

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

RESOLUCIÓN DE UN PROBLEMA DE
OPTIMIZACIÓN MULTI OBJETIVO A TRAVÉS DE
UN ALGORITMO CULTURAL

RICARDO INCHAUSTEGUI LOTINA

INFORME DE TÍTULO.
INGENIERO CIVIL EN INFORMÁTICA

Julio de 2012

Pontificia Universidad Católica de Valparaíso
Facultad de Ingeniería
Escuela de Ingeniería Informática

**RESOLUCIÓN DE UN PROBLEMA DE OPTIMIZACIÓN
MULTIOBJETIVOS A TRAVÉS DE UN ALGORITMO CULTURAL**

RICARDO INCHAUSTEGUI LOTINA

Profesor Guía: **José Rubio León.**

Profesor Co-referente: **Guillermo Cabrera Guerrero.**

Carrera: **Ingeniería Civil Informática.**

Julio de 2012

Agradecimiento

La presente investigación, es el resultado de numerosas revisiones, lecturas, correcciones, opiniones, un sinfín de borradores, apuntes y comentarios.

Debo agradecer a al profesor Guillermo Cabrera Guerrero, profesor que guió este largo proceso, quien con su paciencia, interés y compromiso fue esenciales para la investigación.

Agradecer, al profesor José Rubio León, correferente, que siempre estuvo atento a las consultas realizadas, por sus valiosas observaciones y comentarios durante todas las etapas de la investigación, que permitieron dar otro punto de vista al trabajo realizado.

Dedicatoria

“Quiero dedicar este trabajo de título a mi familia, esas dos personas que han sido mi motivación, mi esposa Sandra Bravo Orellana y a mi hijo Vicente Inchaustegui Bravo, gracias a su paciencia y apoyo incondicional.

También quisiera dedicar este trabajo a mis padres, Ricardo Inchaustegui Poblete y María Lotina Astudillo, por entregar las herramientas necesarias para lograr las metas propuestas.

Y por último a todas los profesores, familia y amigos que me han apoyado durante este largo proceso de formación profesional”.

Índice

Abreviaturas.....	iii
Lista de Ilustración	iv
Lista de algoritmos	iv
Lista de tablas	iv
Lista de ecuaciones.....	v
Resumen	vi
Abstract.....	vi
Introducción.....	1
Capítulo 1 Objetivos.....	2
1.1 Objetivos Generales.....	2
1.2 Objetivos Específicos	2
Capítulo 2 Estado del Arte	3
2.1 Conceptos Básicos.....	3
2.1.1 Optimización	3
2.1.2 Optimización mono y múltiple objetivos	4
2.1.3 Uncapacitated Facility Location Problem (UFLP).....	4
2.1.4 Maximal Covering Location Problem (MCLP).....	5
2.1.5 Multi Objective Problem (MOP).....	6
2.2 Frontera de Pareto.....	6
2.2.1 Dominancia de Pareto.....	6
2.2.2 Frente de Pareto	7
2.2.3 Óptimo de Pareto.....	7
2.2.4 Objetivos de la optimización con múltiples objetivos.....	8
2.3 Métricas	8
2.3.1 Tasa de error	9
2.3.2 Distancia Generacional.....	9
2.3.3 Dispersión.....	9
2.3.4 Espacio Cubierto o Máxima cobertura	11
2.3.5 Cobertura	11
2.4 Computación Evolutiva	11
2.5 Heurísticas alternativas.....	14
2.5.1 Colonia de hormigas.....	14

2.5.2 Algoritmos Meméticos	15
2.5.3 Optimización mediante cúmulos de partículas	17
Capítulo 3 Algoritmos culturales	19
3.1 Componentes del algoritmo cultural.....	19
3.2 Tipos de conocimientos.....	21
3.3 Descripción detallada de los conocimientos trabajados	22
3.3.1 Conocimiento normativo	22
3.3.2 Conocimiento circunstancial	23
Capítulo 4 Diseño del algoritmo cultural propuesto	24
4.1 Arquitectura del Algoritmo Cultural	24
4.1.1 Representación de la solución	24
4.1.2 Población	25
4.1.3 Espacio Creencias.....	27
4.1.4 Selección.....	27
4.1.5 Cruzamiento	28
4.1.6 Mutación.....	31
4.1.7 Función de influencia	31
4.1.8 Función de Aceptación	31
4.1.9 Actualización del espacio de creencia.....	31
4.1.10 Fitness y el cálculo de esta	31
4.1.11 Calculo del área del Frente de Pareto obtenido	32
4.1.12 Parámetros y criterio de parada	34
Capítulo 5 Experimentos computacionales	35
5.1 Instancias utilizadas.....	35
5.2. Afinación de parámetros y funciones	35
5.2.1. Configuración de la Afinación	36
Conclusiones y trabajos futuros.....	41
Referencias	42
Anexos.....	44
Anexo I resultados del conocimiento Circunstancial	44
Anexo II resultados del conocimiento Normativo.....	45
Anexo III Tabla comparativa del conocimiento Circunstancial VS Histórico.....	46
Anexo IV, Tabla comparativa del conocimiento Normativo VS Histórico	47

Abreviaturas

UFLP	: Uncapacitated Facility Location Problem.
MCLP	: Maximal Covering Location Problem.
MOP	: Multi-Objective Problem.
PF	: Frente de Pareto o Frontera de Pareto.
PFcurrent	: Frontera de Pareto actual, obtenida por el algoritmo.
PFtrue	: Frontera verdadera o Frontera Óptima.
GD	: Distancia generacional.
AC's	: Algoritmos Culturales.
MOAQ	: Multi-Objective Ant-Q.
IMMOGLS	: Ishibuchi and Murata Multiples objectives Genetic Local Search.
MOGLS	: Jaskiewicz Murata Multiples objectives Genetic Local Search.
PMA	: Pareto Memetic Algorithm.
M-PAES	: Memetic Pareto Archived Evolution Strategy.
PAES	: Pareto Archived Evolution Strategy.
NSGA	: Non-dominated Sorting Genetic Algorithm
SPEA	: Strength Pareto Evolutionary Algorithm.
SPEA2	: Strength Pareto Evolutionary Algorithm, versión 2.
NSGA2	: Non-dominated Sorting Genetic Algorithm, versión 2.
cNSGA2	: Controlled Non-dominated Sorting Genetic Algorithms.
PSO	: Particle Swarm Optimization.
ACO	: Colonia de Hormigas.

Lista de Ilustración

Ilustración 1. Vilfredo Federico Damaso Pareto.	6
Ilustración 2. Frontera de Pareto.....	7
Ilustración 3. Óptimo de Pareto.....	7
Ilustración 4. Esquema de un algoritmo cultural y sus correspondientes interacciones.....	21
Ilustración 5. Los seis casos posibles para ajustar el conocimiento normativo.....	22
Ilustración 6. Representación gráfica de una solución binaria	25
Ilustración 7. Representación gráfica de una solución no binaria	25
Ilustración 8. Cruce de un punto.....	28
Ilustración 9. Esquema de las variables utilizadas en la función fitness.	32
Ilustración 10. Área de la frontera obtenida o PFCurrent	33
Ilustración 11. Límites para la obtención de la métrica S.	34

Lista de algoritmos

Algoritmo 1. Programación Evolutiva	12
Algoritmo 2. Estrategias Evolutivas.....	13
Algoritmo 3. Genético.....	13
Algoritmo 4. Colonia de Hormigas.....	15
Algoritmo 5. Meméticos con extensiones Múltiple Objetivos.....	16
Algoritmo 6. Cultural	20
Algoritmo 7. Algoritmo Cultural Propuesto.....	24

Lista de tablas

Tabla 1. Distancia entre sucursales y clientes	25
Tabla 2. Distancia entre sucursales y clientes, con distancia menor a 35	26
Tabla 3. Representación de individuo.....	26
Tabla 4. Representación de individuos, con la segunda técnica un individuo una sucursal.	26
Tabla 5. Técnica 1 versus técnica 2, en instancias A.	26
Tabla 6. Técnica 1 versus técnica 2, en instancias B.....	26
Tabla 7. Numero de depósitos y centros de compradores de las instancias.....	35
Tabla 8. Primera configuración de parámetro y funciones.....	36
Tabla 9. Segunda configuración de parámetros y funciones.....	36
Tabla 10. Afinación de las funciones de cruzamiento, espacio Circunstancial.....	36
Tabla 11. Afinación de las funciones de Selección, espacio Circunstancial.....	37
Tabla 12. Afinación del parámetro de porcentaje de mutación en la población, espacio Circunstancial.....	37
Tabla 13. Afinación del parámetro de porcentaje de influencia, espacio Circunstancial.....	37
Tabla 14. Resultado de la afinación de parámetros y funciones.....	37
Tabla 15. Afinación de las funciones de cruzamiento, espacio Distribución Normativo.	37
Tabla 16. Afinación de las funciones de selección, espacio Distribución Normativo.	38

Tabla 17. Afinación del parámetro de porcentaje de mutación en la población, espacio Distribución Normativo.....	38
Tabla 18. Afinación del parámetro de porcentaje de influencia, espacio Distribución Normativo.....	38
Tabla 19. Resultado de la afinación de parámetros y funciones	38
Tabla 20. Afinación de las funciones de selección, espacio Circunstancial.	38
Tabla 21. Afinación de las funciones de cruzamiento, espacio Circunstancial.....	39
Tabla 22. Afinación del Porcentaje influencia, espacio Circunstancial.	39
Tabla 23. Afinación del Porcentaje de mutación, espacio Circunstancial.	39
Tabla 24. Resultado de la afinación de parámetros y funciones, espacio Circunstancial.	39
Tabla 25. Afinación de las funciones de selección, espacio Distribución Normativo.	39
Tabla 26. Afinación de las funciones de cruzamiento, espacio Distribución Normativo.	40
Tabla 27. Afinación del Porcentaje influencia, espacio Distribución Normativo.....	40
Tabla 28. Afinación del Porcentaje de mutación, espacio Distribución Normativo.	40
Tabla 29. Resultado de la afinación de parámetros y funciones, espacio Normativo.	40

Lista de ecuaciones

Ecuación 1. Uncapacitated Facility Location Problem.	5
Ecuación 2. Maximal Covering Location Problem.	5
Ecuación 3. Multi Objective Problem.	6
Ecuación 4. Dominancia de Pareto.....	7
Ecuación 5. Tasa de error.	9
Ecuación 6. Distancia generacional.....	9
Ecuación 7. Dispersión.	10
Ecuación 8. La varianza de los individuos presente en la i-ésima subregión.....	10
Ecuación 9. Ecuación para calcular la dispersión.....	10
Ecuación 10. Efficient Set Spacing.	10
Ecuación 11. Números de elementos del conjuntos de PFCurrent.....	11
Ecuación 12. Conjunto de números reales de un conocimiento normativo.....	22
Ecuación 13. Función para calcular el fitness en un BOUFLP.	32

Resumen

Se va realizar la propuesta de un nuevo algoritmo cultural, que está conformado por los tipos de conocimiento Circunstancial y Normativo que sea capaz, de lidiar con problemas de optimización de múltiples objetivos.

La investigación es el resultado de una acabada investigación, sobre el estado del arte de los algoritmos evolutivos, algoritmos culturales, problemas de optimización de múltiples objetivos, Frente de Pareto y métricas propuestas por diversos investigadores en el área de resolución de problemas de Múltiple objetivos.

Palabras claves: *facility location, computación evolutiva, algoritmos culturales, máxima cobertura, Frente de Pareto Verdadera (PFtrue), Frente de Pareto Obtenida (PFCurrent), MultiObjective Problem (MOP), Uncapacitated Facility Location Problem (UFLP) y Maximal Covering Location Problem (MCLP).*

Abstract

This report presents a new cultural algorithm, which is made of normative and circumstantial types of knowledge, that is able to deal with multiple objective optimization problems.

This work is the result of a research about the state of the art of evolutionary algorithm, cultural algorithm, multiple objective optimization problems, true Pareto front and metrics proposed by a number of researchers on the area of multiple objective problem solution.

Keywords: *facility location, evolutionary computation, cultural algorithm, maximum coverage, True Pareto Front (PFtrue) Obtained Pareto Front (PFCurrent) MultiObjective Problem (MOP), Uncapacitated Facility Location Problem (UFLP) and Maximal Covering Location Problem (MCLP).*

Introducción

Los problemas de optimización con múltiples objetivos han estado presentes siempre en el mundo real, sin embargo sus estudios son recientes, esto es debido a su alta complejidad. Se debe tener en cuenta que al ser funciones con múltiples objetivos, sucede ocurrentemente que estos objetivos se contraponen entre sí.

A medida que las empresas se embarcan en proyectos cada vez más complejos, la solución a las problemáticas planteadas también crece en su complejidad. El nivel de dificultad que alcanza es tan alto, que si no se utiliza una herramienta computacional no es posible llegar a una solución óptima o por lo menos a una buena aproximación a ésta.

El problema de optimización que se va a tratar en esta investigación está compuesto por dos funciones objetivos que se contra oponen entre sí, Uncapacitated Facility Location Problem (UFLP) y Maximal Covering Location Problem (MCLP). En conjunto estas dos funciones objetivos tratan de obtener la mayor cobertura posible, para abarcar la mayor cantidad o el total de los clientes, con el mínimo de instalaciones posibles.

Las instancias utilizadas, han sido resueltas utilizando un algoritmo cultural (AC), propuesto por el Dr. Reynolds, que es una modificación a los algoritmos genéticos. A pesar de que los problemas con múltiples objetivos han sido estudiados, hace ya más de 20 años, las investigaciones realizadas sobre estos problemas de optimización es escasa. Es así, que resulta interesante resolver el problema mediante algoritmos culturales, con mejoras o afinaciones que se realizarán durante la investigación, para mejorar la exactitud y la precisión de este algoritmo.

El capítulo 1, se presentan, el objetivo general, los objetivos específicos y el planteamiento del problema.

Luego el capítulo 2, presenta el estado del arte, de la computación evolutiva, Frente de Pareto, métricas y los problemas optimización de múltiples objetivos.

En el capítulo 3, se realiza el estudio del arte de los AC y su modelado.

