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

RESOLUCIÓN DEL PROBLEMA DE LOCALIZACIÓN DE INSTALACIONES INCORPORANDO DECISIONES DE INVENTARIO CON UNA HEURÍSTICA DE BÚSQUEDA LOCAL

EDUARDO ADOLFO RAMÍREZ VALDERRAMA

**INFORME FINAL DEL PROYECTO
PARA OPTAR AL TÍTULO
PROFESIONAL DE INGENIERO
CIVIL INFORMÁTICO**

DICIEMBRE 2008

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

ACTA DE APROBACIÓN

EDUARDO ADOLFO RAMÍREZ VALDERRAMA

GUILLERMO CABRERA

PROFESOR GUÍA

Dedico este libro a mi madre, padre y hermanos quienes han sido mis alentadores constantes e incondicionales de esta gran tarea personal y familiar

LISTA DE ABREVIATURAS

- CD’s: CENTROS DE DISTRIBUCION
- Dc: DISTANCIA DE COBERTURA
- DRD: DISEÑO DE REDES DE DISTRIBUCION
- EOQ: ECONOMIC ORDER QUANTITY
- LS: BUSQUEDA LOCAL/ LOCAL SEARCH
- PR: PUNTO DE REORDEN
- SCP: SET COVERING PROBLEMS
- SPP: SET PARKING PROBLEM
- SPaP: SET PARTITIONING PROBLEM
- SPLP: SIMPLE PLANT LOCATION PROBLEM
- TS: TABU SEARCH/BUSQUEDA TABU
- UFLP: UNCAPACITATED FACILITY LOCATION PROBLEM

CAPITULO 1 INTRODUCCIÓN

1.1 Introducción

Siempre ha existido en nuestra sociedad un tema de recurrente discusión, el desplazamiento, tema que atañe desde individuos hasta las grandes empresas. Para la gran mayoría de las personas es un inconveniente trasladarse largos trayectos hacia el trabajo, el colegio, al hospital y a cualquier servicio en particular, y desde el punto de vista de las empresas es un tema constante en sus finanzas disminuir los tramos de recorridos hacia sus clientes, proveedores, centros de distribución, etc. Más aún cuando el desplazamiento es cada día más costoso por diferentes factores como son las constantes alzas de precios en los combustibles, alto número de peajes en las autopistas, gastos de seguros, etc. A esto se suma todos los tiempos y esfuerzos acumulados por una mala ubicación geográfica de los sitios.

Es por todos estos motivos antes expuestos que se hace necesario realizar estudios que tengan como fin principal ubicar todos estos tipos de instalaciones en lugares estratégicos, que hagan disminuir las distancias de traslado entre un demandante y una instalación en particular.

Al momento de tomar una decisión sobre el mejor lugar donde se desea ubicar una instalación, son múltiples factores lo que se toman en consideración tales como; aspectos culturales de un país, restricciones gubernamentales y jurídicas, condiciones climáticas y ambientales de un país, mercado de consumidor, abastecimiento de insumo, disponibilidad, costo de terrenos, mano de obra, vías de acceso, entre otras. Sin embargo para el caso de esta investigación, la toma de decisión tendrá un carácter netamente financiero-económico, el cual buscará dar un servicio cercano al 100% al costo más bajo posible.

Los estudios de localización se van a diferenciar de acuerdo a su envergadura en; *superlocalización* que son los casos en donde organizaciones transnacionales que deben escoger a nivel mundial una nación o país que posea ciertos patrones atractivos. *Macrolocalización*, cuando una empresa de carácter nacional analiza varias regiones dentro de una nación para fijar sus operaciones de producción o servicio.

Dentro de los factores económicos que determinarán la ubicación apropiado para situar la localización se pueden considerar los costos: fijos de instalación, mano de obra, de transporte desde una instalación hasta el cliente, distribución, de orden y finalmente de **inventario**. Es este último costo, un punto esencial en el desarrollo de esta investigación, el cual es integrar a la toma de decisión no solo la cantidad de locaciones que minimicen el costo, sino que también un nivel de inventario que disminuya los costos de éste.

Sobre los diferentes problemas de localización existentes en la literatura, son diversos los métodos que se han utilizado para encontrar una óptima solución, si el número de datos que se maneja son pocos, las alternativas de respuestas no son muchas, por lo tanto el nivel de solución no es compleja. Sin embargo para este estudio se buscará encontrar una solución de mayor complejidad, esto ya que se analizará un dominio de solución de gran escala. Es por esta razón que será necesario estudiar alguna heurística que pueda enfrentar con éxito el problema propuesto.

El desarrollo de este documento se inicializará con una descripción del proyecto (capítulo 1), sus objetivos propuestos y la metodología utilizada para su desarrollo. En el segundo capítulo se hará una pequeña investigación de los modelos de localización utilizados como base para el modelo preliminar propuesto [Mark, 1995] y las restricciones en sus modelos matemáticos. En el capítulo 3 se explicará la importancia de integrar los costos de inventario a los modelos de localización. En el cuarto capítulo se fundamentará la importancia de la utilización de las metaheuristicas para la resolución de este problema. Finalmente en el sexto capítulo se buscará integrar el modelo preliminar a la Metaheurística Tabu Search, en donde se fijará la manera de trabajar la memoria tabu, la forma de utilizar la vecindad y los cambios de resultados.

1.2 Descripción del Proyecto

Uno de los problemas de alta estrategia dentro de la gestión en la cadenas de abastecimientos es el *diseño de la red de distribución* (DRD), el cual consiste en crear una topología compuesta por una planta central encargada de abastecer a un conjunto de centros de distribución (CD's), el cual a su vez abastece a un conjunto de clientes conocidos anteriormente (figura 1.1). En ésta red los nodos variables serán los centros de distribución, los cuales deben ser ubicados en un lugar estratégico tal que minimice los costos totales [Miranda, 2007].

Dentro de la literatura existente se presentan diversos problemas tipos de localización de instalaciones [Daskin, 1995], en donde se considera diversas variables como son; demandas (estocásticas o determinísticas), número de centros de distribución a instalar (fijos o con una cantidad máxima), espacios de solución donde se ubicarán los CD's fijos o variables, que un cliente sea servido por uno o más CD, etc. Será parte de la investigación determinar cuáles serán los casos que se modelarán y las restricciones que se van a considerar.

La decisión que determina donde se ubicará la instalación está regida por dos aspectos primordiales, la primera es el nivel de servicio que se entregará a los clientes, esto quiere decir la capacidad que posee el CD's de abastecer a sus clientes. Si la demanda es conocida es posible determinar por lo tanto un nivel de inventario y un punto de reorden (PR) capaz de otorgar un servicio del 100%, sin embargo en caso de que la demanda sea estocástica, es imposible asegurar en su totalidad la capacidad del servicio. Un segundo aspecto que determinará la decisión de instalar el CD's son sus costos asociados, los cuales se dividen en costos de transporte, de instalación y de inventario. Este último costo será el centro de la investigación del proyecto, en donde se buscará integrar este costo al modelo general que genere un costo cercano al óptimo.

Tomando en consideración el inventario para la decisión de ubicar un CD's, es preciso trabajar para maximizar nivel de servicio a los clientes que requieren del acceso del producto, y minimizar los costos que significan tener un inventario con un stock superior al necesario. Es por lo tanto su ubicación, un lugar estratégico tal que optimice ambos resultados.

Tomando en cuenta que el amplio espacio de solución que se puede generar es imposible analizar todas las combinaciones posibles, se ha decidido que la manera de resolverlo será a través de la aplicación de heurísticas de búsqueda local, con lo cual se espera obtener buenos resultados.

Con los resultados obtenidos a través de la implementación se realizará un Benchmarking con lo cual se buscará comparar los resultados obtenidos en la investigación con trabajos semejantes.

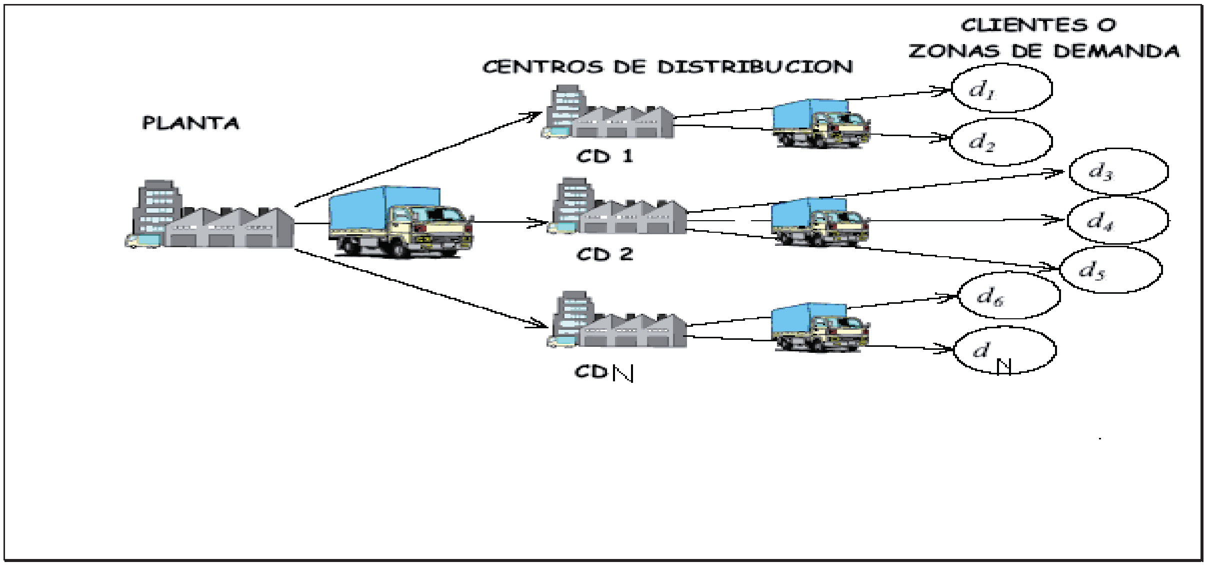

Figura 1.1 Diseño de Redes de Distribuciones

1.3 Definición de Objetivos

1.3.1 Objetivo General

- Resolver el problema de localización de instalaciones integrando los costos de inventario con el fin de ser resuelto a través de una heurística de búsqueda local.

1.3.2 Objetivos Específicos

- Estudiar los diferentes modelos de localización de instalaciones existentes.

- Generar un modelo con una estructura de datos que solucione el problema de localización de instalaciones a través de una heurística de Búsqueda Local

- Implementar la solución a través del lenguaje C.

- Validar los resultados obtenidos a través de benchmarking.

1.4 Metodología.

El desarrollo del presente proyecto de investigación será necesario utilizar tres diferentes metodologías que guíen el éxito del proyecto en su integridad. Las metodologías utilizadas serán: exploratorias, explicativas y descriptivas [Cazau, 2002].

Tomando en cuenta las numerosas publicaciones existentes sobre los modelos de localización, teorías de inventarios y las diferentes heurísticas existentes hoy en día, se utilizará una **metodología exploratoria** para poder investigar en cada uno de estos temas por separados para luego proceder a integrar sus relaciones. Tomando en cuenta que al integrar el costo de inventario a los modelos de localización existentes nos enfrentamos a un modelo aún en investigación, la metodología exploratoria es la más adecuada para su éxito.

En una segunda etapa se procederá a realizar una **metodología descriptiva** para poder generar el modelo matemático que integre los diferentes modelos estudiados, tanto de localización como de inventario. Esto ya que será necesario describir fenómenos de comportamiento homogéneos entre los modelos que hace posible unificarlos.

Finalmente se utilizará una **metodología explicativa** que haga posible justificar los resultados obtenidos. Para probar que los resultados conseguidos poseen un grado satisfactorio de validez será necesario hacer comparaciones con otras publicaciones semejantes a través de benchmarking.

CAPITULO 2 ESTADO DEL ARTE

2.1 Modelos de Localización.

El tema de la localización de instalaciones ha sido un problema presente desde los orígenes en nuestra sociedad, preguntas como donde instalar un poblado, o donde pasar la noche han sido dilemas que siempre han existido. En sus orígenes, este tipo de problemas fue tema de investigación de matemáticos [Carrizosa, 2005], los cuales a través de métodos algebraicos, geométricos o mecánicos respondieron a preguntas como:

- Hallar el punto del plano tal que la suma de las distancias a tres puntos fijos sea mínima,
- Hallar el centro del círculo de mínimo radio que encierra a un conjunto de puntos dado,
- Hallar el punto de un polígono fijo tal que sea máximo el radio del círculo con éste como centro y que no contenga a su interior a ninguno de los puntos de un cierto conjunto P,
- Hallar los centros de N círculos contenidos en un cuadrado, que no tengan más solapamiento entre ellos que en sus bordes, y que el radio de los discos sea máximo,
- Hallar la forma de interconectar N puntos, de modo que se minimice la suma de las longitudes de los segmentos de conexión.

En el siglo XX este y muchos otros problemas han sido estudiados por una amplia gama de profesionales de la ciencia los cuales han encontrado en esta área un gran nicho de estudio para desarrollarse, sin embargo ha sido la investigación de operaciones la que ha centrado en profundidad los temas de localización de instalaciones. Hoy en día los estudios de localización son requeridos en temas tan variados como las telecomunicaciones, finanzas, transporte, producción, etc.

Un problema de localización se puede describir como un conjunto de clientes distribuidos espacialmente en un área geográfica, los cuales demandan un servicio o producto. Esta demanda necesita ser cubierta por una o más instalaciones. El proceso de decisión para establecer donde se ubicarán las instalaciones se basará en decisiones de restricciones geográficas y requerimientos de los clientes. Generalmente los requerimientos tratan de cubrir la demanda total al menor costo posible, sin embargo hay casos, especialmente aquellos del ámbito social en los cuales no solo buscan minimizar los costos, si no que dar la mejor prestación a la sociedad.

En todos los problemas de localización se identifican tres elementos esenciales. Las **instalaciones**, que denotan un conjunto de objetos que serán localizados para proporcionar un servicio o un producto. Las

localizaciones, que son un conjunto de posibles puntos para situar las instalaciones. Y por último, los **clientes**, que son los usuarios de las instalaciones que demandan ciertos servicios o productos.

En los siguientes puntos se presentarán diferentes modelos de localización existente en la literatura actual, los cuales han sido estudiados para solucionar diferentes problemas específicos de localización. Se podrá observar que en todos se busca minimizar los costos de instalación a través de diferentes restricciones que son impuestas para satisfacer las especificaciones impuestas. Es parte de esta investigación conocer estos diferentes modelos, los cuales serán la base de éste proyecto.

2.1.1 Modelo Set Covering.

El Set Covering Problems (SCP) es un problema clásico que consiste en encontrar un conjunto de instalaciones que logren cubrir un conjunto de necesidades al **menor costo posible**. El SCP al igual que el Set Parking Problem (SPP) y El Set Partitioning Problem (SPaP) son problemas pertenecientes a la clase *NP-completo*, esto ya que las soluciones son demasiado complejas y no poseen la certeza que logren encontrar una respuesta exacta en un tiempo razonable. La entrada está dada por varios conjuntos de elementos o datos que tienen algún elemento en común. Estos tres problemas poseen la misma función objetivo y se diferencian por las restricciones que deben cumplir. [Itaim, 2005]

Las aplicaciones del SCP son variadas, siendo las más importantes la localización de servicio, selección de archivos en un banco de datos, simplificación de expresión booleanas, balanceo de líneas de producción, etc. El termino **coverage (**cobertura**)** es asociado con la capacidad que poseen los candidatos Ni de cubrir la demanda de sus clientes. Este sistema puede ser también especificado en términos de coeficientes binarios a_{ij} , el cual toma un valor 1 si el sitio candidato de locación puede cubrir la demanda del nodo i, en caso contrario tendrá el valor cero.

El problema de Set Covering puede ser formulado matemáticamente con la siguiente notación:

Entrada.

a_{ij} = 1, si el sitio candidato j puede cubrir la demanda del nodo i.

a_{ij} = 0, si no.

f_j = costo de localizar el candidato en el sitio j.

También se puede explicar el término a_{ij} como un valor igual a 1 si la vecindad del nodo i es adyacente al nodo j. Y un valor cero en caso contrario.

Variables de decisión.

X_j = 1 si localizamos en el sitio j al candidato.

X_j = 0 en caso contrario.

Con esta notación, podremos formular el problema de set covering como sigue:

Mínimo $$\sum_j f_j * X_j \qquad (1)$$

En donde $$\sum_j a_{ij} * X_j > 1 \quad \forall i \qquad (2)$$

$$X_j = 0,1 \quad \forall j \qquad (3)$$

La función objetivo (1) minimiza el costo total de seleccionar la locación j, la restricción (2) estipula que cada nodo debe ser cubierto por al menos una localización. Nótese que la parte izquierda de las función (2) entrega el número de locaciones que cubre la demanda del nodo i. En el caso de que el costo de todas las instalaciones sea semejante (por ejemplo f_j =1), la función objetivo a minimizar se simplifica a:

Mínimo $$\sum_j X_j \qquad (4)$$

En este caso la función (4), lo que nos responde, es la cantidad de locaciones que minimizan la función objetivo.
En los problemas de localización, frecuentemente los coeficientes a_{ij} se definen en término de la distancia que existe entre la demanda del nodo i y la el candidato de localización. Si se asigna una distancia de cobertura Dc, entonces tendremos que:

$$a_{ij} = 1, \text{ si } D_{ij} < D_c .$$

En donde D_{ij} nos entrega la distancia que existe entre el nodo candidato i, y el cliente que se desea abastecer. Esto quiere decir, que si la distancia de cobertura que se ha impuesto (D_c) es mayor a la distancia que existe entre ambos nodos (D_{ij}), va a significar que se logrará la cobertura mínima deseada para cubrir las distancias. Por lo tanto a_{ij} será 1.

En la literatura existente se ha podido comprobar que la efectividad de los resultados depende de la heurística aplicada y por el número de filas y columnas que se están utilizando (demanda, candidatos). En donde a medida que aumenta la cantidad de filas y columnas que se desean examinar aumenta la complejidad y tiempo de respuesta.

Los métodos usados para resolver el problema de SCP son variadas tipos de heurísticas, en donde se pueden destacar una heurística *dual* con 200 filas y 2000 columnas, combinaciones de heurísticas *Lagrangiana y Branch* mejorada, la cual aumento hasta 400 filas y 4000 columnas. Estos algoritmos cercanos a la optima están basados en procedimientos de árboles de búsqueda [Beasley, 1992].

