



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA
INGENIERÍA DE SISTEMAS – INVESTIGACIÓN DE OPERACIONES

EL ALGORITMO TABÚ SEARCH PARA LA RESOLUCIÓN DE UN
PROBLEMA DE ASIGNACIÓN, SECUENCIACIÓN Y ENTREGAS CON
RESTRICCIONES DE VENTANAS DE TIEMPOS DE ATENCIÓN

TESIS

QUE PARA OBTENER EL GRADO DE:

MAESTRO EN INGENIERÍA

PRESENTA:

GABRIEL POLICRONIADES CHÍPULI

TESIS DIRIGIDA POR:

DRA. IDALIA FLORES DE LA MOTA
FACULTAD DE INGENIERÍA

MÉXICO, D. F. MAYO 2013



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

JURADO

PRESIDENTE: DR. SUÁREZ ROCHA JAVIER

SECRETARIO: M.I. WELLENS PURNAL ANN GODELIEVE

VOCAL: DRA. FLORES DE LA MOTA IDALIA

1ER. SUPLENTE: M.I. RIVERA COLMENERO JOSÉ ANTONIO

2DO. SUPLENTE: DRA. SEGURA PÉREZ ESTHER

Lugar o lugares donde se presentó la tesis: UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

CIUDAD UNIVERSITARIA, MÉXICO, D.F.

TUTOR DE TESIS

DRA. IDALIA FLORES DE LA MOTA

FIRMA

Índice

Resumen	7
Abstract	7
INTRODUCCIÓN.....	8
OBJETIVO	7
HIPÓTESIS	8
LIMITACIONES Y ALCANCE.....	9
CAPÍTULO 1. DESCRIPCIÓN DE LA PROBLEMÁTICA.....	11
1.1 Introducción a los problemas de ruteo de vehículos.....	11
1.2 Definición de la problemática	14
CAPÍTULO 2. MARCO TEÓRICO.....	18
2.1 Revisión de la literatura	18
2.1.1 Programación de vehículos con un único almacén (SDVPS)	18
2.1.2 Programación de vehículos con múltiples almacenes (MDVSP)	19
2.1.3 Problema del agente viajero con ventanas de tiempo de atención (TSPTW).....	20
2.1.4 Problema del camino más corto con ventanas de tiempo (SPPTW).....	20
2.1.5 Modelos VRP y VRPTW	21
2.1.5.2 Problema de ruteo de vehículos con ventanas de atención (VRPTW)	22
2.3 Modelo matemático	22
2.4 Selección del <i>algoritmo de búsqueda tabú</i>	25
2.4.1 Algoritmos heurísticos:	25
2.4.2 Algoritmos exactos:	26
2.4.3 Algoritmos metaheurísticos	26
CAPÍTULO 3. DESCRIPCIÓN DEL ALGORITMO	29

3.1 El algoritmo	29
3.1.1 Breve descripción del algoritmo	29
3.1.2 Supuestos	30
3.2 Algoritmo glotón-tabú.....	31
3.2.1 Esquema general de la aplicación	37
3.2.2 Algoritmo glotón	39
3.2.3 Implementación del algoritmo	40
3.2.3.1 Algoritmo glotón.....	40
3.2.3.2 Algoritmo de búsqueda tabú	52
CAPÍTULO 4. ANÁLISIS NUMÉRICO.....	67
4.1 Obtención de los datos de prueba y comparativos	67
CONCLUSIONES	72
RECOMENDACIONES	73
BIBLIOGRAFÍA	74
ANEXOS.....	76
Anexo 1. Prueba de varianzas iguales para los diferentes algoritmos.....	76
Anexo 2. Análisis de regresión el <i>algoritmo de búsqueda tabú</i>	78
Anexo 3. Análisis de regresión utilizando el algoritmo 1-rbh	99
Anexo 4. Análisis de regresión utilizando el algoritmo cplex	101
Anexo 5. Resultados del <i>algoritmo de búsqueda tabú</i>	103
Anexo 6. Diferentes corridas del algoritmo de búsqueda tabú	109
Anexo 7. Instancias para ejecutar los algoritmos	131
Anexo 8. Horarios de ventana de atención	133
Anexo 9. Análisis de pronósticos de cada uno de los algoritmos	140
Anexo 10. Pronósticos para el algoritmo de búsqueda tabú.....	158
Anexo 11. Diagrama de algoritmo de búsqueda tabú	160

ÍNDICE DE FIGURAS

Figura 1-2	Actividades primarias y secundarias de la logística.	13
Figura 1-3	Gráfica de porcentaje de cumplimiento.	16
Figura 3-1	Interfaz gráfica de los parámetros de la aplicación	32
Figura 3-2	Pestaña para introducir el modelo de Cargas en el software.	36
Figura 3-3	Pestaña ventanas de atención.	37
Figura 3-4	Funcionamiento de la programación del algoritmo	38
Figura 3-5	Representación de las iteraciones.	38
Figura 3-6	Representación de la solución óptima.	38
Figura 3-7	Representación gráfica del Vector Pila	43
Figura 3-8	Intercambios tipo I	53
Figura 3-9	Selección aleatoria del vehículo a eliminar	54
Figura 3-10	Representación de los arcos del vehículo a eliminar en el Vector Dummy	55
Figura 3-11	Intercambios tipo I	56
Figura 3-12	Intercambios tipo II A	57
Figura 3-13	Intercambios tipo II B	58
Figura 3-14	Intercambios tipo II C	58
Figura 3-15	Intercambio tipo II D	59
Figura 3-16	Intercambio tipo III A.....	59
Figura 3-17	Intercambios tipo III B	60
Figura 3-18	Colocación de un arco del vehículo a eliminar en otro camión	61
Figura 3-19	Vehículo eliminado indicado en negro	63
Figura 3-20	Representación gráfica después de eliminar un camión	64

ÍNDICE DE TABLAS

Tabla 1-1	Cumplimiento de viajes en los diferentes meses del año	15
Tabla 2-1	Características de los diferentes algoritmos	27
Tabla 3-1	Horarios reales de las ventanas de atención.....	32
Tabla 3-2	Continuación de tabla 3-1	33
Tabla 3-3	Ventanas de atención ajustadas al algoritmo de González (2004)	33
Tabla 3-4	Continuación de tabla 3-3	34
Tabla 3-5	Representación del horizonte de planeación y vehículos	41
Tabla 3-6	Continuación de tabla 3-3	42
Tabla 3-7	Demanda hipotética para mostrar el funcionamiento del algoritmo	43
Tabla 3-8	Representación de la Lista Tabú.....	53
Tabla 3-9	Representación de un camión candidato.....	60
Tabla 3-10	El movimiento de un arco a otro camión.....	62
Tabla 3-11	Movimiento 2	62
Tabla 3-12	Movimiento 3	62
Tabla 3-13	Movimiento 4	63
Tabla 3-14	Obtención del Valor de la Función Objetivo del primer candidato	64
Tabla 3-15	Representación del movimiento en la Lista Tabú	65
Tabla 4-1	Definición del diseño experimental	67
Tabla 4-3	Análisis de las corridas de cada algoritmo	68
Tabla 4-4	Comparativos condensados de los tres algoritmos parte uno	69
Tabla 4-5	Continuación de la tabla de comparativos.....	70

Agradecimientos

A mis padres por apoyarme a cada paso de mi vida, por ser un ejemplo a seguir. Gracias.

A mi hermanita por apoyarme y estar conmigo a cada momento.

A Idalia Flores de la Mota por su gran apoyo para la elaboración de esta tesis y durante toda mi estancia en la maestría.

Un especial agradecimiento a Rodolfo Torres Matus y al Instituto Tecnológico de Monterrey Campus Toluca por apoyarme incondicionalmente para la realización de esta tesis.

A mis grandes amigos de Pemex que tanto aprecio: Sonia Elizabeth Pérez Tejeda, José David Arce Monroy, Luis Fernando López Cisneros, Fernando Ríos González; quienes me formaron como profesionista en el área laboral.

A Marbel Nuñez Carretero por apoyarme en la elaboración de esta tesis.

A María Ivon por todo el gran amor y apoyo que me ha mostrado a cada momento. Siendo ella la mujer que inspiro la conclusión de esta tesis.

A mis compañeros de la universidad: Vicente Flores Olvera, María del Socorro Guevara Rodríguez.

Resumen

Este trabajo tiene como propósito mostrar una aplicación real del *algoritmo de búsqueda tabú*, para la solución de un problema de asignación, secuenciación y entregas de un producto perecedero a doce centros de distribución; cada centro de distribución cuenta con un horario específico bajo el cual puede recibir al vehículo asignado para descargar material.

Se diseñó un algoritmo específico para la problemática de la compañía dividido en dos etapas; la primera de ellas involucra un algoritmo heurístico que genera soluciones sin un mecanismo para evitar óptimos locales. La segunda de ellas consiste en la aplicación del algoritmo de búsqueda tabú a las soluciones obtenidas por la primera etapa. Este segundo algoritmo cuenta con estrategias propias para evitar óptimos locales y obtener una mejor solución al problema. El objetivo es localizar el menor número de vehículos para cubrir la demanda total del cliente, cumpliendo con la restricción de las ventanas de atención de cada centro de distribución.

Al aplicar el algoritmo de búsqueda tabú se obtuvieron resultados satisfactorios. Sus valores generados presentan una desviación de un vehículo con respecto a los valores óptimos presentados por González (2004). Prácticamente un 4% mejor que el algoritmo 1-RBH que se desvía un promedio de dos vehículos del valor óptimo.

Abstract

This work is intended to show a real application of *tabu search algorithm* for the solution of an assignment problem, sequencing and delivery of a perishable product twelve distribution centers, each distribution center has a specific schedule under which they can receive the vehicle assigned to unload material.

A specific algorithm was designed for the problem of the company, divided into two stages: the first one involves a heuristic algorithm that generates solutions without a mechanism to avoid local optima. The second one is the application of tabu search algorithm to the solutions obtained by the first stage. This second algorithm has its own strategies to avoid local optima and get a better solution. The goal is to find the smallest number of vehicles to cover the total customer demand, under the restriction of time windows for each distribution center must be compliment.

Applying a tabu search algorithm offer a satisfactory results. Their values have generated a standard deviation of one vehicle with regards to the optimal values presented by Gonzalez (2004). The tabu search algorithm is 4% better than algorithm 1-RBH, which diverges an average of two vehicles from the optimal value.

INTRODUCCIÓN

A lo largo del tiempo, una de las inquietudes de la ingeniería ha sido resolver el problema relacionado con la distribución de mercancías. Su mayor interés radica en la búsqueda implacable por reducir los costos logísticos en los que incurren las compañías, y que llegan a reflejarse hasta en un 70% del costo de un producto. Estos costos claves son, para la comercialización de un producto en un mercado particular, razones de éxito o fracaso.

Algunas de las dificultades que se presentan al resolver este tipo de problemas se incrementan cuando existe un producto, el cual se desea desplazar de una ciudad a otra, en modo de recolección o entrega, cuando las distancias entre ellas son distintas. Es decir, supongamos que se tiene un comerciante que desea visitar cuatro ciudades con distancias diferentes entre ellas y conectadas entre sí. El comerciante desea regresar a su punto de origen haciendo uso del mínimo costo posible; pues al existir cuatro ciudades, el comerciante tendrá que elegir entre 4! posibilidades para poder localizar aquella ruta con un costo mínimo. Como puede apreciarse, a medida que la cantidad de ciudades crece, crece la cantidad de rutas a analizar. Esto, para un algoritmo óptimo, es un gran esfuerzo computacional, que en la mayoría de las ocasiones consume una gran cantidad de tiempo, tiempo que para las compañías no existe.

Sin embargo, los algoritmos óptimos no son los únicos destinados a resolver este tipo de problemas. Para ello, es posible hallar algoritmos heurísticos que basan sus soluciones en mecanismos simples, de prueba y error, para hallar buenas soluciones, pero estos algoritmos presentan la desventaja de quedarse sitiados en óptimos locales. Una tercera opción se localiza en los algoritmos metaheurísticos que, a diferencia de los algoritmos heurísticos, contienen mecanismos que le permiten aprender y moverse de un óptimo local a una zona desconocida para iniciar nuevamente una búsqueda, evitando así movimientos que pueden dañar la mejora de una solución.

En los algoritmos metaheurísticos, anteriormente mencionados, se localiza el *algoritmo de búsqueda tabú* creado en la década de los 70's por el doctor Fred Glover. Éste consiste en un algoritmo con la capacidad de rechazar aquellos movimientos que generan una mala solución, almacenándolos en una memoria de corto plazo a la que llama *lista tabú*. En su proceso, el algoritmo solicita la entrada de una solución heurística a la que posteriormente mejora por medio de mecanismos propios; dicho proceso de búsqueda consiste en moverse por medio de intercambios o criterios específicos de la colindancia entre una solución y otra. Cuando una de las soluciones no le permite, al algoritmo, brincar a un óptimo local, ya sea diversificando y enriqueciendo la búsqueda, se llega al criterio de paro del algoritmo.

El *algoritmo de búsqueda tabú* será aplicado a una empresa localizada en el Estado de México dedicada a la distribución de bebidas refrescantes, así como a sus doce centros de distribución. Actualmente, la asignación de viajes no se realiza de manera óptima, pues siempre existen excedentes o faltantes en la demanda del producto. Por lo cual, la compañía solicitó al ITESM Campus Toluca la elaboración de un algoritmo para resolver dicha problemática.

El Doctor Manuel González de la Rosa, en 2004, presentó un modelo matemático y un algoritmo heurístico al que llamó 1-RBH para resolver el problema; sin embargo, en 2008, la compañía solicitó un algoritmo más para solucionar el problema de distribución del producto. Por lo que González creyó conveniente realizar el diseño de un *algoritmo de búsqueda tabú* que brindara la capacidad de una metaheurística para dicha problemática.

Con base en lo anterior se desarrollará el siguiente trabajo de investigación sobre el *algoritmo de búsqueda tabú*.

OBJETIVO

General

Mostrar el desempeño del *algoritmo de búsqueda tabú* en la aplicación del problema de ruteo de vehículos, bajo el régimen de ventanas de tiempo de atención, en comparación con el algoritmo heurístico de asignación 1-RBH descrito en González (2004).

Específicos

Conocer los conceptos generales del ruteo de vehículos.

Desarrollar un modelo conceptual de la búsqueda tabú para el problema de ruteo de vehículos, bajo el régimen de ventanas de tiempo de atención.

Programación del algoritmo en Visual Basic.

Ejecutar las instancias presentadas en González (2004) para la comparación entre los algoritmos.

Comparar los resultados del *algoritmo de búsqueda tabú* respecto al algoritmo 1-RBH y el algoritmo óptimo mostrado por Manuel González (2004)

HIPÓTESIS

El ruteo de vehículos es un problema que permite llegar a soluciones que reducen los costos de las empresas.

LIMITACIONES Y ALCANCE

El algoritmo mostrado, en el presente trabajo de investigación, puede ser aplicado a cualquier compañía que cumpla con las especificaciones y acepte los supuestos del mismo. Se recomienda su uso para compañías de giro alimenticio o de fármacos donde su sistema de entrega sea riguroso.

El algoritmo se diseñó para anchos de ventana específicos dada la problemática de la compañía bajo estudio, así como para una flota de vehículos homogéneos con las mismas características y capacidades de carga.

Para empresas donde no se cuenten con las mismas características deberán modificarse los parámetros del algoritmo para su correcta aplicación.

CAPÍTULO 1. DESCRIPCIÓN DE LA PROBLEMÁTICA

1.1 Introducción a los problemas de ruteo de vehículos

La problemática analizada en el presente trabajo de investigación es englobada en una rama de estudio conocida como: “el flujo de bienes y servicios, y la información asociada que los pone en movimiento.” menciona Ballou (2004), en su libro *Logística: administración de la cadena de suministro*.

Etimológicamente, la palabra logística proviene del griego *logistikos: relativo al cálculo*; sin embargo, su uso comenzó a utilizarse desde mucho antes de que la palabra misma se concibiera de esta forma. Así, a lo largo del desarrollo de la humanidad, la logística ha jugado un papel muy importante, pues, heurísticamente, se han realizado cálculos para supervisar los insumos y consumos a lo largo del tiempo. Posteriormente, al llegar la necesidad o el interés de expansión de territorios, se observa que los recursos técnicos, naturales y operacionales tenían un límite antes de llegar a su fin y que su correcta utilización simbolizaba la victoria o la derrota. La logística, entonces, nace con la necesidad de tener bienes para subsistir. Si se traslada dicha definición al caso de un negocio, sin importar el giro que contenga, pueden observarse una serie de consideraciones hacia ésta, que la convierten en un punto neurálgico de la actividad industrial, en otras palabras, la logística comprende:

- El movimiento de mercancías
- La cobertura de oferta-demanda
- Manejo de la responsabilidad de la empresa
- La satisfacción del cliente
- Eficiencia en las instalaciones

Dada la competencia entre organizaciones, éstas buscarán un mejor manejo de los recursos para diferenciarse unas de las otras y así generar valor para el cliente.

De esta manera es posible dividir la logística o bien el servicio al cliente, que no precisamente es el cliente final sino cualquiera que recibe el producto a lo largo de la cadena de suministro, en las siguientes actividades: transportación, procesamiento de órdenes y almacenamiento de inventario. Así mismo cada una de éstas es dividida en un tercer nivel conformado por: empaque protector, manejo de materiales, almacenaje, adquisiciones, programación de producto y mantenimiento de información. Todas ellas interactúan entre sí para dar un soporte a la cadena de suministro, de toda compañía, y a las relaciones con sus clientes, como se muestra en la figura 1.