A continuación en el capítulo 4, se presenta la solución y el modelado del algoritmo cultural propuesto, definiendo la manera en que se implementó.

Finalmente, en el capítulo 5, se muestran los resultados obtenidos y los afinamientos realizados, para llegar a la mejor configuración.

Capítulo 1 Objetivos

1.1 Objetivos Generales

- Diseñar e implementar un algoritmo cultural, que sea capaz de resolver problemas de localización de instalaciones bi-objetivo.

1.2 Objetivos Específicos

- Investigar sobre el estado del arte de la computación evolutiva.
- Investigar las características de los problemas mono objetivos y múltiples objetivos.
- Investigar problemas UFLP y MCLP.
- Seleccionar los problemas de localización de instalaciones múltiples objetivos más representativos.
- Implementar un algoritmo cultural para la resolución de problema de localización de instalaciones con múltiples objetivos.
- Investigar, seleccionar y /o modificar las métricas a utilizar para medir la calidad de nuestra solución.
- Analizar, comparar y discutir los resultados obtenidos con el algoritmo cultural.

Capítulo 2 Estado del Arte

2.1 Conceptos Básicos

2.1.1 Optimización

La optimización (matemática), es el proceso que se desarrolla para encontrar los mínimos y máximos de una función determinada. Esta área de las matemáticas es la que tiene más aplicaciones en la vida real. En procesos informáticos aparecen constantemente problemas de optimización, de maximizar la cobertura de usuarios y minimizar el tiempo de respuesta de los procesos informáticos, esto también se aplica en la industria, la economía, entre otros [Landa, 2002].

Estos problemas se describen mediante un conjunto de elementos, de los cuales, los más importantes son [Landa, 2002].

- *Variables de decisión.* Son las variables que se alteran en el transcurso de resolución del problema hasta llegar a un óptimo local.
- *Función objetivo.* Se evalúa en esta la variable de decisión, la cual consiste en optimizar (maximizar o minimizar).
- *Restricciones.* Son las condiciones establecidas con anterioridad para que la solución que se obtenga se encuentre en un sector de factibilidad.

La optimización se puede expresar de la siguiente forma matemática:

$$\begin{aligned} & \text{Minimizar o Maximizar } f(\vec{x}) \text{ (función objetivo)} \\ & \text{Variable de decisión } (\vec{x}) \\ & \text{Con } \vec{g}(\vec{x}) \leq 0 \text{ (con } m \text{ restricciones de desigualdad)} \\ & \vec{h}(\vec{x}) = 0 \text{ (con } p \text{ restricciones de igualdad)} \end{aligned}$$

Están restricciones, son las encargadas de delimitar el espacio de donde debe ser buscada la solución, para que se cumpla condición de factibilidad.

Los problemas de optimización son variados, por lo cual presentan ciertas características que permiten realizar la siguiente clasificación [Collette, 2002]:

1. Número de funciones objetivos
 - a. Una → Mono-objetivo
 - b. Varias → Múltiple objetivo
2. Tipo de variable de decisión
 - a. Número real → Continúa
 - b. Número entero → Discreta
 - c. Permutaciones en un conjunto de números de tamaño finito → Combinatorial.

3. Tipo de función objetivo
 - a. Función la cuál es lineal respecto a la variable de decisión → Lineal
 - b. Función la cuál es cuadrática respecto a la variable de decisión → Cuadrática
 - c. Función la cuál es no lineal respecto a la variable de decisión → No lineal
4. Forma del problema
 - a. Con restricciones → Restringido
 - b. Sin restricciones → Sin restricciones.

Según [Collette, 2002] “*las funciones objetivos se dividen en tres grandes grupos y son los siguiente, Programación Lineal (PL), la Programación No Lineal (PNL) (incluyendo cuadrática)*”. La investigación se centra en los PNL, la misión es proporcionar una serie de resultados (óptimos locales) que tienden al punto ideal, cumpliendo con una serie de restricciones para que el resultado sea factible, a diferencia del PL, donde el conjunto de oportunidades y las funciones objetivas tienden a un óptimo global.

2.1.2 Optimización mono y múltiple objetivos

Algunos problemas en el mundo real se pueden reducir a mono objetivo, pero en la mayoría de estos es difícil de poder incluir todos los aspectos, en términos de un solo objetivo. Es aquí cuando el termino Múltiple-objetivos abarca de mejor manera las tareas del mundo real, ya que no solo existe una única solución, sino un conjunto de posibles soluciones, que pueden satisfacer en cierta medida, las necesidades que se ven presentadas en estas tareas con múltiples objetivos. Al contrario que una tarea mono objetivo, que siempre está bien definido dejando, en claro cuál es la función objetivo si es maximizar o minimizar.

Optimización de problemas con múltiples objetivos se ha trabajado ya más de dos décadas. A pesar de que estas tareas, están en variados campos del mundo real, no existen una gran cantidad de técnicas para su resolución. Es muy común encontrar investigaciones, con soluciones a problemáticas que son tratadas como mono-objetivo y que en la realidad tienen más de un objetivo.

2.1.3 Uncapacitated Facility Location Problem (UFLP)

Estos se vinculan directamente en el ámbito industrial, donde instalaciones con producción ilimitada pueden ser colocadas en un conjunto de posibles ubicaciones y con ciertos clientes con ubicación, demanda y costos conocidos.

Los UFLP con restricción, surgieron en los años 70. El objetivo en esta situación es cubrir la mayor medida posible las demandas de los clientes y con ciertas instalaciones, que no tienen una producción ilimitada y que puedan soportar cierta demanda por cierto periodo de tiempo.

Un modelo general definido por [Galvão & Raggi, 1989], está definido de la siguiente forma:

$$v(UFP) = \min \left\{ \sum_{i \in I} f_i y_i + \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} \right\}$$

$$\begin{aligned}
\sum_{i \in I} x_{ij} &= 1, j \in J \\
\sum_{i \in I} y_i &\leq p \\
x_{ij} &\leq y_i, i \in I, j \in J \\
x_{ij} &\in \{0,1\}, i \in I, j \in J \\
y_i &\in \{0,1\}, i \in I
\end{aligned}$$

Ecuación 1. Uncapacitated Facility Location Problem.

Donde $I = \{1, \dots, n\}$ es el conjunto de posibles ubicaciones que una instalación puede situarse, $J = \{1, \dots, m\}$ son los clientes o puntos de demandas, que las instalaciones deben satisfacer, f_i es el costo fijo que incurre la instalación al situarse en un lugar determinado y c_{ij} es el costo que incurre en satisfacer cierta instalación a cierta demanda, y_i adquieren valores binarios, $y_i = 1$ si la instalación se encuentra en $i \in I$, en caso contrario es $y_i = 0$, la variable decisión donde la demanda $j \in J$ es suministrada de $i \in I$.

2.1.4 Maximal Covering Location Problem (MCLP)

Estos problemas, tienen como objetivo determinar el número mínimo de instalaciones, que se necesitan para cubrir todas las demandas, con la restricción de que cada instalación tiene un rango determinado de cobertura.

Estos problemas se pueden encontrar muy habitualmente en organismos públicos, por ejemplo la ubicación de los servicios de emergencias.

La fórmula general del MCLP, está compuesta de la siguiente manera:

$$\begin{aligned}
(MCLP) \quad v(MCLP) &= \max \sum_{i \in N} w_i x_i \\
\text{Sujeto a.} \\
\sum_{j \in S_i} y_j &\geq x_i, \forall i \in N, \\
\sum_{j \in M} y_j &= p, \\
x_i &\in \{0,1\}, \quad \forall i \in N, \\
y_j &\in \{0,1\}, \quad \forall j \in M,
\end{aligned}$$

Ecuación 2. Maximal Covering Location Problem.

Donde $M = \{1, \dots, m\}$ son el conjunto de posibles ubicaciones donde estarían las instalaciones, $N = \{1, \dots, n\}$ son los clientes que debe cubrir, w_i es la demanda de cada cliente que tiene (valor positivo) con $i \in N$ y p es el número de instalaciones que han sido localizadas. Las variables de decisión x_i y y_j se representan de la siguiente forma, donde $x_i = 1$ la demanda i es cubierta $i \in N$ y si $x_i = 0$ no está cubierta y la variable $y_j = 1$ indica que la instalación está situado en j con $j \in M$ y en caso contrario si $y_j = 0$ no se encuentra situada ahí la instalación.

2.1.5 Multi Objective Problem (MOP)

Los MOP, son problemas de optimización que están compuestas por más de una función objetivo, donde los objetivos de cada función se contraponen. Estos problemas de optimización son de uso habitual en cualquier ámbito de la vida real. Por ejemplo un problema compuesto por UFLP y MCLP, si una empresa que quiere instalar sucursales para satisfacer las demandas del cliente, no sólo va tener en mente abarcar la mayor cantidad de clientes. Además deseará satisfacer la demanda completa de esos clientes con el mínimo costo posible u otras funciones objetivos en que se pudiera incurrir.

Ecuación que está compuesta, por UFLP y MCLP: [Fernández y Días, 2010]

$$\begin{aligned} \text{Min } Z_1 &= \sum_{i \in J} \sum_{j \in J} C_{ij} x_{ij} + \sum_{i \in J} f_i y_i \\ \text{Max } Z_2 &= \sum_{j \in J} \left(d_j * \sum_{t \in Q_j} x_{tj} \right) \\ &\sum_{j \in J} x_{ij} = 1, j \in J \\ &x_{ij} \leq y_i, i \in I, j \in J \\ &x_{ij} \in \{0,1\}, i \in I, j \in J \\ &y_i \in \{0,1\}, i \in I \end{aligned}$$

Ecuación 3. Multi Objective Problem.

2.2 Frontera de Pareto

Vilfredo Federico Damaso Pareto, realizó importantes contribuciones al estudio de la economía y de la sociología, en el campo de la distribución de la riqueza y el análisis de las elecciones individuales. Fue el creador del concepto eficiencia de Pareto. Actualmente se utilizan muchas de sus contribuciones, en el campo de la optimización de múltiples objetivos. [A. Carrasco, 2009]



Ilustración 1. Vilfredo Federico Damaso Pareto.

2.2.1 Dominancia de Pareto

La dominancia de Pareto, se utiliza frecuentemente para comparar dos soluciones y de esta manera decidir, si una solución domina a otra. Para el caso en que posea múltiples funciones objetivos, si $f_1(x)$ se debe maximizar y $f_2(x)$ debe minimizar, se dice una solución domina a otra si y sólo si:

Dado una solución $f^1(x)$, domina a otra solución $f^2(x)$, si y sólo si:

$$f_1^1(x) > f_1^2(x) \wedge f_2^1(x) < f_2^2(x)$$

Ecuación 4. Dominancia de Pareto.

2.2.2 Frente de Pareto

Son todas las soluciones que no son dominadas entre las soluciones encontradas, se definen como el conjunto de soluciones encontradas.

El conjunto de estas soluciones no dominadas encontradas, conforman el Frente de Pareto (PF), a continuación la ilustración muestra como los puntos P_1 y P_2 dominan al P_3

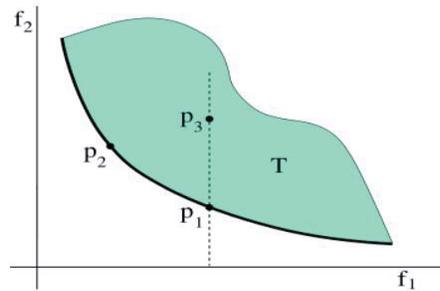


Ilustración 2. Frontera de Pareto

2.2.3 Óptimo de Pareto

En los MOP, cuando los objetivos se encuentran en conflicto entre sí, se buscan normalmente un conjunto de posibles soluciones, en vez de una única solución. Por lo tanto, la noción de óptimo es diferente en estos casos. Dicha noción es comúnmente conocida bajo el término de óptimo de Pareto, como se puede observar en la ilustración 3, para el punto P_1 , ya que es el que domina a todos los demás puntos existentes.

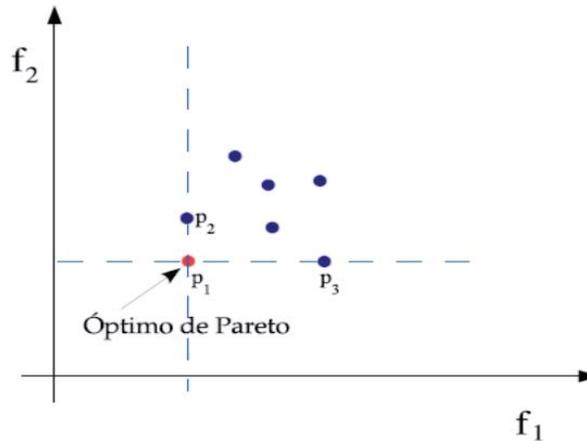


Ilustración 3. Óptimo de Pareto.

2.2.4 Objetivos de la optimización con múltiples objetivos

Cuando existe un problema con dos o más objetivos en conflicto, generalmente el conjunto de soluciones óptimas contienen más de un elemento, por lo tanto, la tarea de elegir una solución entre el resto se vuelve una tarea difícil si no se cuenta con información adicional del problema. En estos casos, todas las soluciones que representan un óptimo de Pareto tienen la misma importancia, por lo que es necesario encontrar tantas soluciones como sea posible, que sean óptimos de Pareto. Por lo tanto, se puede llegar a la conclusión de que existen dos propósitos en la optimización multi-objetivo: [A. Carrasco 2009]

1. Encontrar un conjunto de soluciones lo más cercanas posible al conjunto de óptimos de Pareto real.
2. Encontrar un conjunto de soluciones lo más diversas posible. Es decir, que abarquen todo el frente de Pareto.

Aunque en la mayor parte de las ocasiones el frente óptimo no se conoce, si se puede establecer una relación entre el conjunto de soluciones encontradas y sus características, que serían las siguientes:

1. Encontrar un conjunto de soluciones que minimicen los objetivos lo mejor posible.
2. Encontrar un conjunto de soluciones lo más diversas posible.

2.3 Métricas

Si se desea diseñar una buena métrica para problemas de optimización de múltiples objetivos, se debe tener ciertas consideraciones.

