vida entre tiempo de vida más tiempo de reparación [2].

$$A = \frac{\text{Tiempo de vida}}{\text{Tiempo de vida} + \text{Tiempo de reparación}} \quad (2.9)$$

En donde, el tiempo de vida representa el tiempo promedio entre fallas (MTTF) sigla utilizada para representar al tiempo promedio entre fallas y el tiempo promedio de reparación (MTTR) sigla utilizada para representar el tiempo promedio de reparación [2].

Teniendo en cuenta que un subsistema falla solo si todos los componentes fallan y considerando que solo un equipo de mantenimiento trabaja en un componente, $MTTR1_i$ ver (2.10) sigla que representa el tiempo promedio de reparación en cada subsistema i , el cual es obtenido por medio de la ecuación (2.10);

$$MTTR1_i = \frac{MTTR}{y_i} \quad (2.10)$$

y_i representa el número de componentes redundantes (máquinas) utilizados en cada subsistema.

El tiempo promedio de reparación total para el sistema esta dado por la ecuación (2.11):

$$MTTR1 = f * MTTR1_i \quad (2.11)$$

Donde f es dada por;

$$f = \begin{cases} \frac{y_i}{eq_i} & \text{Si } \frac{y_i}{eq_i} \in \text{Entero} \quad (2.12) \\ \text{int}\left(\frac{y_i}{eq_i}\right) + 1 & \text{Si } \frac{y_i}{eq_i} \notin \text{Entero} \quad (2.13) \end{cases}$$

y_i = número de componentes redundantes (máquinas).

eq_i = número de equipos de mantenimiento.

Int = valor entero de y_i/eq_i .

El factor f ver (2.12) y (2.13) indica el número de equipos de mantenimiento que se pueden utilizar en los subsistemas.

Teniendo calculado el factor f de cada subsistema, se procede a calcular la confiabilidad.

La confiabilidad para cada subsistema es la probabilidad de que una máquina no haya fallado antes de un tiempo t , esta dada por la ecuación (2.14);

$$d_i = \frac{MTTF_i}{f * MTTR_i} \quad (2.14)$$

$MTTF_i$ = tiempo promedio entre fallas.

Calculada ya la confiabilidad, la disponibilidad de un sistema esta dada por la ecuación (2.15).

$$A_s = \prod_{i=1}^n \left[1 - \left(\frac{1}{(1+d)^{y_i}} \right) \right] \quad (2.15)$$

En donde;

y_i = representa el número de componentes redundantes utilizados (máquinas).

d_i = representa la Confiabilidad.

El problema de maximización de la disponibilidad con relación al costo tiene como restricciones;

- 1) Peso del sistema
- 2) Volumen del sistema
- 3) Equipos de mantenimiento disponibles
- 4) El número de equipos de mantenimiento debe ser igual o menor al número de componentes redundantes encontrados.

El peso del sistema y volúmen es calculado por las ecuaciones (2.16) y (2.17);

$$\text{Peso} \quad W = \sum_{i=1}^n w_i * y_i \quad (2.16)$$

$$\text{Volúmen} \quad V = \sum_{i=1}^n v_i * y_i \quad (2.17)$$

Los costos de mantenimiento del sistema son calculados por la ecuación (2.18);

$$CM = \sum_{i=1}^n eqi * ceqi + \sum_{i=1}^n qi * yi * cmi \quad (2.18)$$

Donde:

eqi = Es el número de equipos de mantenimiento.

yi = Es el número de componentes de cada estación.

$ceqi$ = Es el costo de mantenimiento del equipo.

cmi = Es el costo de mantenimiento del subsistema i .

qi = Es la probabilidad de falla de un componente en un subsistema i calculado por la ecuación (2.19).

$$qi = 1 - e^{-\lambda * t} \quad (2.19)$$

El costo de diseño es calculado por la ecuación (2.20);

$$C = \sum_{i=1}^n ci * yi \quad (2.20)$$

Representando un costo total utilizado en la función objetivo como:

$$CT = CM \text{ (Costo de mantenimiento)} + C \text{ (costo de diseño)}$$