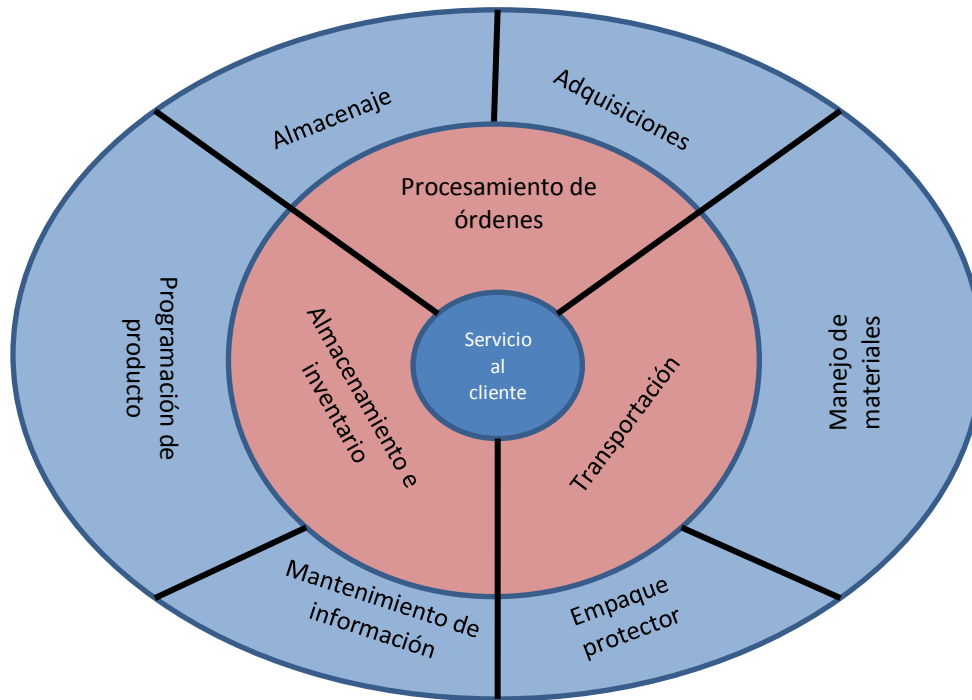


Figura 1-1 Actividades primarias y secundarias de la logística.

Existen muchas opiniones encontradas en cuanto al uso del término logística como cadena de suministro; para ello se mostrarán diversas definiciones que ayudarán a generar un juicio para nuestro fin.

Definiciones de cadena de suministro:

Douglas Lamert: la alineación y encadenamiento de firmas para el correcto flujo de bienes y servicios.

David Simchi-Levi: conjunto de relaciones usadas para integrar eficientemente proveedores, manufacturas, almacenes y tiendas para que las mercancías sean producidas y distribuidas en las cantidades, lugares y tiempos exactos, con el fin de minimizar el costo del sistema mientras se satisfacen los niveles de servicio.

Martín Chistopher: red de organizaciones conectadas e interdependientes que, mutuamente y cooperativamente, trabajan en conjunto para controlar y administrar el flujo de materiales desde los proveedores hasta los usuarios finales.

Definiciones de Logística:

1.- La logística es la administración del flujo de bienes, información y otros recursos, incluida la energía y las personas entre el punto de origen y el punto de consumo a razón de alcanzar los requerimientos de los consumidores (frecuentemente y originalmente, organizaciones militares)

2.- La logística involucra la integración de información, transportación, inventario, almacenaje, manejo de materiales, y empaque. Además es un canal de la cadena de suministro que agrega utilidad de tiempo y espacio, Wikipedia (enero 2013)

Estas definiciones hacen de la logística un punto neurálgico de la actividad industrial:

- Como un asunto vital en el movimiento de mercancías
- En la cobertura de la oferta y la demanda
- Manejo de la responsabilidad de la empresa en la satisfacción del cliente

También, es posible observar las siguientes dificultades que pueden enfrentarse en la logística:

- La optimización del flujo de mercancía
- El uso eficiente de instalaciones
- El manejo del recurso humano
- La modelación de los problemas para su solución

Se observa, entonces, en la figura 1.2 una relación crítica que inicia desde el abastecimiento de materia prima hasta la entrega del producto terminado al cliente; así como el flujo de información en ambos sentidos.

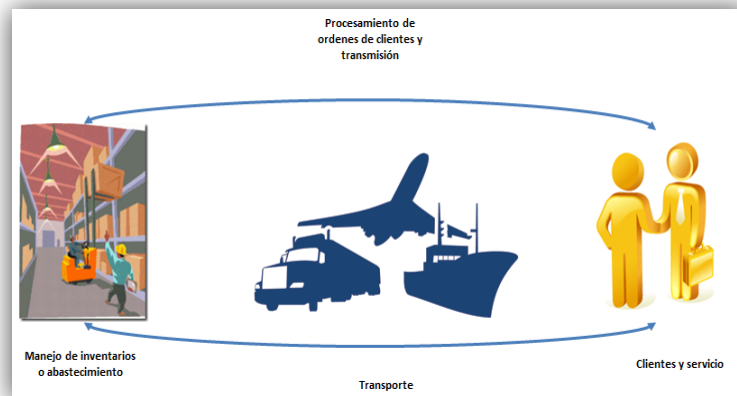


Figura 1-2 Actividades primarias y secundarias de la logística.

Como se observa, el tema del presente trabajo de investigación se orienta hacia el servicio al cliente, que en este caso, es el transporte de materiales de un almacén central, llamado planta, a otros almacenes de distribución minoristas. En otras palabras se centrará en la programación de viajes de vehículos.

Dentro de las actividades principales del transporte pueden mencionarse las siguientes:

- Selección de modo y servicio de transporte
- Consolidación de flota

- Ruteo de transportistas
- Programación de viajes de vehículos
- Procesamiento de reclamos
- Auditorias de tasas de servicio

Con el tiempo se han desarrollado numerosas aplicaciones para mejorar las asignaciones del transporte. Estas aplicaciones van desde métodos heurísticos y metaheurísticos hasta modelos matemáticos complejos que ofrecen valores óptimos. Sin embargo, los tiempos largos de procesamiento de CPU, requeridos para obtener soluciones exactas, cuando el número de rutas posibles aumenta, hacen difícil la selección de alguna de estas aplicaciones.

Ejemplo de ello es el conocido “agente viajero” cuyo objetivo es recorrer una ruta determinada, a un costo mínimo, a través de diferentes ciudades (visitando cada una de ellas) a fin de regresar a la ciudad de donde partió. Si se considera un total de 6 ciudades interconectadas entre sí, y entre las cuales existe un costo determinado, el análisis para seleccionar la mejor ruta implicará un examen de todas las posibilidades existentes a fin de conocer el valor óptimo, lo cual daría un total de 6! rutas a analizar. Si la cantidad de rutas creciera, un algoritmo óptimo tardaría muchísimo tiempo en arrojar la mejor ruta, en dado caso que éste utilice algún método exacto, como el algoritmo Simplex. Por esta razón, los métodos heurísticos son de mayor utilidad debido a su gran aproximación a valores óptimos, cuando la cantidad de ciudades o posibilidades crecen, ofreciendo resultados en un tiempo razonable y menor al de los algoritmos exactos.

1.2 Definición de la problemática

El desarrollo del siguiente trabajo de investigación se centra en la problemática de la compañía ubicada en el Estado de México que produce bebidas refrescantes para satisfacer las necesidades de las comunidades más cercanas. La distribución se realiza por medio de vehículos de iguales características y capacidades, es decir, por una flota homogénea. Estos vehículos parten de un almacén central, llamado planta, a doce centros de distribución, donde cada centro cuenta con un horario específico de ventana de atención para despachar al vehículo que llega dentro de ella.

Los 12 centros de distribución llevan por nombre los lugares en los cuales se localizan:

1. Pacífico
2. Suburbana
3. Metepec
4. Tenango
5. Ixtlahuaca
6. Atlacomulco
7. Villa Cuauhtémoc
8. Lerma
9. Valle de Bravo
10. Tejupilco

- 11. Ixtapan
- 12. Villa Victoria

Mas la demanda del producto ha presentado un crecimiento importante que aumenta con la tasa de población y con ciertos períodos del año donde las personas ingieren más líquido. La compañía X, preocupada por esta problemática, desea ampliar su capacidad de distribución para cumplir con la demanda de producto, pero a su vez busca disminuir los costos de ésta.

Actualmente el sistema de distribución de la compañía se realiza por medio de pronósticos de demanda, donde la persona encargada de los fletes es quien realiza la asignación de los viajes a partir de la experiencia. Mas el sistema anterior no siempre resulta de manera óptima para ello. Además se presenta el temor de que alguna de las unidades sea robada mientras espera, fuera del almacén, para ser atendida.

La mala administración, por parte del departamento de logística, genera incumplimiento en alguno de los viajes de los diferentes CD's, como se muestra en la tabla 1. Aquellos valores cuyo % de cumplimiento promedio es inferior al 100% indican que la demanda no se ha cumplido en su totalidad. Para aquellos valores superiores al 100% se establece que la demanda se excedió. De manera contigua se observa gráficamente esta relación en la figura 1.2.

Tabla 1-1 Cumplimiento de viajes en los diferentes meses del año

Mes	Viajes programados promedio	Envíos en el 1 ^{er} . turno en promedio	Envíos en el 2 ^o . turno en promedio	Envíos en el 3 ^{er} . turno en promedio	Total de envíos en promedio	Diferencia de programados menos total	% de cumplimiento promedio
Enero	1553	801	730	0	1531	22	98.58%
Febrero	1420	674	680	0	1354	66	95.35%
Marzo	1796	844	822	0	1666	130	92.76%
Abril	2068	604	589	436	1629	439	78.77
Mayo	2289	681	713	636	2030	259	88.69
Junio	1442	505	615	359	1479	-37	102.57
Julio	1522	588	603	435	1626	-104	106.83
Agosto	1402	491	559	379	1429	-27	101.93
Septiembre	1485	451	620	402	1473	12	99.19
Octubre	1420	404	559	482	1445	-25	101.76
Noviembre	1420	448	453	416	1317	103	92.75
Diciembre	1557	525	490	467	1482	75	95.18

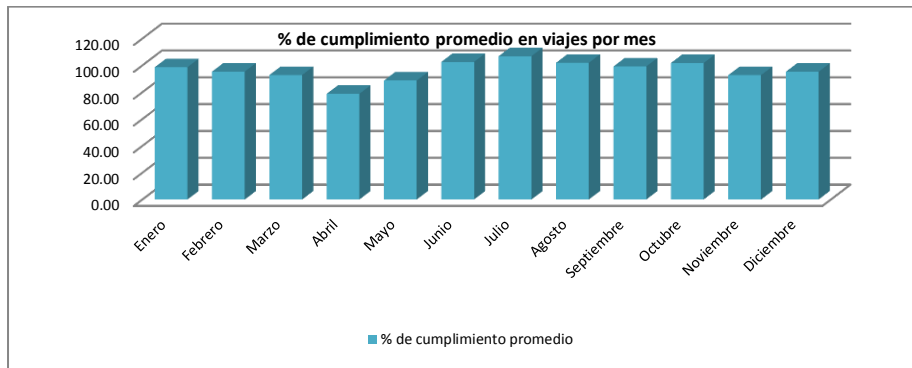


Figura 1-3 Gráfica de porcentaje de cumplimiento.

En la columna de % de cumplimiento es posible observar que en ningún mes se consigue el cumplimiento del 100%, pues en ocasiones existen viajes extras o viajes faltantes, lo que origina pérdidas a la compañía y clientes insatisfechos a lo largo de la cadena de distribución.

La problemática radica en la diferencia entre los recorridos y los anchos de ventanas hacia los diferentes centros de distribución (CD's), lo que provoca una difícil asignación, directa, sin incurrir en alguna falta con alguno de los vehículos que debe visitar a más de un CD's. Esto provoca un incremento en la incertidumbre en los porcentajes de cumplimiento de la demanda, además se le debe sumar el aumento de los distintos productos de la misma marca, 60 productos hoy en día, lo que ha generado una mayor incertidumbre en la satisfacción del cliente.

Los algoritmos propuestos anteriormente son complejos en su interfaz con el usuario, así como lo son la ejecución del algoritmo 1-RBH y la del modelo matemático bajo CPLEX; lo anterior obliga al personal a seguir trabajando de manera tradicional y, por lo tanto, limita la aplicación de alguno de los métodos propuestos. En esta investigación se sugiere una tercera alternativa que permita mejorar los resultados del algoritmo 1-RBH y se aproxime lo más posible al valor óptimo, en un tiempo de ejecución razonable. Por lo que se propone aplicar el *algoritmo de búsqueda tabú* con una interfaz de usuario diferente.

La principal inquietud de la compañía consiste en optimizar la asignación de viajes para cada vehículo, respetando los horarios de atención para cada CD, esto debido a que la planeación se realiza horas antes, sin cambios a lo largo del día, generando un problema de tipo dinámico. Además, al optimizar la asignación de viajes se disminuyen las pérdidas por robo de producto. Objetivo que a su vez generará la disminución de costos de manera general para el departamento de logística.

CAPÍTULO 2. MARCO TEÓRICO

2.1 Revisión de la literatura

La compañía refresquera hace frente a la problemática construyendo heurísticamente un plan de distribución. Una solución factible es elaborada a partir de los siguientes dos principios:

- Qué centros de distribución deben ser asignados a cada ruta para cumplir las ventanas de atención
- En qué orden deberán ser cargados los camiones para cumplir dicho objetivo

Con base en los dos principios anteriores la compañía deberá considerar restricciones de ventanas de atención, ya que cada ruta deberá empezar y terminar dentro del horizonte de planeación, definido a 24 horas; así mismo con los tiempos de las ventanas de atención de cada centro de distribución. Algunos problemas relacionados con las restricciones de ventanas de tiempo son: vehicle routing problem (VRP), single-depot vehicle scheduling Problem (SDVSP), the multiple depot vehicle scheduling problem (MDVPS), the traveling salesman problem with time windows (TSPTW), the shortest path problem with time windows (SPPTW) y vehicle routing problem with time windows (VRPTW).

2.1.1 Programación de vehículos con un único almacén (SDVPS)

El **SDVPS** se considera (Desrosiers J., 1995): un conjunto de n viajes $\tau_1, \tau_2, \dots, \tau_n$, donde el viaje τ_i tiene una duración y un inicio de ventana $a_i, i = 1, 2, \dots, n$. Considerando también un único almacén donde un vehículo v es estacionado. Sea un conjunto de nodos $N = 1, 2, \dots, n$ que representan el conjunto de viajes. Y el nodo $n + 1$ representa el único almacén. Sea también τ_{ij} que denota el tiempo de viaje entre un punto τ_i y un punto τ_j . Para simplicidad se asume que t_{ij} incluye τ_i . Una orden de un conjunto de viajes τ_i, τ_j se dice que son compatibles si se satisface la relación $a_i + t_{ij} \leq \tau_j$, para todo $i, j \in N$. Se denota un conjunto compatible de viajes como I , como subconjunto de $N \times N$. Entonces se define $A = I \cup \{n + 1\} \times N \cup N \times \{n + 1\}$ como un conjunto de arcos. Sea $c_{ij}, i, j \in I$ en el costo incurrido si un vehículo realiza un viaje τ_i seguido de un viaje τ_j . Este costo es usualmente una función de la distancia entre el punto de viaje τ_i y el inicio del punto de viaje τ_j . Sea d el costo fijo asociado a cada vehículo.

En general, el problema consiste en encontrar una asignación de vehículos para los viajes tal que:

- Cada viaje es cubierto exactamente por un vehículo
- Cada vehículo deja el almacén central, realiza la secuencia de viajes y al final de la ruta regresa nuevamente al almacén central
- El número de vehículos que abandonan el almacén no puede exceder la capacidad de vehículos v

- La sumatoria de los costos de los vehículos utilizados es minimizada

El **SDVSP** tiene las siguientes diferencias con respecto al problema de investigación:

- No es un requerimiento que cada viaje deba ser realizado, pero es importante observar que durante el periodo de ventana de atención un centro de distribución puede ser visitado varias veces. El problema requiere que un conjunto de estos viajes sea seleccionado para poder cubrir la demanda
- Una vez cubierta la demanda de un centro de distribución, el vehículo deberá regresar al almacén central
- El número de vehículos a utilizar no es fijo, pero es una variable a determinar

2.1.2 Programación de vehículos con múltiples almacenes (MDVSP)

Estos modelos consisten, principalmente, en la asignación de un conjunto de vehículos ubicados en diferentes centros de distribución esperando a ser asignados a determinados viajes, dicha asignación busca la minimización de los costos. Además es una generalización de los problemas **SDVSP**, donde existe una cantidad de almacenes K con un único almacén central, una flota v^k de vehículos donde $k \in K$. Y de igual modo que el modelo anterior deberán cubrir los viajes asignados.

Están divididos en dos categorías:

- Viajes activos: se definen así cuando en su recorrido en lugar de tener costos se tiene ganancias; un ejemplo de ello podría ser el servicio de transporte colectivo, donde al realizar una ruta determinada se obtiene una ganancia por el cobro del pasaje. Por otra parte, los viajes deben realizarse por la misma ruta, lo cual implica que la compañía incurra en los mismos costos.
- Viajes expresos: se definen como aquellos viajes que se deben realizar para una agencia de viajes, éstos a su vez están divididos en tres categorías:
 - *Pull out trips*: cuando el vehículo debe trasladarse a otra ciudad, diferente, a aquélla donde se guarda para poder empezar sus viajes.
 - *Pull In*: aquellos viajes en los que se requiere, luego de una jornada laboral, llevar al vehículo a su lugar de origen.
 - *Deadhead*: viaje intermedio que se realiza mientras el vehículo va de la última ciudad que visitó a la ciudad donde se guarda.