1. El frente de Pareto obtenido por el algoritmo, este lo más cercano posible del frente de Pareto óptimo, suponiendo que se conoce este frente de Pareto lo cual no ocurre normalmente en la vida real.
2. Obtener la mayor cantidad de individuos, que se encuentren en la frontera de Pareto.
3. Tener una buena distribución de las soluciones encontradas, los individuos no dominados sean lo más suave posible entre sí (uniformidad).

Se presentan variadas métricas existentes en la literatura, los tres puntos que se discutieron anteriormente, son objetivos distintos, es por eso que las métricas también se deben tratar como un problema de múltiple objetivos, es recomendable usar diferentes métricas para evaluar los distintos aspectos de desempeño de un algoritmo. [Coello, 2005]

2.3.1 Tasa de error

Esta métrica fue propuesta por [Van Veldhuizen, 1999], indica el porcentaje de soluciones de PFCurrent que no son miembros de PFTrue.

$$ER = \frac{\sum_{i=1}^n e_i}{n}$$

Ecuación 5. Tasa de error.

N , es el número de individuos que están en la PFCurrent, si $e_i = 0$, es porque el individuo de PFCurrent si pertenece a la PFTrue, en caso contrario si los individuos no pertenece al PFTrue $e_i = 1$.

El resultado ideal es $ER = 0$, para poder aplicar esta métrica se debe conocer la frente de Pareto óptimo, lo que resulta ser impracticable para problemas compuestos por UFLP y MCLP, debido a su tamaño y complejidad. Por esta razón esta métrica queda descartada en la presente investigación.

2.3.2 Distancia Generacional

La distancia generacional (GD), fue propuesta por [VanVeldhuizen & Lamont, 1998], buscando algún método para ver qué tan lejos está el PFCurrent del PFTrue. La ecuación está compuesta de la siguiente manera.

$$GD = \frac{\sqrt{\sum_{i=1}^n d_i^2}}{n}$$

Ecuación 6. Distancia generacional.

La variable “ n ” es el número de individuos no dominados en cierta generación, por ende es el PFCurrent y d_i es la distancia Euclidiana de una PFCurrent y el individuo más cercano que se encuentre en PFTrue.

El valor ideal es $GD=0$, de igual manera que en la métrica anterior es impracticable debido a que no se conoce el PFTrue en la vida real.

[Rudolph, 1998], [Schott ,1995] y [Zitzler et al, 2000] han propuesto métricas similares. [Coello, 2005].

2.3.3 Dispersión

[Deb & Srinivas, 1994], propusieron el uso de una distribución chi-cuadrada:

$$SP = \sqrt{\sum_{i=1}^{q+1} \left(\frac{\mu_i - \bar{\mu}_i}{\sigma_i} \right)^2}$$

Ecuación 7. Dispersión.

La variable “q” es el número de puntos óptimos de la PFTrue, se presupone que la subregión (q+1)-ésima es dominada por la q-ésima subregión, μ_i es el número de individuos en el i-ésimo nicho (o subregión) de la región no dominada, $\bar{\mu}_i$ es el número esperado de individuos presente en el i-ésimo nicho, y σ_i^2 es la varianza de los individuos presente en la i-ésima subregión de la región no dominada.

[Deb, 1989] había usado teoría de la probabilidad anteriormente para estimar que:

$$\sigma_i^2 = \bar{\mu}_i \left(1 - \frac{\mu_i}{P} \right), i = 1, 2, \dots, q,$$

Ecuación 8. La varianza de los individuos presente en la i-ésima subregión.

Donde P es el tamaño de la población, suponiendo que la (q+1)-ésima subregión es la región dominada, entonces $\bar{\mu}_{q+1} = 0$, entonces no se desea tener individuos en esta región.

Los estudios de Deb demostró que.

$$\sigma_{q+1}^2 = \sum_{i=1}^q \sigma_i^2$$

Ecuación 9. Ecuación para calcular la dispersión.

Si $SP=0$, significa que el algoritmo ha encontrado la distribución ideal de puntos, implica, si SP tiene valores bajos tiene una buena dispersión para el algoritmo.

Para analizar la distribución con esta métrica, la región no dominada se subdivide en regiones de igual tamaño. Esto puede ser hecho de manera aleatoria o bien, ingresando el número de regiones con anterioridad. Además con el conocimiento de la población se puede determinar la cantidad de individuos que se espera en cada subregión.

[Schott, 1995], propuso la métrica “Efficient Set Spacing” (ESS), similar a la propuesta por Deb.

$$ESS = \sqrt{\frac{1}{e-1} \sum_{i=1}^e (\bar{d} - d_i)^2}$$

Ecuación 10. Efficient Set Spacing.

Dónde:

$$d_i = \min_j \left\{ |f_1^i - f_1^j| + |f_2^i - f_2^j| \right\}$$

Ecuación 11. Números de elementos del conjuntos de PFCurrent.

La variable \bar{d} , se refiere a las medidas de todas las d_i y e , es el número de elementos del conjunto de PFCurrent. El valor ideal es $ESS = 0$, significa que el algoritmo ha encontrado la distribución ideal de vectores no dominados.

Al contrario de las otras métricas mencionadas, la dispersión, no se necesita conocer PFtrue, se parte de la premisa de que sea llegado a este. A pesar de que esta métrica es practicable, comparadas con las anteriores, hay que tener en cuenta los siguientes casos especiales y poder adaptarse para los siguientes. Como PF disjuntos y si sólo producimos dos soluciones, esta métrica nos dirá que su distribución es la ideal.

2.3.4 Espacio Cubierto o Máxima cobertura

Esta métrica, calcula el área del espacio de las funciones objetivo, cubierta por lo vectores no dominados generados por el algoritmo, tomando como referencia el origen (0,0) hasta la función $(f_1(x), f_2(x))$ donde estas funciones son no dominadas, si la frontera de Pareto resulta ser *no convexas*, esta métrica dará resultados erróneos.

El valor que se obtiene a través de esta métrica, depende directamente del número de soluciones no dominadas y su distribución a lo largo del PF. Esta métrica intenta combinar los tres puntos anteriormente discutidos. Por lo tanto, esta métrica no resulta efectiva en aquellos casos en los que dos algoritmos difieran en más de uno de estos criterios antes mencionados (distancia, dispersión y número de elementos del conjunto de Pareto).

2.3.5 Cobertura

[Zitzler & Thiele, 1999]. Proponen comparar dos conjuntos de vectores no dominados, calculando la fracción de cada uno que es cubierta o dominada por el otro.

Problema con la métrica, puede darse que tome como mejor resultado, un PF conformado por un sólo individuo, pero más cercano a PFTrue, en vez de varios individuos más lejanos del PFTrue pero si con una mejor distribución.

Otro problema con esta métrica propuesta, es que puede resultar que ningunos de los dos conjuntos, se dominen entre sí.

[Zitzler & Thiele, 1999], propuso otra métrica llamada “Coverage difference of two sets”, que resuelve algunos de los problemas de la métrica de cobertura.

2.4 Computación Evolutiva

Durante los años 50, se comenzaron aplicar los principios de *Charles Darwin* en la resolución de PNL y combinatoriales. Los años 60 y 70, varias corrientes de investigación

empezaron a conformar lo que ahora se conoce como computación evolutiva. Los algoritmos genéticos fueron propuestos por *John H. Holland* en 1975, su motivación principal fue proponer un modelo de un algoritmo computacional que se fuera adaptando mediante el tiempo hasta llegar a un buen óptimo local.

La computación evolutiva, trabaja con una población de individuos donde cada uno tiene características distintas entre sí e independientes, de esta manera con una función objetivo se puede calcular el fitness de cada uno y a través de algún método de selección elegir los individuos que representarían, una mejor solución y pasándolo a la siguiente generación, al poseer un conjunto de soluciones se puede explorar muchas más soluciones y de esta manera evita quedarse atrapados en óptimos locales y así tiene más posibilidades de acercarse a un óptimo global.

La ejecución del evolutivo, involucra métodos probabilísticos, que son las mutaciones que sufren los individuos, estas mutaciones que tienen los algoritmos genéticos son al azar, gracias a estos se puede explorar de mejor manera el espacio de soluciones, pero a su vez lo hace altamente inestable, si el mismo algoritmo se ejecuta dos veces es muy posible que no lleguen al mismo resultado. [Landa ,2002].

- Programación evolutiva, fue propuesta por Lawrence J. Fogel, modelado para máquinas de estados finitos. Posteriormente David Fogel, lo modifico para optimización numérica [Landa ,2002].

Algoritmo. Programación evolutiva.
Generar la población inicial
Evaluar la población inicial
Repetir
Aplicar operador de mutación
Evaluar algoritmo cada hijo
Realizar la selección
Mientras (no se cumpla la condición de finalización)

Algoritmo 1. Programación Evolutiva

En el algoritmo de programación evolutiva, se genera una población inicial de tamaño N , cada individuo se evalúa y se aplica el operador de mutación, si se tiene N padre la cantidad de hijo será N , se evalúa todas las nuevas soluciones, el tamaño de la población será de $2*N$, mediante una serie de torneos estocásticos. Al final del proceso de selección se tiene nuevamente N individuos, estos procesos se repiten hasta obtener la condición de finalización, el cual puede realizarse mediante un cierto tiempo ya transcurrido, hasta que llegue un óptimo local que se considere aceptable.

Estrategias evolutivas, el creador Ingo Rechenberg, su primera propuesta fue el 1+1, que un solo individuo va evolucionando en el tiempo sin la interacción de más soluciones. Posteriormente la propuesta cambió de un individuo a una población de individuos, si esta es

de tamaño N , a través de recombinación y mutación la población de individuos queda $N + M$, individuos.

El método de selección es determinístico.

Algoritmo Estrategias Evolutivas. Generar población inicial Evaluar población inicial Repetir Aplicar operador de mutación Aplicar operador de recombinación Evaluar cada hijo. Realizar la selección. Mientras (no se cumpla la condición de finalización)
--

Algoritmo 2. Estrategias Evolutivas.

La estrategia evolutiva se distingue de la programación evolutiva por los siguientes aspectos, recombinación y por el método de selección estocástico el cual solo pasan los mejores a la siguiente generación, completamente distinto al torneo de la programación evolutiva donde puede pasar a la siguiente generación el peor individuo.

Algoritmo genético, son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización. Están basados en el proceso genético de los organismos vivos. En los años 1970, de la mano de John Henry Holland, surgió una de las líneas más prometedoras de la inteligencia artificial, la de los algoritmos genéticos. Son llamados así porque se inspiran en la evolución biológica y su base genético-molecular. Estos algoritmos hacen evolucionar una población de individuos sometiéndola a acciones aleatorias semejantes a las que actúan en la evolución biológica (mutaciones y recombinaciones genéticas), así como también a una selección de acuerdo con algún criterio, en función del cual se decide cuáles son los individuos más adaptados, que sobreviven, y cuáles los menos aptos, que son descartados.

Algoritmo Genético. Generar población inicial Evaluar población inicial Repetir Realizar la selección. Aplicar operador de recombinación Aplicar operador de mutación Evaluar cada hijo. Mientras (no se cumpla la condición de finalización)
--

Algoritmo 3. Genético.

Existe una gran propuestas de alternativas en la selección, recombinación y mutación, a diferencias de los anteriores algoritmos la mutación sólo se toma como una acción secundaria dándole la mayor importancia a la recombinación, a través de este algoritmo surge la propuesta de Reynolds el cual denomina Algoritmo Cultural (AC).

2.5 Heurísticas alternativas

En los últimos años, distintas heurísticas que fueron desarrolladas para resolver problemas de optimización mono objetivo, se han adaptado para los nuevos problemas de optimización que están presente en la vida real, los cuales son los problemas con múltiples objetivos. A continuación se realiza una breve descripción de estos algoritmos.

2.5.1 Colonia de hormigas

Está metaheurística, esta basado en el comportamiento de las colonias de hormigas, las cuales buscan comida mediante diversos camino y de manera desordenada, después de llevado un tiempo las hormigas empiezan a cambiar las rutas que eran poco eficiente, hasta llegar a una ruta optima, la cual tiene la trayectoria más cercana, desde su hogar hasta la comida que desean obtener. Esto se logra a través de una feromona, es una segregación y de esta manera si una ruta tiene concentrado más feromona, significa que el tiempo no ha afectado tanto como las otras rutas, por ende se deduce que es la mejor ruta.

Algoritmo Colonia de hormigas

```

1. Initialize  $Q(s, a)$  arbitrarily
2. For  $i = 1$  to  $N$  ( $N =$  number of episodes)
  For  $i = 1$  to  $m$  ( $m =$  number of agents)
  Initialize  $s = s_0$  for the  $m$  agents
  Repeat for  $f$  steps in the episode
  For  $i = 1$  to  $m$ 
  Select  $a$  ins using acceptance rule
  Apply  $a$  and observe  $r, s'$ 
   $Q(s, a) \leftarrow Q(s, a) + \partial[\theta \max_{a'} Q(s', a') - Q(s, a)]$ 
   $s \leftarrow s'$ 
  End Loop
  Until  $s$  is a terminal state
  End Loop
3. Compare the  $m$  solutions found and select best
For all the  $Q(s, a)$  in the best solution
 $Q(s, a) \leftarrow Q(s, a) + \partial[r + \theta \max_{a'} Q(s', a') - Q(s, a)]$ 
( $\partial$  is the learning step)
( $\theta$  is the discount factor)

```

```
End Loop
4. Report best solution found
```

Algoritmo 4. Colonia de Hormigas.

2.5.1.1 Colonia de Hormigas con extensiones Múltiples objetivos

Mariano y Morales en 1999, propusieron el algoritmo MOAQ (Multi-Objetivo Ant-Q), que usa una familia de agente para resolver un problema multiobjetivo.

Esta técnica se tiene que jerarquizar las funciones objetivos y se resuelve de manera incremental desde la menos importante hasta la más importante, se utiliza el concepto de dominancia y a su vez se tiene un pueblo auxiliar que está conformado por todos los individuos no dominados globales.