Las múltiples opciones que cubren toda la demanda se diferencian por los diferentes costos que significa ubicar el servicio en un lugar u otro. Es por esta razón que la misión del problema Set Covering busca encontrar aquellas posiciones estratégicas que minimicen los costos totales. Para el caso que el problema se restrinja a costos de instalaciones iguales, se busca por lo tanto buscar el mínimo número de locaciones.

Un ejemplo real y sencillo es el que se grafica en la figura 2.1, en donde una ciudad ha sido dividida en 20 sectores, colindantes entre ellos. Dentro de esta ciudad se han elegido 10 lugares posibles donde sería posible instalar algún tipo de servicio como un cuerpo de bomberos, estación de carabineros, colegios, centros de Internet, etc. Esta decisión estratégica donde ubicar este servicio en particular, es un claro ejemplo de los problemas de Set Covering.

Para este caso va existir un vector de costos f_j = [f_1 , f_2 , f_3 …, f_{10}] que representan los costos que significa instalar en cada sitio el servicio. Los valores a_{ij} de la matriz tendrán el valor 1 si existe una cobertura entre la columna j y la fila i, por ejemplo la columna j=9 cubre la fila i=14, i=15, i=18, i=19.

Luego tomando en cuenta que x_j , será igual a 1 cuando instala el servicio en la locación j y cero en caso contrario se buscará encontrar el vector X= [x_1, x_2 , …., x_{10}] que minimice la función $\sum_{1}^{10} f_j * X_j$. De esta forma será posible ubicar los servicios que cubran todos los sectores a un costo mínimo.

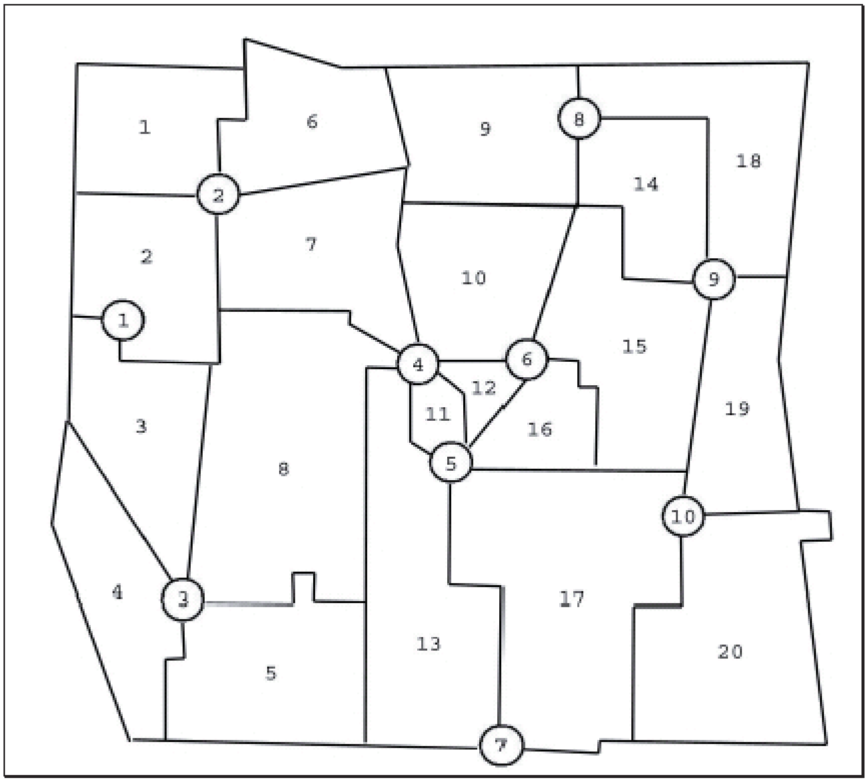

Figura 2.1 Problema de Set Covering

Es posible concebir que la cantidad de respuestas son variadas, sin embargo, es tarea de una adecuada heurística encontrar aquella solución que nos cubra toda la demanda al menor costo posible.

2.1.2 Problemas de Máxima Cobertura.

Los problemas de máxima cobertura son una modificación de los problemas clásicos de Set Covering. Una modificación más real de lo que ocurre en las empresas, en donde por diversas razones, generalmente financiera existen limitaciones respecto al número de instalaciones que es posible situar. Es por esta razón, que este nuevo modelo agrega una nueva variable (P), la cual indicará la cantidad máxima de instalaciones que se pueden ubicar, haciéndolo un modelo mucho más real que el anterior.

Es importante destacar que bajo esta restricción es imposible asegurar que el nivel de servicio sea abastecido en su totalidad, esto debido a que se puede dar el caso que el número de instalaciones máxima que es posible asignar sea una cantidad imposible de cubrir la totalidad de la demanda. Por esta razón la función objetivo (5) busca cubrir solo el máximo posible de la demanda solicitada, no su totalidad.

El presente modelo se puede representar matemáticamente a través de la siguiente notación:

Entrada:

μ_i = Demanda al nodo i.

P = Número de instalaciones a localizar.

Variables de decisión.

Z_i = 1 si el nodo i es cubierto.

Z_i = 0 si no es cubierto.

Con esta notación adicional podemos formular el modelo de máxima cobertura como:

Máximo $$\sum_i \mu_i * z_i \qquad \textbf{(5)}$$

En donde $$Z_i < \sum_j ai * Xj \quad \forall i \qquad \textbf{(6)}$$

$$\sum_j X_j \leq P \qquad \textbf{(7)}$$

$$X_j = 0,1 \qquad \forall_j \qquad \textbf{(8)}$$

$$Z_i = 0,1 \qquad \forall_i \qquad \textbf{(9)}$$

Se puede concluir que la restricción (7), nos permite mantener la condición de que el número de instalaciones no supera la cantidad exigida (P). Y las restricciones (8) y (9) mantienen la integridad del modelo planteado.

2.1.3 Problemas de Localización de Instalaciones con Carga Fija.

Los costos fijos asociados a la instalación de un servicio en particular, tienen obviamente diversos precios, dependiendo del sitio donde se ubiquen. Esto ya que múltiples factores inciden en su costo; precios del terreno, costo de manos de obra, permisos de instalación, entre otras. Por ejemplo el costo de instalar una antena de celular va a tener diferentes costos situarlo en el centro de la ciudad de Santiago que ubicarla en un cerro alejado de la capital. Es por esta razón, la necesidad de considerar estos costos fijos al momento de pensar en qué lugar ubicar un servicio que se desea instalar, una mala decisión va a significar elevar los costos de una manera que los resultados finales dejen de ser rentables, al contrario una perdida para la empresa.

Para solucionar esta situación hemos asignado un costo fijo, f_j el cual indicará el costo de asignar una instalación en el lugar j.

Los problemas de carga fija se pueden dividir en 2 tipos, *problema de carga fija con capacidad (capacitated) y de carga fija sin capacitado (uncapacitated)*, en donde su diferencia se radica en considerar o no, el hecho de que la instalación tenga el soporte suficiente para dar abasto a toda la demanda que se exige. Por ejemplo un centro de distribución de automóviles tiene solo una capacidad limitada para almacenar.

La importancia de los costos fijos asignados a la instalación de localizaciones se pueden observar a través de la figura 2.2, la cual demuestra el comportamiento de los costos totales. Primero se puede observar en la serie 1 que los costos del número de instalaciones crece de manera lineal a medida que aumenta el número de locaciones, de manera de simplificar el análisis asumimos que el costo de cada instalación es fijo. Por otra parte el costo correspondiente al transporte y distribución, va decreciendo fuertemente a medida que aumenta el número de instalaciones, esto ya que obviamente al existir una amplia cantidad de servicios va ser más rápido y eficiente la distribución. Sin embargo esta disminución de costos se va ver castigado por la gran cantidad de instalaciones, las cuales aumentan fuertemente el costo fijo de cada locación. (S3)

Es por este motivo que es necesario ubicar una cantidad apropiada que regule el costo total de las instalaciones a ubicar como se ve en la figura 2.2. La cual a través de la intersección de las 2 curvas encuentra un punto de equilibrio total de las 2 funciones. Esto se puede observar en el gráfico al momento de tener 5 instalaciones se logra un punto óptimo.

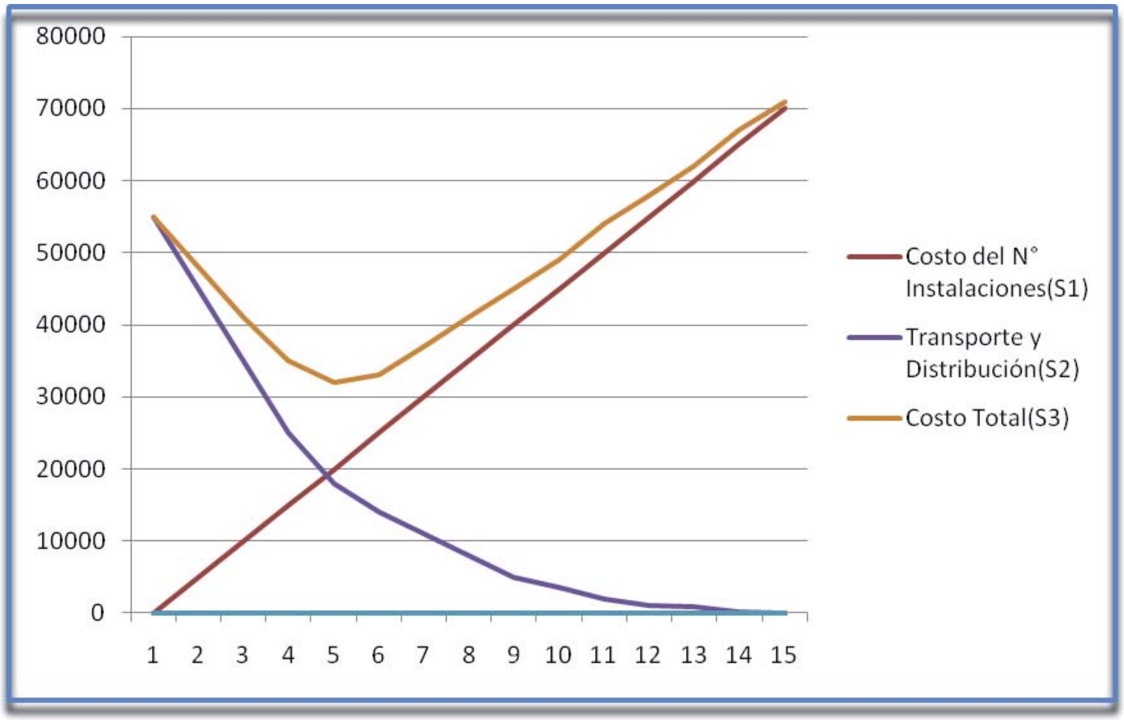

Figura 2.2 Punto de Equilibrio

2.1.4 Problema de localización de Instalaciones con carga fija uncapacitated.

Para este tipo de modelo se considera que la localización no posee la restricción de capacidad en su almacenamiento, situación que tiene sentido para casos en donde el producto que se desea almacenar no son de gran tamaño, en la literatura son llamados como *Simple Plant Location Problem* (SPLP), o por *Uncapacitated Facility Location Problem* (UFLP)

La función objetivo consiste en minimizar la sumatoria de localización de instalaciones incluida su distribución o costo de transporte. Para esto usaremos la siguiente notación matemática:

Entrada

f_j = Costo fijo de localizar un candidato en el sitio j.

μ_i = Demanda del nodo i.

d_{ij} = Distancia desde la demanda del nodo i al candidato j de localizar.

α = Costo por unidad de distancia demandada.

Variables de decisión.

X_j = 1 si localizamos en el sitio j al candidato.

X_j = 0 en caso contrario.

Y_{ij} = 1, si el cliente es abastecido por el CD. En caso contrario será 0.

Con esta notación, podremos formular el problema de localización de instalaciones con carga fija uncapacitated como:

Mínimo $$\sum_j fj * Xj + \alpha \sum_i \sum_j \mu_i * d_{ij} * Y_{ij} \qquad \textbf{(10)}$$

En donde:

$$\sum_j Y_{ij} = 1 \quad \forall_i \qquad \textbf{(11)}$$

$$Y_{ij} \leq X_j \quad \forall_{i,j} \qquad \textbf{(12)}$$

$$x_j = 0{,}1 \quad \forall i, j \qquad \textbf{(13)}$$

$$Y_{ij} \geq 0 \quad \forall_{i,j} \qquad \textbf{(14)}$$

En la literatura algunos autores [Ozen, 2004] hacen distinción a la distancia que existe entre la planta y el CD's de manera separada con la existente entre el CD's y los clientes. En el modelo matemático planteado (10) solo se considera la distancia que hay desde el CD's hasta los clientes [Daskin, 1995].

El objetivo de la función (10), es minimizar el costo total, el cual es la suma de los costos fijos de localización y el costo de transporte. La restricción (11) estipula que cada nodo i de la demanda es servido solo por un centro de distribución. La condición (12) nos permite asegurar que para que un cliente sea asignado a un centro de distribución, necesariamente esa instalación debe ser considerada válida.

2.1.5 Problema de localización de Instalaciones con carga fija capacitada.

La capacidad es un término constantemente mencionado y estudiado en diversos problemas de localizaciones de instalaciones, esto debido a que obviamente todas las instalaciones poseen una capacidad fija, la cual determina en ocasiones la velocidad de los flujos de sus procesos. Por ejemplo una empresa que ensambla 100 automóviles diarios en turnos de 8 horas, puede quizás aumentar su cantidad a 200 vehículos si trabaja en doble turnos diarios o triplicar el número si realiza 3 turnos diarios, sin embargo el hecho de que tenga una bodega que es capaz de almacenar solo120 vehículos diarios hace que nuestra restricción de capacidad sea significativa en la manera que funciona nuestra empresa. A través de este simple ejemplo es posible demostrar la importancia que posee esta restricción en la problemática de localización.

Para este problema se considerará la localización de instalaciones, minimizando la sumatoria de los costos de instalaciones y el costo de viaje del cliente a la instalación, sujeta a la restricción estipulada de que toda la demanda debe ser satisfecha desde servicios abiertos.

Para este problema debemos sumar una nueva entrada k_j.

Entrada:

$k_j =$ Es la capacidad de la instalación del sitio j, si el local se ubica ahí.

Luego con esta nueva definición podemos formular el problema de localización de instalaciones con carga fija capacitado como:

Mínimo $$\sum_j f_j * X_J + \alpha \sum_i \sum_j \mu_i * d_{ij} * Y_{ij} \qquad \textbf{(15)}$$

En donde:

$$\sum_j Y_{ij} = 1 \qquad \forall_i \tag{16}$$

$$Y_{ij} \leq X_j \qquad \forall_{i,j} \tag{17}$$

$$\sum_j \mu_i * Y_{ij} \leq k_j * X_j \quad \forall_j \tag{18}$$

$$x_j = 0,1 \qquad \forall i, j \tag{19}$$

$$Y_{ij} \geq 0 \qquad \forall i, j \tag{20}$$

Se puede observar que para este caso la función objetivo y algunas restricciones son iguales, solo hace la diferencia la nueva condición (21). La cual nos asegura que la suma de toda la demanda abastecido por un centro de distribución es menor a la cantidad k_j.

2.2 Modelo de Inventario.

Desde un punto de vista operativo, los inventarios representan aquellos bienes almacenados por una organización para los que se prevé una demanda futura. En cambio, desde un punto de vista económico, los inventarios suponen un capital invertido que es recuperado cuando se satisface la demanda para un artículo o servicio específico. [Gutiérrez, 2003]

Otra definición de inventario es la de un conjunto de productos y/o recursos utilizados en una organización (materias primas, productos terminados, repuestos, productos en procesos) empleados para satisfacer una demanda futura). Hay que considerar que el inventario representa un porcentaje importante del capital de trabajo de una empresa, por lo tanto es de vital importancia para las empresas una correcta utilización de ellos.

El problema de la gestión óptima de inventario se ve por primera vez a comienzos del siglo XX. La manera de afrontarla ha variado considerablemente este último siglo, en sus principios se consideraba que una estrategia correcta era mantener grandes cantidades de inventarios para hacer frente a fluctuaciones de demanda. Hoy en día se busca manejar niveles de inventario al mínimo, ya que de esta forma reduce el *costo* en cantidades que en algunas ocasiones son extremadamente considerables (productos refrigerados, congelados o de alto valor). Es por esta razón que el control de inventario involucra aquellas actividades y procedimientos utilizados para asegurar el mantenimiento de la cantidad correcta del stock. Para esto necesita por lo tanto manejar un nivel de sincronización de cuatro factores: *tiempo, discontinuidad, incertidumbre y económico*.

La gestión de inventario se desarrollo a partir del trabajo Harris de 1913, quien propuso el modelo **EOQ** (Economic Order Quantity), sistema que ha tenido diversas extensiones al considerarse diferentes hipótesis.

2.2.1 Modelo EOQ.

También llamado en ocasiones como la cantidad económica de pedido (CEP), es una herramienta utilizada para determinar el monto de pedido que reduzca al mínimo el costo total del inventario de la empresa. Por lo tanto nos entregará la cantidad exacta que es necesario solicitar.

Este modelo se basa en tres supuestos fundamentales:

- La empresa conoce cuál es la utilización anual de los artículos que se encuentran en el inventario. Por lo tanto usa una demanda determinísticas.
- La frecuencia con la cual la empresa utiliza el inventario no varía en el tiempo.
- Y por último los pedidos que se colocan para reemplazar las existencias de inventario se reciben en el momento exacto en que los inventarios se agotan.

El modelo EOQ, lo que busca es minimizar el costo total de inventario que está compuesto por:

Costo Total de Inventario= Costo de Compras+ Costo de Preparación+ Costo de Almacenamiento+ Costo Faltante

- El costo de compra se basa en el precio por unidad del artículo. Puede ser constante, o se puede ofrecer con un descuento que dependa del volumen del pedido.

- El costo de preparación representa el cargo fijo en el cual se incurre cuando se hace un pedido. Este costo es independiente del volumen del pedido.

- El costo de almacenamiento representa el costo de mantener suficientes existencias en el inventario. Incluye el interés sobre el capital, así como el costo de mantenimiento y manejo

- El costo de faltante es la penalidad en la cual se incurre cuando nos quedamos sin existencias. Incluye la perdida potencial de ingresos, así como el costo más subjetivo de la perdida de la buena voluntad de los clientes.