Algunos otros modelos realizados para éste tipo de problema los presenta Bertosi, *et al.* (1978) quien formula el problema **MDVS** como un problema de multiservicio enlazado, proponiendo un algoritmo heurístico para solucionarlo. Lamatsch (1990) propone una solución con base en una red tiempo- espacio; sin embargo, el heurístico más exitoso es realizado por Lôbel (1997) mediante su algoritmo llamado *lagrangean pricing*.

2.1.3 Problema del agente viajero con ventanas de tiempo de atención (TSPTW)

El problema del agente viajero con ventanas de tiempo (TSPTW) consiste en encontrar el camino con el mínimo costo asignado a un vehículo. Éste debe, a partir de un conjunto de nodos dado (exactamente una vez), regresar al nodo de inicio. El servicio en un nodo se inicia con una ventana de atención definida, y el nodo está activo únicamente durante este tiempo. Si un vehículo llega antes de la apertura de dicha ventana, a un nodo determinado, entonces, se le permitirá esperar a que se realice la apertura del mismo. Este modelo es un problema de ruteo de vehículos (VRP) con la única diferencia de que existe un vehículo con capacidad limitada. Savelsberg (1985) probó que, encontrar una solución factible, lo convierte en un problema NP completo.

La relación de este modelo, con el propuesto, se da gracias a las aportaciones de complejidad computacional propuestas para problemas con restricción de ventanas de tiempo de atención. La complejidad computacional depende de los intervalos de las ventanas y de las técnicas de eliminación de arcos.

2.1.4 Problema del camino más corto con ventanas de tiempo (SPPTW)

El SPPTW fue introducido por J. Desrosiers (1983) como una solución al problema de múltiples almacenes para el agente viajero. En él se propone el uso de un sistema de etiquetado permanente para las rutas seleccionadas al que llama: generalized permanent labeling algorithm (GPLA), del cual se obtiene una solución para 250,000 ciudades en pocos minutos.

En (Desrosiers J., 1995) se define como: una gráfica $G = V, A$ donde A es el conjunto de arcos y V es el conjunto de nodos. $N \cup o, d$, donde N consiste en nodos que pueden ser visitados de un origen o a un destino d . Con un tiempo de ventana $a_i b_j$, $i \in V$ es asociado a cada nodo. Un camino en la gráfica G es definido como una secuencia de nodos i_0, i_1, \dots, i_H , de modo que cada arco i_{k-1}, i_k pertenece a A . Todos los caminos se inician en un tiempo a_0 del nodo $i_0 = o$ y termina en un nodo $i_H = d$ no después de b_d . Un camino es elemental si no contiene ciclos. Cada arco i, j tiene un valor negativo o positivo c_{ij} y una duración positiva t_{ij} , para todo $i \in N$. Un arco i, j es definido en el conjunto A si éste es factible. El problema consiste en encontrar el mínimo costo de la ruta entre un recurso y los nodos, mientras se cumplen las ventanas de tiempo.

Su relación con el problema consiste en la cantidad de restricciones de asignación con ventanas de tiempo y el problema de ruteo. Es decir, un conjunto de nodos N puede ser construido considerando los instantes de tiempo del horizonte de planeación. Por otra parte, los arcos pueden ser vistos como arcos de espera y de entrega. Así los períodos de entrega deberán ser calculados considerando las ventanas de atención de cada centro de distribución.

2.1.5 Modelos VRP y VRPTW

Este tipo de modelos se relaciona con el problema debido a que se pueden convertir en él, si se introduce la duplicidad de nodos y se incluyen los arcos en el camino de regreso a la planta, así como los tiempos de carga y de descarga de cada vehículo; incluyendo de manera análoga las restricciones de ventanas de tiempo de atención de cada nodo.

2.1.5.1. Problema de ruteo de vehículos (VRP)

Uno de los algoritmos más importantes y populares que se han desarrollado para modelar el comportamiento de la administración de flotas, y que captura la esencia de asignar vehículos a rutas, además de atender a un conjunto de clientes de manera eficiente, es relacionado con el problema de **VRP**.

El **VRP** representa un área basta de investigación, su estudio recopilado por Bodin (1983), contiene alrededor de 700 referencias, en el cual se documentan métodos de optimización y propuestas de métodos de aproximación para resolver este problema y sus diferentes versiones. Estas referencias, también, documentan las aplicaciones prácticas más recientes del problema, así como su relevancia económica.

El **VRP** está definido genéricamente en función de variables tales como: el número de vehículos que integran una flotilla, el número de clientes a los cuales se les debe atender, la capacidad de cada vehículo (peso/volumen), el tamaño de la orden para un cliente determinado y el costo del viaje directo entre dos puntos. Busca, además, el diseño de una serie de rutas para una flota de vehículos que atienden un conjunto de clientes con demandas conocidas, al mínimo costo. Estas rutas se originan y terminan en un punto o depósito central y sus algoritmos de solución pueden ser formulados en una de las siguientes dos estructuras:

- Con sistemas de ruteo variable: los clientes pueden colocar sus demandas de mercancía hasta un día antes de programar las rutas de los vehículos, de tal forma que al final del día se sabrá con precisión qué clientes requieren ser atendidos y, en consecuencia, aplicar el algoritmo correspondiente para optimizar las rutas.
- Con sistemas de ruteo fijo: las demandas de los clientes son suficientemente estables a tal grado que se puede usar el mismo programa de rutas en varias ocasiones.

El origen del **VRP** fue presentado, inicialmente, por Dantzig (1959) como un problema de despacho de vehículos de clasificación NP-hard; sin embargo, la gran necesidad de resolver este tipo de problemas ha hecho que los investigadores busquen métodos para su resolución. Este tipo de modelo considera un conjunto de vehículos con una capacidad limitada que debe visitar un conjunto de clientes con una demanda específica. Los vehículos parten de un almacén central a realizar su ruta y siempre regresan a este nodo. A cada arco que indica la ruta, existente, entre los nodos se le asigna un costo específico, el cual deberá ser minimizado.

2.1.5.2 Problema de ruteo de vehículos con ventanas de atención (VRPTW)

En 1977 Russell representó una problemática más apegada a la realidad. Él, incorporó una restricción al modelo, donde cada cliente tiene un período de tiempo en el cual podrá ser atendido. Si bien el modelo se resolvió óptimamente, el proceso dependía de modelos heurísticos.

Solomon (1983, 1987) y Salvendy (1984) demostraron que este tipo de problemas es, fundamentalmente, más difícil de resolver que los **VRP** simples. Esto obligó a los investigadores a desarrollar modelos heurísticos de mejor calidad para poder sortear la nueva restricción. Christofides, Mingozzi y Tóth (1981) y Becker (1983) propusieron un método de optimización para el caso de un solo vehículo basados en el método del "agente viajero". Posteriormente se desarrollaron métodos basados en esquemas de ruta más corta como los lagrangianos, hasta que en 1986, Kolen, Rinnooy y Trienekens propusieron el método de "ramificar y acotar" dando origen a la segunda generación.

Este tipo de modelo considera un conjunto de vehículos con una capacidad limitada que debe visitar un conjunto de clientes con una demanda específica. Los vehículos parten de un almacén central a realizar su ruta y siempre regresan a este nodo.

A cada arco que indica la ruta existente entre los nodos se asigna un costo específico, el cual deberá ser minimizado.

Una diferencia notoria entre este modelo y la realidad consiste en que el modelo considera la característica de la desigualdad triangular, es decir, tomar la menor distancia entre dos puntos, situación que en la realidad puede llegar a quebrantarse. Por ejemplo, puede presentarse una mayor distancia y un menor tiempo de recorrido.

2.3 Modelo matemático

Como se sabe todos los problemas de enrutamiento y asignación de vehículos tienen una base común que es posible hallar en todos los modelos matemáticos. Dicha base se describirá a continuación. Los componentes mencionados aquí se hayan plasmados en González (2004).

- Sea un conjunto de vehículos homogéneos K , con $K = m$ con una capacidad q . Para cuestiones de cálculo en nuestro problema esto no se considera útil debido a que el conteo de la demanda se basa en número de viajes, no de una capacidad específica.
- Un conjunto de clientes N , con $N = n$ y una cantidad de producto requerido D^l indica la cantidad de demanda por centro de distribución l , con $l \in L$. En este caso, no existe una desigualdad triangular debido a que un vehículo sale del almacén central, viaja con el cliente, descarga, y retorna al almacén central utilizando la misma ruta, formando una matriz simétrica.
- El tiempo de viaje t . Parte del supuesto que los tiempos de recorrido son iguales, tanto de ida como de regreso.

- La demanda pronosticada: la demanda total debe ser igual al número de viajes de todos los vehículos por flota homogénea.
- Para cada cliente existe un intervalo de tiempo dentro del cual puede atender al vehículo k . Dicho intervalo a, b se define como ventana de tiempo de atención.

Estas bases forman parte del modelo matemático empleado en este trabajo de investigación, mostrado a continuación, para el cual se utilizó el modelo presentado por González (2004):

Notación del modelo:

L : conjunto que representa a los centros de distribución, identificados por l

K : conjunto que representa a los vehículos utilizados, identificados por k

N : conjunto de nodos identificados por i, j

$s \in N$: nodo fuente

$t \in N$: nodo destino

$A_e = (i, j) | i, j : \text{es el arco de espera}, (i, j) \in N$

$A^l = (i, j) | i, j : \text{es un arco de entrega para el centro de distribución } l, l \in L, (i, j) \in N$

$B = i | i \text{ es un nodo inicial de un arco de entrega que va al CD } l, l \in L, (i) \in N$

D^l : demanda del centro de distribución $l \in L$

D : demanda total, $D = \sum_{l \in L} D^l$ (número total de viajes a realizarse para el centro de distribución l).

C^l : costo unitario del transporte en el que se incurre por hacer uso del arco de entrega $i, j \in A^l, l \in L$

T^l : tiempo de viaje en el que se incide por usar el arco de entrega $i, j \in A^l, l \in L, T^l = (tc + 2tv^l + td^l)$; donde, esta expresión representa el tiempo de carga, el tiempo de viaje de ida y vuelta, al centro de distribución, y el tiempo de descarga en el CD específico.

Las variables de decisión son:

$$x_{ijk} = \begin{cases} 1, & \text{si el arco de espera } i, j \text{ es usado por el vehículo } k \\ 0, & \text{en otra forma} \end{cases}; i, j \in A_e, k \in K$$

$$w_{ijk}^l = \begin{cases} 1, & \text{si el arco de entrega } i, j \text{ es usado por el vehículo } k \\ 0, & \text{en otra forma} \end{cases}; i, j \in A^l, k \in K, l \in L$$

$$y_k = \begin{cases} 1, & \text{si el vehículo } k \text{ es usado} \\ 0, & \text{de otra forma} \end{cases} \quad \text{donde } k \in K$$

$r_k =$ Períodos de tiempo en espera del vehículo; $k \in K$

Definición de costos:

G_k : costos incurridos por los períodos de espera de un vehículo k , $k \in K$

F_k : costo fijo en el que se incide al usar el vehículo k , $k \in K$

P : períodos totales en el horizonte de planeación

E : constante seleccionada

El modelo de programación matemática por resolver es:

Función Objetivo:

$$\text{Min } \sum_{l \in L} \sum_{(i,j) \in A^l} \sum_{k \in K} C^l w_{ijk}^l + \sum_{k \in K} F_k y_k + \sum_{k \in K} G_k r_k \quad (2.1)$$

En esta definición de función objetivo podemos encontrar los costos incurridos por la utilización de los arcos, los costos de utilización de los vehículos, más otros costos de espera de los vehículos.

Sujeto a:

$$\sum_{l \in L} \sum_{(i,j) \in A^l} \sum_{k \in K} w_{ijk}^l \leq 1, \quad i \in B \quad (2.2)$$

En esta restricción se asegura que a un camión determinado se le asigne un viaje.

$$\sum_{(i,j) \in A^l} \sum_{k \in K} w_{ijk}^l = D^l \quad l \in L \quad (2.3)$$

Con esta restricción se asegura que la sumatoria, de todos los arcos de entrega, sea igual a la demanda total de todos los centros de distribución.

$$\sum_{(i,j) \in A_e} x_{ijk} - \sum_{l \in L} \sum_{(i,j) \in A^l} w_{ijk}^l = \begin{cases} 1, & \text{si } i = s \\ 0, & \text{si } i \in N - s, t \\ -1, & \text{si } i = t \end{cases}, \quad i \in N; k \in K \quad (2.4)$$

Con esta restricción se define que un vehículo puede hallarse en planta, en un nodo de espera o en un nodo de entrega a un cliente, pero no en una combinación de ellos al mismo tiempo.

$$\sum_{l \in L} \sum_{(i,j) \in A^l} w_{ijk}^l - E y_k \quad k \in K \quad (2.5)$$

Esta restricción garantiza que si un arco j es utilizado, entonces un vehículo k es utilizado de igual manera.

$$\sum_{l \in L} \sum_{(i,j) \in A^l} P^l w_{ijk}^l + r_k = P, \quad k \in K \quad (2.6)$$

Indica los costos por vehículo, si se presenta un arco de espera.

$$r_k \geq 0 \quad (2.7)$$

$$x_{ijk}, w_{ijk}^l, y_k \in \{0, 1\} \quad \forall i, j \in N; k \in K, l \in L$$

Estas últimas tres restricciones se refieren a los controles de las ventanas de tiempo de atención. Por lo que este modelo cumple con las siguientes características: “las asignaciones a los centros de distribución incluyen restricciones de tiempos de rutas. En las cuales, se incluye el tiempo de carga del vehículo en el almacén central”, menciona González (2004).

2.4 Selección del *algoritmo de búsqueda tabú*

Como se mencionó, anteriormente, este tipo de problemas de ruteo se ha resuelto por numerosos métodos de búsqueda, que van desde métodos heurísticos hasta los metaheurísticos o técnicas que ofrecen mejores resultados. También pueden hallarse en otros casos, por mencionar algunos más, como: recocido simulado, el algoritmo de las hormigas, algoritmos genéticos, entre otros. De manera general, estos algoritmos pueden clasificarse de la siguiente manera.

2.4.1 Algoritmos heurísticos:

Dentro de las técnicas heurísticas para resolver este tipo de problemas pueden hallarse métodos de búsqueda local y aleatoria, o de construcción, como pueden ser una técnica voraz de descomposición, entre otras. Sin embargo, el mayor problema de este tipo de algoritmos radica en la susceptibilidad de quedarse estancados en óptimos locales debido a la falta de criterios más avanzados para salir de ellos. Por ejemplo, en problemas donde la cantidad de posibilidades aumenta, la cantidad de óptimos locales crece a su vez, provocando que estos algoritmos sean técnicas poco recomendables para situaciones de este tipo.

2.4.2 Algoritmos exactos:

Este tipo de algoritmos ofrecen un valor óptimo a un problema dado, pero el tiempo de ejecución para este tipo de problemas es excesivo, ya que dependerá de la cantidad de candidatos que deberá analizar a lo largo de su búsqueda. Un ejemplo de ello, puede ser el problema del agente viajero. Supongamos que un cartero debe visitar 7 comunidades interconectadas entre sí, con costos de rutas completamente distintos, y regresar al lugar de origen, cualquiera que este fuese. Si el cartero decide visitar la ruta (1, 4, 6, 7, 2, 3, 5, 1) seguramente le ofrecerá un costo diferente a que si decide tomar otra ruta (1, 4, 2, 7, 6, 3, 5, 1). Ahora, si se observa detenidamente, como un grafo interconectado entre sí, entonces podrá descubrirse que la cantidad de rutas posibles existentes será de $7!$, dando un total de 4200 rutas diferentes. Lo que hará un algoritmo exacto será evaluar cada una de ellas y de allí seleccionar aquella que ofrezca un mínimo costo. Pero, si el número de ciudades aumenta, el tiempo de ejecución será exponencial y la toma de decisiones, de manera eficiente, ya no será factible en aquellos casos donde las compañías necesitan tener una respuesta inmediata a su problema de asignación de vehículos. Por esta razón, este tipo de algoritmos no fueron incluidos en este trabajo de investigación.

2.4.3 Algoritmos metaheurísticos

Francisco Herrera, profesor de la Universidad de Granada, comenta que los algoritmos metaheurísticos guían una heurística por medio de métodos inteligentes para encontrar una solución óptima o aproximada. En la actualidad, los algoritmos metaheurísticos han cobrado fuerza en la resolución de problemas de ruteo, en gran parte, porque a diferencia de los algoritmos heurísticos y exactos no se detienen en óptimos locales y su tiempo de ejecución es razonable, respectivamente. Dentro de este tipo de técnica, se cuenta con tres grandes ramas, las cuales están definidas en la figura 2:

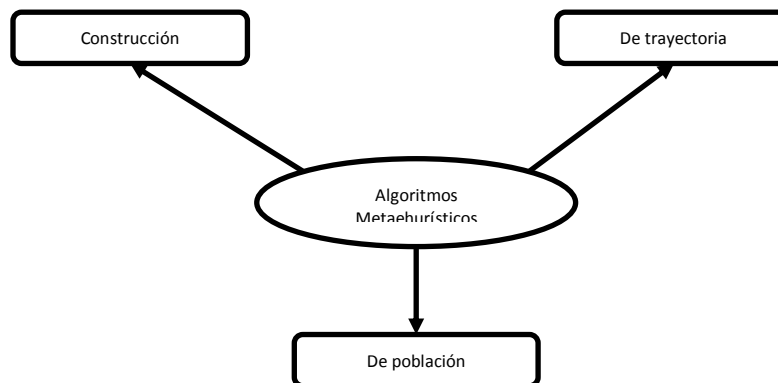


Figura 2-1 Tipos de algoritmos metaheurísticos.