Iredi, propuso un algoritmo basado en la colonia de hormigas, con la diferencia que no sería solamente una población, sino varias con diferentes pesos, teniendo una función cooperativa en este caso es un intercambio de soluciones entre colonias o poblaciones diferentes, de esta manera agrandando el espacio de exploración. Esta técnica presenta también algunas similitudes con Pareto Archived Evolution Strategy (PAES), debido al tratamiento que hace con el PF, donde segmenta de manera similar a las rejillas de PAES con el fin de tener un mejor resultado de distribución de las soluciones.

2.5.2 Algoritmos Meméticos

El creador de los algoritmos meméticos es Pablo Moscato [1989], para indicar el uso de una heurística local combinada con una estrategia poblacional, los memes a diferencias de los genes, estos últimos pasan de una generación a otra sin alteración y los memes sufre alteraciones en el traspaso de un individuo a otro, llegando la información con ciertas alteraciones [Báez, 2006].

- **Algoritmos Meméticos con extensiones Múltiple Objetivos.**

Los algoritmos meméticos, en la literatura relacionada con problemas de múltiples objetivos podremos encontrar una gran variedad de estos algoritmos que se difieren, de la forma de realizar ciertas operaciones, el orden de estas operaciones y estrategias de diseños más profundas, aun con estas diferencias se puede encontrar un algoritmo patrón, que al tomar esta algoritmo patrón puede expresarse como una instancia de las distintas variedades de algoritmos meméticos.

Algoritmo meméticos con extensión multiobjetivos.

```
01:   Inicializar conjunto de soluciones Cs
02:   Actualizar conjunto de soluciones no dominadas Pc desde Cs
03   Mientras no se cumpla criterio de parada
04:       C = seleccionar soluciones de Cs U Pc
05:       C' = variar individuos de C
06:       C'' = aplicar búsqueda local a soluciones de C'
```

```
07:      Actualizar soluciones de Cs desde C''
08:      Actualizar Pc desde C''
09:      Fin mientras
```

Algoritmo 5. Meméticos con extensiones Múltiple Objetivos.

- **IMMOGLS: Ishibuchi and Murata Múltiples objetivos Genetic Local Search.**

Ishibuchi and Murata [1996], fueron los primeros en proponer un algoritmo meméticos con múltiples objetivos. Este algoritmo utiliza una función escalar definida gracias a un vector de pesos que se ingresa de manera aleatoria, obtenido para combinar los k objetivos del problema en una única función de aptitud.

- **MOGLS: Jaskiewicz Murata Multiplés objetivos Genetic Local Search.**

Propuesta por Jaskiewicz [2002], igual manera que la función IMMOLGS, utiliza una función escalar definida gracias a un vector de pesos, que se da de manera aleatoria e intenta mejorar el valor de esta función en cada iteración a diferencia del algoritmo que se mencionó en anterioridad y la manera de seleccionar las soluciones para recombinaciones. Este algoritmo no posee la línea número 5 y se salta inmediatamente a la línea número 6 (Algoritmo 5).

- **PMA: Pareto Memetic Algorithm.**

Este algoritmo el cual fue propuesto por Jaskiewicz [2003], es muy similar a los enfoques mencionados con anterioridad, difiriendo a ellos en el esquema de selección de soluciones y en la estrategia de supervivencia del descendiente generado.

- **M-PAES: Memetic Pareto Archived Evolution Strategy.**

Knowles y Corne propusieron M-PAES, que es una versión memética de PAES, a diferencia de esta, los autores propusieron trabajar con una población de soluciones diferenciándose al PAES, que solo trabaja con una sola solución y agregando dos vectores, la primera guardar vectores dominados localmente y vectores no dominado de manera global. Se usa una población intermedia de esta manera que el costo computacional no se ha demasiado alto.

- **NSGA: Non-dominated Sorting Genetic Algorithm**

Algoritmo propuesto por Srinivas y Deb. Se basa en la clasificación de varias capas o frente. Consiste en agrupar todos los individuos no dominados en un frente, con un valor de fitness igual para todos los individuos, este valor es proporcional al tamaño de la población, es grupo que fue clasificado pasa a segundo plano y obteniendo el nuevo frente con la nueva generación.

- **SPEA: Strength Pareto Evolutionary Algorithm.**

Los autores son Zitzler y Thiele, Este algoritmo va recogiendo todas las soluciones no dominadas en un vector global, el cual se va actualizando mediante cada iteración y de esta manera al fin de las iteraciones nos encontraremos con un Pareto que contiene todas las soluciones no dominadas. Durante el tiempo de ejecución del algoritmo, el fitness de SPEA, cada miembro de la población actual se calcula según la fuerzas de todas las soluciones no dominadas externas que la dominen

- **SPEA2: Strength Pareto Evolutionary Algorithm, versión 2.**

Las grandes diferencias que tiene con su predecesor, son las siguientes, la primera es la asignación del fitness que considera cada individuo que domina y cada individuo que lo domina y la segunda diferencia utiliza la técnica del vecino más cercano para la valoración de la densidad.

- **NSGA2: Non-dominated Sorting Genetic Algorithm, versión 2.**

Deb propuso una versión revisada del NSGA, computacional mente más eficiente y elitista.

- **cNSGA2: Controlled Non-dominated Sorting Genetic Algorithms.**

Algoritmo propuesto por Deb y Goel, propusieron una variación del NSGA2, en vez de elegir los mejores como el NSGA2, este algoritmo utiliza una proporción geométrica para elegir n individuos de cada frente, la razón de esta proporción geométrica es dado por el usuario.

2.5.3 Optimización mediante cúmulos de partículas

Los autores Kennedy y Eberhart, propusieron un algoritmo llamado *particle swarm optimization* (PSO), que está inspirado en el vuelo migratorios de aves, simulando el movimiento grupal de una determinada población.

Existen variadas propuestas para extender PSO de problemas mono objetivo a múltiples objetivos [Coello, 2005].

- Moore & Chapman [1999], utilizando la dominancia de Pareto, esta propuesta se da importancia tanto como efectuar una búsqueda individual y grupal no se adopta ningún esquema específico para mantener diversidad en la población.

- Ray & Liew [2002], utilizando la dominancia de Pareto y combinando conceptos de las técnicas evolutivas con los de la técnicas cumulares. La técnica usa un operador de agrupamiento para mantener la diversidad y un filtrado de varios niveles para manejar restricciones.

- Parsopoulos & Vrahatis [2002], se implementa tres tipos de técnicas diferentes, una función agregativa lineal convencional, agregativa dinámica y la técnica agregativa “bang-bang”.
- [Hu & Eberhart, 2002], el método propuesto por los autores es ordenar las funciones en un esquema lexicográfico, es decir trabaja una función objetivo a la vez. Esto suele ser extremadamente sensible en el ordenamiento de las funciones objetivos y además de que su funcionamiento es bueno si se utiliza no más de 3 funciones objetivos, cabe mencionar que al trabajar la función objetivo por separado en costo computacional crece.
- [Deb, 2000]. En este algoritmo la población descendiente es creada en la primera instancia, utilizando la población de los padres más la población de la generación siguiente y de esta manera construye un frente de Pareto de manera global y la nueva población se crea basándose en la población resultante no dominada.

Capítulo 3 Algoritmos culturales

En la literatura existen una gran cantidad de algoritmos que están motivados en los grupos sociales. Estos se han utilizados para solucionar problemas complejos de optimización.

Los algoritmos que han utilizado estos métodos son, PSO creador por [Kennedy y Ebehart, 1995], MOAQ [Dorigo, Maniezzo y Colorni, 1996], AG [John Henry Holland], algoritmos meméticos, AC's [Reynolds 1978, 1994]. Los algoritmos que se mencionaron anteriormente son los más conocidos y utilizados para la solución de problemas de optimización en múltiples objetivos, aunque la mayoría de estos algoritmos se crearon para solucionar problemas mono objetivos, estos han sido reajustados para proporcionar soluciones de problemas de optimización de múltiples objetivos, el modelo de cada uno está basado en una determinada población, que van interactuando entre si hasta llegar a un posible optimo global o un buen optimo local.

Los algoritmos culturales creados por el Dr. Reynolds en 1978, están inspirados en el estudio de algunos sociólogos y arqueólogos, que tratan de modelar la evolución cultural que tienen las sociedades durante un largo periodo de tiempo. Los estudios formales de la cultura datan de principio del siglo XX, Durham (1990) propuso un modelo evolutivo cultural de dos niveles.

- *Nivel Micro-Evolutivo*: Es el componente biológico que determina las características heredadas.
- *Nivel Macro-Evolutivo*: Es el componente ideológico (factor ambiental).

La cultura de cada una de las sociedades, se identifica como un conjunto de fenómenos ideológicos compartidos por una población y de estas ideologías el individuo puede dar una orientación de su camino a tomar en su vida, de esta manera como las creencias que son traspasadas de manera Macro-Evolutivo y altera el comportamiento de los individuos, es este factor ambiental lo que el algoritmo cultural desea simular, de esta forma no solo posee el conocimiento que es heredado por sus antepasados directos sino también de las generaciones anteriores.

3.1 Componentes del algoritmo cultural

Los algoritmos culturales están compuesto por:

Espacio de la población: Se encuentran un conjunto de individuos que tienen características entre si distintas e independientes entre el resto de la población, es aquí donde uno puede obtener la aptitud de cada uno con el fitness y de esta manera con algún método de selección elegir los representantes de esta población, los individuos que pertenecen al espacio población, no solo se ven afectado por el cruzamiento entre sus pares, también se ven influenciado con la población que se encuentra en el espacio de las creencias. A través del tiempo, tales Individuos podrían ser reemplazados por algunos de sus descendientes [Soza, 2005].

Espacio de las creencias: En este espacio estará constituido por el o los representantes de todas las generaciones anteriores. Toda solución se verá influenciada sin excepción por el espacio de creencia.

Protocolo de comunicación: Es el encargado de comunicar los conocimientos entre los dos espacios ya mencionados anteriormente con reglas ya definidas. La comunicación del conocimiento en el espacio de las creencias hacia el espacio de la población se llama *función de influencia* y el conocimiento que va desde el espacio de población al espacio de las creencias se llama *función de aceptación*.

- *Función de aceptación:* Se encarga de llevar las soluciones elegidas en el espacio de la población hacia el espacio de las creencias y de esta manera conformando un nuevo espacio de creencias dependiendo del método de selección que se obtenga y las restricciones correspondientes obtenidas mediante el conocimiento elegido.
- *Función de influencia:* Esta función realiza que todos los individuos en el espacio de población, se alejen de los indeseables y se acerque a los deseables, se asume que los deseables son los individuos pertenecientes al espacio de creencia.

Reynolds al modelar el primer algoritmo cultural trata de captar este fenómeno, que ocurre en la sociedad que de herencias dobles, el objetivo de esto es incrementar la tasa de aprendizaje mejorando la exploración y la explotación el mapa de soluciones.

Como se demostró anteriormente los algoritmos culturales operan en dos espacios, el primero es donde se capturan la información de la generación anterior los padres y las recombinaciones que pueden ocurrir dentro del espacio de la población y el segundo es el espacio de las creencias, el que contiene la información de las generaciones pasadas.

El pseudocódigo de un algoritmo cultural es el siguiente.

```
Algoritmo Cultural  
1. Generar población inicial.  
2. Evaluar población inicial.  
3. Actualizar el espacio de creencias  
4. Repetir.  
    *Cruzamiento  
    *Mutación  
    *Aplicar operadores de variación (influenciado del espacio  
    de creencias).  
    Evaluar cada hijo.  
    Realizar la selección (aceptación en el espacio de  
    creencias).  
Mientras no se cumpla la condición de finalización.
```

Algoritmo 6. Cultural

A simple vista se aprecia que es muy parecido a los algoritmos de computación evolutiva y la única diferencia está en la información de las generaciones antepasadas.

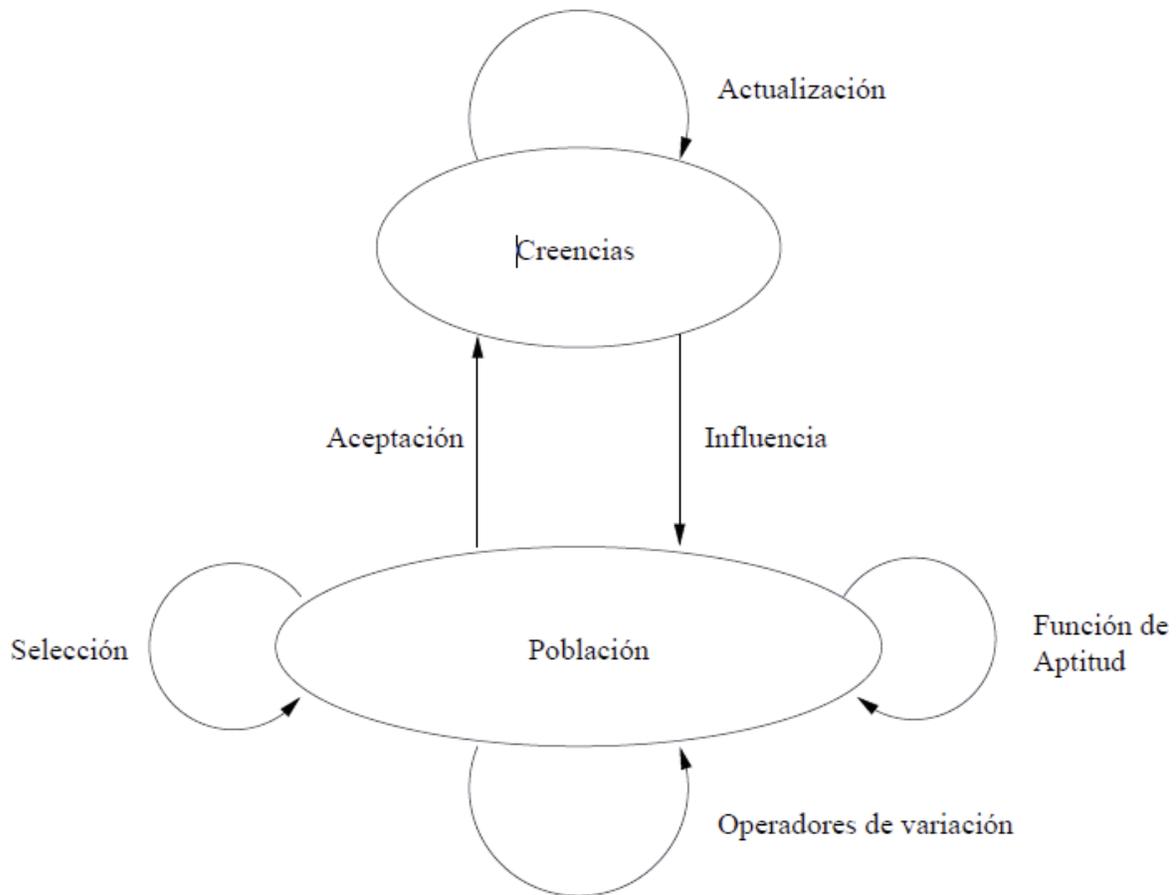


Ilustración 4. Esquema de un algoritmo cultural y sus correspondientes interacciones.