Tomando en cuenta que se está modelando una situación en donde se está tratando un solo producto, el inventario es reducido a una sola localización y que la demanda es continua a una velocidad constante, podemos definir:

λ = demanda esperada (cantidad/ tiempo).

LT_i = plazo de tiempo o leadtime (tiempo).

$I(t)$ = Inventario en el tiempo t. $t \geq 0$

$IO(t)$ =Inventario en orden, que es la cantidad solicitada de stock pero que aún no es recibida.

q = Orden o tamaño de batch (cantidad).

Si tomamos en cuenta que el inventario va decreciendo linealmente a través del tiempo como en la figura 2.3, a una velocidad constante de -λ, va a llegar un instante de tiempo que el nivel del inventario llegará a cero, es por lo tanto necesario hacer una solicitud a los proveedores del producto carente, en donde la solicitud debe indicar la cantidad necesaria para nivelar el inventario a la cantidad deseada. Por lo tanto LT_i, *será un tiempo de espera entre la emisión del producto y la llegada de esta*.

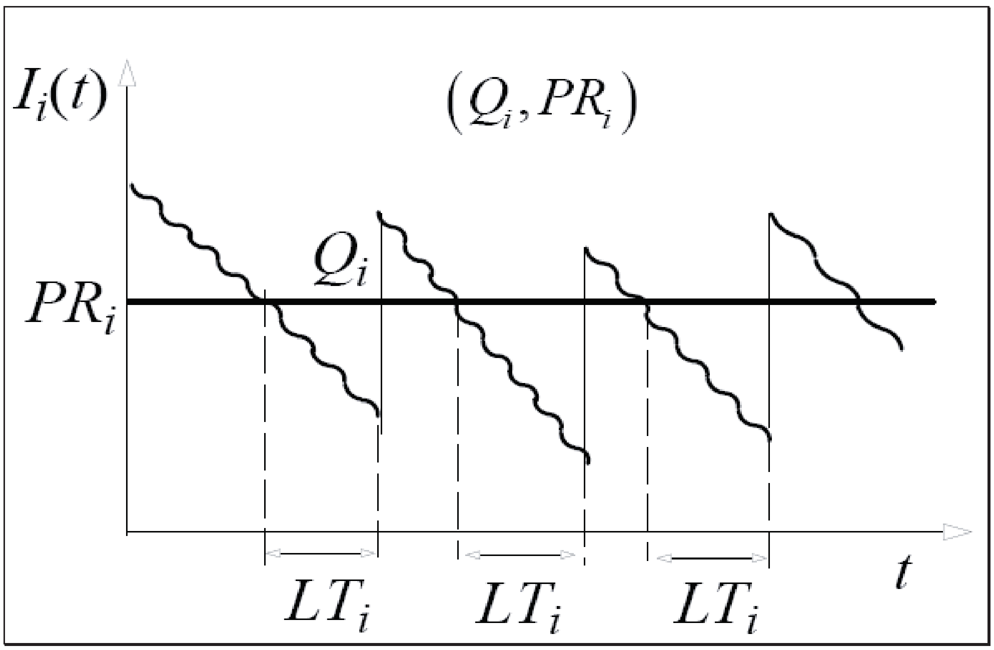

Figura 2.3 Punto de Reorden

En la figura 2.3, se observa en el eje y la cantidad que posee el inventario del producto en cuestión y en el eje x el tiempo. Es importante destacar que a pesar de que para el modelo EOQ se ha restringido que la demanda es lineal y constante durante el tiempo como se muestra en la figura, esto es difícil en su realidad ya que el comportamiento de la demanda varía con una frecuencia inesperada.

Si no se realiza un pedido en los momentos adecuados es posible que el nivel del inventario llegue a cero y eso va a significar que la empresa va a sufrir *costos de faltantes*, al no poder vender productos de su existencia. Esto va a causar que el cliente realice la compra en otra empresa, por lo tanto se perderían todas las posibles ventas hasta que llegue el pedido del producto requerido. También se puede dar el caso que se pida una orden con mucha anticipación, esto traería consigo tener que sostener costos de inventario innecesario por parte de la empresa. Es por lo tanto resolver la interrogante de qué *cantidad pedir y en qué momento exacto*, una respuesta fundamental para el éxito financiero de cualquier empresa.

Otros criterios importantes son la orden de frecuencia (OF) y el inventario promedio (IP), los cuales son datos que ayudan a tomar decisiones de la solicitud de un producto y valores de gran importancia para la función de costo total. Estos datos se obtienen a través de la siguiente fórmula:

$$\mathrm{OF} = \lambda / q \qquad \textbf{(21)}$$

$$\mathbf{IP} = q/2 \qquad \textbf{(22)}$$

Nótese que el inventario promedio aumenta en q, cuando la orden de frecuencia disminuye, estos dos criterios entran en un directo conflicto al momento de tomar la decisión de una orden [Zipkin, 2000]. En respuesta a la interrogante de las condición que se ocupan para decidir la cantidad de q a ordenar, son generalmente las restricciones exteriores quienes definen los criterios a considerar, en donde ambos casos se guían en un criterio de no exceder un parámetro predefinido. Por lo general se busca minimizar las frecuencias de las órdenes y mantener un promedio de inventario superior al necesario para abastecer sin inconvenientes a la demanda.

Una manera adecuada es trasladar estos dos criterios a una escala común (variable monetaria), en donde uniendo los costos de inventario y de orden es posible satisfacer el objetivo común, el cual es mantener el mínimo de costo y un servicio óptimo. Es necesario estimar un costo de factor para la adquisición y mantención del inventario (holding), este costo se mantiene constante durante el tiempo como se ve en la figura 2.4. Es necesario medir todos los costos en algún estándar monetario, las cuales se especificarán en las siguientes variables:

k = Costo fijo de situar una orden (monetario).

c = Costo variable de situar una orden (monetario/cantidad-unitaria).

h = Costo de tener una unidad en inventario por una unidad de tiempo (monetaria/ [cantidad-unidad*unidad de tiempo]).

El costo fijo k representa todo el costo de orden, el cual es independiente del tamaño del pedido. Generalmente se encuentran en este tipo de costo los equivalentes a: procesamiento, administración, recepción de orden, etc. El costo variable c representa el costo unitario de compra y los costos asociados al tamaño de la orden. El costo total por orden es por lo tanto $k + c*q$ (en un ciclo).

El costo holding h está compuesto generalmente por dos componentes. El primero comprende todo los costos directos asociados con los inventarios tales como; costos físicos de mantención, refrigeración, seguros, renta de almacenamiento, etc. El segundo costo es el referente al costo promedio del holding. De esta forma es posible definir el costo total promedio de (q) inventarios como la suma del costo de orden $(k+c*q)*OF$ y el costo promedio de holding $h*IP$. Con lo cual se desprende la siguiente formula $C(t)$, la cual indica el costo total promedio, graficado en la figura 2.4.

$$C(q) = (k + c*q)OF + h*IP$$
$$= c*\lambda + k*\lambda/q + 1/2*h*k \quad q > 0 \qquad \textbf{(23)}$$

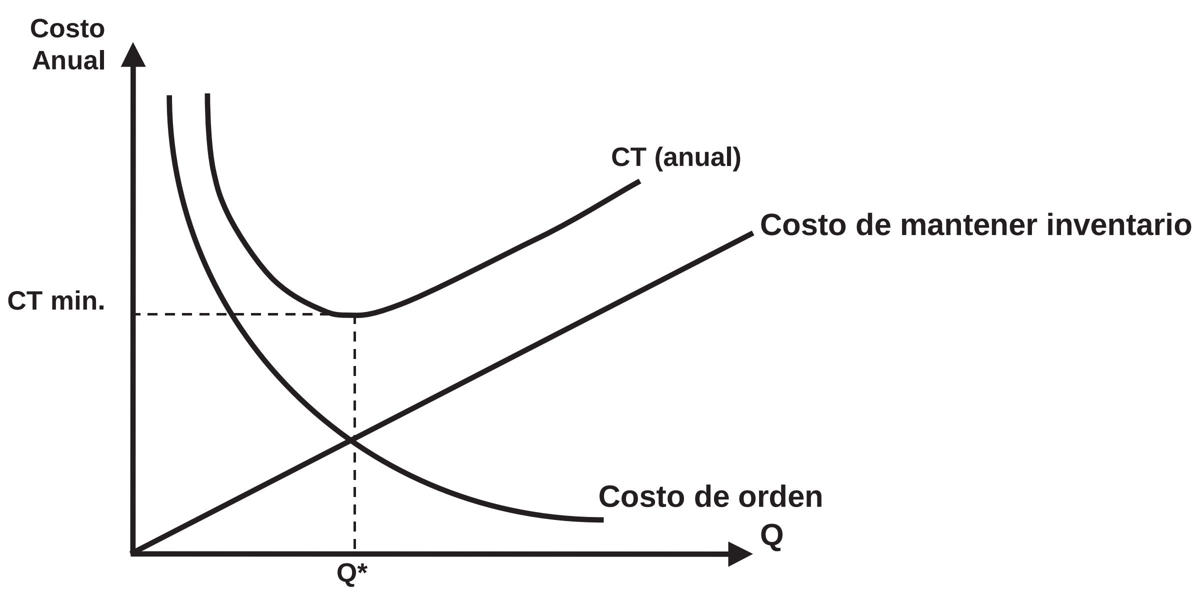

Figura 2.4 Costo Total

La función C (q) no entrega el costo total para las posibles ordenes de tamaño Q. La siguiente misión es encontrar aquella cantidad óptima que minimice los costos totales, esto se realiza derivando la función objetivo del costo total y luego igualando cero. De esta forma es posible encontrar la cantidad óptima de Q. De esta forma es posible encontrar el valor óptimo q*, el cual nos entregará la cantidad con la cual es posible minimizar los costos de inventario. Esta fórmula es comúnmente llamada (EOQ), o cantidad económica a ordenar.

$$C'(q) = -k\lambda / q^2 + 1/2 * h = 0$$

$$q^* = \sqrt{2k\lambda / h} \qquad \textbf{(24)}$$

2.2.2 Punto de Reorden (PRi).

Es una práctica que consiste en la presencia de una señal dirigida hacia el departamento encargado de colocar los pedidos. Esta señal indica que cierto material del inventario ha llegado a un nivel crítico y necesita ser abastecido. Este punto debe ser tal que permita seguir cumpliendo la demanda mientras llega el siguiente pedido. Este punto se puede ver en la figura 2.3, en donde se puede observar que se encuentra directamente relacionado con el *lead time.* Para su aplicación ocuparemos las siguientes funciones:

$I(t)$ = Inventario en tiempo t.

$B(t)$ = Pedidos pendientes al tiempo t, también llamado como Backorders.

$IN(t)$ = Inventario neto en tiempo t = $I(t) - B(t)$

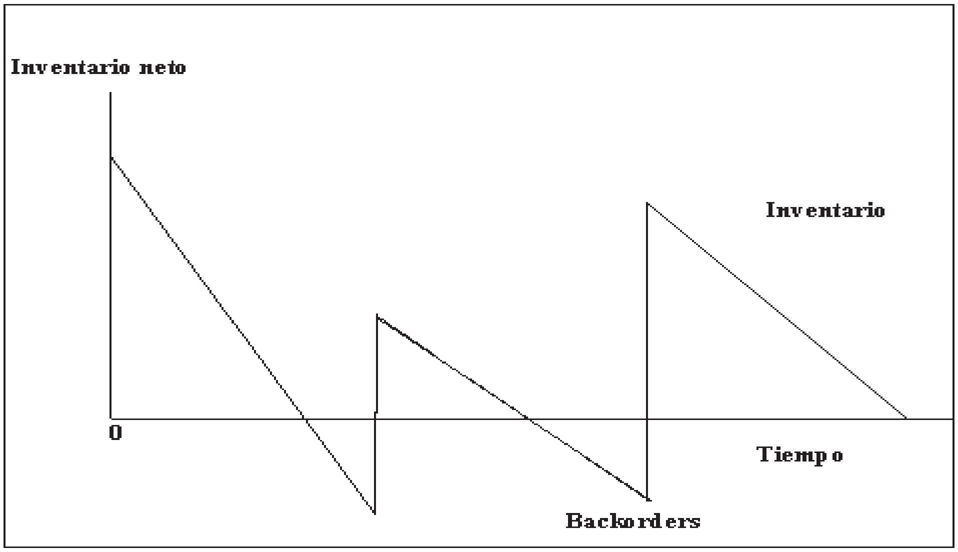

Figura 2.5 Punto de Reorden

En la figura 2.5 se puede graficar lo que ocurre en el nivel de inventario en momentos que el stock se encuentra con capacidad suficiente (mayor a cero), y en casos en donde el inventario a llegado a cero y se continúan haciendo solicitudes por parte de la demanda. Esta circunstancia es alejada por momentos de la realidad, ya que en muchos casos el cliente al encontrarse con un proveedor incapacitado busca un nuevo lugar donde abastecerse [Zipkin, 2000].

El inventario neto tiene la siguiente restricción: $IN(t)$ = Inventario neto en tiempo t = $I(t) - B(t)$

$IN(t)$ = $I(t)$ cuando $IN(t) \geq 0$

- $B(t)$ cuando $IN(t) \leq 0$

Por lo tanto tenemos que:

$I(t)$ = [$IN(t)$]+

$B(t)$ =[$IN(t)$]-

El punto de reorden tiene una relación directa con el Lead Time como se ve en la figura 2.6, cuando la planta central se encuentra más cerca de la bodega o del centro de distribución, el tiempo que se demora en llegar la orden desde un punto a otro disminuye linealmente, gráficamente se puede ver como si la línea del punto de reorden disminuyera en la cantidad del nivel de inventario.

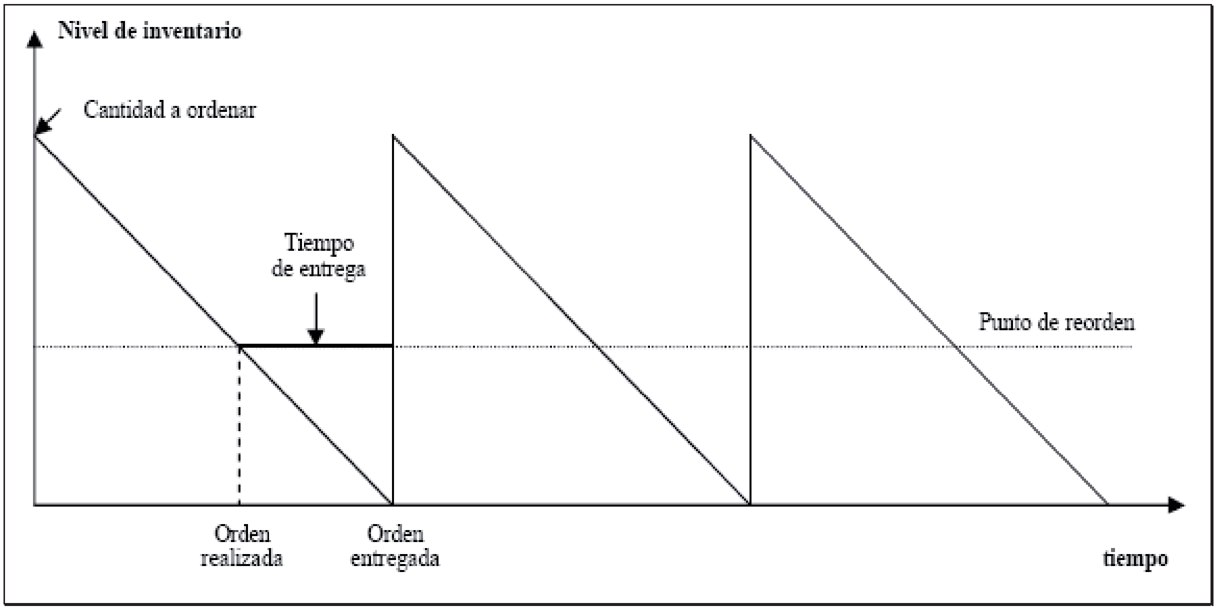

Figura 2.6 Punto de Reorden

De esta forma la distancia que existe entre la orden realizada y la orden entregada tiende a cero a medida que la fabrica y los centros de distribución de acercan.

2.3 Heurísticas.

La localización de instalaciones es descrito como un problema del tipo optimización combinatorial, tema que es ampliamente estudiado en diversas áreas tales como: economía, industria, medicina, comercio, entre otros. En el contexto científico la optimización es el proceso de tratar de encontrar la mejor solución posible dentro una gama de opciones, donde el criterio utilizado para obtener el mejor resultado se basa en la función objetivo (valores máximos y mínimos) y el conjunto de restricciones que el problema plantea.

Las técnicas existentes para resolver los problemas de optimización combinatorial se pueden dividir en *métodos exactos y métodos aproximados*. Los algoritmos exactos buscan una solución óptima y demostrar que la solución obtenida es de hecho el óptima global, debido a que los algoritmos exactos muestran un rendimiento pobre para muchos problemas se han desarrollado múltiples tipos de algoritmos aproximados que proporcionan soluciones de alta calidad para estos problemas combinatorios (aunque no necesariamente la óptima) en un tiempo computacional breve [Alonso, 2005].

Considerando el alto número de problemas de alta complejidad (NP-Completos) que necesitan ser resuelto en forma eficiente, se han desarrollado procedimientos para encontrar buenas soluciones aunque no sean necesariamente las óptimas. Estos métodos, en donde *la rapidez* de la solución es tan importante como la *calidad* obtenida se concentran las heurísticas, las cuales se pueden definir de la siguiente forma:

"Un método heurístico es un procedimiento para resolver un problema de optimización bien definido mediante una aproximación intuitiva, en la que la estructura del problema se utiliza de forma inteligente para poder obtener una buena solución. [Martí, 2003]"

La función objetivo que se desea buscar, es parte de un conjunto finito de soluciones denotados como *S*, tomando en cuenta la característica finita del conjunto S, las variables toman valores discretos, restringiendo su dominio a una serie finita de valores. Habitualmente, el número de elementos de S es muy elevado, haciendo impracticable la evaluación de todas sus soluciones para determinar el óptimo.