Metaheurísticas de construcción

Son aquellos que no parten de una solución inicial, pues a lo largo de la búsqueda la van construyendo hasta hallar el mejor valor. Ejemplos: **GRASP**, optimización basada en colonia de hormigas.

Metaheurísticas basadas en trayectorias

Parten de una solución inicial ya conocida y la mejoran a lo largo de la búsqueda para hallar un mejor valor. Ejemplos: recocido simulado, búsqueda tabú, búsqueda local.

Metaheurísticas de población

Evalúa una población de soluciones iterativamente, ejemplos: algoritmos genéticos, recocido simulado.

Bullnheimer explica que los *algoritmos de búsqueda tabú* ofrecen mejores rendimientos y resultados que los algoritmos recocidos simulados y los algoritmos de colonias de hormigas, tanto en tiempo de ejecución como en consistencia de resultados. Estos análisis fueron obtenidos al realizar pruebas a un algoritmo **VRPTW**, con diferentes algoritmos, en el cual la búsqueda tabú obtuvo un 75% de mejor rendimiento que los algoritmos mencionados anteriormente.

Por otra parte, entre las desventajas de la utilización de los algoritmos de tipo constructivo es que no cuentan con la capacidad de cambiar o mejorar aquella parte de la solución que ya ha sido creada, y por lo tanto no le ofrece al algoritmo una flexibilidad en la búsqueda.

Tabla 2-1 Características de los diferentes algoritmos

ALGORITMOS METAHEURÍSTICOS	HEURÍSTICAS DE TRAYECTORIA	ALGORITMOS DE POBLACIÓN	ALGORITMOS DE CONSTRUCCIÓN
<p>Objetivo: Explorar eficientemente el espacio de búsqueda, de forma de encontrar soluciones (sub)óptimas.</p> <p>Son estrategias para guiar los procesos de búsqueda.</p> <p>Incorporan mecanismos para evitar óptimos locales.</p> <p>Amplio espectro, desde técnicas sencillas como búsqueda local hasta técnicas complejas (Procesos de aprendizaje)</p> <p>No son específicas al problema que intentan resolver, incorporan el conocimiento específico del problema o la experiencia (Memoria) desviando la búsqueda.</p>	Malos exploradores	Buenos exploradores	Debido a que, como su nombre lo dice, construyen la solución a lo largo de la búsqueda, les es imposible corregir lo ya construido, o buscar mejores alternativas en otras regiones.

Con base en la tabla anterior se ha decidido elegir el *algoritmo de búsqueda tabú* debido a que se desea obtener una mejor precisión a lo largo de la búsqueda y en el resultado final. Por otra parte, como se mencionó, los *algoritmos de búsqueda tabú* ofrecen un mejor desempeño en los problemas de **VRPTW** que los algoritmos de origen poblacional.

Otra razón por la cual se decidió realizar este trabajo de asignación de vehículos a rutas con ventanas de tiempo, con base en un *algoritmo de búsqueda tabú* (TS) fue por la lectura del artículo sobre el tema: transporte de materiales peligrosos. Este artículo fue presentado en el *symposium* de computación aplicada en 2005. Aquí, los investigadores buscan aproximarse al *óptimo* por medio de un heurístico TS. (Zhang, Zuo, & Lim, 2005) Anteriormente, este tipo de problemas se habían solucionado con los siguientes algoritmos:

- Multi-objetivo: minimización del riesgo, daños, tiempo de viaje.
- Algoritmo de “Branch and Bound”.
- Problema de flujo con costo mínimo.
- Problema del ruteo de vehículos y calendarización, entre otros.

Sin embargo, estos modelos, al parecer, no proporcionaron una solución óptima, de acuerdo a Zhang, *et al.* (2005), por lo que decidieron tratar con un **TS** obteniendo, no un óptimo, pero sí superando los antiguos heurísticos y acercándose mucho al idóneo.

Por lo anterior, la búsqueda quedó aún más motivada, pues se parte de un algoritmo ya utilizado, en este tipo de problemas, y comprobado con una mejora eficiente de soluciones.

CAPÍTULO 3. DESCRIPCIÓN DEL ALGORITMO

3.1 El algoritmo

Recordando que el problema abordado en este tema de investigación tiene como principal objetivo: determinar el número mínimo de vehículos necesarios para atender a la demanda de 12 centros de distribución. Y con base en lo presentado en el capítulo anterior, se apuntar que, el proceso para encontrar soluciones factibles al problema de la serie **VRP**, es muy complicado (Bodin *et al*, 1983; Golden & Asad, 1986; Solomon, 1988; Laporte, 1992; Desrochers *et al*, 1992 y Ainily, 1999). Por lo tanto, se propone resolver el problema haciendo uso del *algoritmo de búsqueda tabú* y mostrar una aplicación real del mismo. En este capítulo se describe de manera detallada, y a modo de ejemplo, la estructura del algoritmo propuesto, que incorpora un procedimiento glotón de búsqueda local por medio de inserciones aleatorias de arcos para generar un conjunto de soluciones, que con ayuda de los mecanismos del *algoritmo de búsqueda tabú* permita determinar un número mínimo de vehículos para cumplir con una demanda determinada.

Fred Glover & Laguna (1997) establecen que el *algoritmo de búsqueda tabú* está basado en métodos de mejora. Es decir, se traslada de la vecindad de una solución conocida a otra, por medio de intercambios, buscando un valor que minimice o maximice el valor de una función objetivo, bajo los principios de arranque o terminación del mismo. Del conjunto de estas vecindades es seleccionada aquella que ofrece un mejor valor a la función objetivo, de manera que es almacenada en una pila llamada *lista tabú*, donde, a partir de ella, se evitan posibles ciclos durante un tiempo determinado, en otras palabras, se retome la misma solución.

A lo largo de la búsqueda es posible que el algoritmo encuentre peores o mejores soluciones, pero este proceso le permitirá salir de óptimos locales con una mayor facilidad mientras busca a lo largo de las vecindades. La búsqueda tabú evita los ciclos, marcando las soluciones, para impedir regresar a ellas almacenándolas en una *lista tabú*, mencionada anteriormente, que dependiendo de su tamaño y características permitirá la búsqueda de manera más eficiente a través de sus elementos.

3.1.1 Breve descripción del algoritmo

El algoritmo que se mostrará, a continuación, es un algoritmo que contiene de forma básica los conceptos del *algoritmo de búsqueda tabú*. La característica más significativa consiste en incorporar la *lista tabú* y un proceso para diversificar la búsqueda, que consiste en regresar a una solución inicial para iniciar nuevamente el algoritmo. El proceso mostrado a continuación, describe a este algoritmo:

Generar una solución inicial X_0 .

La solución se genera por medio de un algoritmo glotón seleccionando una salida de forma aleatoria X_0 del conjunto de soluciones S_0 (soluciones iniciales).

Inicializar:

Lista tabú (T) con $k = 0$ elementos.

t (Contador de elementos en lista tabú = 0)

Parámetro máximo de elementos permitidos en lista tabú, al llegar a este valor, se eliminará el elemento más antiguo para permitir el acceso a un nuevo elemento.

A (Cantidad de candidatos por iteración en el algoritmo tabú).

B (Contador de iteraciones seguidas en las que el valor de $F''(x)$ no es mejorado) al llegar a su valor máximo se reiniciará el algoritmo en el paso 1.

Hacer $F''(x) = F(x)$.

El valor de solución inicial X_0 , se nombra como el mejor valor de F al iniciar el algoritmo.

Mientras la condición de finalización no se encuentre:

Identificar la vecindad $N(x)$

La vecindad se generará por medio de la eliminación de un vehículo seleccionado al azar e insertando sus viajes en los vehículos restantes. Cada solución o intercambio se almacenará en una memoria interna llamada $S1$. Se tendrán tantas $S1$ como valores A estén establecidos.

Se comparará el valor de $F''(x)$ con el valor después de realizar el intercambio, si:

$F''(x)$ es mayor al valor de F encontrado después del paso 2.1, entonces buscar otra $N(x)$ y aumentar en una unidad el valor de B .

$F''(x)$ es menor al valor de F encontrado después del paso 2.1, entonces actualizar el valor de $F''(x)$ con este nuevo valor y guardar el movimiento en la *lista tabú*, aumentar en una unidad el valor de T .

Si cada solución $F''(x)$ es verdadera entonces utilizar las características de esta solución para repetir el paso 2.1 hasta que se cumpla el paso 2.1.1.2.

Repetir el procedimiento hasta que se llegue al número máximo de iteraciones.

3.1.2 Supuestos

Para llevar a cabo la descripción del algoritmo es necesario establecer los siguientes supuestos:

1. Todos los viajes realizados por los vehículos inician y terminan en un mismo origen, denominado almacén central o planta.

2. Todos los vehículos que atienden al conjunto de CD's tienen la misma capacidad, es decir, son parte de una flota homogénea.
3. Se establece que el tiempo máximo de carga, en planta para un vehículo, es de veinte minutos. Este dato obliga a ajustar el horario de operación de veinticuatro horas a setenta y dos, en períodos de 20 minutos.
4. En planta no se permite llevar a cabo dos operaciones de carga en el mismo período.
5. En cada CD están predefinidos uno o varios intervalos de tiempo (ventanas de atención) para llevar a cabo operaciones de carga/descarga de producto.
6. Todos los viajes asignados a cada uno de los CD's deberán llevarse a cabo dentro de las ventanas de atención.
7. Cada CD puede ser visitado varias ocasiones durante el período de planeación (72 períodos y dentro de su ventana de atención).
8. Todos los viajes demandados por los 12 CD's se deben realizar siempre y cuando no excedan los 72 períodos.
9. Cada vehículo que es asignado dispone de un número máximo de 72 períodos de 20 minutos para realizar los viajes.
10. La activación de un vehículo se da en el momento en el que se realiza su primer viaje a cualquier CD.
11. El criterio para la secuenciación de vehículos se basa en el principio de la minimización de sus períodos de ocio.
12. La ruta para cada vehículo es la secuencia de viajes a CD's, utilizando total o parcialmente sus períodos disponibles.
13. El orden de asignación y de acomodo se basará siempre en dar prioridad a un CD con un menor ancho de ventana, es decir, si existe un arco asignado a un CD con un menor ancho de ventana y se asigna otro viaje, siempre se dará prioridad a colocar en orden descendente el arco con menor ventana seguido del de mayor ventana de atención.
14. El objetivo es determinar el número mínimo de vehículos para atender la demanda de los doce centros de distribución.

3.2 Algoritmo glotón-tabú

La aplicación, a la que se ha dado el nombre de glotón-tabú, se encuentra conformada por dos grandes algoritmos integrados con la finalidad de organizar el contenido de centros de carga (CD) dentro de ventanas de atención (**VATN**). Para obtener los resultados que se procesan con esta aplicación, se hace uso de un medio de almacenamiento intermedio de datos (Base de Datos) para el cual se ha elegido la edición SQL Server 2005 Express por su versatilidad y por ser una aplicación robusta derivada de la aplicación comercial, completa, de este producto; con la ventaja de ser compacta y, sobre todo, de distribución libre. La aplicación se vale de una pequeña interfaz en la que se encuentran agrupados tres grupos de parámetros, con los que los algoritmos parten, para lograr el objetivo del programa. En la siguiente imagen se muestra dicha interfaz y en seguida se hace una breve descripción de sus componentes:

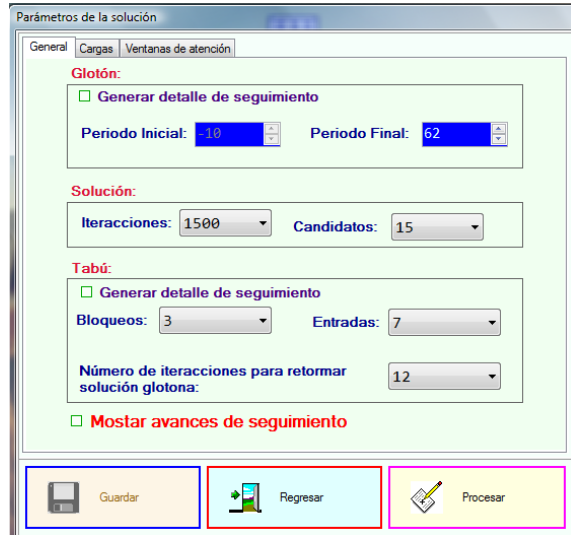


Figura 3-1 Interfaz gráfica de los parámetros de la aplicación

PESTAÑA GENERAL: **Grupo glotón.** En este grupo se puede seleccionar el período inicial y período final de las ventanas de atención (por el momento se establecieron en 10 y 62, respectivamente; por ser estos los parámetros reales del problema que se quiere resolver). Posteriormente se prevé, para posibles cambios futuros de la aplicación, hacerlos modificables. La selección de estos períodos, inicial y final, dependen del ancho del horizonte de planeación del algoritmo. El cual representa un día laboral, es decir, 24 horas. Los horarios de los centros de atención se muestran a continuación.

Tabla 3-1 Horarios reales de las ventanas de atención

CENTRO DE DISTRIBUCIÓN	HORARIO DE VENTANA
A	7:30 AM - 2:30 PM
	8:30 PM - 9:30 PM
B	7:00 AM - 3:00 PM
C	7:00 AM - 3:00 PM
D	7:00 AM - 4:00 PM
	7:00 AM - 3:00 PM
E	8:00 PM - 10:00 PM
F	8:00 AM - 3:00 PM
	7:00 PM - 11:00 PM

Tabla 3-2 Continuación de tabla 3-1

CENTRO DE DISTRIBUCIÓN	HORARIO DE VENTANA
G	8:00 AM - 5:00 PM
	12:00 PM - 4:00 AM
H	8:00 AM - 3:00 PM
	10:00 PM - 5:30 AM
I	7:00 AM - 3:00 PM
	10:30 PM - 5:00 AM
K	8:00 AM - 11:00 AM
	3:00 PM - 6:00 PM
L	8:00 AM - 4:30 AM
	8:00 AM - 2:00 PM
	4:00 PM - 9:00 PM
	11:00 PM - 4:30 AM

Tabla 3-3 Ventanas de atención ajustadas al algoritmo de González (2004)

CENTRO DE DISTRIBUCIÓN	HORARIO DE VENTANA
A	7:40 AM - 2:20 PM
	8:40 PM - 9:20 PM
B	7:00 AM - 3:00 PM
C	7:00 AM - 3:00 PM
D	7:00 AM - 4:00 PM
	7:00 AM - 3:00 PM
E	8:00 PM - 10:00 PM
F	8:00 AM - 3:00 PM
G	7:00 PM - 11:00 PM
H	8:00 AM - 5:00 PM
	12:00 PM - 4:00 AM
I	8:00 AM - 3:00 PM
	10:00 PM - 5:20 AM
J	7:00 AM - 3:00 PM
	10:30 PM - 5:00 AM
	8:00 AM - 11:00 AM

Tabla 3-4 Continuación de tabla 3-3

CENTRO DE DISTRIBUCIÓN	HORARIO DE VENTANA
J	3:00 PM - 6:00 PM
	11:00 PM - 6:00 AM
K	8:00 AM - 4:20 AM
	8:00 AM - 2:00 PM
L	4:00 PM - 9:00 PM
	11:00 PM - 4:20 AM

Sin embargo, con el objetivo de parear los valores obtenidos por Manuel González (2004) se hace uso del mismo horario de atención.

Como se observa, los anchos de ventana, reales, no corresponden de manera directa a los utilizados en este algoritmo, debido a la realización de un estudio, previo, de tiempos y movimientos para estandarizar los tiempos de recorrido, carga y descarga. Después del estudio, se llegó a la conclusión de que los tiempos deberían estandarizarse a períodos de 20 minutos, tiempo requerido para realizar una carga en un camión de tipo Torton en el almacén central. De esta manera, si se divide un día, es decir 24 horas en períodos de 20 minutos, se tendría un total de 72 períodos, los cuales se utilizan para representar los horarios de ventana de atención. Para aquellos CD's cuyo inicio se presenta, por ejemplo, a las 7:30 su carga sería a las 7:40; de igual manera si un ancho de ventana termina a las 2:30, su equivalente en períodos de 20 minutos sería de 2:20. De esta forma, no se penalizan las cargas fuera de período, sino se asegura que todas estén dentro del período de atención. Obteniéndose así, la tabla de horarios de atención mostrada en el anexo de ventanas de atención.

Se hace uso de una variable llamada "soluciones iniciales" que corresponde a la cantidad de soluciones glotonas iniciales, que se generaran, para ser utilizadas dentro del *algoritmo de búsqueda tabú*. Las soluciones elegidas de forma aleatoria serán utilizadas para establecer la tasa interna de recuperación de solución glotona dentro del *Tabú Search*. Esta aplicación se encuentra diseñada para permitir el seguimiento del algoritmo, mediante la generación de archivos de texto intermedio con formato, que es reconocido por una macro diseñada en Excel, para convertir a una representación gráfica la información, así como un archivo de detalle de cada etapa del algoritmo. La aplicación se activa o desactiva mediante la caja de verificación que se encuentra en la parte superior izquierda del grupo de parámetros.