3.2 Tipos de conocimientos

En la literatura, los investigadores han tratado de desarrollar un conocimiento capaz de interpretar a la perfección o por lo menos simular las modificaciones que sufren las generaciones durante el tiempo. El conocimiento es un conjunto prometedor de rasgos variables que proporcionan las normas de los comportamientos individuales y las directrices en el que los ajustes se pueden hacer [Chung y Reynolds, 1996].

Las fuentes de conocimiento se describen aquí en términos de su capacidad, para coordinar la difusión de los individuos sobre la naturaleza de un problema:

1. **Conocimiento Circunstancial:** Este conocimiento selecciona el mejor individuo de cierta generación y la compara con la ideología presente en ese momento, si el individuo de la generación actual es mejor, se reemplaza, en caso contrario se espera

hasta que se encuentre uno que supere al individuo perteneciente al espacio de creencia.

2. **Conocimiento Normativo:** Se almacena el rango de cada variable de decisión, estos rangos van aumentando o disminuyendo en cada iteración.
3. **Conocimiento del Dominio:** Este tipo de conocimiento es poco factible de utilizar en la vida real, se debe conocer el espacio de búsqueda y varía dependiendo del problema y de esta manera utilizar las variables.
4. **Conocimiento Topográfico:** Consiste en crear un mapa y dividirlo por zonas en factibles, infactibles, semi-factibles y desconocidas, este tipo de conocimiento también es poco factible en la vida real, ya que muchas veces se toparan con problemas que desconocen su topografía.
5. **Conocimiento Histórico:** Guardar una lista de los mejores individuos en cada uno de los cambios de entorno, con el fin de no perder los buenos eventos encontrados en generaciones anteriores [Soza, 2005].

3.3 Descripción detallada de los conocimientos trabajados

3.3.1 Conocimiento normativo

El conocimiento normativo está compuesto por cuatro elementos:

- Un conjunto de intervalos I_1, \dots, I_n cada uno correspondiente a la variable j . Un intervalo es un conjunto de números reales y se define como:

$$I_j = [l_j, l_u] = \{x | l_j \leq x \leq u_j, x \in R\}$$

Ecuación 12. Conjunto de números reales de un conocimiento normativo.

Donde l_j y u_j son los límites inferiores y superiores, respectivamente.

- Un conjunto de valores L_1, \dots, L_n que representan la aptitud del límite inferior l_j para el parámetro j .

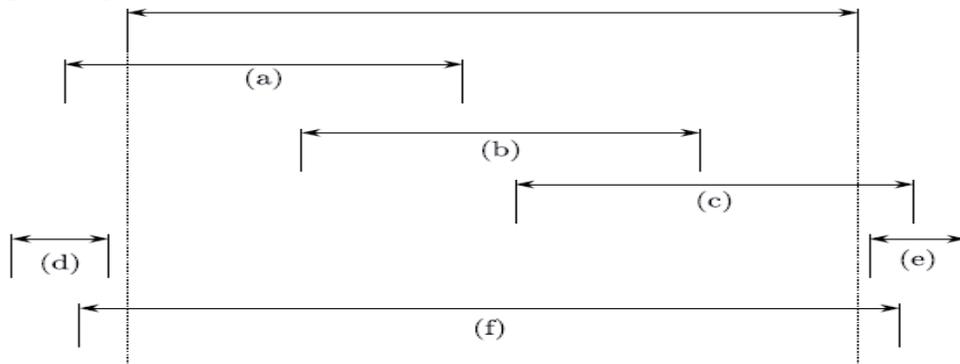


Ilustración 5. Los seis casos posibles para ajustar el conocimiento normativo.

- Un conjunto de valores U_1, \dots, U_n con la medida de aptitud del límite superior u_j para el parámetro j .
- Una función de ajuste encargada de actualizar el conocimiento normativo.

Antes de comenzar con la primera iteración del algoritmo, los intervalos I_j son definidos como el dominio de la variable correspondiente. Las medidas de aptitud L_j y U_j son inicializadas en $-\infty$ (suponiendo que se desea maximizar la aptitud).

La función de ajuste del conocimiento normativo recibe el conjunto de individuos aceptados y calcula los nuevos límite inferior y superior para cada una de las variables. En la Ilustración 5 se presenta un ejemplo de lo que puede ocurrir. La primera línea representa el intervalo antes de aplicar la función de ajuste y las seis líneas inferiores representan los posibles casos que pueden presentarse al calcular el nuevo intervalo.

El caso (b) representa un intervalo que se encuentra contenido en el anterior. En los casos (a), (c) y (f) el nuevo intervalo se intersecta con el anterior. Finalmente, en los casos (d) y (e) el nuevo intervalo cae fuera del intervalo actual.

La idea principal al actualizar el conocimiento normativo se describe a continuación. Si el nuevo límite (inferior o superior) se encuentra fuera del intervalo actual, entonces éste es extendido. Si el nuevo límite cae dentro del intervalo entonces se reduce únicamente si la aptitud de la solución límite es mayor que la del límite anterior [Gutiérrez, 2011].

3.3.2 Conocimiento circunstancial

Consiste en almacenar el mejor ejemplar del proceso evolutivo hasta ese momento, con el fin de considerarlo como líder para la siguiente generación. La influencia de este conocimiento se hace efectiva al hacer tender a las variables de decisión de los nuevos individuos, a los valores del almacenado en el espacio de creencia [Soza, 2005].

Capítulo 4 Diseño del algoritmo cultural propuesto

En la literatura actual, se puede observar que los problemas de optimización con múltiples objetivos son relativamente nuevos. Aun así existen variados algoritmos que provienen desde el desarrollo de soluciones a problemas de mono objetivos. Estos han sido modificados para su funcionamiento en problemas de múltiples objetivos, obteniendo buenos resultados.

Algoritmo Cultural Propuesto

Comienzo

```
g=0 //índice de generaciones
P(g)=Inicializar_poblacion_con_restricción()
Candidatos = Función_de_aceptacion()
C(g) = Actualizar_espacio_creencia(Candidatos)
PFcurrent(g) = No_Dominados(P(g))
Limites=Obtener_limites()
Calcular_fitness(Nueva_poblacion, Limites)
```

Repetir

```
    Nueva_poblacion = Cruzamiento(P(g))
    Nueva_poblacion = Funcion_de_influencia(Nueva_Poblacion)
    Nueva_poblacion = Mutacion(Nueva_Población)
    Calcular_fitness(Nueva_poblacion, Limites)
    g=g+1
    P(g) =Nueva_poblacion
    Candidatos = Función_de_aceptacion()
    C(g) = Actualizar_espacio_creencia(Candidatos)

    PFcurrent(g) = No_Dominados(P(g))
```

Hasta que (cumpla condición de parada)

```
PFcurrent_final = No_Dominados(PFcurrent(g))
Area_de_Pareto = Calcular_area(PFcurrent_final, Limites)
```

Algoritmo 7. Algoritmo Cultural Propuesto

4.1 Arquitectura del Algoritmo Cultural

4.1.1 Representación de la solución

Cada individuo del algoritmo cultural representará una solución al problema. Las formas en que se puede representar esta solución son las siguientes.

- *Representación Binaria:* Está representación de los individuos está conformada por una matriz de tamaño $(n \times m)$, n es el número de sucursales que pueden ser habilitadas y m es el número de clientes que está asignado en el problema. De esta manera si un cliente K (donde $\{K \in M\}$) se le asignó la sucursal J (donde $\{J \in N\}$), la matriz en la fila J columna K tendrá valor igual a 1 y en el resto de la columna los valores serán 0 representando que esta instalación no fue asignada al cliente K .

Un Ejemplo gráfico de una solución o individuo con una representación binaria sería la siguiente.

0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	1	0	0
0	0	1	1	1	0	0	0	1	1
1	0	0	0	0	1	1	0	0	0

Ilustración 6. Representación gráfica de una solución binaria

En la representación gráfica de una solución se les asignó a los clientes, las instalaciones 2,3 y 4. Dejando de lado la instalación 1 debido a los altos costos que tenía correspondientemente con cada uno de los clientes.

- *Representación No Binaria:* Está compuesta por un vector, donde el largo estará definido por el número de clientes que posee el problema y cada posición se guardara el número de la instalación que le fue asignada a los clientes. Tomando como ejemplo la misma solución del punto anterior la solución quedaría representada de la siguiente forma.

4	2	3	3	3	4	4	2	3	3
---	---	---	---	---	---	---	---	---	---

Ilustración 7. Representación gráfica de una solución no binaria

4.1.2 Población

Se evaluó distintas técnicas para generar la población. La primera técnica utilizada fue generar la población inicial de manera aleatoria, de igual manera que los algoritmos evolutivos, con la salvedad que al generar la población, el gen de cada individuo va ser elegido aleatoriamente. Por ejempló.

Se tiene una matriz de distancia con 4 filas que serán las sucursales y con 10 columna correspondiente a los clientes.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
Sucursal 1	40	20	35	32	39	42	41	26	34	60
Sucursal 2	34	15	25	38	48	58	51	24	56	34
Sucursal 3	15	20	80	25	15	36	17	5	45	15
Sucursal 4	42	50	24	26	27	38	43	52	18	42

Tabla 1. Distancia entre sucursales y clientes

La condición es la siguiente, para asignar un cliente a una sucursal, debe tener una distancia máxima de 35, en caso contrario no se puede asignar el cliente a la sucursal. En este caso la matriz de distancia quedaría de la siguiente manera.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
Sucursal 1	X	20	35	32	X	X	X	26	34	X
Sucursal 2	34	15	25	X	X	X	X	24	X	34
Sucursal 3	15	20	X	25	15	X	17	5	X	15
Sucursal 4	X	X	24	26	27	X	X	X	18	X

Tabla 2. Distancia entre sucursales y clientes, con distancia menor a 35

Se generan los individuos de manera aleatoria, pero puede ocurrir que cierto cliente no cumple la condición para que se haya asignado a una sucursal.

Individuo 1	S2	S1	S4	S1	S3	S4	S3	S1	S1	S3
-------------	----	----	----	----	----	----	----	----	----	----

Tabla 3. Representación de individuo.

En la segunda técnica, de inicialización de la población se opta por crear un individuo y solo asignarle una sucursal, como aparece a continuación.

Individuo 1	S1									
Individuo 2	S2									

Tabla 4. Representación de individuos, con la segunda técnica un individuo una sucursal.

La segunda técnica de inicialización de población, mostro mejores resultados que al inicializar la población de manera aleatoria. Mostrando diferencias hasta un 20% aproximadamente.

Instancia	S' , Técnica 1	S' , Técnica 2
A10-25C1	0.5281	0.7058
A30-75C2	0.5729	0.7798
A30-75C5	0.5981	0.7628
A50-150C3	0.6348	0.7171
A50-150C6	0.4182	0.5849

Tabla 5. Técnica 1 versus técnica 2, en instancias A.

.Instancia	S' , Técnica 1	S' , Técnica 2
B10-25C3	0.6154	0.7278
B30-75C2	0.5973	0.7133
B30-75C5	0.6145	0.6869
B50-150C3	0.5748	0.7048
B50-150C6	0.624	0.756

Tabla 6. Técnica 1 versus técnica 2, en instancias B.

4.1.3 Espacio Creencias

Las creencias, en vez de inicializarlas de manera aleatorias, lo que se hace es evaluar el primer espacio de población, mediante el tipo de conocimiento elegido, en este caso son “*Circunstancial y Normativo*”. En el tipo de conocimiento “*Circunstancial*”, el espacio de creencia está conformado por el que tiene la mayor cobertura y el que tiene el mínimo costo. Mientras que el tipo de conocimiento “*Normativo*” se tomó dos conjuntos, el de costo y el de cobertura ambos estaban dentro de los intervalos $[u \pm \theta]$ donde u es la media y θ el porcentaje asignado. El que obtuvo mejores resultado en las pruebas ejecutadas fue el “*Circunstancial*”.

La propuesta que se tiene en esta investigación, relacionado con los espacios de creencia, es no tomar como referencia el fitness para los individuos del espacio de creencia, lo que se tomó en consideración es quien tiene la mayor cobertura y el menor costo, es así como se desea obtener una mejor exploración.

4.1.4 Selección

Las funciones de selección serán los encargados de elegir la nueva población, los cuales pasaran a la función de aceptación que esta evaluará si es ingresado o es descartado en el espacio de las creencias. El tipo de selecciones existentes son diversas pero más utilizadas son las siguiente.

1. *Selección por ruleta*: A cada uno de los individuos de la población se le asigna una parte proporcional a su ajuste de una ruleta, de tal forma que la suma de todos los porcentajes sea la unidad. Los mejores individuos recibirán una porción de la ruleta mayor que la recibida por los peores.

Es un método muy sencillo, pero ineficiente a medida que aumenta el tamaño de la población su complejidad es $O(n^2)$, cabe la posibilidad de elegir el peor más de una ocasión. Este tipo de selección no se tomó en cuenta.

2. *Selección por torneo*: Consiste en realizar comparaciones directas entre dos individuos. Existen dos versiones de selección mediante torneo.

En la versión determinística se selecciona al azar un número p de individuos (generalmente se escoge $P = 2$). De entre los individuos seleccionados se selecciona el más apto para pasarlo a la siguiente generación.