Al abordar el estudio de los algoritmos heurísticos podemos comprobar que dependen en gran medida del problema concreto para el que se han diseñado. En otros métodos de resolución de propósito general, como pueden ser los algoritmos exactos de Ramificación y Acotación, existe un procedimiento conciso y preestablecido, independiente en gran medida del problema abordado. En los métodos heurísticos esto no es así. Las técnicas e ideas aplicadas a la resolución de un problema son específicas de éste y aunque, en general, pueden ser trasladadas a otros problemas, han de particularizarse en cada caso. Así pues, es necesario referirse a un problema concreto para estudiar con detalle los procedimientos heurísticos [Martí, 2003]".

Otra de las razones que hacen necesario utilizar métodos heurísticos son:

• El problema es de una naturaleza tal que no se conoce ningún método exacto para su resolución.
• Aunque existe un método exacto para resolver el problema, su uso es computacionalmente muy costoso.
• El método heurístico es más flexible que un método exacto, permitiendo, por ejemplo, la incorporación de condiciones de difícil modelización.

Existen hoy en día una gran variedad de algoritmos heurísticas, cada uno con una naturaleza diferente, lo que hace muy complejo poder agruparlos según alguna característica en especial, a eso se suma el hecho de que mucho de ellos fueron creados para resolver problemas tan particulares que se hace imposible poder utilizarlo en otro tipo de problemas, eliminando cualquier posibilidad de generalidad. Aún así se pueden agrupar de manera amplia y no excluyente de la siguiente forma:

- Métodos inductivos.
- Métodos de descomposición.
- Métodos de reducción.
- Métodos constructivos.
- Métodos de búsqueda local.

De este grupo de métodos heurísticas, centraremos el estudio en el método de *búsqueda local*, ya que esta técnica es la más acertada para resolver el problema de localización de instalaciones y por ser junto a *los métodos constructivos* base de los procedimientos metaheurísticos.

La calidad de una heurística se verá reflejado primeramente si su resultado es ***bueno***, segundo si cumple un resultado ***eficiente*** y finalmente si el algoritmo es ***robusto***. Los procedimientos utilizados para medir su calidad serán:

- Comparación con una solución óptima.
- Comparación con una cota.
- Comparación con otras heurísticas.

2.3.1 Métodos Constructivos.

Consisten en construir literalmente paso a paso una solución del problema. Usualmente son métodos deterministas y suelen estar basados en la mejor elección en cada iteración. Estos métodos han sido muy utilizados en problemas clásicos como el del viajante. Un ejemplo son las heurísticas de construcción voraz o Greedy [Alonso, 2005].

Su gran ventaja es su velocidad, generalmente dan una solución rápida y de buenos resultados, sin embargo, no puede garantizarse que dichas soluciones sean óptimas con respecto a pequeños cambios a nivel local. En consecuencia, una mejora típica es refinar la solución obtenida por la heurística voraz utilizando una búsqueda local.

2.3.2 Métodos de Búsqueda Local.

El algoritmo de búsqueda local parte con una solución inicial S (generalmente aleatoria), la cual busca iterativamente mejorar con movimiento a soluciones vecinas que posean mejor resultado. Si la solución actual es S y en su vecindad se encuentra una mejor solución (S'), esta reemplazara la solución actual y se continuará la búsqueda a partir de S'. Si no se encuentra una mejor solución en el vecindario el algoritmo termina de buscar quedándose con el óptimo local.

Un ejemplo es lo que ocurre en la figura 2.7, en donde se parte con una solución inicial la cual va modificándose constantemente a medida que encuentra una mejor solución. Este movimiento se detiene cuando no logra mejorar su solución actual. El éxito de este algoritmo depende de 2 factores importantes los cuales son la **solución inicial** y el **tamaño del conjunto solución**, esto implica que dada una misma entrada la solución que se obtendrá será siempre la misma, sin embargo si la solución inicial se modifica es muy probable que el resultado final sea diferente.

Existe el inconveniente de que el algoritmo puede estancarse en un óptimo local indeseado, el cual dará un resultado lejano al óptimo global, una de las principales razones puede ser que la solución inicial se encuentra muy lejos del óptimo global, lo que hace imposible acceder a su valor. Este tipo de inconveniente es posible solucionarlo a través de las metaheuristicas, las cuales entregan una mejora sustentable a la heurística original.

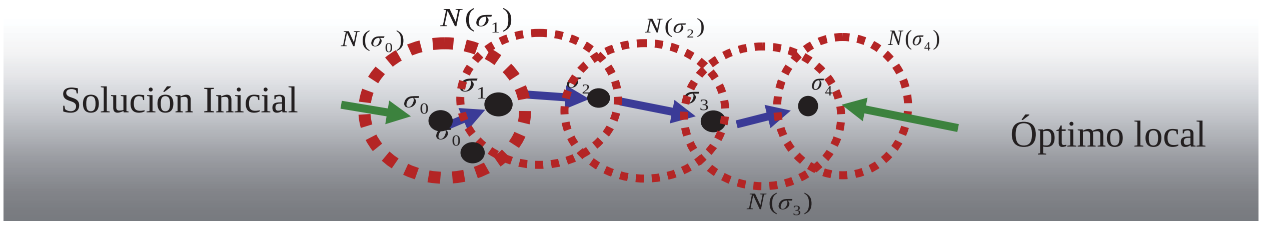

Figura 2.7 Óptimo Local

2.3.3 Metaheurística.

Este método, también llamado en algunos libros como heurísticas modernas fue introducido en la comunidad científica por Fred Glover en 1986. Esta técnica nace para dar mejores soluciones encontradas en las heurísticas tradicionales, esto debido a que las metaheuristicas guían el diseño de las heurísticas. Una definición clara es la que entregan los profesores Osman y Nelly [Osman, 1996]:

"Los procedimientos Metaheurísticos son una clase de métodos aproximados que están diseñados para resolver problemas difíciles de optimización combinatoria, en los que los heurísticos clásicos no son efectivos. Los Metaheurísticos proporcionan un marco general para crear nuevos algoritmos híbridos combinando diferentes conceptos derivados de la inteligencia artificial, la evolución biológica y los mecanismos estadísticos."

Tomando en cuenta que el número de soluciones que es posible encontrar es en algunos casos demasiado grandes, en ocasiones las metaheuristicas son diseñadas para que puedan ser interrumpidas después de un tiempo especificado por el usuario. Si no son detenidas en ocasiones podrán examinar todos los candidatos posibles. Las metaheuristicas resuelven problemas más generales que las heurísticas, y en algunas ocasiones son menos eficientes que estas, esto debido a que las heurísticas están creadas para resolver problemas de orden más específicos.

Las metaheuristicas incorporan conceptos de muchos y diversos campos como son la genética, la biología, la inteligencia artificial, las matemáticas, la física y la neurología, entre otras [Alonso, 2005]. Algunos ejemplos de metaheuristicas son: ***Enfriamiento simulado***, ***búsqueda tabú***, ***búsqueda local iterativa*** (*"iterated localsearch"*), ***algoritmos de búsqueda local con vecindario variable*** (*"variable neighborhood search"*), **GRASP** (*"greedy randomized adaptative search procedures"*), ***algoritmos evolutivos, algoritmos genéticos***, ***Optimización basada en Colonias de Hormigas*** (OCH), etc.

2.3.4 Búsqueda Tabú (TS).

Tomando en consideración las diferentes investigaciones que han utilizado la Metaheurística Tabu Search para los problemas de localización de instalaciones, como son el caso de V. Filho y R. Galvao [Filho, 1999] y Laurent Michel, Pascal Van Hentenryck [Michel, 2002] y considerando los buenos resultados obtenidos en los problemas de optimización duro [Glover, 1998], se ha decidido optar por ésta estrategia como base para la solución del problema en cuestión.

La búsqueda Tabu (Tabu Search), es una Metaheurística diseñada para cruzar cotas de factibilidad y poder explotar regiones no consideradas en otros casos. Su característica principal es el uso de memoria adaptativas y estrategias especiales para la resolución de problemas. TS es también responsable de enfatizar el uso de los diseños estructurados para explotar los patrones históricos de la búsqueda, de forma opuesta a los procesos que confían casi exclusivamente en procesos aleatorios [Glover, 1998].

Esta Metaheurística combina la búsqueda local con una heurística para evitar mínimos locales, esto se puede observar al analizar el siguiente algoritmo de búsqueda local:

1. Seleccionar una $x \in X$ inicial.
2. Seleccionar algún $s \in N(x)$ (vecino) tal que: $c(s) < c(x)$
3. *Si* no existe, x es un óptimo local y para.
4. *Si no*, sea $x \leftarrow s$ y regresa al punto 2.

En la búsqueda local existe el problema de quedar muchas zonas muertas imposible de explorar debido a que no existen vecinos con una mejor condición (generalmente costos). Por lo tanto el óptimo local queda muy distante del óptimo global. Es por esta razón que la búsqueda tabú busca mejorar éste error continuando la exploración de soluciones aunque su resultado no mejore con los cambios. Con el fin de evitar ciclos infinitos esta Metaheurística maneja una *memoria temporal* (lista tabú) de los lugares históricos ya explorados, de esta forma se evita seguir realizando búsquedas en lugares examinados. Por esta razón es que generalmente es llamada esta Metaheurística como una búsqueda local con memoria de corto plazo.

Se pueden encontrar caso en donde la lista tabú prohíba movimientos deseables que no produzcan ciclos, o también llevarnos a puntos en donde no sea posible moverse. Es por esta razón que el algoritmo de búsqueda tabú permite cancelar algunos movimientos considerados como tabú. A estos se les llama *criterio de aspiración*.

Dos criterios importantes utilizados en esta Metaheurística son los relacionados con la *intensificación* y la *diversificación*. La primera consiste en realizar exploraciones más exhaustivas en porciones del espacio que se consideran más importantes o prometedoras, este proceso se puede realizar cada cierto tiempo, la diversificación en cambio consiste en realizar búsqueda en espacios no explorados hasta el momento.

Los criterios utilizados para detener la búsqueda tabu son variados, dependen tanto del problema planteado como de los impuestos por el diseñador. Dentro de los más importantes se pueden mencionar; la cantidad de iteraciones, tiempo máximo del CPU, alcanzar una solución i que sea mejor que un cierto valor fijado al inicio, no obtener una nueva mejor solución i* luego de una cierta cantidad de iteraciones, todos los vecinos del paso actual están incluidos en la lista tabú.

Finalmente se puede concluir que el éxito de esta estrategia va a depender del modelado del problema, del largo de la lista que se va utilizar, de la manera como se define la vecindad y la función objetivo, del criterio al elegir una solución inicial i que sea capaz de llegar al óptimo *i.

Un algoritmo general de la Metaheurística de búsqueda tabú se puede examinar en la figura 2.8.

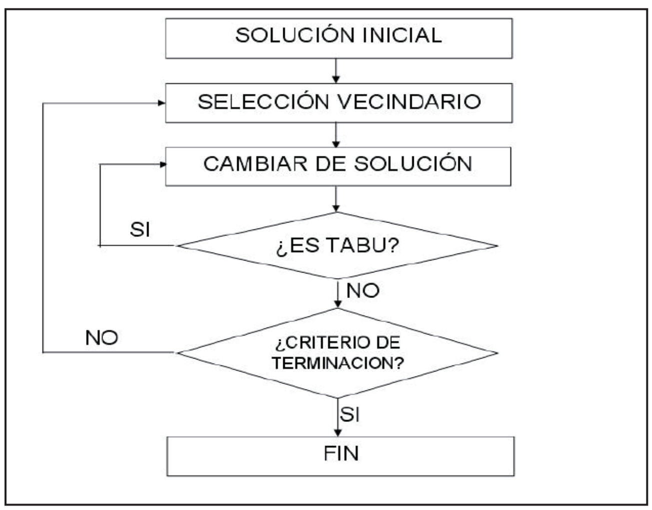

Figura 2.8 Algoritmo Búsqueda Tabu

CAPITULO 3 SOLUCIÓN PROPUESTA

3.1 Modelo Preliminar.

Luego de analizar los diferentes modelos de localización existentes en la literatura [Daskin, 1995], los modelos de inventarios [Zipkin, 2000] y tomando en consideración la importancia que significan los costos de inventarios en los problemas reales [Ferhat, 2005], en donde según estudios los costos de inventarios llegan al 15% de las ventas totales, se hace necesario considerar integrar los costos de inventarios al modelo final dado su importancia en los costos totales.

El modelo que se busca diseñar en esta investigación está bajo la primera restricción de cubrir la totalidad de la demanda. Esta condición hace necesario tener una cantidad ilimitada del número de instalaciones que se desea localizar, ya que si nos guiamos bajo el modelo de máxima cobertura, (ver capitulo 2.1.2) la cantidad P que se asigne puede ser insuficiente para abastecer la totalidad de la demanda.

La segunda restricción de nuestro modelo es considerar una demanda determinística (μ_i), conocida y diferenciada en cada uno de los clientes. Con esta demanda conocida es posible manejar de manera cierta el stock del inventario en todo momento, con lo cual es posible determinar de manera el punto de reorden necesario para solicitar a la planta central una nueva cantidad Q para abastecer al cliente con la cantidad requerida .Al manejar una demanda conocida es posible eliminar los costos asociados al stock de seguridad, los cuales son utilizados para protegerse de la incertidumbre de la demanda [Miranda, 2004].

Una tercera restricción propuesta para éste modelo preliminar sugiere que la distancia máxima que será posible abastecer desde un centro de distribución hasta los clientes sea asignada por parte del usuario. Esta condición es posible observarse en el capítulo 2.1.2.

Una última condición para este modelo insta a que todo cliente sea abastecido tan solo por un centro de distribución, esta restricción es posible obtenerse a través del siguiente modelo matemático formalizado en el capitulo 2.1.5:

$$\sum_j Y_{ij} = 1 \qquad \forall_i$$

Con respecto al modelo de localización, se ha determinado que el modelo que más se aplica al problema en desarrollo es el que plantea el problema de localización con carga fija sin capacidad [Ozen, 2004]. A través de éste modelo es factible poder integrar de manera adecuada los costos de inventario descrito en el planteamiento de la investigación.

Finalmente el modelo preliminar se pude definir como:

Entrada

f_j = Costo fijo de localizar un candidato en el sitio j.

μ_i = Demanda del nodo i.

d_{ij} = Distancia desde la demanda del nodo i al candidato j de localizar.

a_i = Distancia desde la planta central al centro de distribución i.

c_i = costo por unidad transportada.

k_i =costo fijo de ordenamiento.

Q_i = Cantidad del lote i.

Variables de decisión.

X_j = 1 si localizamos en el sitio j al candidato.

X_j = 0 en caso contrario.

Y_{ij} = 1, si el cliente es abastecido por el CD.

Y_{ij} = 0, en caso contrario.

Mínimo $$\sum_j fj * Xj + \sum_i \sum_j \mu_i * Y_{ij} * (d_{ij} + a_j) + \sum_i \sum_j \mu_i * Y_{ij} * (c_i + k_i / Q_i)_j \qquad \textbf{(25)}$$

En donde:

$$\sum_j Y_{ij} = 1 \qquad \forall_i$$

$$Y_{ij} \leq X_j \qquad \forall_{i,j}$$

$$x_j = 0,1 \qquad \forall i, j$$

$$Y_{ij} \geq 0 \qquad \forall i, j$$

De esta forma es posible finalmente desprender en el primer componente los costos fijos de instalación, el segundo componente los costos asociados al transporte y por último el tercer componente los costos de inventario. Las explicación de las restricciones pueden observarse en el capitulo 2.1.4

Es importante destacar que para nuestro modelo (25), solo estamos considerando una demanda media (no estocástica). Esta es la razón por la cual no será utilizado los inventarios de seguridad, el cual es usado frecuentemente en modelos de otros autores.

3.2 Aplicación de Búsqueda Tabu al problema.

3.2.1 Estructura de Datos.

Tomando en consideración la Metaheurística seleccionada para resolver el problema en cuestión y utilizando la experiencia de la bibliografía existente se ocuparan las siguientes estructuras de datos para los datos de entradas y de salida y procesamiento.

3.2.2 Datos de Entrada.

Los datos de entrada serán simulados con el fin de poder observar el comportamiento del modelo tomando en cuenta variados casos. Estos datos serán ingresado a través de un archivo de texto los cuales se considerará para evaluar el modelo preliminar (formula 25).Los datos de entrada serán los siguientes:

$X[n]$= Este vector va servir como un arreglo de variables de decisiones, el cual entregará la salida final del problema.

$f[n]$ = Es un vector de largo n, correspondiente a los costos fijos de instalación de cada uno de los centros de distribución.

$\mu[m]$ = Corresponde a un vector de largo m (número de clientes del modelo) el cual indicará la demanda que necesita ser cubierta por cada cliente por un periodo de 1 mes.

$d[n][m]$ = Es una matriz que indica la distancia existente entre cada cliente y todas las opciones de centros de distribución. Es importante destacar que para este modelo en particular va existir una restricción de máxima cobertura la cual eliminará por lo tanto las distancias que superen dicho valor establecido en el futuro.

$a[n]$ = Es un vector de largo n, correspondiente a la distancia que existe entre cada centro de distribución y la planta central.

$c[n]$ = vector de largo n, correspondiente al costo por unidad transportada desde cada centro de distribución.

$k[n]$ =Vector de largo n que indica los costos fijos de ordenamiento de cada centro de distribución.

$Q[n]$ = Vector que maneja en sus datos el tamaño del lote que solicitará cada centro de distribución para abastecer a cada cliente.

Este conjunto de datos tienen como objetivo ser utilizado como valor de entrada para calcular el costo total obtenido a través del modelo preliminar propuesto en el capitulo anterior (formula 25)

3.2.3 Datos de Procesamiento.

Durante la ejecución del programa se estarán utilizando una Matriz Binaria $X[n][4]$ la cual irá registrando los movimientos que se encuentran en una condición Tabú. El periodo tabú dependerá del valor N, el cual indicará el tiempo necesario que se necesita mantener un movimiento en espera. Esta condición sin embargo es posible ser modificada por causa el criterio de aspiración.

3.2.4 Datos de Salida.

Los resultados finales que se van a obtener de la investigación será un vector de un largo n dependiente del número de posibles lugares existentes para ubicar los Centros de Distribución. Cada valor X_j va ser asignado a una posición en particular del conjunto de opciones entregadas. De esta forma se puede declarar la estructura de salida como:

$$X[n] = [X1, X2, X3, X4, ..., Xn] \qquad \textbf{(26)}$$

Este vector $X[n]$ como fue declarado anteriormente es una variable de decisión que posee valores binarios en donde sus datos determinarán:

$$X_j = \begin{cases} 1 \quad si \quad localizamos \quad en \quad el \quad sitio \quad j \quad al \quad candidato \\ 0 \quad en \quad caso \quad contrario \end{cases}$$

En conclusión el vector final entregará datos con valor 1 en las ubicaciones en donde es más factible instalar el centro de distribución.