Grupo solución y grupo tabú. En este grupo se encuentran sólo dos parámetros, mismos que sirven para indicar a la aplicación la cantidad de iteraciones que se ejecutarán en el proceso de verificación dentro del algoritmo de *búsqueda tabú*. Por otra parte el parámetro *candidatos* indicará la cantidad de vehículos que se considerarán por interacción. A partir de dicha verificación se escogerá al que mejore, el mejor valor encontrado hasta el momento, es decir, se realizará una validación de la siguiente manera:

1. Si el valor de la función objetivo, de la iteración, es menor al mejor valor encontrado, entonces se realizará otra iteración.
2. Si el valor de la función objetivo, de la iteración, es mayor al mejor valor encontrado hasta el momento, entonces su valor se identificará como la mejor solución encontrada hasta el momento, y su solución se almacenará en una lista de memoria a corto plazo llamada *Lista Tabú*. Los valores permanecerán en esta lista, con la cantidad de iteraciones indicadas en el valor *Bloqueos*. El valor *Entradas* indicará, a su vez, la cantidad de elementos que podrán almacenarse como máximo en esta lista. Al llegar al valor máximo, y hallar otro valor, al ingresar, saldrá de la lista el valor con más antigüedad en ella.
3. El indicador de número de iteraciones antes de regresar a una solución glotona, hará que el algoritmo, después de no hallar una mejor solución dentro de una vecindad, brinque a otra, regresando a una solución glotona diferente, para iniciar nuevamente el proceso de búsqueda.

PESTAÑA CARGAS:

Dentro de la ventana de selección de parámetros, se encuentra una segunda pestaña bajo el nombre de "Cargas", como se muestra en la figura 3.1. En esta retícula de opciones se muestra una línea por cada uno de los centros de distribución (CD), en seguida, se muestra la demanda con la que se quiere asociar al CD correspondiente, y en la última columna se encuentra una lista de selección de períodos asociados al centro de distribución correspondiente. Esta ventana de períodos indica la cantidad de períodos conformados por los siguientes tiempos:

1. Carga
2. Viaje de recorrido del almacén central al centro de distribución X
3. Descarga en el centro de distribución X
4. Viaje de regreso del centro de distribución X al almacén central

Esta cantidad de períodos, como se mencionó anteriormente, es la cantidad de períodos de 20 minutos cada uno. Dicha información permite crear los *Objetos Arcos* que se utilizarán a lo largo del algoritmo. Es decir, existirán tantos arcos como demandas hayan a un centro de distribución, obteniéndose con su suma, la demanda total.

CD	Demanda	Periodos
CD01	6	25
CD02	6	14
CD03	1	26
CD04	4	20
CD05	8	14
CD06	6	11
CD07	5	17
CD08	1	10
CD09	2	16
CD10	7	10
CD11	6	12
CD12	13	5

Figura 3-2 Pestaña para introducir el modelo de Cargas en el software.

PESTAÑA VENTANAS DE ATENCIÓN:

En esta pestaña se muestra una tabla cuyo número de renglones corresponde a la cantidad de períodos que se han seleccionado en la *PESTAÑA GENERAL* y con el número de columnas, igual al número de centros de distribución, que se están considerando para la evaluación de los algoritmos *glotón* y *búsqueda tabú*. La intersección de esta matriz está conformada por cajas de verificación elegibles por simple botón del dispositivo apuntador (mouse o tablet) y aquellas que queden ser seleccionadas con una palomilla se considerarán como parte de los períodos que serán asociados al centro de distribución en turno. La figura siguiente es una extracción de dicha pestaña. Las ventanas completas de cada centro de distribución, puede hallarse en el Anexo número 8.

Periodo	CD01	CD02	CD03	CD04	CD05	CD06	CD07	CD08	CD09	CD10	CD11	CD12
-10	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
-9	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
-8	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
-7	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
-6	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
-5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
-4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
-3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
-2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
-1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Figura 3-3 Pestaña ventanas de atención.

En la parte baja, de la ventana de parámetros, se encuentran 3 botones de significado intuitivo: *guardar*, *regresar* y *procesar*, cuya finalidad y particularidad se mencionan a continuación:

Guardar. Esta opción permite guardar en la base de datos los parámetros con que operará la ejecución de la aplicación. Es importante señalar que este botón permanecerá desactivado mientras no se modifique alguno de los parámetros, si es el caso, el botón se activará y no se podrá iniciar el procesamiento del algoritmo hasta que no se guarden los cambios efectuados.

Regresar. Este botón se utiliza para terminar la ejecución de la aplicación o simplemente si no se ejecuta el proceso, regresar el control al Sistema Operativo.

Procesar. Éste se encarga de iniciar la ejecución de los algoritmos *glotón* y *búsqueda tabú*. Hay que considerar que para iniciar la ejecución, primero se deberán guardar los cambios efectuados a los parámetros de la aplicación, si es que los hay. En cualquier caso, si no se hace, se pedirá al usuario que lo haga, primero, antes de continuar.

3.2.1 Esquema general de la aplicación

En el siguiente esquema se presentan los componentes de la aplicación. Aquí se aprecia que todas las operaciones, de entrada y salida, entre la interfaz de la aplicación y los

algoritmos se encuentran centradas en la *base de datos* cuyo nombre es SOLUCIÓN. En ésta, como ya se ha indicado, se guardan los parámetros de configuración de la aplicación: las soluciones glotonas iniciales y las soluciones que se utilizan por interacción. Éstas se toman de la *base de datos* así como las soluciones que mejoran el resultado óptimo por interacción y se vuelven a guardar en la *base de datos*, al mismo tiempo que se guardan en la ventana de avance del proceso y se actualiza en la ventanilla indicadora de avance. Al final del proceso se toma, de manera aleatoria, la solución que será dada como la mejor solución de la corrida. Los resultados se dan en formato de archivo plano, en forma texto, que pueden graficarse mediante la macro en Excel antes citada.

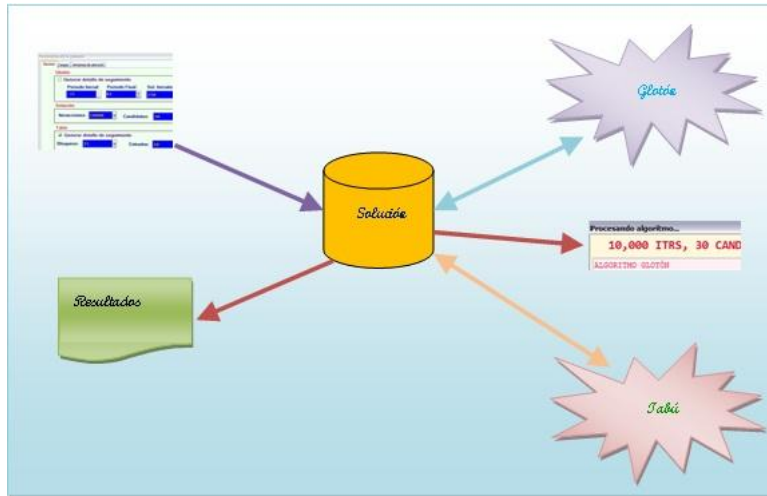


Figura 3-4 Funcionamiento de la programación del algoritmo

Al final de la una ejecución de la aplicación en la ventanilla de avance, aparece un texto que muestra la cantidad de camiones, de la solución óptima obtenida, el tiempo, que tardó en ejecutarse el algoritmo, y los resultados, así como los archivos intermedios generados, cuando así se indiquen, se guardarán en la carpeta: “Mis documentos/Glotón”

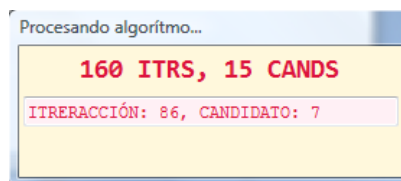


Figura 3-5 Representación de las iteraciones.

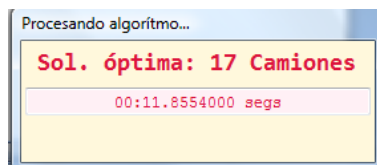


Figura 3-6 Representación de la solución óptima.

Los archivos generados para los propósitos indicados son los siguientes:

MODELO DE INTERACCIÓN XXXXX.modelo. Contiene el archivo de formato texto plano diseñado para que la macro de Excel, ya indicada, para que lo interprete como un modelo intermedio de alguna interacción.

PROCESO DE ITERACIÓN XXXXX.txt. Es el archivo de seguimiento de la ejecución del algoritmo.

SOLUCIÓN ÓPTIMA.txt. Es un archivo en formato texto plano que indica la representación del modelo óptimo seleccionado para la solución de la corrida de la aplicación.

SOLUCIONES ÓPTIMAS.txt. Es un archivo que contiene la representación de todas las soluciones óptimas obtenidas en la ejecución de la aplicación.

SOLUCIONES.final. Es un archivo de texto plano con la representación de la solución óptima elegida y reconocible por la macro de Excel que se encarga de hacer la representación gráfica de la solución.

ESTADÍSTICAS GLOTONAS.csv. Archivo de texto plano con formato delimitado por comas y comillas que contiene los detalles de la ejecución de la aplicación para que pueda ser manipulada en Excel para una mejor interpretación.

INTERCAMBIOS TABÚ.csv. Archivo que contiene la lista de todas las operaciones de evaluación e intercambios realizadas durante toda la corrida de la aplicación con los parámetros establecidos, con el formato de la tabla *Tabú* en formato delimitado por comas y comillas que pueden ser importados en Excel.

LISTA TABÚ.csv. Archivo que contiene la lista *Tabú* de toda la ejecución de la aplicación en formato delimitado por comas y comillas para que, de igual forma al caso previo, se pueda manipular en Excel.

LOG.csv. Este archivo es similar a la tabla *Tabú*, con la diferencia que indica qué tipo de operación se realizó en el modelo, el candidato elegido por interacción, así como los intercambios que se realizaron: tipo de acción, origen y destino de vehículo y períodos, valor de la función objetivo y la carga que se repite en lista *Tabú*. Se encuentra en formato delimitado por comas y comillas con la misma finalidad que los archivos anteriores.

3.2.2 Algoritmo glotón

Una técnica muy sencilla de resolución de problemas (que, lamentablemente, no siempre da resultado) es la de los algoritmos voraces o glotones. Se trata de algoritmos que resuelven problemas de optimización, es decir, tenemos un problema que se quiere resolver de manera óptima: el mejor camino que une dos ciudades, el valor máximo alcanzable seleccionando ciertos productos, el costo mínimo para proveer un cierto servicio, la menor cantidad de efectivo para pagar un cierto importe, etc. Los algoritmos

glotones, lo que hacen es construir una solución, paso a paso, para generar al final una solución factible al problema.

Pero para el algoritmo bajo estudio, el criterio de factibilidad de una solución, está definido por:

FACTIBILIDAD DE UNA SOLUCIÓN

Una solución será factible, si y sólo si:

- La demanda total es cumplida al cien por ciento
- Cada asignación de un arco, cumplirá con las ventanas de atención de cada centro de atención.

En esta aplicación se busca acomodar los centros de distribución (CD) establecidos en los parámetros iniciales, considerando el criterio de factibilidad de una solución, para proveer soluciones de partida al *algoritmo búsqueda tabú*.

CREACIÓN DE LA CANTIDAD DE SOLUCIONES

Debido a que el *algoritmo tabú* conlleva diferentes parámetros es importante destacar que la cantidad de soluciones tabú generadas dependerán de la cantidad de veces que se le solicita al algoritmo regresar a una solución inicial, en otras palabras, la ecuación para crear soluciones iniciales es la siguiente:

$$\text{Cantidad de soluciones glotonas} = \frac{A}{B},$$

Dónde:

A= número total de iteraciones tabú.

B = cantidad de veces que se desea regresar a una solución glotona.

3.2.3 Implementación del algoritmo

3.2.3.1 Algoritmo glotón

Toda la aplicación que se describe fue elaborada en Visual Basic.NET 2005 por ser una plataforma de desarrollo de software que aprovecha la potencialidad con que se ha provisto a los Sistemas Operativos de Microsoft y porque ofrece una amplia serie de clases y objetos, que se encuentran agrupados de acuerdo al enfoque de las tareas que realizan en conjunto, es decir, tiene componentes optimizados para sacar el máximo provecho de los recursos físicos y lógicos de un equipo de cómputo.

En la implementación del algoritmo

Entradas:

Demanda: corresponde a la cantidad de viajes asignados a cada centro de distribución (CD) en los parámetros de configuración de la aplicación.

Ventanas de Atención: Son los períodos asignados a cada centro de distribución, cuyos valores son establecidos en la ventana principal de asignación de parámetros.

Procedimiento:

Al iniciar el algoritmo se crea una tabla bidimensional que representará el modelo de solución inicial en el que cada entrada, en la primera dimensión, representa a cada uno de los camiones y, a su vez, cada una de éstas representará las cargas que son asignadas en el camión en turno; tomando en cuenta, siempre, las ventanas de atención y los criterios para evitar traslapes y desplazamientos de cargas.

Tabla 3-5 Representación del horizonte de planeación y vehículos

PERIODO	CAMIÓN 1	...	CAMIÓN N
-10			
-9			
-8			
-7			
-6			
-5			
-4			
-3			
-2			
-1			
0			

Tabla 3-6 Continuación de tabla 3-3

PERIODO	CAMIÓN 1	...	CAMIÓN N
1			
2			
3			
4			
5			
6			
7			
8			
.			
.			
.			
62			

Conforme se va ejecutando el procedimiento del algoritmo, simultáneamente se va creando un archivo de seguimiento de la cantidad de intentos por generar una solución inicial, que se tuvo que efectuar para ser considerado como una solución válida de inicio.

Para generar una solución inicial, de la ejecución del algoritmo, se consideran dos listas auxiliares que nombraremos Modelo A (Vector Pila) y Modelo B. El modelo A se utiliza para tener, en él, todos los Arcos con sus respectivos atributos, que serán tomados aleatoriamente, para colocarse en el camión en turno y evitar que, los arcos que se han evaluado, sean colocados en el camión sin haber cumplido con las condiciones de prioridad, así como con los períodos de las respectivas ventanas de atención, y por lo tanto se vuelvan a incluir en el modelo A, para volver a intentar colocarlos. En este caso para evitar evaluaciones que conduzcan a ciclos indefinidos, se ha hecho uso del modelo B para evitar que, cuando se hayan determinado, no se pueda volver a acomodar un arco más y entonces se cree un camión nuevo para repetir el proceso; donde los arcos que se encuentran en el modelo B se agregan a los que quedan en el modelo A.

Para comprender mejor los siguientes pasos, se realizarán tres iteraciones mostrando los elementos y conceptos del algoritmo. Sin desarrollar el ejercicio completo, se podrá observar, a lo largo del algoritmo, que el proceso aleatorio no permitirá al lector obtener los mismos resultados mostrados a continuación, razón por la cual se ha omitido el ejercicio completo. El funcionamiento general del algoritmo puede hallarse en el anexo 11.

Supóngase la siguiente demanda:

Tabla 3-7 Demanda hipotética para mostrar el funcionamiento del algoritmo

CENTRO DE DISTRIBUCIÓN	DEMANDA
A	1
B	2
C	1
D	1
E	1
F	1
G	1
H	3
I	1
J	4
K	3

Paso 1. Se inicia el modelo A y el modelo B como listas vacías. El modelo A obtiene los valores, automáticamente, de la ventana de entrada donde se realizan las *cargas* de las demandas a cada centro de distribución (CD)

Paso 2. Se agrega al modelo A una representación de cada uno de los *arcos* definidos, mediante la selección de parámetros en la que cada arco se define de la siguiente manera:

ARCO: CD, período inicial, período final, ancho de ventana ó

ARCO: CD, período inicial, período final, ancho de ventana, referencia.

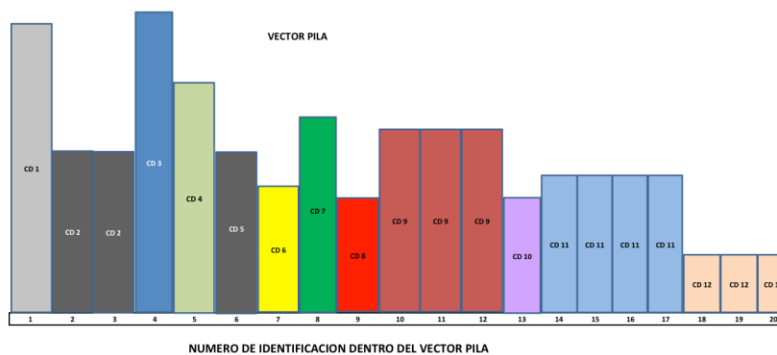


Figura 3-7 Representación gráfica del Vector Pila

En este caso, como se está creando un modelo a partir de la definición de los Arcos, los períodos iniciales serán siempre 0 y los finales se ajustarán al ancho de ventana, definido por el usuario, aplicado únicamente para el modelo A.

Paso 3. Se inicia al estado vacío el modelo de solución inicial.

Paso 4. Se inicia el contador de camión en turno a cero, también se utiliza una lista para determinar los ocios o huecos que quedan entre arcos, para ser evaluados, cuando se intenta insertar un arco nuevo dentro de la estructura del camión. Además, se utiliza una

pila para la evaluación temporal de la estructura del camión y se inicializa el estado de vacío.

Paso 5. Se incrementa en 1 el contador de camiones (en este momento es el camión 1) y en la estructura que representa al modelo se le agrega un camión vacío y asigna el número de contador que se encuentra en el momento.

Paso 6. Se verifica si en el modelo B hay arcos pendientes por procesar. Si hay arcos, entonces se mueve cada uno de los arcos en el modelo B hacia el modelo A.

Paso 7. Se establece el estado vacío al modelo B.

Paso 8. En la lista de ocios, se crea un camión para ir actualizando los ocios que quedan al acomodar los arcos.

Paso 9. Se incrementa en 1 al valor del contador *Intentos*.

Paso 10. Se inicia un bucle. Mientras haya algún arco, por procesar, en el modelo B o bien en el modelo A, todavía se encuentre algún arco por colocar, se realizan los pasos siguientes, en caso contrario, se pasa el control al Paso 11.