La versión probabilística únicamente se diferencia en el paso de selección del ganador del torneo. En vez de escoger siempre el mejor se genera un número aleatorio del intervalo $[0..1]$, si es mayor que un parámetro p (fijado para todo el proceso evolutivo) se escoge el individuo más alto y en caso contrario el menos apto. Generalmente p toma valores en el rango $0,5 < p \leq 1$.

3. *Selección Elitista*: Este es el más simple de implementar, se comparan todos los individuos con su fitness y se selecciona el mejor.

4. *Selección Destructiva*: Esta selección toma como nueva población los hijos dejando de lado cualquier tipo de segregación por el fitness o por un mejor ajuste.

4.1.5 Cruzamiento

Una vez seleccionado los individuos, estos individuos que representan la generación $g(t - 1)$, se realizara un cruzamiento, de tal manera de producir descendencia que se insertara en la siguiente generación $g(t)$, lo que realiza el cruce es una estrategia de reproducción sexual.

Existen diferentes métodos o estrategias de implementación del cruce.

Existen diversos métodos de realizar este cruce:

1. *Cruce de 1 punto*: Es una técnica realmente sencilla de comprender y sencilla de implementar, una vez seleccionado los dos cromosomas se selecciona un punto del cromosoma de manera aleatoria, diferenciando la cabeza y la cola lo cual intercambian la cola conformando a dos individuos diferentes.

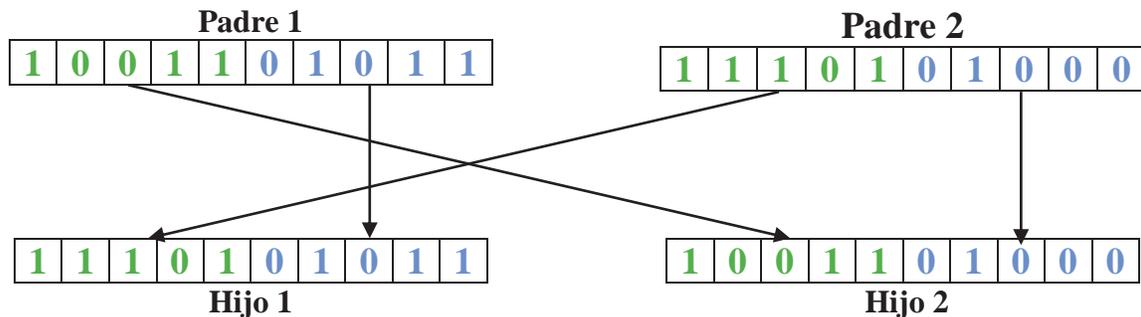


Ilustración 8. Cruce de un punto.

2. *Cruce de 2 puntos*: En vez de cortar un cromosoma por un punto se selecciona de manera aleatoria dos puntos, obligando que los dos puntos del cromosoma que fue elegido de manera aleatoriamente no se han el mismo, de esta manera se crean cabeza, tronco y cola. Para generar una descendencia lo que se realiza es remplazar los troncos de esta manera generando a dos nuevos individuos diferentes.

Generalizando se pueden añadir más puntos de cruce dando lugar a algoritmos de cruce multipunto. Sin embargo existen estudios que desaprueban esta técnica [Jong, 1975]. Aunque se admite que el cruce de 2 puntos aporta una sustancial mejora con respecto al cruce de un solo punto, el hecho de añadir un mayor número de puntos de cruce reduce el rendimiento del Algoritmo Genético, como el algoritmo cultural tiene un diseño muy similar al algoritmo genético, se deja de lado la opción de generar cruce de multipunto.

4.1.6 Mutación

La mutación de un individuo provoca que alguno de sus genes, varíe su valor de forma aleatoria, la mutación más usual es el reemplazo aleatorio. Este consiste en variar aleatoriamente un gen de un cromosoma. Si se trabaja con codificaciones binarias consistirá simplemente en cambiar un bit.

Los individuos de la población que sufren la mutación, puede ser una parte del total de la población o el 100% de ésta. En esta investigación se tomó a todos los individuos y se le modifico al azar uno de sus gen, solo si el bit que se cambiaba estaba dentro del espacio de solución obtenido al principio de la ejecución del algoritmo.

4.1.7 Función de influencia

La función de influencia, se comportara muy similar al cruzamiento uniforme, se creara una máscara de igual manera, se realiza el cruce entre el individuo y con uno o todos los individuos perteneciente al espacio de las creencias y de esta manera el individuo tendrá ciertas características similar al espacio de creencia, estas características nuevas ingresada gracias al macro entorno (“Espacio de creencia”). Las pruebas que se realizó en esta investigación, están dentro de un rango del 5 al 20 por ciento correspondientes al individuo.

4.1.8 Función de Aceptación

La función de aceptación, está directamente relacionado con el tipo de conocimiento que tendrá el espacio de creencia, por ejemplo en el “*Circunstancial*”, se elegirán el individuo con mejor cobertura de cliente y el que tenga menor costo de instalación. De similar manera ocurre con el espacio de creencia “*Normativo*”, con la diferencia que se calcula la media y se toman los límites inferior y superior. Donde todo individuo que este dentro de ese intervalo son seleccionados por la función de aceptación y llevados al espacio de creencia.

4.1.9 Actualización del espacio de creencia

Inmediatamente después de haber ocurrido la función de aceptación, se actualiza el espacio de creencia. En el caso del “*Circunstancial*”, se comparan los individuos que tengan la mayor cobertura y el mínimo costo. Estos quedan seleccionados como el nuevo espacio de creencia. En la situación de que se estuviera en frente de un espacio de creencia como “*Normativo*”, se obtiene la media de todo el espacio de creencia tanto para el costo y como la cobertura, obteniendo la media μ de cada uno $\pm \theta$. El parámetro θ , que es un porcentaje asignado que, en el caso de las pruebas realizadas corresponde al 5%. Después de obtener los límites, se crean dos conjuntos, uno que serán los representantes de cobertura y de costo.

4.1.10 Fitness y el cálculo de esta

La función fitness, en los algoritmos evolutivos y culturales, refleja la calidad de la solución obtenida. En un problema de optimización de mono-objetivos, la función objetivo es la misma que la función fitness, en el caso de que el problema de optimización es de múltiples objetivos, la función objetivo son más de una y muchas veces estas funciones se contra oponen entre sí, es debido que para estos problemas no existe una única solución, sino más bien un conjunto de posibles soluciones, debido a esto las soluciones no son comparables entre sí.

Se realiza una ponderación de la siguiente manera:

Del punto que se desea obtener el Fitness, se saca los valores de las dos funciones correspondientes a este, f_1 que será en esta investigación correspondiente a la función costo y f_2 a la función cobertura, y las otras variables de los demás puntos existente en la frontera de Pareto o población, lif_1 (Límite inferior costo), lsf_1 (Límite superior de costo), lif_2 (Límite inferior de cobertura), lsf_2 (Límite superior de costo), dif_1 la diferencia que existe entre el valor de costo versus el mínimo costo existente y dif_2 la diferencia que existe entre el valor de cobertura versus la máxima cobertura existente. Obteniendo la siguiente función de fitness.

$$fitness = \frac{dif1 * 100}{lsf1 - lif1} + \frac{dif2 * 100}{lsf2 - lif2}$$

Ecuación 13. Función para calcular el fitness en un BOUFLP.

Mientras menor es el valor del fitness, es más apto el individuo.

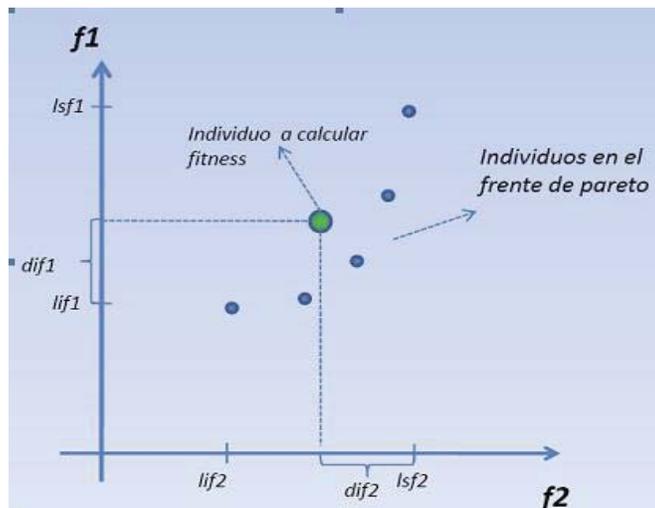


Ilustración 9. Esquema de las variables utilizadas en la función fitness.

En la investigación presente, no se requirió el uso del fitness, debido que no solo se necesita una solución si no un conjunto de estas soluciones, se tomó la dominancia de los individuos y todos los que no estén dominados por otro pertenecen al frente de Pareto.

4.1.11 Cálculo del área del Frente de Pareto obtenido

Según la investigación realizada por [Zitzler, 1998]. La métrica proporcionada no mide la uniformidad ni la cantidad de individuos en la Frontera de Pareto obtenida, lo que mide es la aproximación a la Frontera de Pareto Verdadera “*PFtrue*”.

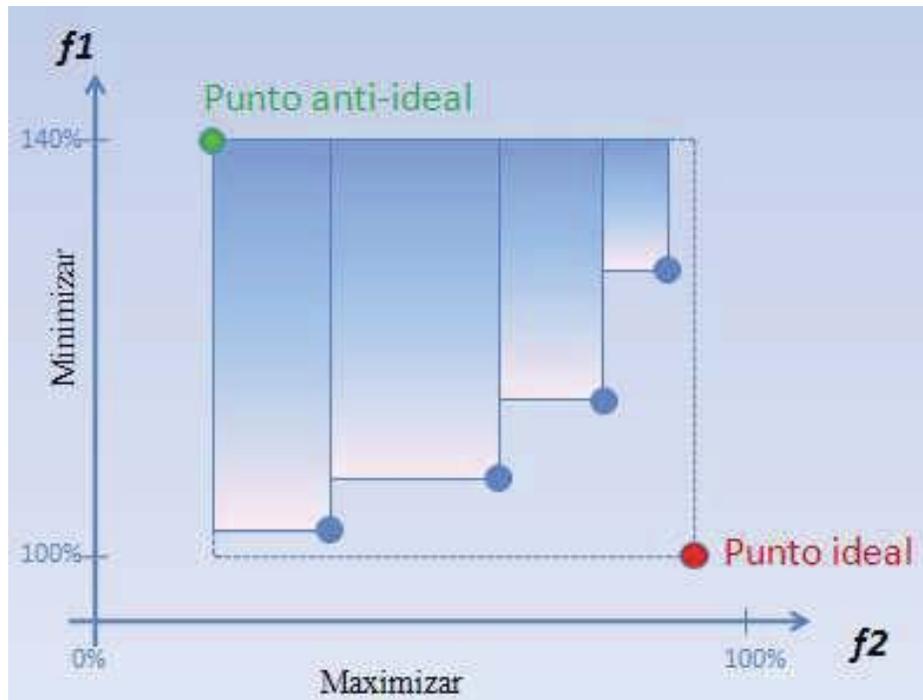


Ilustración 10. Área de la frontera obtenida o *PFCurrent*

Tomando los puntos de referencias que aparecen en la ilustración, como los puntos de la Frontera de Pareto obtenida “*PFCurrent*”.

El valor de la métrica S' dependerá de las unidades que se utilicen para medir cada uno de los criterios de decisión. Utilizando el enfoque de [Medaglia, 2003], se obtiene la métrica modificada S' con valores entre cero y uno. Esta se define como la división del espacio dominado por la aproximación del frente, PF^* . La solución ideal $v(x)$ es un vector formado por las soluciones óptimas de cada función objetivo. Estas fueron obtenidas optimizando cada función objetivo de manera independiente. La solución ideal es rara vez factible y solo es alcanzable si no existe conflicto entre las funciones objetivo. [Villegas, 2003].

La métrica S' está definida por:

$$S' = \frac{S(PF^*)}{S(v(x^*))}$$

El valor de la solución ideal sería $S' = 1$, y cuando no exista ninguna solución factible el valor de S' será cero, puesto que no hay ninguna solución que defina un espacio dominado.

El espacio de los dominados y el cálculo de la área de este espacio, se obtiene tomando los puntos que conforman el *PFCurrent* y sus extremos que son los individuos que tiene máxima cobertura y mínimo costo (marcadas con ● en la ilustración 10). En esta problemática no hubo inconveniente ya que los dos extremos pertenecían a la *PFCurrent*, solo se recorrió esta Frontera buscando la máxima cobertura y el mínimo costo.

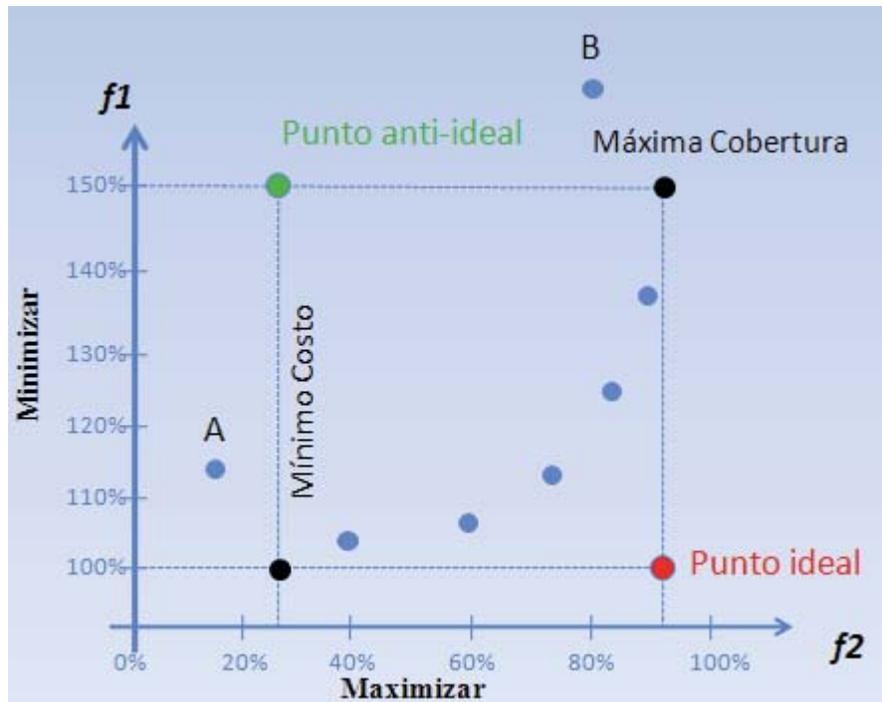


Ilustración 11. Límites para la obtención de la métrica S.