Una manera de graficar esta situación es la que ocurre en la figura 3.1 en donde se han seleccionado 3 Centros de Distribución de un grupo de 5 lugares posibles. De esta forma el vector debería tener los siguientes datos de salida:

1	0	1	0	1

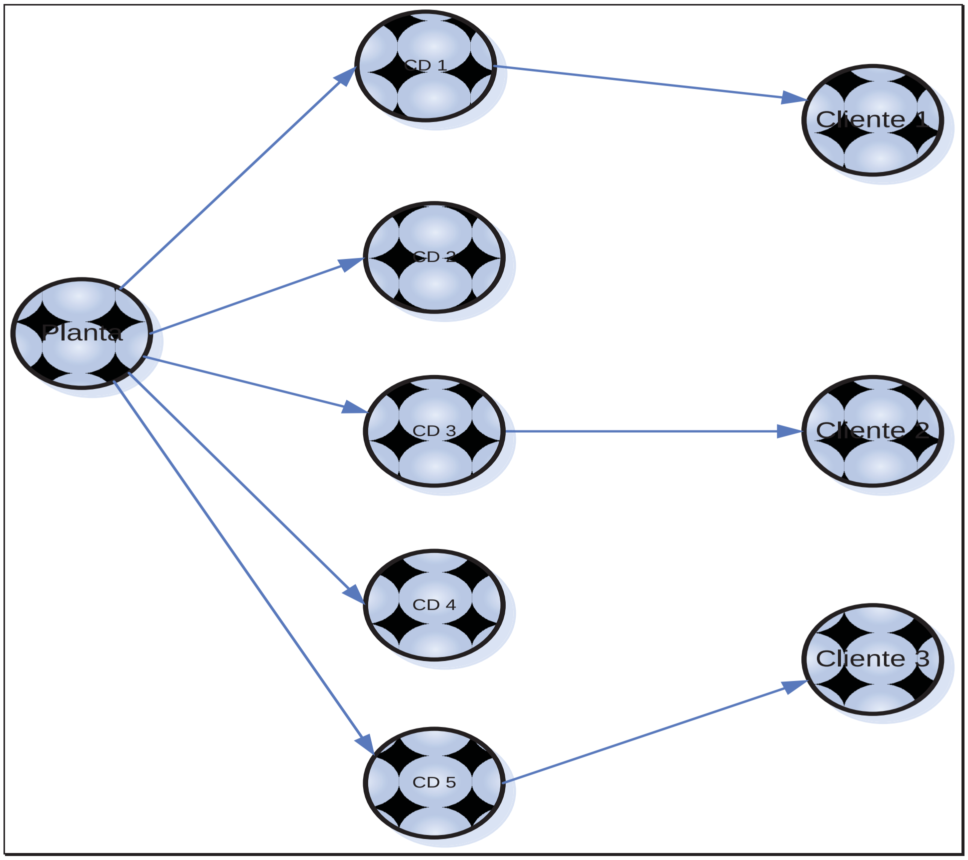

Figura 3.1 Datos de Salida

3.2.5 Vecindad.

La forma de representar la vecindad es de una manera simple, esta consiste en realizar un cambio de la condición de un Centro de Distribución $X[n] = [X1, X2, X3, X4, ..., Xn]$. Este cambio, también llamado movimiento es desde un estado *abierto a cerrado* o viceversa. La vecindad V es definida como:

$$V(X) = \{Movimiento(X, w) \quad / w \in N\}$$

En donde

$$Movimiento(\langle x_1, x_2, x_3, \ldots x_n \rangle, w) = \langle x_1, x_2, x_3, x_{w-1}, !x_w, x_{w+1}, x_n \rangle \quad \textbf{27)}$$

El procedimiento planteado lo que realiza es realizar sobre el vector $X[n]$ un intercambio de valor en la posición w. De esta manera si el valor de $X[w]$ es igual a 1 (se encuentra abierto) se modificará a cero, de igual forma en caso contrario.

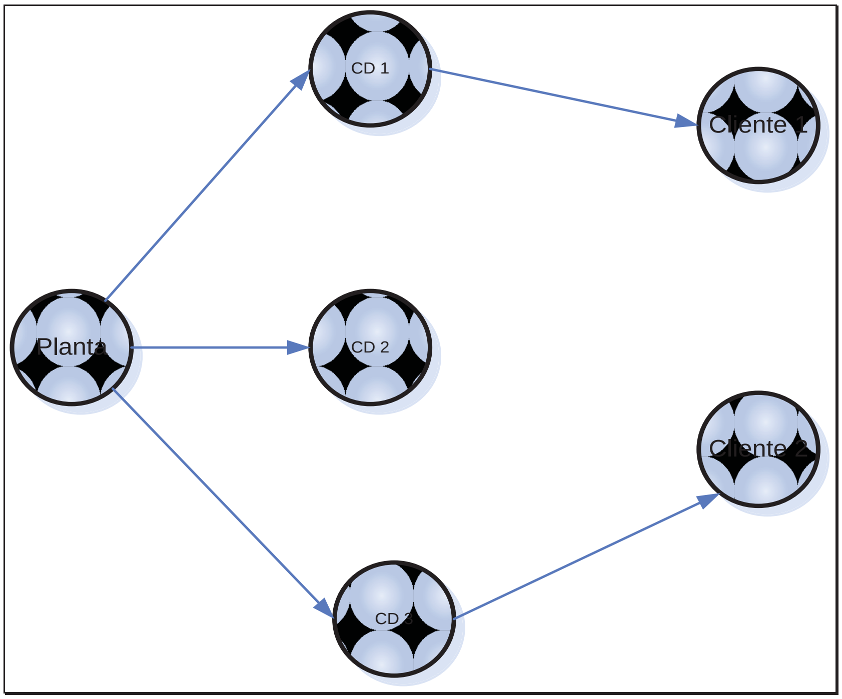

Figura 3.2 Vecindad

Un ejemplo de este procedimiento es lo que ocurre en la figura 3.2, en este caso se encuentran abiertos el CD1 y el CD2, los cuales abastecen a los 2 clientes. Esta situación se puede plantear de la siguiente forma:

1	0	1

Un movimiento factible en este caso sería abrir el CD2, el cual apoyaría el abastecimiento de uno de los 2 clientes. Es importante destacar que para esta situación en particular este sería el único movimiento posible, ya que al cerrar uno de los 2 CD se dejaría de cumplir una restricción del modelo, el cual indica que todos los clientes tienes que ser cubierto en todo momento.

Por lo tanto el movimiento consistente en abrir el segundo CD se puede ver gráficamente a través de la figura 3.3 y vectorialmente se plantearía como:

1	1	1

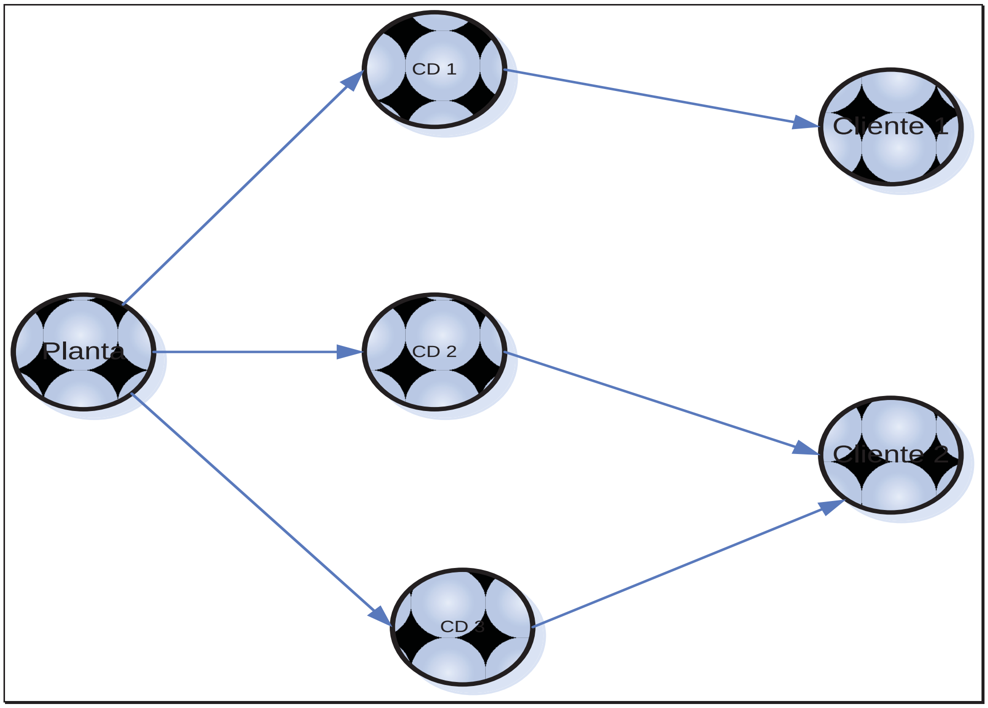

Figura 3.3 Movimiento

3.2.6 Función Objetivo.

El fin último del desarrollo de esta investigación es encontrar una combinación de condiciones que minimice los costos totales del diseño de la red de distribución, es por esta razón que lo que se buscará en cada iteración del algoritmo con base a los datos de entrada, los costos mínimos del modelo preliminar. Es por esta razón por lo tanto que se fijará la función objetivo como:

$$\text{Objetivo (X)} = \sum_j fj * Xj + \sum_i \sum_j \mu_i * Y_{ij} * (d_{ij} + a_j) + \sum_i \sum_j \mu_i * Y_{ij} * (c_i + k_i / Q_i)_j$$

3.2.7 Solución Inicial.

La solución inicial en un proceso de búsqueda como el Tabu Search es un punto de vital importancia para el resultado que se pueda encontrar. Diferentes estrategias de inicio guiarán hacia diferentes resultados, es por este motivo que iniciar el proceso de iteración con un resultado cercano al optimo es lo más oportuno.

Tomando en consideración que la solución inicial, al igual que todos los resultados siguientes deben cumplir con las restricciones establecidas en el modelo preliminar (capitulo 3.1) nos acota las posibles opciones, es por esta razón que se ha concluido que los candidatos factibles serán los siguientes 3, los cuales se evaluarán en el proceso de implementación:

La primera opciones, usada constantemente en la aplicación de esta meta heurística es partir con una solución inicial con *datos aleatorios* (Random). Esta estrategia tiene el inconveniente de que el vector que entregue el random no cumpla con las condiciones básicas establecidas en el modelo preliminar (distancia máxima de cobertura de un CD), lo que hará necesario repetir aleatoriamente la solución inicial hasta encontrar con un vector permitido.

La segunda opción consiste en partir con todos los Centros de Distribución abiertos, es decir el vector $X[n]$ solo con valores igual a 1. De esta manera se parte asegurando que la opción es factible, pero sin duda la más costosa dentro de las posibles.

Finalmente la última opción aprovechando que los datos de entradas que se darán al inicio del programa están incluido las distancias existente entre cada cliente y lo CD ($d[n][m]$) se buscará por cada cliente cual es su centro de Distribución más cercano, dejándolo este como abierto. Esta estrategia va dar una solución inicial con unos costos totales menores a la alternativa anterior y dará muy buenos resultados si los costos de traslado desde la planta a los CD no son muy altos respecto al costo de traslado desde el CD a los clientes.

Una buena estrategia de solución inicial también traerá como beneficio un menor tiempo de procesamiento, lo que hace aun más importante su elección, situación que será resuelta en el momento de su implementación.

3.2.8 Memoria Tabú.

La característica principal de la Metaheurística Tabu Search, es el uso de una memoria la cual permita solucionar el problema de zonas muertas encontradas en la búsqueda local. Esta memoria almacenará información de aquellos sectores que ya han sido visitados, lo cual solucionará el problema existente de concentrar la exploración solo en unas determinadas zonas fijas. De esta forma expandimos la búsqueda a nuevas zonas que antes no habían sido cubiertas.

La manera de utilizar la memoria depende exclusivamente del problema y la estrategia que se aplicará para su solución. Por lo general se utiliza una estructura de dato capaz de soportar la cantidad de datos necesaria y por una cantidad de tiempo que sea apropiada para el buen desempeño del algoritmo.

En el caso particular de nuestro problema se utilizará una matriz $X[n][4]$, donde n es igual al tiempo que se mantendrá tabú un movimiento. Esta matriz tendrá como nombre: $Tabu[n][4]$

Este vector manejará en su contenido valores Reales enteros, el cual será inicializado en cero. Cuando un CD cambia de disposición (de abierto a cerrado o viceversa) el valor correspondiente a su posición se le asignará como tabú por un periodo de tiempo definido por el usuario, de esta manera a medida que se va realizando una nueva iteración su valor se decremento en 1 hasta llegar a cero, con lo cual deja de ser una condición tabú, por lo cual se permite volver a cambiar de estado.

A medida que se realizan nuevas iteraciones los valores que no son afectados también disminuyen su valor en 1, inclusive llegando a tomar valores negativos, de esta forma es posible determinar cuáles son aquellos CD's que no ha sido modificado su estado por periodos de tiempo más prolongado. Esta condición hace más fácil realizar procesos de nuevas búsquedas en sectores nunca antes explorados.

No existe un criterio exacto para determinar la cantidad de tiempo óptimo que es preciso mantener tabú un estado en particular, sin embargo el número 7 lo han tomado varios autores para diferentes casos. Más recientemente, se toman valores dependientes del tamaño del problema. En cualquier caso, constituye un parámetro importante cuya influencia habría que analizar y del cual dependerá la evolución del algoritmo en gran medida.

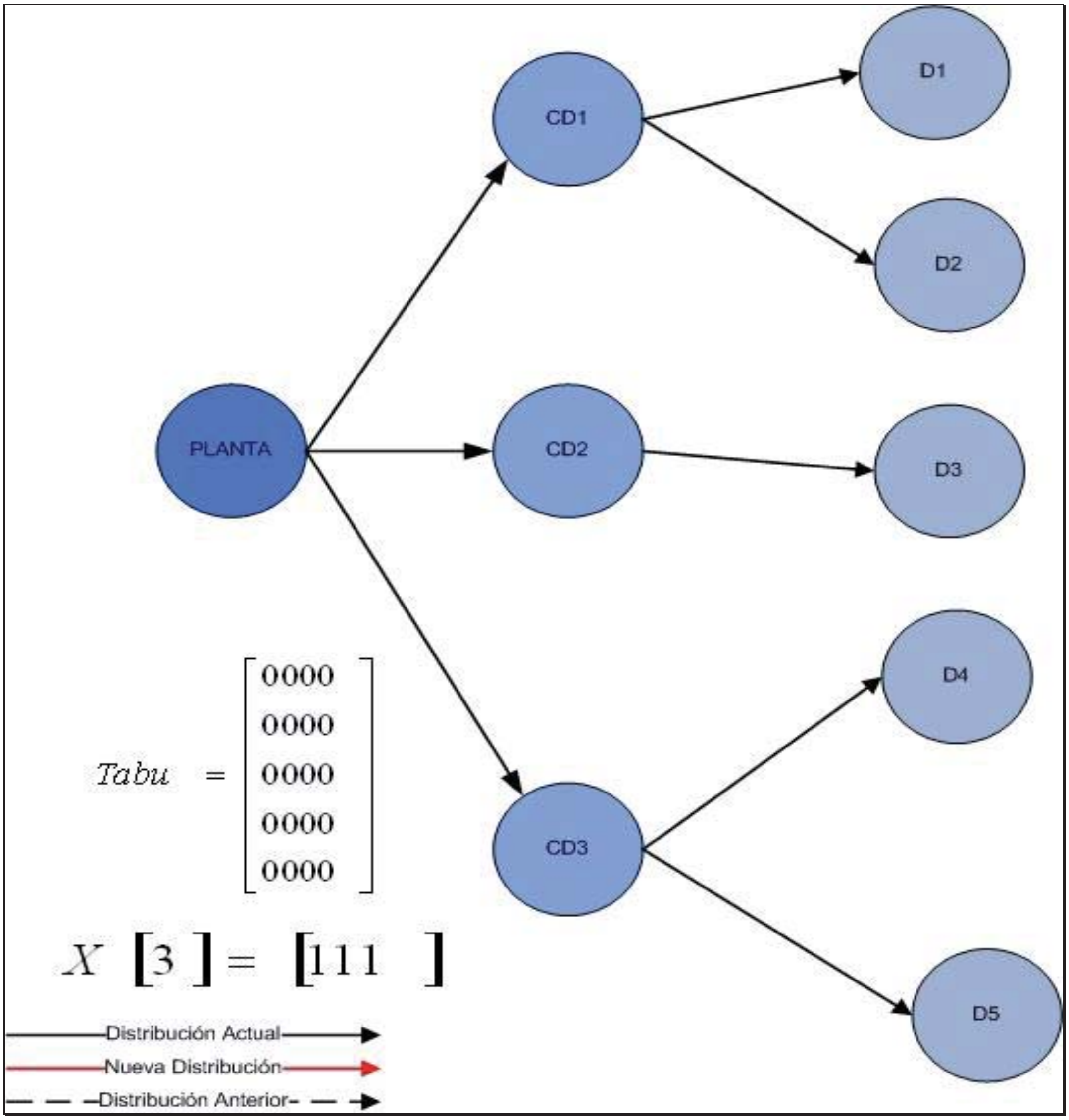

Figura 3.4 Memoria Tabu

En la figura 3.4 se puede observar la situación inicial de un DRD, en donde 3 Centros de Distribución abastecen a los 5 clientes. Esta situación inicial muestra una matriz tabú en cero, esto significa que no existen movimientos restrictivos para futuros cambios. El número de filas para nuestro ejemplo nos indica que serán 5 los momentos que un movimiento se mantendrá como Tabu.