Paso 10.1. Se genera un número aleatorio cuyo valor se encuentre entre 0 y el número de arcos que se encuentran en el modelo A, menos 1 que servirá como el índice para el Arco candidato a acomodar (nota: en el lenguaje de programación, toda estructura que tenga comportamiento de lista o arreglo de objetos, se indexa a partir de 0).

Paso 10.2. Se verifica que haya arcos en el modelo A, si no hay entonces se realiza lo siguiente:

Paso 10.2.1. Se crea un camión nuevo (se incrementa el contador de camión y se anexa un camión vacío con el número recién incrementado en el modelo de la solución)

Paso 10.2.2. Se vuelve a generar un valor aleatorio para indexar al arco candidato.

Paso 10.2.3. Se verifica si hay más de un camión vacío en el modelo, de ser así quiere decir que se ha estado intentando acomodar los arcos que aún no han sido colocados, si éste es el caso se realiza lo siguiente:

Paso 10.2.3.1. Se desecha el modelo. Pues debido a su estructura ya no será posible hacer el acomodo, por lo que se vuelve a iniciar todo el proceso, conservando el valor del contador *Intentos* y el número de soluciones. Una vez realizado lo expresado, se continúa con el proceso normal de los arcos.

Paso 10.2.4. En la variable *strEsqAntes* se crea una representación, simbólica, de la manera en que se encuentra el modelo de solución en el momento. La variable indica que contiene una cadena con el esquema del modelo antes de comenzar a hacer cambios en el éste.

Paso 10.2.5. Se solicita al método *acomodar entrega*, al que se pasa como parámetro, el valor aleatorio, generado, que representa al arco seleccionado para ser acomodado

Paso 10.2.6. En la variable *strEsqDespués* se crea una representación, simbólica, de la manera en que se encuentra el modelo de solución en el momento. La variable al igual que en el caso del paso 10.2.4, alude a su contenido.

Paso 10.2.7. Se solicita al método *cabe otra carga*, al que se le proporciona como parámetro el número de camión de interés (en este caso es el camión en turno), preguntar si ya no cabe otra carga o Arco, si es así se sigue el siguiente paso, de lo contrario se continúa en el paso 10.2.8

Paso 10.2.7.1. Se pide el proceso de creación de otro camión: incremento del contador de camión y además se agrega un nuevo camión al modelo, con el contador actualizado.

Paso 10.2.8. El proceso continúa en el paso 10

Paso 11. Se incrementa el contador *Soluciones* en 1

MÉTODO ACOMODAR ENTREGA

Entrada: Índice de Arco Seleccionado (*Arco*)

Paso 1. En la variable *objCamión* se coloca la referencia al camión, en turno, dentro del modelo.

Paso 2. En la variable *objArco* se coloca la referencia física al arco que se ha seleccionado (vía parámetro *Arco*).

Paso 3. Se verifica: si hay Ocios en el camión en turno, de ser así se continúa con lo que sigue, de no ser así, se continúa con el Paso 3.2.

Paso 3.1. Para cada Ocio, dentro de la lista de ocios del camión en turno, se realizan los siguientes Pasos:

Paso 3.1.1. Se asigna a la variable *intPIBase* el valor del período inicial del ocio que se está verificando (la variable hace alusión a que se trata de una variable con valor numérico que contiene el período inicial base del ocio dentro del modelo).

Paso 3.1.2. De manera semejante, se procede para el período superior en la variable *intPSBase*.

Paso 3.1.3. Se busca si hay algún período libre, cercano al ciclo seleccionado al azar, dentro de los períodos inicial y final, validando que encaje dentro del período inicial global y final, así como el ancho de ventana, respectivo, para determinar si el arco puede encajar en alguno de los ocios primero.

Paso 3.1.4. Si el arco cabe en el ocio evaluado, entonces se hace el siguiente paso, si no, se pasa al Paso 3.1.5

Paso 3.1.4.1. Se mueve el periodo inicial del arco en turno hacia el período libre que se ha encontrado

Paso 3.1.4.2. Se agrega el arco en evaluación al camión en turno

Paso 3.1.4.3. Se quita el arco del modelo A.

Paso 3.1.4.4. Se actualiza la lista de ocios dentro del camión en turno, considerando el arco se incluyó en el paso 3.1.4.3

Paso 3.1.4.5. Se ordenan los arcos en el camión tomando como criterio de orden, los períodos iniciales de cada arco dentro del camión en turno.

Paso 3.1.4.6. Se sale del método y el flujo del programa continúa en el punto donde fue invocado el método

Paso 3.1.5. Se pasa al punto 3.1 para verificar el siguiente ocio en la lista.

Paso 3.2. Se vacía el contenido de la pila

Paso 3.3. Se inicia una verificación múltiple en relación al número de arcos dentro del camión en turno

Paso 3.3.1. Si no tiene arcos se realiza lo siguiente, de lo contrario se continúa con el paso 3.3.1.2

Paso 3.3.1.1. Se asigna a la variable *intPI* el primer valor disponible del CD en turno dentro de su ventana de atención.

Paso 3.3.1.2. Si hay un arco, se realiza lo que sigue; de no ser así se continúa en el Paso 3.

Paso 3.3.1.3. Se verifica si el ancho de ventana del arco, en turno, es menor al ancho de ventana del primer arco en el camión, además, si el primer período dentro del ancho de ventana, del arco en turno, es menor que el primer período de la ventana de atención del primer arco en el camión en turno, entonces se realiza lo que sigue; de no ser así, se continúa con el paso 3.3.1.4

Paso 3.3.1.3.1. Se asigna a la variable *objPrimerArco* la referencia del primer arco dentro del camión en turno

Paso 3.3.1.3.2. Se agrega a la pila la información del primer arco en el camión

Paso 3.3.1.3.3. Se asigna a la variable *intPI* el primer período disponible dentro del ancho de ventana del arco en turno

Paso 3.3.1.3.4. Continuar en el paso 3.3.1.5

Paso 3.3.1.4. Se asigna a la variable *intPI* el valor del período superior + 1 del primer arco en el camión en turno

Paso 3.3.1.5. Se continúa en el paso 3.

Paso 3.3.2. En caso de que haya 2 o más arcos entonces se realiza lo que sigue:

Paso 3.3.3. Se inicia un proceso de verificación sobre el Ancho de Ventana (AV) del arco en turno

Paso 3.3.3.1. Si es menor al ancho de ventana del primer arco en el camión en turno, se realiza lo que sigue, sino, se continúa en el Paso 3.3.3.

Paso 3.3.3.2. Si el primer período libre, dentro del ancho de ventana del arco en turno, es menor al primer período disponible dentro del ancho de ventana del primer arco, en el camión en turno, entonces se realiza lo que sigue, sino se continúa en el paso 3.3.3.

Paso 3.3.3.3. Se asigna -1 a la variable *intBase*.

Paso 3.3.3.4. Se asigna a la variable *intPI* el primer período disponible dentro del ancho de ventana del arco en turno.

Paso 3.3.3.5. Se continúa en el paso 3.3.4.

Paso 3.3.3.6. Se asigna a la variable *intBase* el valor de 0.

Paso 3.3.3.7. Se asigna a la variable *intPI* el valor del período superior + 1 del primer arco en el camión en turno.

Paso 3.3.4. Si el ancho de ventana del arco, en turno, es mayor o igual al ancho de ventana del último arco, en el camión en turno, se realiza lo que sigue, de lo contrario se continúa con el paso 3.

Paso 3.3.4.1. Se asigna a la variable *intBase* el valor entero más negativo.

Paso 3.3.4.2. Continuar en el paso 3.4.

Paso 3.3.4.3. En cualquier otro caso, se realiza lo que sigue:

Paso 3.3.4.4. Se hace un recorrido desde el penúltimo arco, en el camión, hacia el primero para lo que sigue:

Paso 3.3.4.4.1. Si el ancho de ventana del arco en turno es mayor o igual al ancho de ventana del arco que se está recorriendo y además, si el ancho de ventana del arco en turno, es menor que el arco superior al que se está recorriendo en su ancho de ventana, entonces se hace lo que sigue, sino se continúa en el paso 3.3.

Paso 3.3.4.4.1.1. Se asigna a la variable *intBase* el valor de la variable *intArco* (posición del arco en evaluación)

Paso 3.3.4.4.1.2. Continuar en el paso 3.3.4.6.

Paso 3.3.4.5. Continuar en el paso 3.3.4.4.

Paso 3.3.4.6. Se asigna a la variable *intPI* el valor del Período Superior + 1 del arco ubicado en la posición indicada en *intBase*. (En base a índice 0 como inicial).

Paso 3.3.4.7. Continuar en el paso 3.4.

Paso 3.3.4.8. Si el valor de *intBase* es diferente al valor más negativo, se realiza lo que sigue, de lo contrario se continúa en el Paso 3.4.

Paso 3.3.4.9. Se realiza el recorrido de todos los arcos en el camión en turno hasta la posición *intBase* +1, mediante la variable *intArco*.

Paso 3.3.4.9.1. Se asigna a la variable *objArcoEn*, la referencia al arco referenciado por la variable *intAr*.

Paso 3.3.4.9.2. Se coloca en la pila al arco referenciado con *objArcoEn* con sus atributos

Paso 3.3.4.9.3. Continuar en el paso 3.3.4.9.

Paso 3.3.4.10. Se asigna a la variable *intPI* el valor del período superior + 1 del último arco en el camión en turno.

Paso 3.4. Se agrega a la pila la referencia al Arco en turno.

Paso 3.5. Se solicita al método *Ejecutar Entregas*. Con parámetro Arco que representa al arco en turno.

Paso 4. Regresar al punto de llamada.

MÉTODO EJECUTAR ENTREGAS

Parámetro: arco a procesar

Paso 1. Se asigna a la variable *objCamión*, la referencia al camión en turno.

Paso 2. Se asigna a la variable *intPI* el valor del período inicial del último arco que se encuentra en la pila.

Paso 3. Se recorre la pila desde el último arco hasta el primero y se usa como índice la variable *intPila*.

Paso 3.1. Si el valor de la propiedad *secuencia* de la pila, asociado al índice *intPila* es negativo, se ejecuta lo siguiente, sino se continúa en el Paso 3.2.

Paso 3.2. Se asigna a la variable *objArco* la referencia al arco colocado en la posición actual de la pila.

Paso 3.3. Continuar en el paso 4.

Paso 3.4. Se asigna a la variable *objArco* el arco colocado en la pila indicado por la variable *intPila* y la propiedad *secuencia*.

Paso 4. Se busca si hay disponible un período de inicio cercano al arco en turno dentro de su ventana de atención, el período encontrado se coloca en la variable *intBase*.

Paso 5. Se actualiza la propiedad API del arco en turno indicado por la variable *intPila* con el valor de *intBase*.

Paso 6. Se verifica si el valor *intPSBase* es diferente a el valor más negativo (si se encontró lugar para colocar al arco) entonces se realiza lo que sigue, de lo contrario, se continúa en el paso 8.

Paso 7. Se asigna a la variable *intPI* el valor *intPSBase +1*

Paso 8. Continuar en el recorrido del paso 3.

Paso 9. Se verifica si el valor al que se mueve el *PeríodoSuperior* del arco referenciado al final de la pila es mayor al *PeríodoSuperior* del modelo o si el valor del *PeriodoInferior* referenciado al final de la pila es igual al valor más negativo (no puede colocarse el arco), entonces se realiza lo que sigue; de lo contrario, se continúa en el Paso 10.

Paso 9.1. Se limpia la pila.

Paso 9.2. Se agrega al modelo B el arco que se está referenciando en el modelo A.

Paso 9.3. Se quita del modelo A, el arco enlazado en el paso 9.2.

Paso 9.4. Se regresa al lugar de procedencia.

Paso 10. Se crea una lista temporal contenedora de arcos llamada *IstSalidas*.

Paso 11. Se inicia un ciclo para recorrer todos los arcos dentro de la pila en forma de lista FIFO con índice en la variable *intPila*.

Paso 11.1. Se asigna a la variable *intSecuencia* el valor de la propiedad *secuencia* en el arco de la pila indicado en la variable *intPila*.

Paso 11.2. Si el período de destino del arco que se encuentra en la posición indicada por la variable *intPila*, en la pila, es igual al valor más negativo o si el valor del período superior excede el período superior del modelo, entonces se continúa con los pasos subsecuentes, sino, se continúa en el Paso 11.

Paso 11.2.1. Se asigna a la variable *objArco*, la referencia al Arco en la pila que se encuentra indicado en la variable *intPila*.

Paso 11.2.2. El período inicial del arco en *objArco* se mueve temporalmente al valor más negativo.

Paso 11.2.3. Se agrega el arco en *objArco* a la lista de salida temporal.

Paso 12.2.4. Del camión en turno se quita al arco indicado en la variable *intSecuencia*.

Paso 12.2.5. Continuar en el paso 12.4.

Paso 12.3. Se inicia una comparación múltiple del valor de *intSecuencia*.

Paso 12.3.1. Si contiene el valor -1, continuar con los pasos siguientes, sino, en el paso 12.4

Paso 12.3.1.1. Asignar a la variable *objArco* el arco seleccionado para su procesamiento.

Paso 12.3.1.2. Mover al arco en *objArco* al período inicial que se encuentra en el arco de la pila en la posición actual en la propiedad API.

Paso 12.3.1.3. Agregar al camión en turno el arco referenciado en *objArco*.

Paso 12.3.1.4. Quitar de la lista temporal el arco del paso 12.3.1.3.

Paso 12.3.1.5. Ir al paso 12.4

Paso 12.3.2. En cualquier valor diferente continuar con los siguientes pasos

Paso 12.3.2.1. Asignar a la variable *objArco* la referencia al arco que se encuentra referenciado por la variable *intSecuencia* dentro del camión actual.

Paso 12.3.2.2. Dentro del camión en turno, mover el arco del paso anterior hacia el período que tiene referenciado en la pila

Paso 12.4. Se continúa con el ciclo iniciado en el paso 11

Paso 12.5. Si hay arcos en la lista *IstSalidas* entonces continuar con los pasos siguientes, sino, ir al paso 12.5.6.

Paso 12.5.1. Iniciar un ciclo

Paso 12.5.1.1. Asignar a la variable *bolSemáforo*, el valor True

Paso 12.5.1.2. Asignar a la variable *intOcio* el valor 0

Paso 12.5.1.3. Actualización de la lista de ocios.

Paso 12.5.1.4. Si no hay ocios en el camión en turno o si la lista *IstSalidas* no está vacía, continuar con los pasos, de lo contrario, continuar en el paso 12.5.6.

Paso 12.5.1.4.1. Asignar a la variable *bolSemáforo* el valor False.

Paso 12.5.1.4.2. Continuar en el paso 12.5.6.

Paso 12.5.1.5. Asignar a la variable *bolColocación* el valor False.

Paso 12.5.1.6. Iniciar un ciclo mientras *intOcios* es menor al total de ocios en el camión en turno, realizar los siguientes pasos:

Paso 12.5.1.6.1. Iniciar a *intArco* con 0.

Paso 12.5.1.6.2. Iniciar un ciclo mientras *intArco* sea menor que la posición del último arco en los ocios del camión en turno, realizar los siguientes pasos

Paso 12.5.1.6.2.1. *objArco* hace referencia al arco indicado en *intArco* dentro de la lista temporal de salida.

Paso 12.5.1.6.2.2. *intPIBase* se iguala con el período inicial del Ocio indicado por la variable *intOcio*.

Paso 12.5.1.6.2.3. *intPSBase* se iguala con el período final del Ocio indicado por la variable *intOcio*.

Paso 12.5.1.6.2.4. Buscar el período de inicio libre en su ancho de ventana cercano del arco indicado en *objArco*.

Paso 12.5.1.6.2.5. Si se encontró libre un período entonces continuar con los pasos, sino, ir al paso 12.5.1.7.

Paso 12.5.1.6.2.6. El arco referenciado se mueve al período localizado y establecido en *intPI*

Paso 12.5.1.6.2.7. Agregar el arco al camión en turno.

Paso 12.5.1.6.2.8. Quitar el arco de la lista de salida temporal.

Paso 12.5.1.6.2.9. Asignar a *bolColocación* el valor True.

Paso 12.5.1.6.2.10. Continuar en el paso 12.5.2.

Paso 12.5.1.6.3. Asignar False a la variable *bolColocación*.

Paso 12.5.1.7. incrementar en 1 a la variable *intArco*

Paso 12.5.1.8. Continuar con el ciclo en el paso 12.5.1.6.2.

Paso 12.5.2. Incrementar en 1 a la variable *intOcio*

Paso 12.5.3. Si es cierta la variable *bolColocación* entonces ir al paso 12.5.6.

Paso 12.5.4. Si *intOcio* es mayor que al contador de ocios en el camión, entonces continuar con los siguientes pasos, de lo contrario, ir al Paso 12.5.2.

Paso 12.5.4.1. Asignar a bolSemáforo el valor False

Paso 12.5.4.2. Continuar en el paso 12.5.6.

Paso 12.5.5. Continuar en el paso 12.5.1.6.

Paso 12.5.6. Continuar mientras bolSemáforo sea true.

Paso 12.6. Si aún hay arcos en la lista de salida entonces mover todos los arcos al modelo B.

Paso 12.7. Ordenar los arcos del camión en turno.

Paso 12.8. Vaciar la pila.

Paso 13. Si (Período Superior en el último arco del camión en turno) es menor que 5 o si no cabe otra carga en el camión, entonces continuar con los siguientes pasos, si no continuar con el paso o si el modelo A está vacío entonces, crear un camión nuevo.

Paso 14. Regresar al punto de llamada, es decir, se repetirá el método anterior hasta cumplir con la cantidad de iteraciones establecidas por los criterios definidos por el algoritmo.