4.1.12 Parámetros y criterio de parada

El parámetro que se utilizó, para determinar el criterio de parada fue el de números de individuos que se encontraban en la frontera de Pareto. El criterio actúa de la siguiente manera, si se lleva más de un 70% de las generaciones y sufre algún cambio la $PFC_{Current}$ en la cantidad de individuos, la generación total aumenta en un 20% y así sucesivamente hasta que el último 30% de las generaciones asignadas en ese momento no sufra modificaciones en la cantidad de individuos en el $PFC_{Current}$.

Se seleccionó la métrica, Espacio Cubierto o Máxima cobertura esta métrica calcula el área del espacio de las funciones objetivo cubierta por los vectores no dominados generados por nuestro algoritmo, tomando como referencia el origen (0,0) hasta la función $(f_1(x), f_2(x))$ donde estas funciones son no dominadas, si la frontera de Pareto resulta ser *no convexa*, esta métrica dará resultados erróneos, pero debido a la naturaleza del problema que se va a tratar, está demostrado que la frontera de Pareto es *convexa*, por ende no surge complicaciones por este estilo.

Cabe notar, como se mencionó antes, esta métrica no mide la suavidad (uniformidad), ni la cantidad de individuos que se encuentran en el Frente de Pareto.

Capítulo 5 Experimentos computacionales

5.1 Instancias utilizadas

Las instancias que se utilizaron en esta investigación, fueron las mismas utilizadas por la tesis de [Fernández y Díaz, 2010], cuyos datos fueron obtenidos de la investigación realizada en [Villegas, 2003]. Estas instancias han sido desarrolladas aleatoriamente usando un generador de problemas que sigue la metodología “UflLib” y fueron propuestas en 1977 [Bilde 1977].

Estas instancias están basadas tomando un problema en particular que trata de una red de recolección de café en el país de Colombia, donde existen depósitos (instalaciones), centros compradores (clientes) y granjeros a los que un centro compra café. Por lo tanto el número de granjeros que el centro comprador atiende es la demanda de este centro [Fernández y Díaz, 2010]. Los problemas que se poseen en este informe fueron de tres tamaños como aparece en el siguiente recuadro.

	Depósitos "m"	Centros de compradores "n"
Tamaño 1	10	25
Tamaño 2	30	75
Tamaño 3	50	150

Tabla 7. Numero de depósitos y centros de compradores de las instancias.

Además del tamaño que están catalogadas estas instancias, corresponde a dos tipos (A y B). Las del tipo A son de distribución uniforme de los centros, con cuadra de 190 unidades de distancia de ancho y las instancias del tipo B, el mismo centro de compradores son elegidos como depósito, es decir que el traslado de la fábrica hasta ese centro es 0. Además están catalogadas por el costo, las que están divididas en C1, C2 y C3 todos los depósitos presentan el mismo costo fijo, correspondiente a 400, 700 y 1000 respectivamente y las instancias etiquetadas como C4, C5 y C6, tienen una distribución de los costos respectivamente $U(100,400)$, $U(400,700)$, $U(700,1000)$. [Villegas, 2003].

Si una etiqueta presenta el siguiente nombre, A35-70C1, quiere decir que esta instancia, tiene 35 depósitos uniformemente distribuidos en cuadradas de 190 unidades de distancia, con 75 centros de compradores y todos los depósitos tienen un costo fijo de 400.

5.2. Afinación de parámetros y funciones

En la afinación del algoritmo cultural propuesto, se tomó dos criterios. El primero consta de la configuración a nivel de funciones que tiene este algoritmo, observando con que funciones tuvo un mejor comportamiento. No se tomó en cuenta el tiempo, sino el porcentaje de cobertura que se obtenía con el $PFC_{Current}$ y los números de individuos encontrados.

5.2.1. Configuración de la Afinación

En la afinación de los algoritmos culturales con conocimientos de Normativo y Circunstancial, se utilizó la instancia A30-75C2. Los parámetros, funciones a evaluar y los resultados de las pruebas son los siguientes.

Configuración

En la realización de la configuración de parámetros y funciones, se ocupó las siguientes combinaciones y del siguiente orden. En la tabla se varía en los dos tipos de cruzamiento dejando fijo los tipos de selección y el porcentaje de mutación e influencia, cuando el mejor resultado obtenido por la variación de cruzamiento, este se deja fijo y a continuación se varía las funciones de selección y así sucesivamente.

Cruzamiento	Selección	P_{mp} (% mutación)	P_i (% influencia)
1 Punto	Elite	2	2
Uniforme	Destructiva	3	4
		4	6
		6	8
		8	

Tabla 8. Primera configuración de parámetro y funciones.

Selección	Cruzamiento	P_i (% influencia)	P_{mp} (% mutación)
Elite	1 Punto	2	2
Destructiva	Uniforme	4	3
		6	4
		8	6
			8

Tabla 9. Segunda configuración de parámetros y funciones.

Conocimiento Circunstancial “Tabla 1”:

Cruzamiento	Selección	P_{mp} (% mutación)	P_i (% influencia)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
1 Punto	Elite	2	2	77.83	26	0.96441
Uniforme	Elite	2	2	78.09	28	0.5712

Tabla 10. Afinación de las funciones de cruzamiento, espacio Circunstancial.

Cruzamiento	Selección	P_{mp} (%) mutación)	P_i (%) influencia)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Uniforme	Destructivo	2	2	75.29	24	0.5845
Uniforme	Elite	2	2	78.09	28	0.5712
Uniforme	Torneo	2	2	76.97	22	1.274

Tabla 11. Afinación de las funciones de Selección, espacio Circunstancial.

Cruzamiento	Selección	P_{mp} (%) mutación)	P_i (%) influencia)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Uniforme	Elite	2	2	78.09	28	0.8386
Uniforme	Elite	3	2	78.38	20	0.5712
Uniforme	Elite	4	2	78.50	25	0.8875
Uniforme	Elite	6	2	78.0	18	0.8134
Uniforme	Elite	8	2	78.13	20	0.5985

Tabla 12. Afinación del parámetro de porcentaje de mutación en la población, espacio Circunstancial.

Cruzamiento	Selección	P_{mp} (%) mutación)	P_i (%) influencia)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Uniforme	Elite	4	2	78.50	25	0.8875
Uniforme	Elite	4	4	78.36	24	0.4548
Uniforme	Elite	4	6	77.78	18	0.6925
Uniforme	Elite	4	8	77.84	24	0.7873

Tabla 13. Afinación del parámetro de porcentaje de influencia, espacio Circunstancial.

Resultado final en el algoritmo cultural con tipo de conocimiento Circunstancial.

Cruzamiento	Selección	P_{mp} (%) mutación)	P_i (%) influencia)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Uniforme	Elite	4	2	78.50	25	0.8875

Tabla 14. Resultado de la afinación de parámetros y funciones.

Conocimiento Distribución Normativo “Tabla 1”:

Cruzamiento	Selección	P_{mp} (%) mutación)	P_i (%) influencia)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
1 Punto	Elite	2	2	79.09	30	0.467
Uniforme	Elite	2	2	78.26	23	0.562

Tabla 15. Afinación de las funciones de cruzamiento, espacio Distribución Normativo.

Cruzamiento	Selección	P_{mp} (%)	P_i (%)	Porcentaje	PF del	Desviación
-------------	-----------	--------------	-----------	------------	---------	------------

		mutación)	influencia)	del mejor resultado	mejor resultado	estándar
1 Punto	Destructivo	2	2	79.09	37	0.467
1 Punto	Elite	2	2	72.95	37	0.592
1 Punto	Torneo	2	2	79.82	31	0.483

Tabla 16. Afinación de las funciones de selección, espacio Distribución Normativo.

Cruzamiento	Selección	P_{mp} (%) mutación)	P_i (%) influencia)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
1 Punto	Torneo	2	2	79.09	30	0.467
1 Punto	Torneo	3	2	78.36	28	0.9737
1 Punto	Torneo	4	2	77.88	33	0.420
1 Punto	Torneo	6	2	77.84	32	0.5868
1 Punto	Torneo	8	2	77.0	31	0.589

Tabla 17. Afinación del parámetro de porcentaje de mutación en la población, espacio Distribución Normativo.

Cruzamiento	Selección	P_{mp} (%) mutación)	P_i (%) influencia)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
1 Punto	Torneo	2	2	79.09	30	0.467
1 Punto	Torneo	2	4	78.24	28	0.4322
1 Punto	Torneo	2	6	78.94	33	0.4154
1 Punto	Torneo	2	8	78.13	32	4.88

Tabla 18. Afinación del parámetro de porcentaje de influencia, espacio Distribución Normativo.

Resultado final en el algoritmo cultural con tipo de conocimiento Distribución Normativo.

Cruzamiento	Selección	P_{mp} (%) mutación)	P_i (%) influencia)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
1 Punto	Torneo	3	4	79.09	30	0.467

Tabla 19. Resultado de la afinación de parámetros y funciones

Conocimiento Circunstancial “Tabla 2”:

Selección	Cruzamiento	P_i (%) influencia)	P_{mp} (%) mutación)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Elite	1 Punto	2	2	78.42	30	0.95
Destructivo	1 Punto	2	2	77.11	24	1.61
Torneo	1 Punto	2	2	76.95	28	1.10

Tabla 20. Afinación de las funciones de selección, espacio Circunstancial.

Selección	Cruzamiento	P_i (%)	P_{mp} (%)	Porcentaje	PF del	Desviación
-----------	-------------	-----------	--------------	------------	---------	------------

		influencia)	mutación)	del mejor resultado	mejor resultado	estándar
Elite	1 Punto	2	2	78.42	30	0.95
Elite	Uniforme	2	2	76.38	30	0.609

Tabla 21. Afinación de las funciones de cruzamiento, espacio Circunstancial

Selección	Cruzamiento	P_i (% influencia)	P_{mp} (% mutación)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Elite	1 Punto	2	2	78.42	30	0.95
Elite	1 Punto	4	2	78.07	28	0.716
Elite	1 Punto	6	2	78.64	35	0.9509
Elite	1 Punto	8	2	79.26	31	1.69

Tabla 22. Afinación del Porcentaje influencia, espacio Circunstancial.

Selección	Cruzamiento	P_i (% influencia)	P_{mp} (% mutación)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Elite	1 Punto	8	2	79.06	31	1.69
Elite	1 Punto	8	3	78.29	36	1.30
Elite	1 Punto	8	4	78.79	28	1.072
Elite	1 Punto	8	6	77.96	30	1.004
Elite	1 Punto	8	8	78.25	24	1.567

Tabla 23. Afinación del Porcentaje de mutación, espacio Circunstancial.

Resultado final en el algoritmo cultural con tipo de conocimiento Circunstancial.

Selección	Cruzamiento	P_i (% influencia)	P_{mp} (% mutación)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Elite	1 Punto	8	2	79.06	31	1.69

Tabla 24. Resultado de la afinación de parámetros y funciones, espacio Circunstancial.

Conocimiento Distribución Normativo “Tabla 2”:

Selección	Cruzamiento	P_i (% influencia)	P_{mp} (% mutación)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Elite	1 Punto	2	2	76.095	21	1.31
Destructivo	1 Punto	2	2	76.82	20	0.9891
Torneo	1 Punto	2	2	78.22	34	1.047

Tabla 25. Afinación de las funciones de selección, espacio Distribución Normativo.

Selección	Cruzamiento	P_i (% influencia)	P_{mp} (% mutación)	Porcentaje del mejor	PF del mejor	Desviación estándar
-----------	-------------	----------------------	-----------------------	----------------------	---------------	---------------------

				resultado	resultado	
Elite	1 Punto	2	2	78.22	34	1.047
Elite	Uniforme	2	2	74.92	28	1.20

Tabla 26. Afinación de las funciones de cruzamiento, espacio Distribución Normativo.

Selección	Cruzamiento	P_i (% influencia)	P_{mp} (% mutación)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Elite	1 Punto	2	2	78.22	34	1.047
Elite	1 Punto	4	2	76.9	23	1.18
Elite	1 Punto	6	2	77.26	30	1.139
Elite	1 Punto	8	2	76.86	28	1.374

Tabla 27. Afinación del Porcentaje influencia, espacio Distribución Normativo.

Selección	Cruzamiento	P_i (% influencia)	P_{mp} (% mutación)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Elite	1 Punto	2	2	78.22	34	1.047
Elite	1 Punto	2	3	76.95	23	1.211
Elite	1 Punto	2	4	76.16	21	0.783
Elite	1 Punto	2	6	76.96	24	0.988
Elite	1 Punto	2	8	77.41	29	1.02

Tabla 28. Afinación del Porcentaje de mutación, espacio Distribución Normativo.

Resultado final en el algoritmo cultural con tipo de conocimiento Normativo.

Selección	Cruzamiento	P_i (% influencia)	P_{mp} (% mutación)	Porcentaje del mejor resultado	PF del mejor resultado	Desviación estándar
Elite	1 Punto	2	2	78.22	34	1.047

Tabla 29. Resultado de la afinación de parámetros y funciones, espacio Normativo.

Conclusiones y trabajos futuros

En esta investigación, se trabajó en la resolución de problemas de múltiples objetivos, con funciones objetivos que se contra oponen entre sí, en las instancias utilizadas las funciones que se trataron son, UFLP *Uncapacitated Facility Location Problem* y MCLP *Maximal Covering Location Problem*.