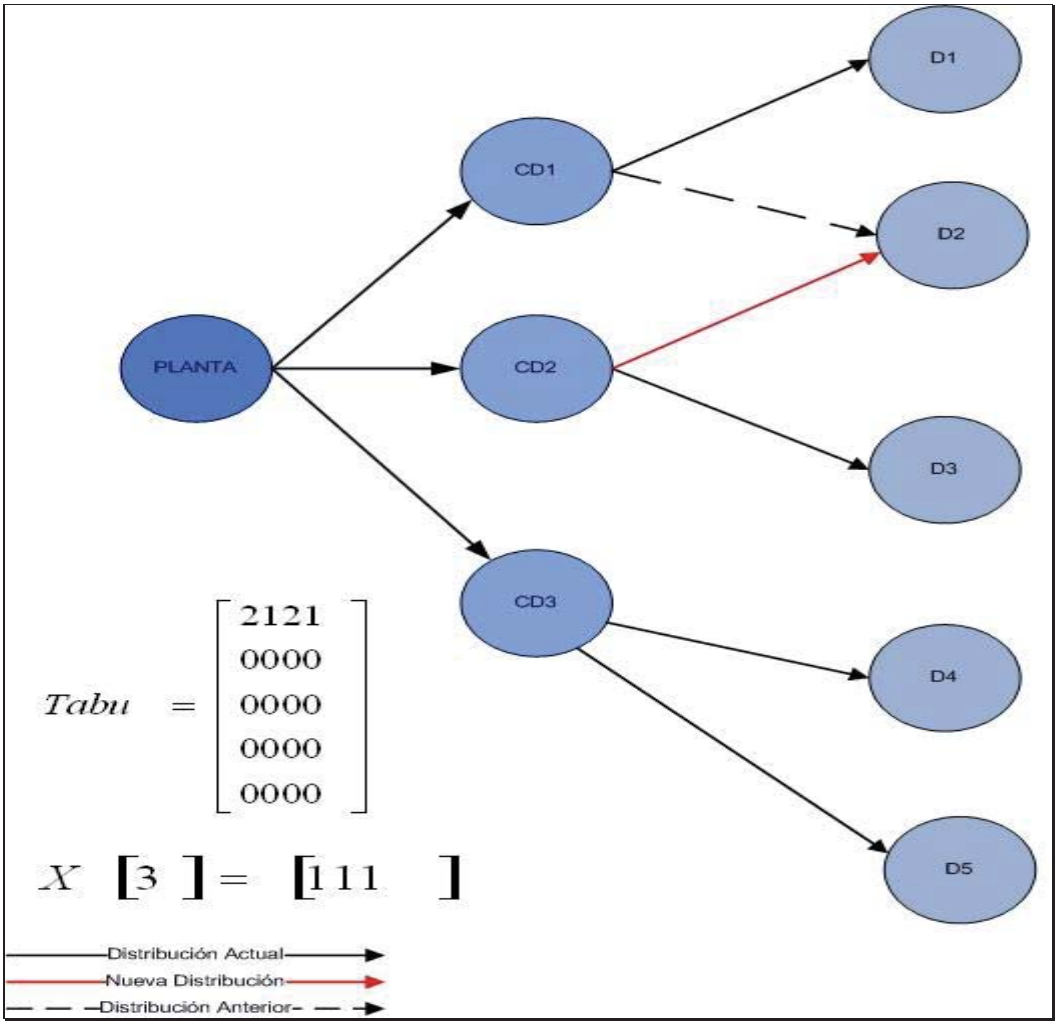

Figura 3.5 Memoria Tabu

En la figura 3.5 podemos verificar los efectos que suceden al ejecutarse el primer movimiento. En esta ocasión el Cliente 2 dejó de abastecerse del CD 1 y lo hace a través del CD2 . La primera fila refleja el movimiento efectuado a través de sus cambios, en donde el primer valor de la fila (2) nos indica que el cliente 2 ha efectuado un cambio**,** la información de la segunda columna (1) nos indica que originalmente el cliente 2 era abastecido por el CD 1, la tercera columna (2) nos indica cual es nuevo CD que abastece al cliente y finalmente la última columna nos indica el tiempo que ese movimiento se encuentra tabú. Para nuestro caso es 1, debido a que es el movimiento más reciente.

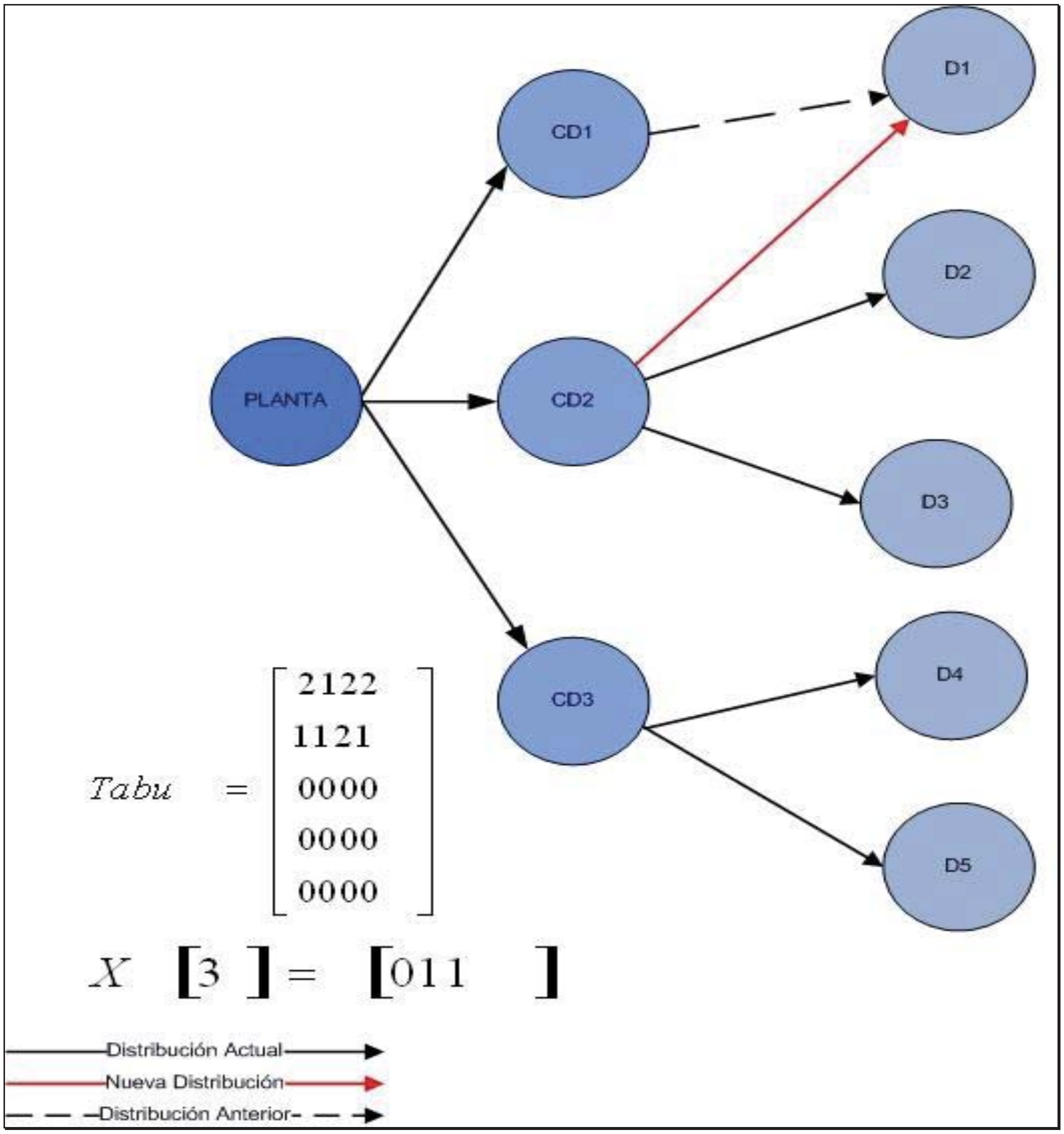

Figura 3.6 Memoria Tabu

El segundo movimiento refleja el efecto que sucede en la matriz al efectuarse un cambio en los Centros de Distribución. El movimiento tabú que estaba registrado en la primera fila modifica la última columna a 2, esto significa que ese movimiento en particular lleva dos tiempos tabú, su valor irá incrementando cada vez que se realice un movimiento hasta que se efectúe el quinto cambio en donde dejará de ser un movimiento restrictivo.

Al igual que en el caso anterior el primer valor de la fila (1) nos indica que el cliente 1 se verá afectado por un cambio, el segundo valor (1) nos indica que originalmente era abastecido por el CD 1, el tercer valor (2) indica que será cubierto por el CD2 y finalmente el último valor (1) nos indica que se encuentra en su primer movimiento Tabu.

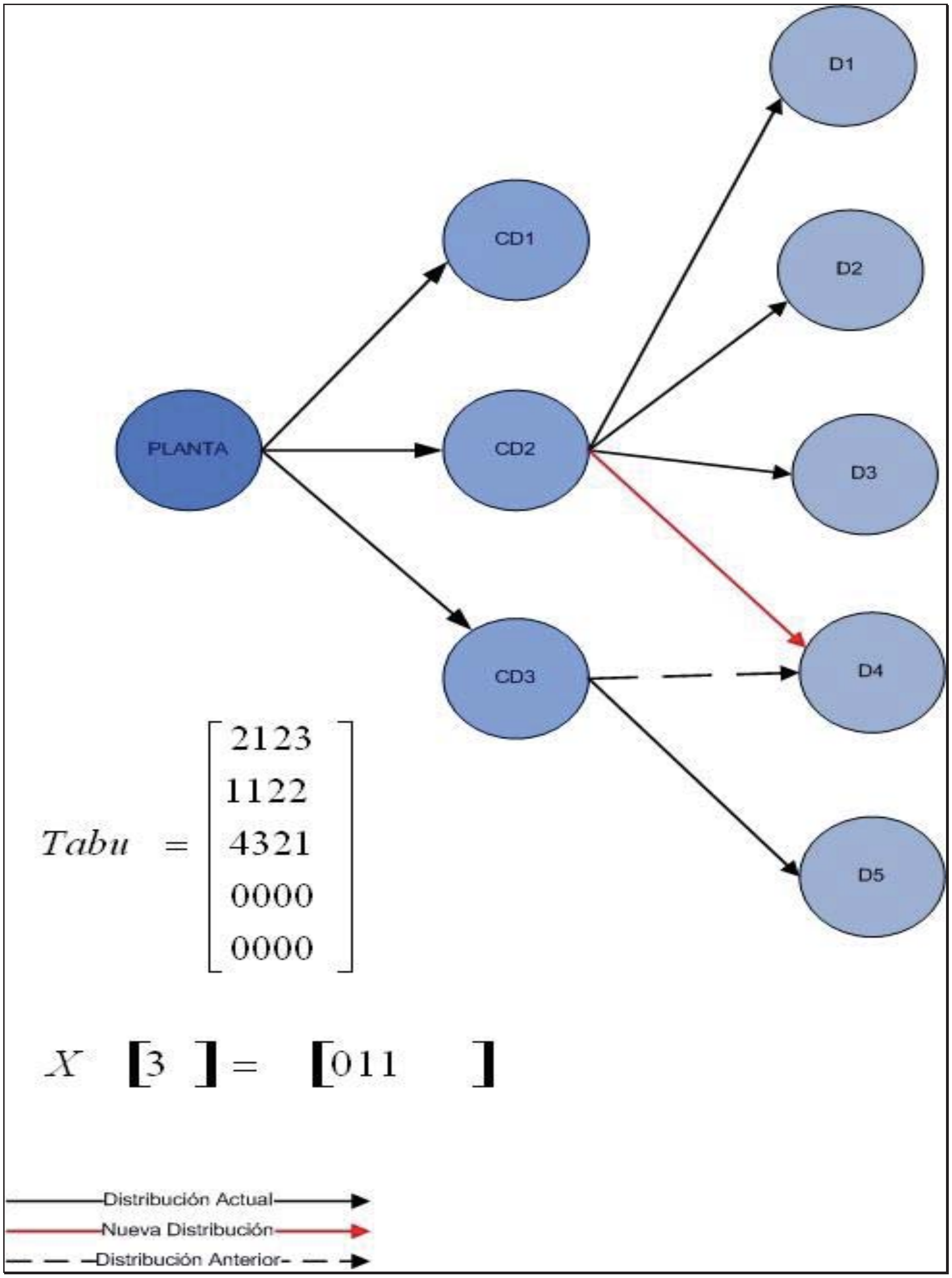

Figura 3.7 Memoria Tabu

Finalmente a través del movimiento reflejado en la figura 3.7es posible apreciar como el valor correspondiente al tiempo tabú (última columna) se incrementa en los movimientos antiguos y es igual a uno en el nuevo movimiento.

3.2.9 Criterio de Parada.

La característica principal de la Metaheurística Tabu Search es la de poder continuar una búsqueda constante de manera infinita, incluso cuando los resultados no se están mejorando. Es por esta razón que se hace necesario crear un criterio que determine la política que se debe efectuar para detener la búsqueda.

La manera de predecir un acertado criterio de parada es tan compleja como buscar una solución inicial, es por esta razón que solo se darán algunos posibles criterios que serán evaluados en el momento de la implementación. De esta forma el criterio elegido para detener la búsqueda es poseer una cota superior de iteraciones posibles, esta cota N, puede ser una cantidad de iteraciones en donde el algoritmo no ha podido mejorar los resultados obtenidos anteriormente. Esta cantidad N será definido en etapas superiores, ya que solo en el proceso de pruebas de los resultados se podrá determinar la efectividad del criterio.

3.2.10 Criterio de Aspiración.

Todo algoritmo permite revocar la condición Tabu de un movimiento, a esto se le llama *criterio de aspiración*. Para el problema de DRD se va utilizar este criterio cuando un movimiento tabu genere un resultado con los datos de entrada con mejor resultado (menor costo) que el mejor caso guardado anteriormente. Esto puede observarse en la figura 3.8, perteneciente a un segmento del algoritmo preliminar de la figura 4.10.

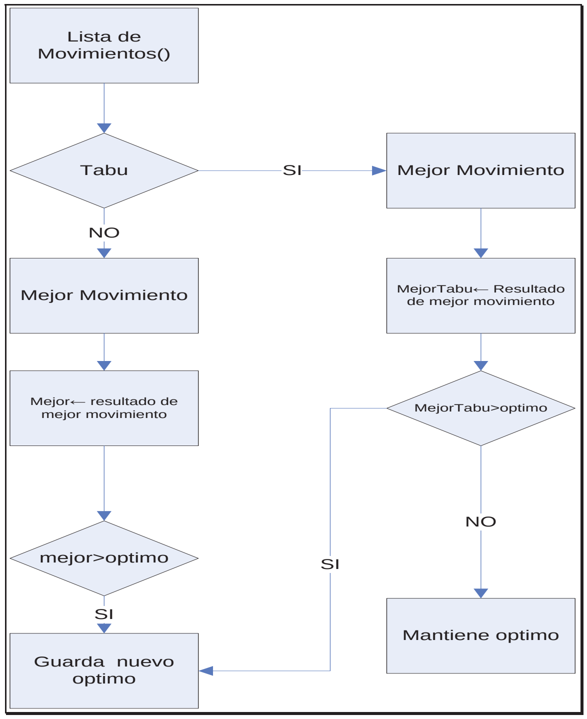

Figura 3.8 Criterio de Aspiración

Luego de obtenerse la lista de movimientos posibles, esto quiere decir la lista de movimiento que cumplen con los requisitos impuestos en el modelo preliminar (ver capitulo 5), se dividen los movimientos que se encuentran como tabú y permitidos. Para la utilización del criterio de aspiración se consideran del conjunto de movimientos tabú, de este conjunto se busca el que nos entregue el mejor resultado (menor costo), si esta combinación disminuye el menor costo almacenado históricamente se aplicará el criterio de aspiración a este movimiento. Por lo tanto se realizará el movimiento que anteriormente no estaba permitido.

3.2.11 Intensificación y Diversificación.

Con el fin de ir mejorando las soluciones posibles es posible aplicar algunas estrategias que generen una respuesta cada vez más óptima. La *intensificación* es un caso el cual consiste en realizar búsqueda en porciones de espacio que aparentan ser mejores, una alternativa altamente utilizada es re-iniciar la búsqueda cada un cierto tiempo desde una solución con buenos resultados.

El proceso de diversificación consiste en realizar exploraciones en combinaciones antes no realizada. Para este problema en particular se puede utilizar la memoria tabú, la cual va manejar de manera paralela la información de aquellos movimientos que son tabú y aquellos CD's que se han mantenido constante durante más tiempo.

1	0	**-5**	-3	1	1	-2	**-5**	**3**	0	0	-1

Figura 3.9 Intensificación y Diversificación

Como se puede observar en la figura 3.9, la memoria tabu maneja dos informaciones relevantes. La primera información relacionada con todos valores **positivos** incluidos el cero, indican todos aquellos movimientos que se encuentran restringidos sus movimiento por la cantidad de iteraciones que indica su valor, para este caso en particular el noveno CD será el que se mantendrá por mayor cantidad de iteraciones impedido de modificarse (durante 3 iteraciones). La segunda información tiene que ver con los valores **negativos**, estos datos indicarán cuales son las posiciones que se han mantenido sin movimiento por mayor cantidad de tiempo. En la figura 3.9 se puede notar que existen dos posiciones con valor -5 que se han mantenido por más largo periodo de tiempo sin ser modificado, esta información es de vital importancia para aplicar el criterio de diversificación en este problema.