La cantidad de soluciones glotonas se determinarán por el siguiente coeficiente.

3.2.3.2 Algoritmo de búsqueda tabú

El objetivo principal del *algoritmo de búsqueda tabú* radicará en minimizar la cantidad de vehículos, haciendo una suma de la cantidad de camiones presentados al finalizar la eliminación de un vehículo.

Paso 1. Seleccionar una solución Glotona de forma aleatoria del conjunto SOLUCIÓN generada previamente por el algoritmo Glotón.

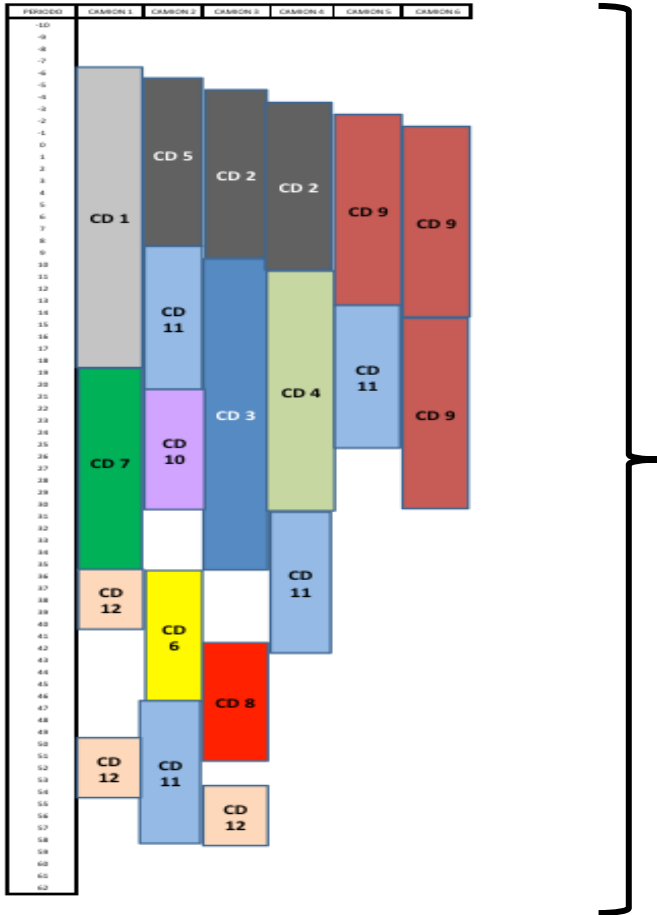


Figura 3-8 Intercambios tipo I

En la figura 3.8 está representado a modo de objetos, el período y el vehículo en el cual se realizará el viaje para cumplir con la demanda requerida.

Paso 2. La lista tabú se inicia con cero elementos. Se realiza un respaldo de la mejor solución encontrada hasta el momento.

Tabla 3-8 Representación de la Lista Tabú.

ITERACIÓN	DE			A			NUMERO DE ITERACIONES BLOQUEADAS	NUMERO DE ITERACIONES RESTANTES ANTES DE REGRESAR A UNA SOLUCIÓN INICIAL	VALOR DE LA ITERACIÓN
	CAMIÓN	PERIODO	CD	CAMIÓN	PERIODO	CD			

Paso 3. Almacenar el número de camiones de la solución seleccionada en el paso 1, en una variable llamada F^* , es decir $F^* = 6$ Camiones.

El objetivo del algoritmo se centrará en minimizar el valor de F a lo largo de la búsqueda.

Paso 4. Se seleccionará un vehículo al azar de la solución seleccionada en el paso 1.

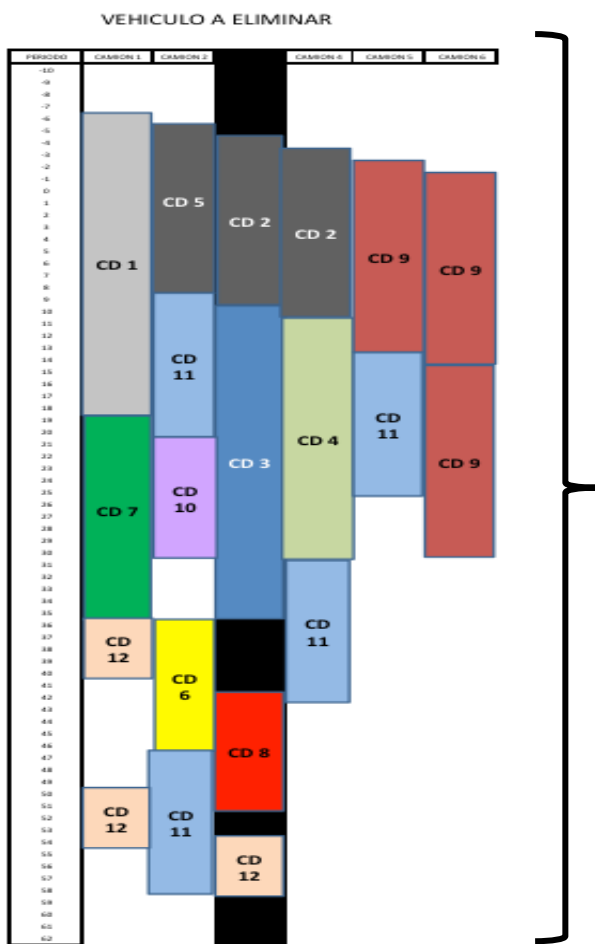


Figura 3-9 Selección aleatoria del vehículo a eliminar

Paso 5. Colocar sus arcos en el vector denominado Dummy.

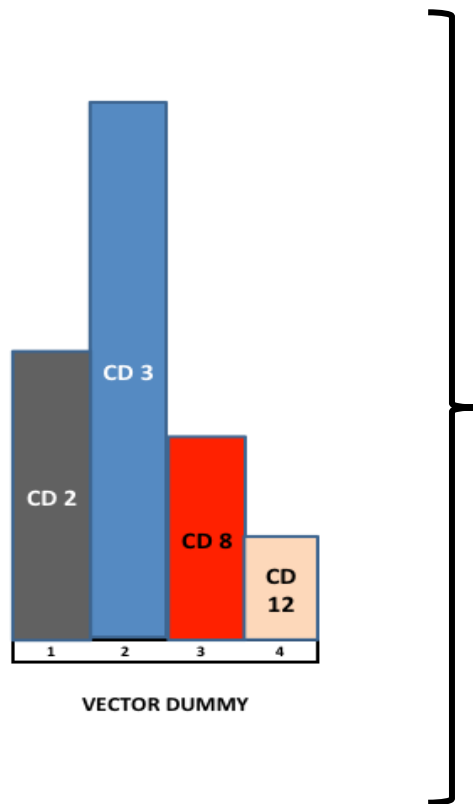


Figura 3-10 Representación de los arcos del vehículo a eliminar en el Vector Dummy

Este vector tendrá el mismo funcionamiento que el Vector Pila.

Paso 6. Las iteraciones normales se irán controlando por medio de una tabla parecida a la *lista tabú*, a excepción de la columna: iteraciones bloqueadas.

El objetivo será trasladar los arcos hallados en el vector Dummy, a alguno de los otros vehículos, haciendo uso de los períodos de ocio y funcionando de manera parecida al algoritmo glotón. A excepción de algunos intercambios agregados a fin de optimizar el mecanismo.

Paso 7. Seleccionar un arco de manera aleatoria del vector Dummy.

Paso 8, Seleccionar un camión y un período de manera aleatoria del modelo B o solución seleccionada en el paso 1.

Paso 9. Realizar los siguientes tipos de intercambios para colocar el arco. Si no es posible colocarlo en las posiciones seleccionadas, se creará otro vehículo para cumplir su demanda.

INTERCAMBIOS TABÚ, AGREGADOS A LOS YA EXISTENTES EN EL ALGORITMO GLOTÓN.

Se marcará con un círculo rojo los cambios significativos, sin embargo, el intercambio debe de ser factible para ambos arcos con respecto a sus ventanas de atención. Se marca con una columna roja el período seleccionado al azar para colocar el arco que se moverá del vehículo a eliminar.

INTERCAMBIO TIPO I

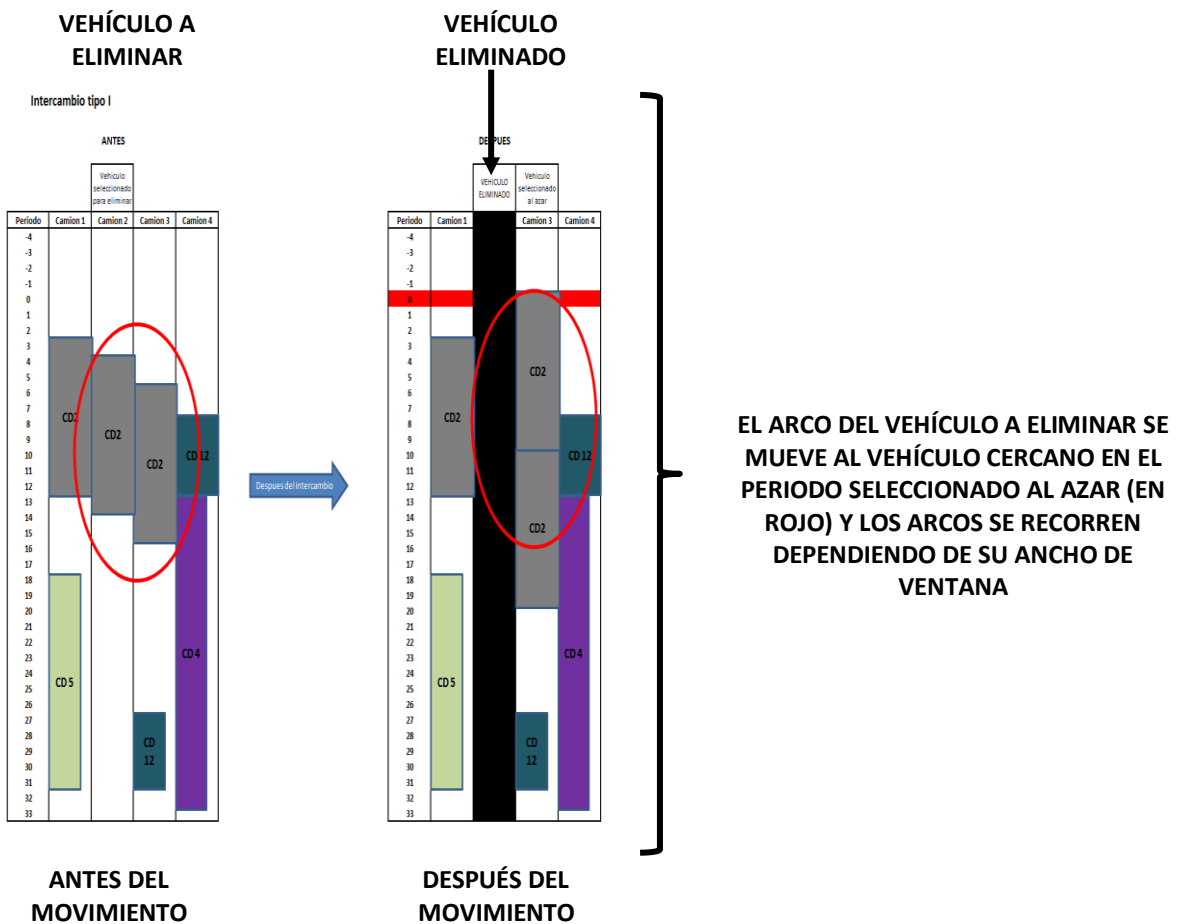


Figura 3-11 Intercambios tipo I

Se presenta cuando se selecciona al azar un período para el cual no existe carga y no se traslapa con ningún otro envío, como se muestran en los siguientes intercambios. En este caso, obsérvese la colocación de los envíos al final del intercambio, los cuales, muestran una secuencia de las cargas, es decir, al terminar un envío, inicia el próximo.

INTERCAMBIOS DEL TIPO II

En los intercambios, de tipo II, el período de carga seleccionado no coincide con el de ningún otro envío, sin embargo, el período esta en medio de las cargas de algún vehículo seleccionado. Es importante observar, cómo se presentan los cambios al final de los intercambios.