Si los problemas con múltiples objetivos, son relativamente nuevos, la investigación sobre este, ha quedado demostrado por varios estudios, que los algoritmos culturales, son unos de los mejores para obtener soluciones que tiene dos o más funciones objetivos que se contra oponen entre sí. En el trabajo desarrollado se demostró la importancia que tiene el espacio de creencia, el cual al cambiar el algoritmo al Circunstancial se obtuvieron buenos resultados en el porcentaje del área obtenida por el PFCurrent, pero un mal resultado en la cantidad de individuos en esta. Mientras tanto que el espacio Normativo, obtuvo peores resultados al calcular el área del PFCurrent, pero si obtuvo mayor número de soluciones en esta.

Un trabajo interesante que puede continuar con esta investigación sería resolver el problema BOUFLP o Multi-Objetivos mediante otras heurísticas, realizando una comparación entre el algoritmo cultural y otra metaheurística, debido a que este problema en la literatura ha sido poco abarcado y sólo ha podido ser comparado con benchmarks obtenidos de [Villegas 2003]

Referencias

- [A. Carrasco, 2009] Evolución de functional link networks con un enfoque multiobjetivo, pág 26-30.
- [Báez, 2006] Marcos A. Báez, Derlis A. Zárate y Benjamín Barán “Algoritmos Meméticos Adaptativos para Optimización Multi-Objetivo”, pág 2-7.
- [Bilde, 1977] O.Bilde and J. Krarup (1977). “Sharp lower bounds and efficient algorithms for the simple plant location problem Annals of Discrete Mathematics”, pág 79-97.
- [Chung y Reynolds, 1996] ChanJin Chung, Robert G. Reynolds (1996). “A self-adaptive approach to representation shifts in cultural algorithms”.
- [Coello, 1998] Carlos Coello 1998 “Optimización Evolutiva con Objetivos Múltiples: Estado del Arte y Tendencias Futuras”.
- [Coello, 2005] Introducción a la Optimización Multiobjetivo Usando Metaheurísticas. Disponible vía web en <http://elavio2005.uniandes.edu.co/MiniCursosTutoriales/Coello3.pdf>
- [Collette, 2002] Yann Collette, Patrick Siarry (2002). “Multiobjective Optimización. Principales and Case Studies”, pág 20-30.
- [Fernández y Díaz, 2010] Boris Fernández y Daniela Díaz, “Resolución de un problema de bi-objetivo a través de un algoritmo cultural”, pág 36-49.
- [Galvão & Raggi, 1989] A method for solving to optimality uncapacitated location problems, pág 225-244.
- [Gutierrez, 2011] Optimización multiobjetivos utilizando algoritmos geneticos culturales, pág 36-37.
- [Jong, 1975] An analysis of the behavior of a class of genetic adaptive systems. PhD thesis, University of Michigan.
- [Landa, 2002] Ricardo Landa 2002 “Algoritmos culturales aplicados a optimización con restricciones y optimización Multiobjetivo”.Tesis magister, Centro de Investigación y Estudios avanzados del Instituto Politécnico Nacional, Mexico D.F.2002.58, pág 5-11.

- [Landa, 2004] Ricardo Landa, Carlos Coello (2004). “A cultural algorithm with differential evolution to solve constrained optimization problems”
- [Landa 2004a] Ricardo Landa, Carlos Coello (2004). “A cultural algorithm for solving the job shop scheduling problem”
- [Reynolds, 1994] Robert G. Reynolds (1994). “An Introduction to Cultural Algorithms”.
- [Reynolds, 1999] Robert G. Reynolds (1999). “Cultural algorithms: Theory and applications”. In David Corne, Marco Dorigo, y Fred Glover, editors, *New Ideas in Optimization*, pág 367–377. McGraw-Hill, London.
- [Soza, 2005] Carlos Soza (2005) Tesis “Algoritmos Culturales para resolución de problemas complejos de Asignación Horaria”.
- [Van Veldhuizen, 1999] On Measuring Multiobjective Evolutionary Algorithm Performance.
- [Villegas, 2003] Juan G. Villegas (2003) “Problemas de Localización multiobjetivo”,pág 69-83.
- [Zitzler, 1998] EckartZitzler and Lothar Thiele (1998). “Multiobjective Evolutionary Algorithms: A Comparative Case Study and theStrength Pareto Approach”

Anexos

Anexo I resultados del conocimiento Circunstancial

Instancia	Tiempo	Porcentaje de Pareto	PF	Desviación estándar
A10-25C1	167	0,7058	13	0,589
A10-25C2	176	0,6136	11	0,7782
A10-25C3	184	0,6992	9	0
A10-25C4	193	0,5729	5	0,5728
A10-25C5	168	0,6352	8	0
A10-25C6	186	0,5961	8	0
A30-75C1	793	0,7828	16	1,5
A30-75C2	598	0,7798	23	0,675
A30-75C3	611	0,7374	21	0,772
A30-75C4	960	0,6307	10	3,42
A30-75C5	610	0,7628	32	1,68
A30-75C6	536	0,6656	14	2,3
A50-150C1	2386	0,6685	27	2,54
A50-150C2	4266	0,6418	27	2,1
A50-150C3	1644	0,7171	24	1,03
A50-150C4	1437	0,5849	17	1,09
A50-150C5	2333	0,6725	21	1,88
A50-150C6	2399	0,6535	26	0,7509
B10-25C1	196	0,7795	7	3.139
B10-25C2	128	0,5813	4	0,48
B10-25C3	229	0,7278	12	1,03
B10-25C4	222	0,7246	14	0,336
B10-25C5	185	0,725	4	0,482
B10-25C6	245	0,634	15	0,475
B30-75C1	736	0,6945	26	1.037
B30-75C2	882	0,7133	30	0,831
B30-75C3	683	0,6263	28	0,6624
B30-75C4	857	0,4947	27	0,845
B30-75C5	917	0,6869	37	0,97
B30-75C6	565	0,8502	24	0,665
B50-150C1	885	0,8105	18	0,745
B50-150C2	1405	0,7134	24	0,891
B50-150C3	1706	0,7169	13	0,489
B50-150C4	604	0,672	17	0,482
B50-150C5	482	0,547	27	0,812
B50-150C6	1745	0,756	34	0,745

Anexo II resultados del conocimiento Normativo

Instancia	Tiempo	Porcentaje de Pareto	PF	Desviación estándar
A10-25C1	37	0,6963	11	0,2158
A10-25C2	27	0,6135	10	3,16
A10-25C3	33	0,6992	9	0
A10-25C4	28	0,5728	3	1,08
A10-25C5	32	0,6352	8	0
A10-25C6	34	0,5944	8	0,9
A30-75C1	1908	0,73	12	4,15
A30-75C2	1201	0,709	30	0,5385
A30-75C3	1305	0,7098	29	0,9488
A30-75C4	1450	0,6048	15	0,874
A30-75C5	1725	0,7078	28	0,5163
A30-75C6	446	0,701	28	0,2629
A50-150C1	3481	0,714	42	0,812
A50-150C2	5412	0,62	32	0,871
A50-150C3	3941	0,716	27	0,458
A50-150C4	3419	0,601	20	0,315
A50-150C5	2745	0,7158	35	0,413
A50-150C6	2811	0,704	48	0,514
B10-25C1	36	0,7721	8	1,11
B10-25C2	48	0,5807	4	1.148
B10-25C3	101	0,717	13	0,5653
B10-25C4	182	0,7218	12	5,2
B10-25C5	25	0,802	5	6,89
B10-25C6	31	0,5967	12	2,09
B30-75C1	1208	0,7119	20	1,28
B30-75C2	1014	0,7115	24	1.731
B30-75C3	1076	0,6106	28	0,675
B30-75C4	0,472	0,4824	29	0,487
B30-75C5	1820	0,6647	25	1,31
B30-75C6	781	0,831	35	0,972
B50-150C1	2371	0,809	20	0,713
B50-150C2	2678	0,705	35	0,489
B50-150C3	2791	0,694	45	0,178
B50-150C4	3841	0,642	15	0,955
B50-150C5	5712	0,587	25	0,845
B50-150C6	4958	0,715	54	0,469

Anexo III Tabla comparativa del conocimiento Circunstancial VS Histórico

Instancia	Tiempo (T1)	Porcentaje de Pareto (%1)	PF (PF1)	Tiempo (T2)	Porcentaje de Pareto (%2)	PF (PF2)	Porcentaje de Pareto (%1-%2)/%2
A10-25C1	167	0,7058	13	252	0,7057	13	0,000141703
A10-25C2	176	0,6136	11	197	0,6136	11	0
A10-25C3	184	0,6992	9	186	0,6993	9	-0,000143
A10-25C4	193	0,5729	5	115	0,5729	5	0
A10-25C5	168	0,6352	8	152	0,6351	8	0,000157456
A10-25C6	186	0,5961	8	183	0,5982	9	-0,003510532
A30-75C1	793	0,7828	16	464	0,8071	18	-0,030107793
A30-75C2	598	0,7798	23	1247	0,8013	47	-0,026831399
A30-75C3	611	0,7374	21	1177	0,7552	50	-0,023569915
A30-75C4	960	0,6307	10	450	0,7471	21	-0,155802436
A30-75C5	610	0,7628	32	940	0,779	46	-0,020795892
A30-75C6	536	0,6656	14	740	0,7245	37	-0,081297447
A50-150C1	2386	0,6685	27	1473	0,7878	52	-0,151434374
A50-150C2	4266	0,6418	27	1043	0,7611	37	-0,156746814
A50-150C3	1644	0,7171	24	1165	0,7883	53	-0,090320944
A50-150C4	1437	0,5849	17	735	0,6345	17	-0,078171789
A50-150C5	2333	0,6725	21	1459	0,8039	53	-0,163453166
A50-150C6	2399	0,6535	26	1434	0,7903	43	-0,173098823
B10-25C1	196	0,7795	7	135	0,7795	7	0
B10-25C2	128	0,5813	4	130	0,5856	5	-0,007342896
B10-25C3	229	0,7278	12	238	0,741	14	-0,017813765
B10-25C4	222	0,7246	14	260	0,7261	14	-0,002065831
B10-25C5	185	0,725	4	115	0,8117	6	-0,106812862
B10-25C6	245	0,634	15	285	0,6517	18	-0,027159736
B30-75C1	736	0,6945	22	683	0,765	38	-0,092156863
B30-75C2	882	0,7133	30	944	0,7323	59	-0,025945651
B30-75C3	683	0,6263	28	1351	0,6335	72	-0,01136543
B30-75C4	857	0,4947	27	618	0,5214	32	-0,051208285
B30-75C5	917	0,6869	37	905	0,7473	45	-0,080824301
B30-75C6	565	0,8502	24	800	0,8775	42	-0,031111111
B50-150C1	885	0,8105	18	802	0,8345	23	-0,028759736
B50-150C2	1405	0,7134	24	1221	0,7634	53	-0,065496463
B50-150C3	1050	0,7048	27	1410	0,7915	58	-0,10953885
B50-150C4	604	0,672	17	567	0,7498	10	-0,103761003
B50-150C5	482	0,547	27	939	0,6393	37	-0,144376662
B50-150C6	1745	0,756	34	1904	0,8016	67	-0,056886228

Anexo IV, Tabla comparativa del conocimiento Normativo VS Histórico

Instancia	Tiempo (T1)	Porcentaje de Pareto (%1)	PF (PF1)	Tiempo (T2)	Porcentaje de Pareto (%2)	PF (PF2)	Porcentaje de Pareto (%1-%2)/%2
A10-25C1	37	0,6963	11	252	0,7057	13	-0,01332011
A10-25C2	27	0,6135	10	197	0,6136	11	-0,00016297
A10-25C3	33	0,6992	9	186	0,6993	9	-0,000143
A10-25C4	28	0,5728	3	115	0,5729	5	-0,00017455
A10-25C5	32	0,6352	8	152	0,6351	8	0,00015746
A10-25C6	34	0,5944	8	183	0,5982	9	-0,00635239
A30-75C1	1908	0,73	12	464	0,8071	18	-0,0955272
A30-75C2	1201	0,709	30	1247	0,8013	47	-0,11518782
A30-75C3	1305	0,7098	29	1177	0,7552	50	-0,06011653
A30-75C4	1450	0,6048	15	450	0,7471	21	-0,19046982
A30-75C5	1725	0,7078	28	940	0,779	46	-0,09139923
A30-75C6	446	0,701	28	740	0,7245	37	-0,03243616
A50-150C1	3481	0,714	42	1473	0,7878	52	-0,0936786
A50-150C2	5412	0,62	32	1043	0,7611	37	-0,18538957
A50-150C3	3941	0,716	27	1165	0,7883	53	-0,09171635
A50-150C4	3419	0,601	20	735	0,6345	17	-0,05279748
A50-150C5	2745	0,7158	35	1459	0,8039	53	-0,10959075
A50-150C6	2811	0,704	48	1434	0,7903	43	-0,10919904
B10-25C1	36	0,7721	8	135	0,7795	7	-0,00949326
B10-25C2	48	0,5807	4	130	0,5856	5	-0,00836749
B10-25C3	101	0,717	13	238	0,741	14	-0,03238866
B10-25C4	182	0,7218	12	260	0,7261	14	-0,00592205
B10-25C5	25	0,802	5	115	0,8117	6	-0,01195023
B10-25C6	31	0,5967	12	285	0,6517	18	-0,08439466
B30-75C1	1208	0,7119	20	683	0,765	38	-0,06941176
B30-75C2	1014	0,7115	24	944	0,7323	59	-0,02840366
B30-75C3	1076	0,6106	28	1351	0,6335	72	-0,03614838
B30-75C4	0,472	0,4824	29	618	0,5214	32	-0,07479862
B30-75C5	1820	0,6647	25	905	0,7473	45	-0,11053125
B30-75C6	781	0,831	35	800	0,8775	42	-0,05299145
B50-150C1	2371	0,809	20	802	0,8345	23	-0,03055722
B50-150C2	2678	0,705	35	1221	0,7634	53	-0,07649987
B50-150C3	2791	0,694	45	1410	0,7915	58	-0,12318383
B50-150C4	3841	0,642	15	567	0,7498	10	-0,14377167
B50-150C5	5712	0,587	25	939	0,6393	37	-0,08180823
B50-150C6	4958	0,715	54	1904	0,8016	67	-0,10803393