3.2.12 Algoritmo Final.

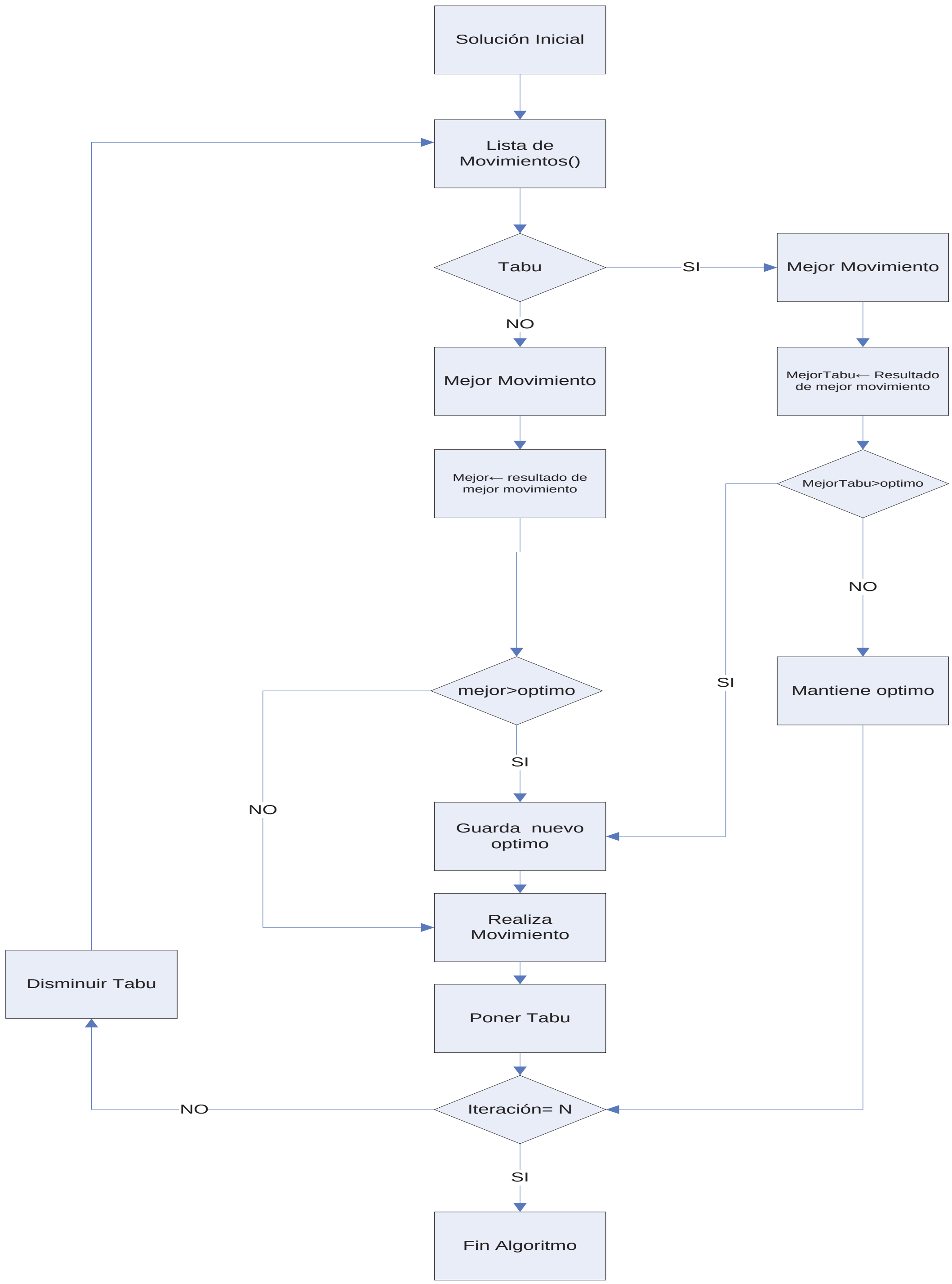

Figura 3.10 Algoritmo Final

3.3 Implementación del Algoritmo.

La implementación del algoritmo propuesto en el capitulo anterior fue desarrollado a través del *lenguaje de programación C*. La ejecución del programa fue creado a través de un desarrollo estructurado, es importante destacar que el código se encuentra en un proceso inicial, el cual necesita mayores depuraciones en las siguientes versiones.

3.3.1 Solución Inicial.

La solución inicial genera una matriz de (CD*CL), en donde CD es el número de centros de Distribución y CL el número de Clientes. Estos dos valores irán modificando su valor para futuras pruebas y comprobar el comportamiento del programa bajo diferentes condiciones.

La matriz inicial tendrá valores binarios, las cuales indicarán cuando un cliente es cubierto por un Centro de Distribución. Esta matriz inicial necesita cumplir las siguientes condiciones para que sea aceptada como correcta:

La sumatoria de todas las filas tiene que ser igual a 1. De esta forma se cumple que todos los clientes son cubiertos por un CD. ($\sum_j Y_{ij} = 1 \quad \forall_i$)

Cada Centro de Distribución posee una capacidad de Distribución, la cual no puede ser superada por la suma de los clientes que han sido asignados aleatoriamente. Si la capacidad no permite cubrir un cliente, se buscará una segunda opción que satisfaga las condiciones impuestas.

Cada cliente será cubierto inicialmente por un CD de manera aleatoria.

Algoritmo Matriz Inicial

```
int solInicial(struct datosEntrada entrada[CL][CD],struct datosInstalacion instalacion[CD] ){  // genera la matriz inicial CL * CD
   int i,r,j;
    Desde(i=0;i<CL;i++){
            r← (rand() % (CD));  // un valor aleatorio entre todos los centros de distribucion
                    Si(capacidad[r]<cliente[r].demanda){/si la capacidad del CD es menor a la demanda de un cliente
                            i←i-1;
                   }Fin
                  Si(capacidad[r]>cliente[r].demanda){// capacidad mayor a la demanda
                            capacidad[r]←capacidad[r]-cliente[i].demanda; // la capacidad se resta con la demanda del cliente
                            entrada[i][r].ent←1;      // el cliente es cubierto por el CD (r)
                  }Fin
    }Hasta
}Hasta
```

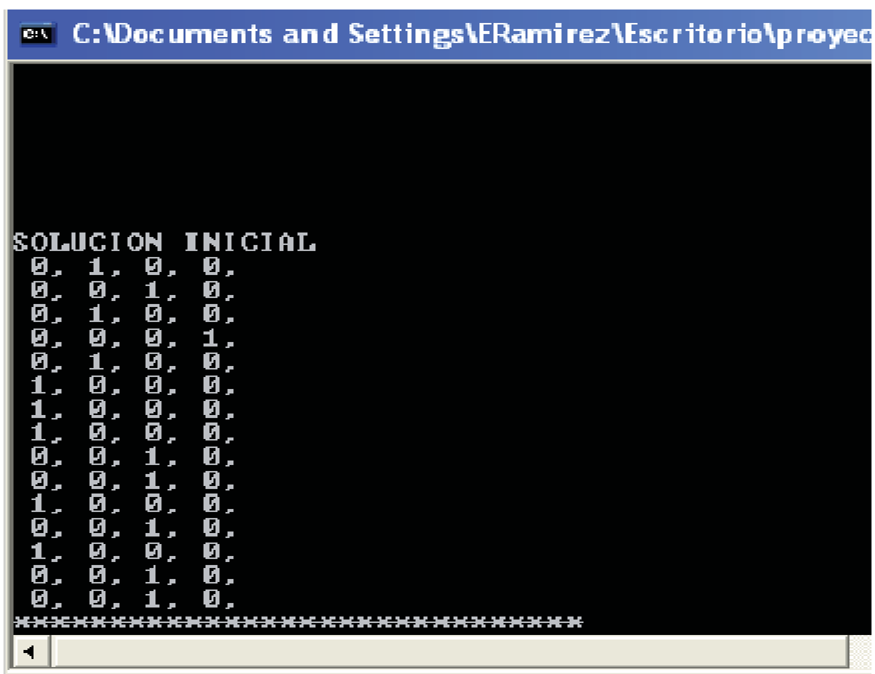

Figura 3.11 Solución Inicial

.

3.3.2 Estructura Tabu.

La estructura tabú que permite solucionar el problema de sectores de áreas muertas está compuesta por una matriz **Tabu [N][4].** Esta matriz funciona de la siguiente manera:

El valor N es el tiempo por el cual un movimiento se encontrará tabú. Este valor optimiza su comportamiento según las referencias cuando está cercano a 7.

- La columna 1 indica la *fila* que se modificó.
- La columna 2 indica la *columna* que se modificó.
- La columna 3 indica la ubicación *donde se modificó* la columna anterior
- Finalmente la columna 4 indica el *tiempo* que un movimiento se ha mantenido tabú. Este movimiento se mantendrá hasta que llegue al valor N.

Figura 3.12 Estructura Tabu

En la figura 3.12 es posible observar el funcionamiento de la matriz Tabu. En este caso existen cuatro movimientos que se encuentran registrados, los cuales no podrán repetirse mientras estén en la lista.

El primer Movimiento registrado es correspondiente a la tercera fila (tercer cliente), la cual originalmente está asignado a la tercera columna (tercer Centro de Distribución) y fue modificado por la primera columna (Primer Centro de Distribución), el cual lleva cuatro iteraciones tabú. En el caso de que el valor N sea igual a 7, le quedarían tres movimientos más de restricción.

De esta forma el movimiento inverso al realizado no podrá ejecutarse mientras se encuentre registrado en la lista tabú.

Algoritmo Tabu

```
Desde (j=0;j<CL;j++){
      Desde (n=0;n<CD;n++){
           Si (entrada[j][n].ent==1){ // busco donde está el 1
                  r1← n;   // queda en r1
           }Fin
      }Hasta
 }Hasta
Desde (i=0;i<CD;i++){
      Si (entrada[j][i].ent==1){
          r← i;
      }Fin
 } Hasta
 Desde (i=0;i<CD;i++){
      Desde(n=0;n<CD;n++){
            Si (entrada[j][n].ent==1){
                   r← n;
             } Fin
      } Hasta
      Si(i!=r1){
          objetivo2(entrada,asignacion,instalacion,i,r,j);
          xxx← objetivo(entrada,asignacion,instalacion);
         contador← contador+1;
         Si(xxx< menor && capacidad[i]>cliente[i].demanda ){   // comparo si es menor y la capacidad es permitida
                  Menor← xxx;
                   Desde (x2=0;x2<CD;x2++){  // ver el origen del cambio
                          Si(entrada2[j][x2].ent2==1){
                                   Origen← x2;
                          }Fin
                   }Hasta
                   Desde (i2=0;i2<CL;i2++){
                         Desde (j2=0;j2<CD;j2++){
                              entrada2[i2][j2].ent2= entrada[i2][j2].ent // copia de datos de asignación
                              fila← j+1;
                              columna←  vector[j]+1;/// una de mas
                         }Hasta
                   } Hasta
           }Hasta
           }Fin
        }Fin
        Desde (i4=0;i4<CD;i4++){
            entrada[j][i4].ent← 0;       // dejo igual al origen, lleno de 0
```

```
 }Hasta
 c1← vector[j];
 entrada[j][c1].ent← 1;
 }Hasta
     Desde (i2=0;i2<CL;i2++){
                    Desde (j2=0;j2<CD;j2++){
                          Si(entrada[i2][j2].ent != entrada2[i2][j2].ent2){
                                f← i2;
                                f2← j2; // copia de datos de asignación
                          }
                    }
          }
```

Procedimiento Tabu

```
  Desde (j=0;j<CD;j++){
        Si (entrada2[fila-1][j].ent2==1){
              bandera2← j;
        }Fin
  }Hasta
  Si(tt=N){
        tt← 0;
  }Fin
  Desde (i=0;i<7;i++){
              Si(tabu[i][3]!=0){
                    tabu[i][3]=tabu[i][3]+1;
              }Fin
  }Hasta
  tabu[tt][0] ← fila;
  tabu[tt][1] ← columna;
  tabu[tt][2] ← bandera2+1;//f2 +1
  tabu[tt][3] ←  1
  capacidad[j] ← capacidad[j]-cliente[i].demanda;
  Escribir("\n demanda, %d \n", cliente[i].demanda);
  Escribir("\n capacidad final, %d \n", capacidad[j]);
  capacidad[fila] ← capacidad[fila]-cliente[modific].demanda;
```

3.3.3 Movimientos.

Los movimientos permitidos serán aquellos que cumplan los requisitos establecidos anteriormente. Estos señalan que el cliente solo puede ser asignado a un Centro de Distribución que logre cubrir la demanda del cliente en cuestión.

Para efectos de los futuros análisis se utilizarán 2 grupos de vecinos:

- Uno en donde se realice una búsqueda exhaustiva de todos los vecinos posibles.

- Otra en donde genere solo un grupo de aleatorio de posibles movimientos permitidos.

En ambos casos el movimiento realizado será aquel que disminuya el costo de la función objetivo.

3.3.4 Función Objetivo.

Tomando como base la matriz de asignación es factible obtener el resultado correspondiente al costo que significa implementar el último movimiento realizado. Con dicha información se podrá determinar si dicho movimiento es el adecuado según el algoritmo. El algoritmo de la función objetivo se puede obtener a través del siguiente algoritmo.

Algoritmo Función Objetivo

```
 int objetivo(struct datosEntrada entrada[CL][CD], struct datosAsignacion asignacion[CL][CD],struct datosInstalacion
instalacion[CD]){  // costo de funcion objetivo
  int i,j,transporte,bandera,fijo, ord, alm,total;
  int Q[CD]={1500,1800,1200};  //
  transporte← 0;
  int almacenamiento←  0;
  ord ← 0;
  bandera ←  0;
  fijo←  0;
  alm← 0;
  Desde (i=0;i<CL;i++){ // sumo los costos de transportes
     Desde (j=0;j<CD;j++){
        Si(entrada[i][j].ent==1){
             Transporte←transporte + asignacion[i][j].TC*cliente[i].demanda ;
        }Fin
     }Hasta
  }Hasta
  Desde (i=0;i<CL;i++){ // sumo los costos de alamcenamiento
     Desde (j=0;j<CD;j++){
        Si(entrada[i][j].ent==1){
             almacenamiento←  almacenamiento + cliente[i].demanda*(instalacion[i].HC +  instalacion[i].OC/Q[0] ) ;
        }Fin
     } Hasta
 } Hasta
 Desde (i=0;i<CD;i++){  // sumo los costos fijos
     Desde (j=0;j<CL;j++){
        Si(entrada[i][j].ent==1){
              bandera←  1;
        }Fin
     } Hasta
```

```
      Si(bandera =1){
              fijo<--   fijo + instalacion[i].CF;
      }Fin
      bandera<-  0;
    }Hasta
    total ←   transporte + fijo + almacenamiento ;
return (total);
}Fin
```

CAPITULO 4 EXPERIMENTO Y DISCUSIÓN DE RESULTADOS

4.1 Datos de Entrada.

El estudio de los casos fue realizado tomando datos de entradas con una población de 15 Clientes y 4 Centros de Distribuciones (15*4) y un segundo datos de entradas de (16*4). (Tabla 4).

Demanda	Varianza	Transporte 1	Transporte 2	Transporte 3	Transporte 4
57	553	940	1029	313	22
59	808	661	455	11	19
48	356	2356	246	264	375
65	1325	1349	507	389	242
60	1063	2521	332	127	310
66	532	3174	51	369	267
65	1183	892	363	2	234
66	1459	43	904	763	74
61	852	840	37	77	870
61	980	315	621	655	117
59	816	988	305	211	134
59	710	126	181	205	137
53	269	565	317	134	431
77	3293	1462	629	153	871
61	1398	663	529	277	30

C. Fijo	C. Ord.	Abastec.	Capacidad	Lead Time
589	177	35	435	1
812	244	49	734	3
123	37	7	357	3
67	20	4	983	4

Tabla 4 Datos de Entrada

Los datos de entradas contiene información de la demanda que posee cada cliente, la varianza de la demanda, el costo de transporte desde cada Centro de Distribución hasta cada cliente, los costos fijos de instalación, de ordenamiento, abastecimiento de cada Centro de Distribución, la capacidad de abastecimiento de cada CD's y finalmente el Lead Time. Estos datos se capturan desde un archivo plano de nombre "Datos Simulación". Los resultados obtenidos de estos datos de entradas nos permiten realizar las primeras comparaciones del comportamiento de la heurística.

4.2 Relación Costo Iteración.

Los resultados obtenidos tanto en la muestra de 15*4 como 16*4 nos indican que la heurística converge de manera rápida y efectiva al costo mínimo de los datos muestreados.

La imagen de la figura 4.1 grafica la forma en que la heurística converge hacia el valor óptimo, el cual para este caso en particular es de $ 5600. Considerando la aleatoriedad de la solución inicial y de la forma en que se desenvuelve el algoritmo es imposible determinar de manera exacta el número de iteraciones exactas en las que se encuentra el valor óptimo. Para los datos de entradas de la tabla 1 es posible determinar luego de un número considerable de pruebas que la cantidad necesarias de iteraciones que convergen al valor mínimo es 20, esto considerando todas las restricciones propuesta en los capítulos anteriores.

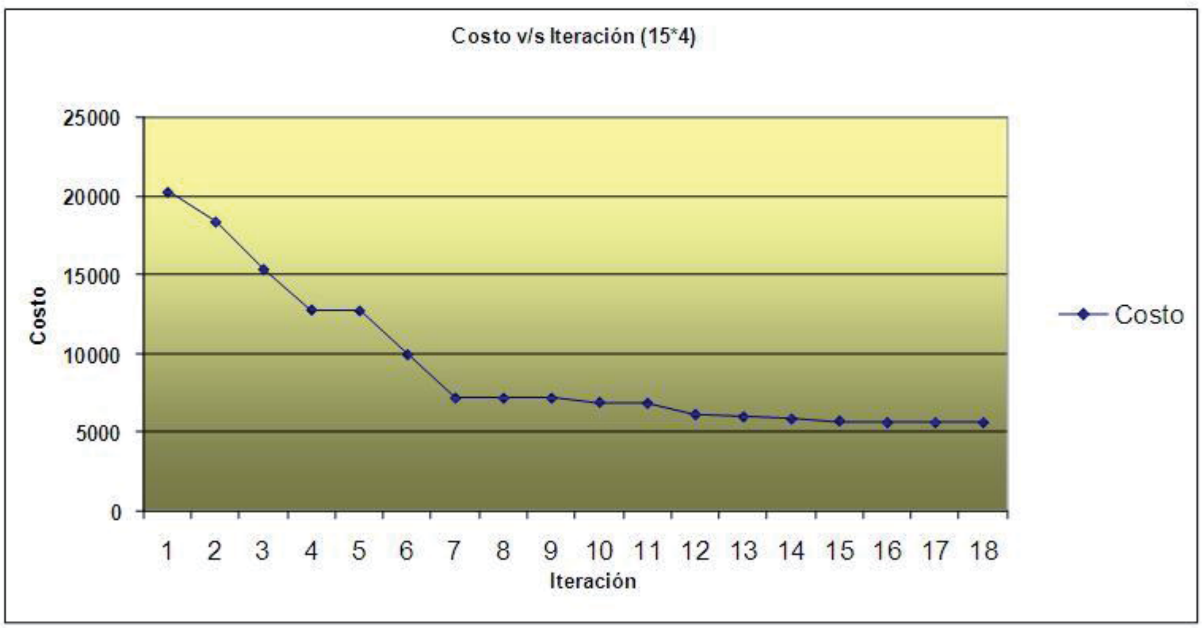

Figura 4.1 Costo v/s Iteración (15*4)

Para el caso de la figura 8.2.2, en donde se aumenta el número de cliente es posible determinar que el número de iteraciones necesarias para llegar al óptimo es cercano a 25, lo que implica mayor tiempo de procesamiento. Para poder replicar el resultado obtenido es necesario utilizar una semilla de valor 7, la cual entregará idénticos resultados que los graficados. Esto considerando que los valores obtenidos son resultado de datos aleatorios que se generan de manera automática por parte del sistema (considerando la variable tiempo de ejecución).

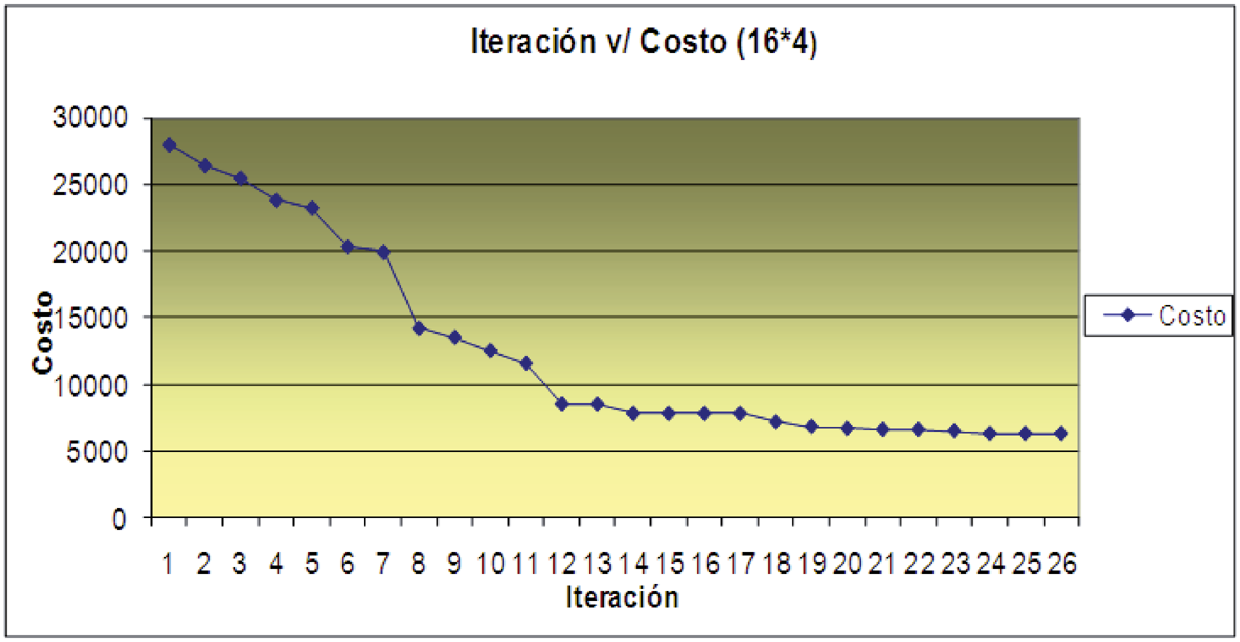

Figura 4.2 Costo v/s Iteración (16*4)

Al comparar los resultados obtenidos con el trabajo desarrollado por Pablo Miranda[Miranda, 2004] se puede demostrar que el algoritmo converge al mismo resultado, sin embrago esto ocurre de una manera más violenta, en donde los cambios que se generar en cada iteración son de una diferencia más considerada, los cambios son más gráficos en las primeras iteraciones en donde la pendiente toma una mayor fuerza (Figura 4.2).Esto hace que el algoritmo finalmente converja de manera más rápida que el resultado de Miranda[Miranda, 2004].

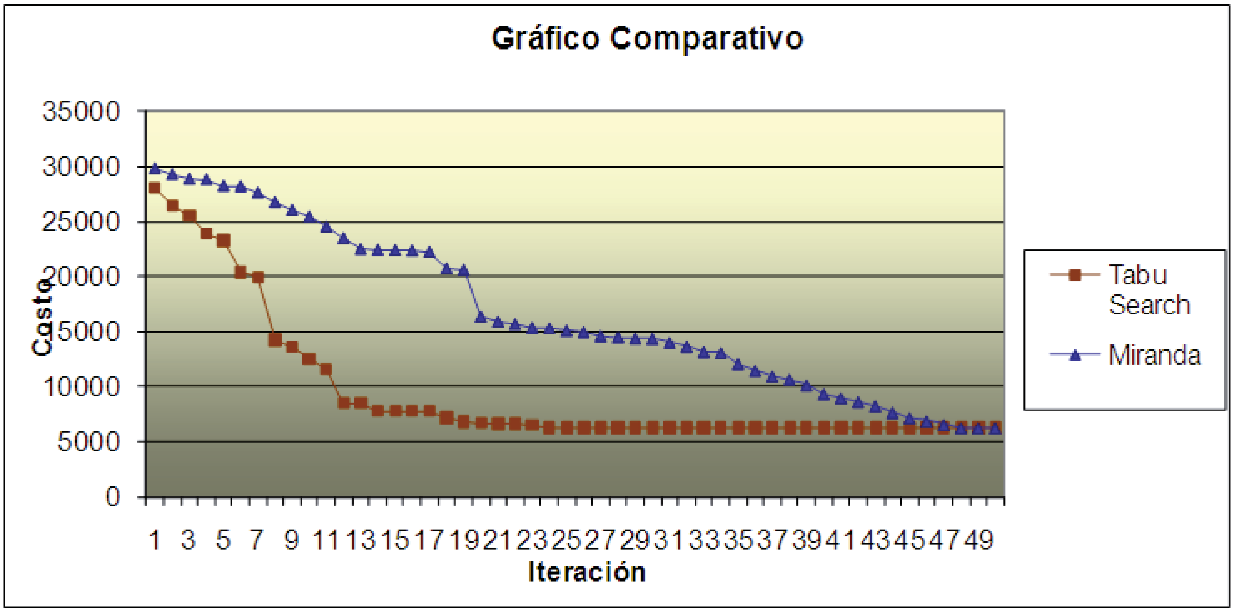

Figura 4.3 Gráfico Comparativo

4.3 Utilización por Centros de Distribución.

Luego de variadas pruebas se ha logrado determinar que la cantidad óptima de Centros de Distribución que genera el costo mínimo de la Red de Distribución está asociada al mínimo números de centros. Esto se debe principalmente a que al eliminar centros baja considerablemente los costos fijos asociados a cada Centro de Distribución.

En la figura 4.3 es posible observar que la solución inicial genera una utilización de los 4 Centros, a medida que se van generando iteraciones el modelo tiende a eliminar el Centro 1 (iteración 5), en las siguientes iteraciones el modelo elimina el segundo Centro. De esta forma el Diseño final queda compuesto tan solo de 2 Centros de Distribuciones que son los que cubren toda la demanda de los 15 clientes.

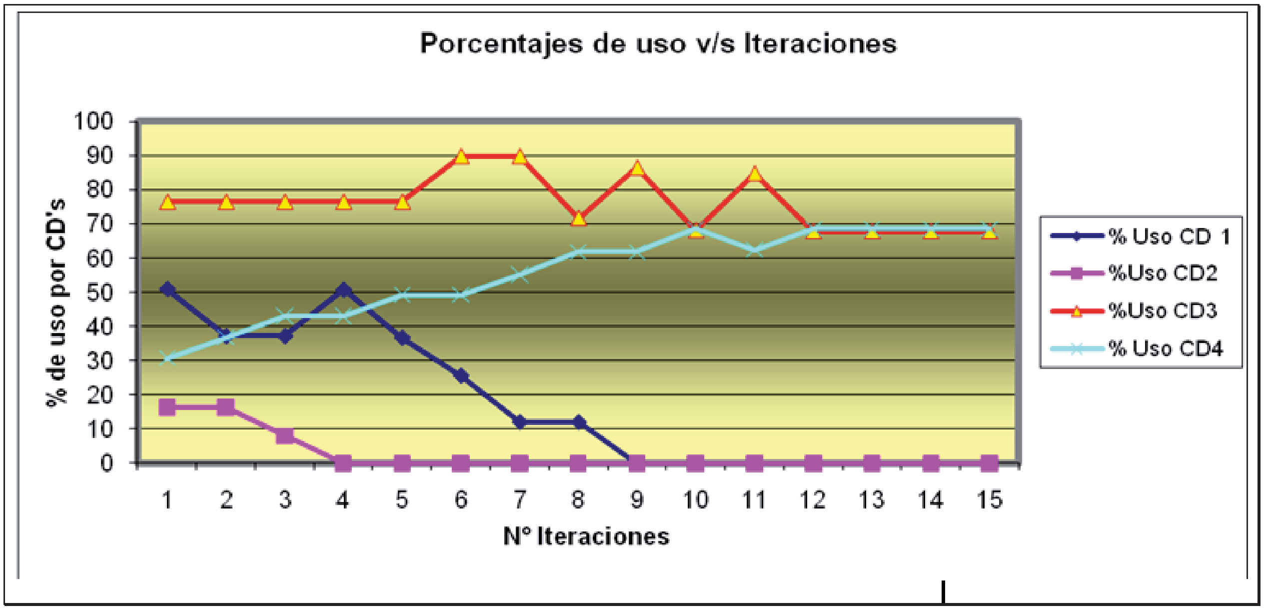

Figura 4.4 Porcentajes de Uso

Los resultados obtenidos anteriormente están totalmente relacionada con la restricción de capacidad de posee cada Centro de Distribución, esta restricción indica que cada Centro tiene una capacidad máxima de cobertura para la totalidad de sus clientes. En la figura 4.4 se ejecuta el programa con los mismos datos de entrada y el mismo número de clientes y CD's, su única diferencia radica en la capacidad que posee cada Centro. El resultado obtenido cambia radicalmente el preforman del algoritmo, tanto así que aumenta el número de Centros que son necesarios para cubrir la totalidad de la demanda de los clientes.

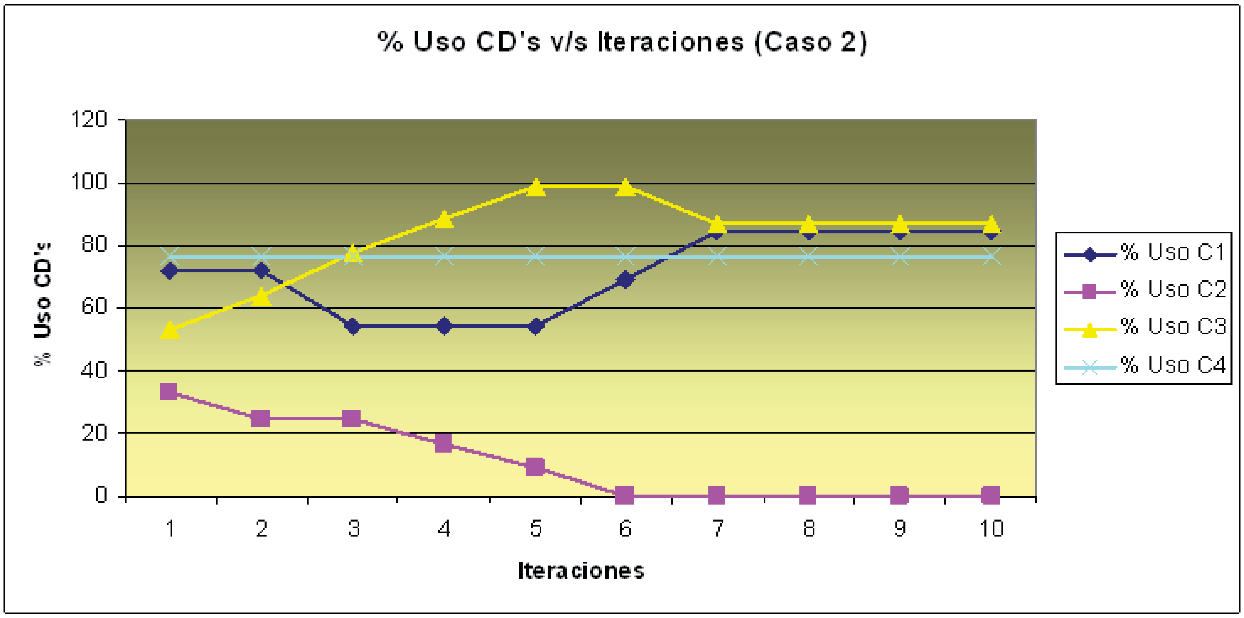

Figura 4.5 Porcentaje de Uso, caso 2

CAPITULO 5 CONCLUSIONES

Conclusiones Generales.

Finalizada esta investigación es posible concluir de manera global que las Metaheurística Tabú Search en esta problemática planteada, así como en muchas más expuestas por otros autores nos aporta resultados valiosos y esperados, los cuales nos permiten confirmar que es una estrategia robusta ante problemas de gran envergadura.

Las conclusiones que se han logrado obtener a través del capítulo anterior es verificar el buen performance de la Heurística Tabu Search para el problema de Diseño de Redes de Distribuciones. Esta estrategia ha logrado converger sus resultados de manera rápida y efectiva al óptimo deseado, generado de los datos de entrada. Estos resultados se pueden verificar en el capítulo 4 en donde los datos de pruebas utilizados convergen de manera expedita a los óptimos deseados.

La segunda conclusión importante que se ha permitido demostrar en la presente investigación es la velocidad con la que el algoritmo logra converger a su valor óptimo, en donde a través de cerca de 20 iteraciones aproximadamente lograr llegar al valor esperado. Esta velocidad sin embargo se ve influenciada directamente por el tamaño del vecindario que se considera, si el vecindario que se toma es alto, existen un mayor grupo de datos que comparar, por lo tanto la velocidad que se logra es más alta. Si el tamaño del vecindario es mayor también puede repercutir en el resultado final, ya que es probable que no se logre llegar al óptimo deseado.

El hecho de integrar a esta investigación los costos asociado al inventario (Capitulo 2.2) nos permite generar resultados más cercanos a la realidad que concuerden con los verdaderos costos que implica generar en un caso real el diseño de una red de distribución, por lo tanto poder utilizarlo con mayor grado de confiabilidad si fuera necesario.

Se logró demostrar en el capitulo anterior que la Red de Distribución tiende a generar mallas en donde el número de centros de Distribución es la mínima posible, esto debido a que un menor número de Centros disminuye drásticamente el costo total al eliminar los costos asociados a los Costos Fijos de instalación.

Finalizada esta investigación es posible proyectar futuros trabajos, los cuales podemos enfocarlos en dos grandes ítems:

Generar pruebas con datos de entradas de mayor tamaño, esto quiere decir con un mayor número de Centros de Distribución y Clientes. Estas pruebas con datos de mayor envergadura permitirían comprobar el funcionamiento del algoritmo bajo una situación más compleja y generar conclusiones más acabadas de la heurística Tabu Search para el problema en cuestión. Para datos de entradas superiores a 20*10 se esperaría obtener resultados cercanos al 96 % de una heurística de buen funcionamiento.

Un segundo trabajo futuro que sería interesante realizar es la de comparar el funcionamiento del algoritmo Tabu Search frente a otras heurísticas aplicadas al tema en cuestión. Estrategias tales como el Algoritmo Genético han logrado buenos resultados, los cuales permitirían realizar nuevos Beanchmarking que permitirían ampliar las conclusiones generadas en esta investigación.

CAPITULO 6 REFERENCIA

[Abdul-Jalbar, 2003] B. Abdul-Jalbar, J. Gutiérrez, J. Sicilia, "Análisis de Sistemas Multiniveles de Inventario con demanda determinística", 27 Congreso Nacional de Estadística e Investigación Operativa, (2003).

[Alonso, 2005] Sergio Alonso, "La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques" (2005).

[Beasley, 1992] J. Beasley, "Enchancing an algorithm for set covering problem. European Journal of Operatonal Research", (1992).

[Carrizosa, 2005] E. Carrizosa, "Algunas aportaciones de la investigación operativa a los problemas de localización", Revista Internacional de ciencia y tecnología de la información geográfica (2005).

[Cazau, 2002] Pablo Cazau,"Introducción a la Metodología de la Investigación", Etapas de una investigación típica, Capítulo 1, Buenos Aires, 2002.

[Daskin, 1995] Mark S. Daskin, "Network and Discrete location Models, Algorithms and applications", (1995).

[Ferhat, 2005] Mehmet Ferhat Candas, Erhan Kutanoglu, "Benefits of Considering Inventory in Service Parts Logistics Network Design Problems with Time-based Service Constraints",(2005).

[Filho, 1999] V. Filho, R Galvao, "A Tabu Search heuristic for the concentrador location problem",(1999).

[Itaim, 2005] Pablo Itaim Ananias, "Resolución del Problema de Set-Covering utilizando un Algoritmo Genético", (2005).

[Glover, 1998] Fred Glover,Belén Melián, "Búsqueda Tabu", Leeds School of Business University of Colorado, (1998).

[Gutiérrez, 2003] J. Miguel Gutiérrez Expósito, "Eficient Aproaches for some extensions of the EOQ Model", (2003).

[Miranda, 2004] Pablo Miranda, "Un enfoque integrado para el diseño estratégico de redes de distribución de carga", Tesis para optar al grado de Doctor en Ciencias de la Ingeniería, PUCV, (2004).

[Miranda, 2006] Pablo Miranda, Rodrigo Garrido, "A Simultaneous Inventory Control and Facility Location Model with Stochastic Capacity Constraints", (2006).

[Miranda, 2007] Pablo Miranda, Rodrigo Garrido, "Niveles de Servicio y Diseño de la Red de Distribuciónen un Enfoque de Optimización Iterativo", Proyecto FONDECYT N°1060945, y por la Universidad Nacional Andrés Bello, Chile, a través del Proyecto DI: 31–05/R.,(2007).

[Martí, 2003] Rafael Martí, "Algoritmos Heurísticos en problemas de optimización combinatorial" (2003).

[Michel, 2002]Laurent Michel, Pascal Van Hentenryck, "A simple tabu search for warehouse location" (2002).

[Osman, 1996] I.H. Osman y J. Kelly, "Meta-Heuristics: Theory and Applications" (1996).

[Ozen, 2004] L. Ozen, M. Daskin ,"Capacitated Facility Location Model with Risk Pooling" (2004).

TABLA DE ILUSTACIONES

TABLA DE CONTENIDO.