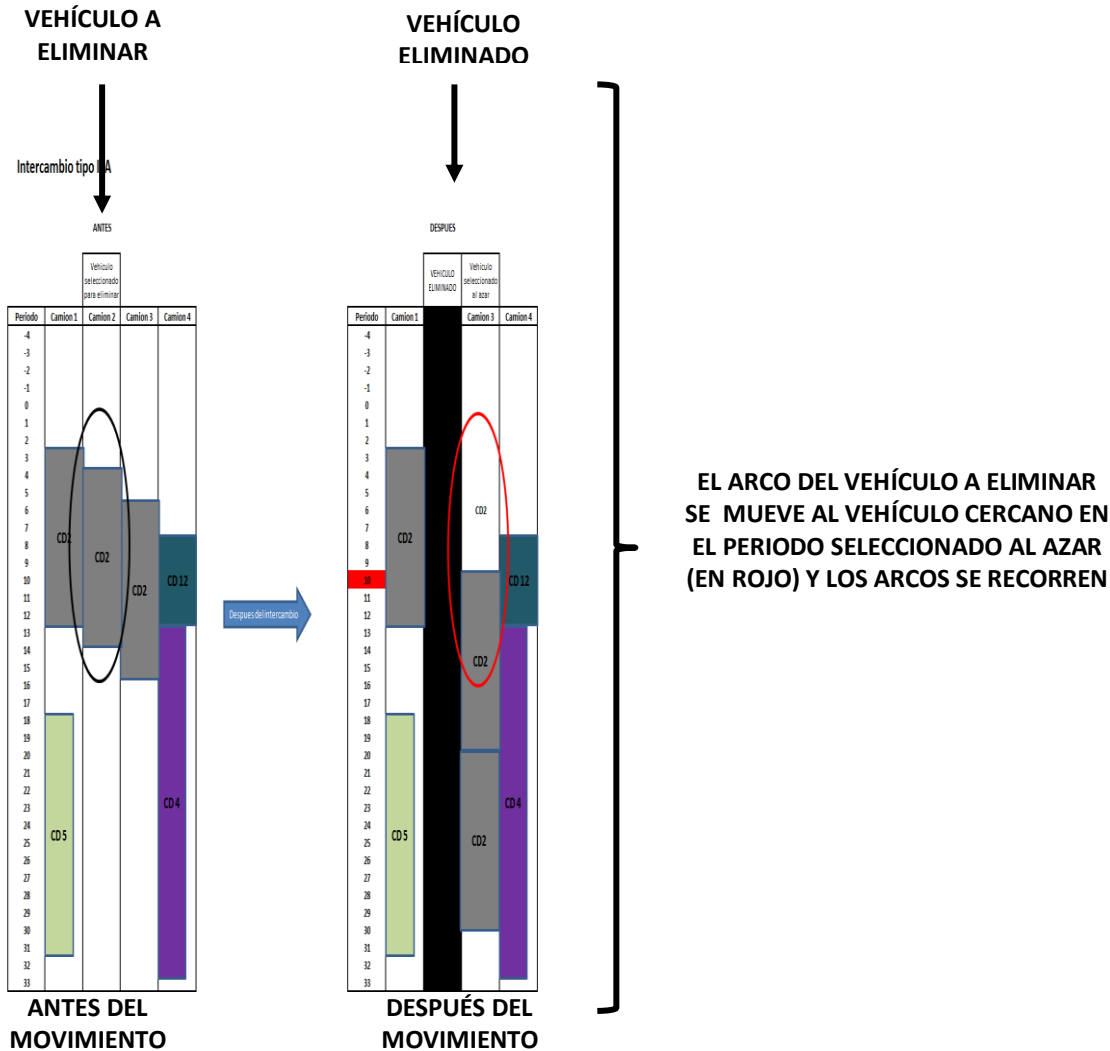


Figura 3-12 Intercambios tipo II A

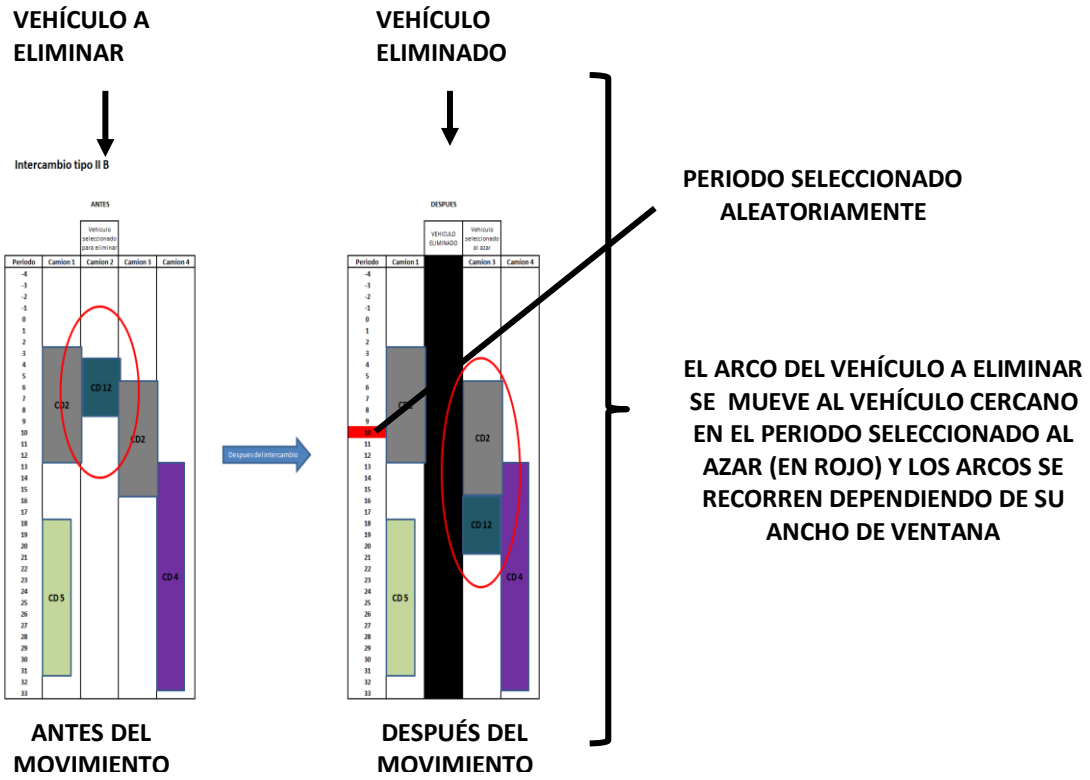


Figura 3-13 Intercambios tipo II B

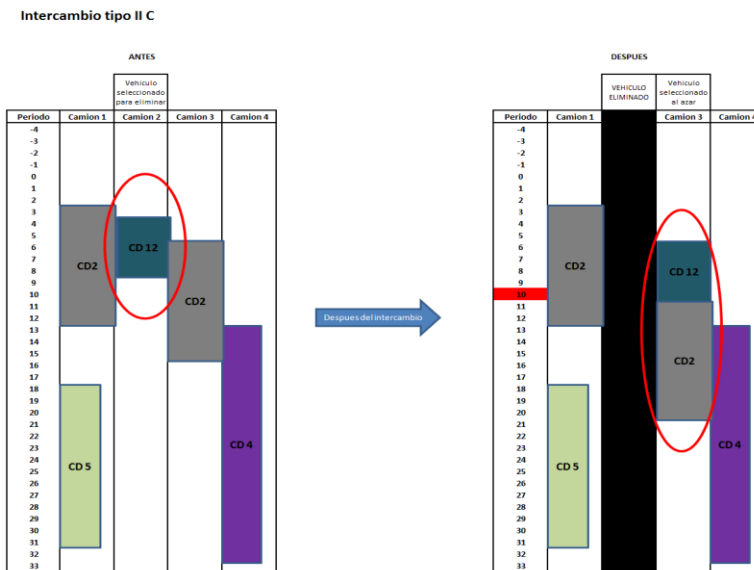


Figura 3-14 Intercambios tipo II C

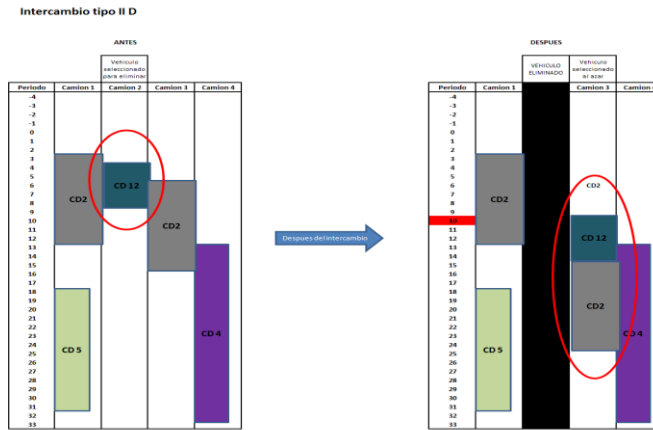


Figura 3-15 Intercambio tipo II D

INTERCAMBIOS DEL TIPO III

Se presentan cuando se selecciona un período de carga al azar y éste es utilizado por otro vehículo.

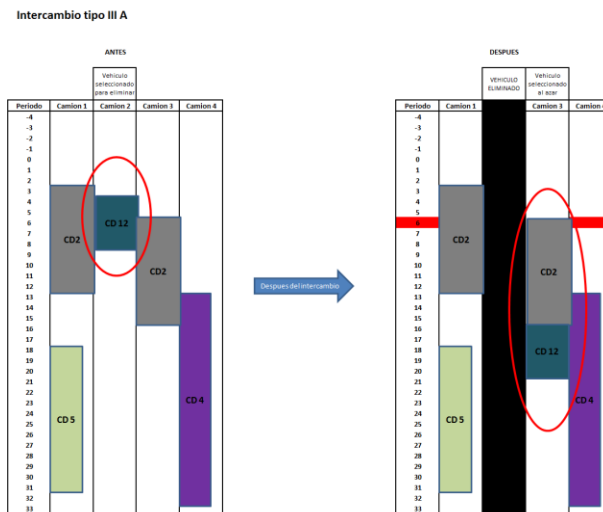


Figura 3-16 Intercambio tipo III A

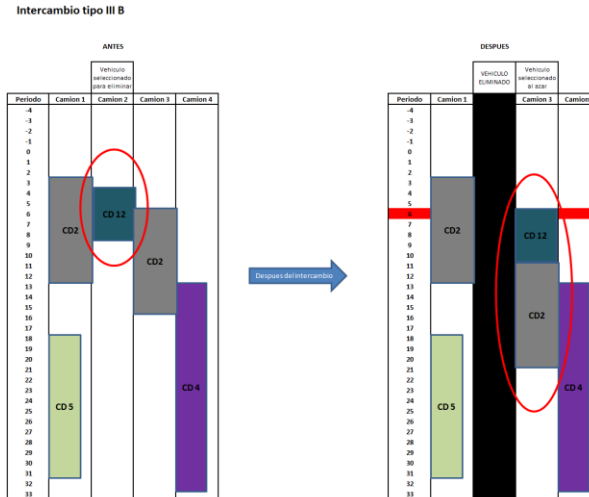


Figura 3-17 Intercambios tipo III B

Paso 10.

Una iteración está integrada por varios candidatos o intentos, es decir, los candidatos indicarán cuántas veces por iteración se intentará eliminar un vehículo seleccionado de forma aleatoria, y a su vez, un candidato será definido como el movimiento de todos los arcos de un camión a otros vehículos. De una iteración se compararán los valores de la función objetivo de los candidatos con respecto a la F^* si alguno de estos valores es mejor, o mejora la solución, se ingresará a la *Lista Tabú* y su solución será, ahora, el eje para las siguientes búsquedas hasta que sea mejorada. Es decir, esto se realiza para la misma solución glotona hasta que el contador de **número de iteraciones antes de regresar a una solución glotona** sea igual al valor introducido en ella, en la ventana de solicitud de datos.

De esta forma se realiza el siguiente proceso:

Supongamos que se desean dos candidatos seleccionados al azar.

Análisis para el primer candidato. Tabla para el primer candidato

Tabla 3-9 Representación de un camión candidato

ITERACION	CANDIDATO	DE			A			NUMERO DE ITERACIONES RESTANTES ANTES DE REGRESAR A UNA SOLUCION INICIAL	VALOR DE LA FUNCION OBJETIVO
		CAMION	PERIODO	CD	CAMION	PERIODO	CD		
1	1	3	-4	2					
		3	10	3					
		3	42	8					
		3	54	12					

La tabla contiene la información de dónde a dónde se moverán los arcos, para así, al final, poder evaluar el movimiento y asignarle un valor a su función objetivo. Se selecciona el CD

3, del vector Dummy, para posteriormente colocarlo de manera aleatoria en un camión. Al momento de hacer la inserción se le da preferencia a aquellas ventanas de atención con menor tamaño, por esta razón, se observa que después del movimiento quedan en orden descendente con base en su ancho de venta. Se recorren los CD's hacia abajo, siguiendo las reglas de los intercambios mencionados anteriormente.

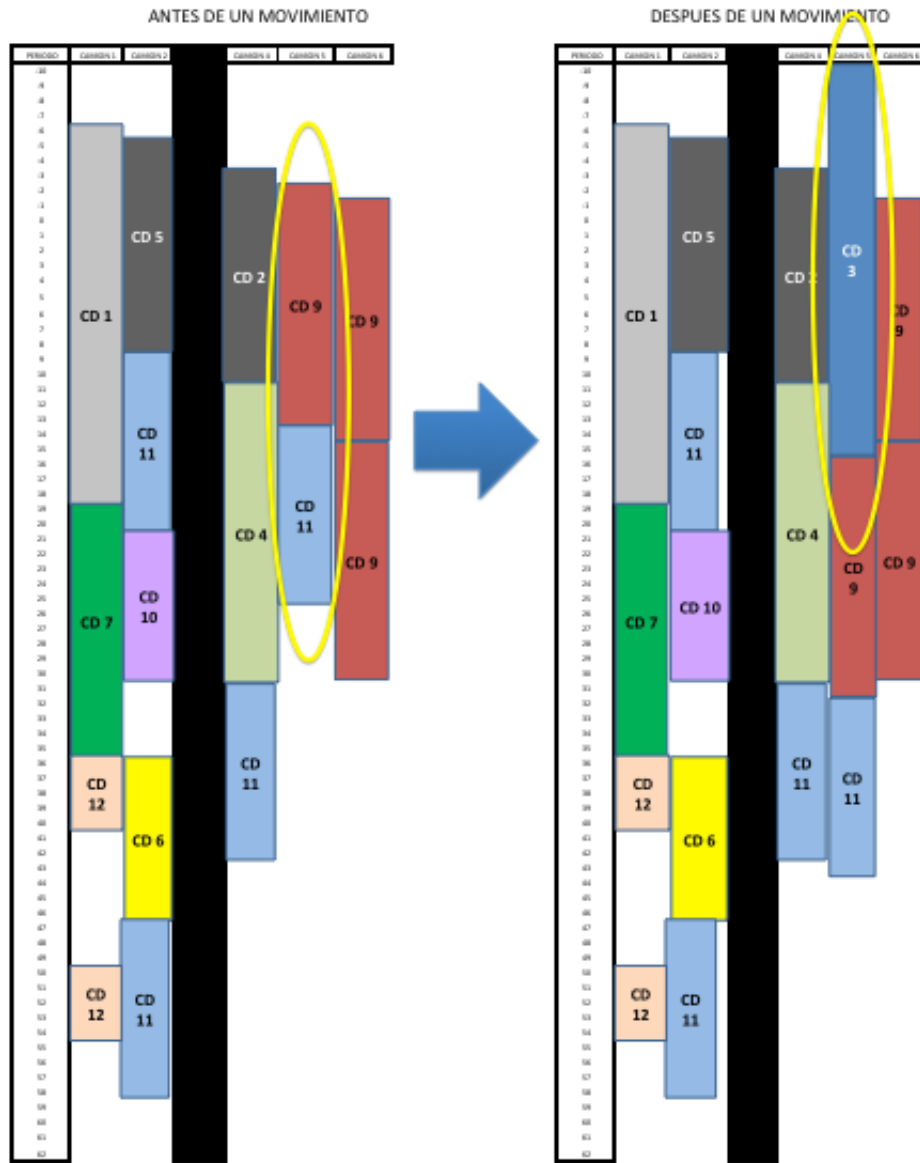


Figura 3-18 Colocación de un arco del vehículo a eliminar en otro camión

De esta manera, se registra el primer movimiento en la tabla de control de iteraciones:

Movimiento 1

Tabla 3-10 El movimiento de un arco a otro camión

NUEVA INSERCIÓN DEL ARCO ANTIGUAMENTE EN EL CAMIÓN 3, PERIODO 10 Y CD 3, EN UN CAMIÓN DIFERENTE Y DIFERENTE PERIODO

ITERACION	CANDIDATO	DE			A			NUMERO DE ITERACIONES RESTANTES ANTES DE REGRESAR A UNA	VALOR DE LA FUNCION OBJETIVO
		CAMION	PERIODO	CD	CAMION	PERIODO	CD		
1	1	3	-4	2	5	-10	3		
		3	42	8	5	16	9		
		3	54	12	5	32	11		
		5	-2	9	5	16	9		
		5	14	11	5	32	11		
		6	-1	9	6	12	9		
		6	15	9	6	45	9		

AL INSERTAR ESE ARCO EN EL CAMIÓN, RECORRE LOS DEMÁS ARCOS, GENERANDO ESTE TIPO DE MOVIMIENTO

AL FINALIZAR LOS INTERCAMBIOS DE LOS CANDIDATOS, SE REALIZA LA SUMATORIA DE CAMIONES, LA CUAL SE COLOCARÁ EN ESTA CASILLA. Y SERÁ EL VALOR DE LA FUNCIÓN OBJETIVO

INDICA EL VALOR DE ITERACIONES RESTANTES ANTES DE TOMAR OTRA SOLUCIÓN GLOTONA, SI NO EXISTE UNA MEJORA EN LA SOLUCIÓN. SU VALOR ESTÁ DETERMINADO POR LOS VALORES INICIALES E IRA DISMINUYENDO HASTA LLEGAR A CERO. MOMENTO EN EL CUAL SE DEBERÁ TOMAR OTRA SOLUCIÓN GLOTONA

Se realizan los siguientes movimientos de la misma forma:

Movimiento 2

Tabla 3-11 Movimiento 2

ITERACION	CANDIDATO	DE			A			NUMERO DE ITERACIONES RESTANTES ANTES DE REGRESAR A UNA SOLUCION INICIAL	VALOR DE LA FUNCION OBJETIVO
		CAMION	PERIODO	CD	CAMION	PERIODO	CD		
1	1	3	-4	2	6	-2	2		
		3	10	3	5	-10	3		
		3	42	8	5	16	9		
		3	54	12	5	32	11		
		5	-2	9	5	16	9		
		5	14	11	5	32	11		
		6	-1	9	6	12	9		

Movimiento 3

Tabla 3-12 Movimiento 3

ITERACION	CANDIDATO	DE			A			NUMERO DE ITERACIONES RESTANTES ANTES DE REGRESAR A UNA	VALOR DE LA FUNCION OBJETIVO
		CAMION	PERIODO	CD	CAMION	PERIODO	CD		
1	1	3	-4	2	6	-2	2		
		3	10	3	5	-10	3		
		3	42	8	4	48	8		
		3	54	12					
		5	-2	9	5	16	9		
		5	14	11	5	32	11		
		6	-1	9	6	12	9		
		6	15	9	6	45	9		
		4	-3	2	4	-3	2		
		4	11	4	4	11	4		

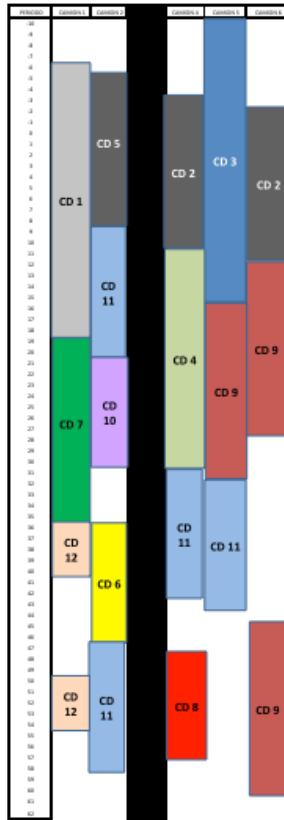


Figura 3-19 Vehículo eliminado indicado en negro

Movimiento 4

Tabla 3-13 Movimiento 4

ITERACION	CANDIDATO	DE			A			NUMERO DE ITERACIONES RESTANTES ANTES DE REGRESAR A UNA SOLUCION INICIAL	VALOR DE LA FUNCION OBJETIVO
		CAMION	PERIODO	CD	CAMION	PERIODO	CD		
1	1	3	-4	2	6	-2	2		
		3	10	3	5	-10	3		
		3	42	8	4	48	8		
		3	54	12	6	33	12		
		5	-2	9	5	16	9		
		5	14	11	5	32	11		
		6	-1	9	6	12	9		
		6	15	9	6	45	9		
		4	-3	2	4	-3	2		
		4	11	4	4	11	4		
		4	31	11	4	31	11		
		6	-2	2	6	-2	2		
		6	12	9	6	12	9		
		6	45	9	6	45	9		

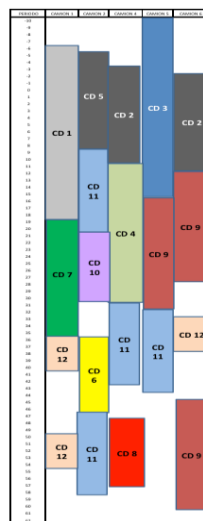
El valor de la función objetivo será la suma de los camiones utilizados para cubrir la demanda.

Tabla 3-14 Obtención del Valor de la Función Objetivo del primer candidato

DE ESTA FORMA QUEDAN ASIGNADOS TODOS LOS ARCOS DEL VEHÍCULO ELIMINADO

ITERACION	CANDIDATO	DE			A			NUMERO DE ITERACIONES RESTANTES ANTES DE REGRESAR A UNA SOLUCION INICIAL	VALOR DE LA FUNCION OBJETIVO
		CAMION	PERIODO	CD	CAMION	PERIODO	CD		
1	1	3	-4	2	6	-2	2	0	5
		3	10	3	5	-10	3		
		3	42	8	4	48	8		
		3	54	12	6	33	12		
		5	-2	9	5	16	9		
		5	14	11	5	32	11		
		6	-1	9	6	12	9		
		6	15	9	6	45	9		
		4	-3	2	4	-3	2		
		4	11	4	4	11	4		
		4	31	11	4	31	11		
		6	-2	2	6	-2	2		
		6	12	9	6	12	9		
		6	45	9	6	45	9		

Con la siguiente solución:



**REPRESENTACIÓN
DESPUÉS DE ELIMINAR
UN VEHÍCULO. TODOS
LOS ARCOS SE
MOVIERON A LOS
DIFERENTES VEHÍCULOS**

Figura 3-20 Representación gráfica después de eliminar un camión

Debe notarse que la solución obtenida siempre será diferente debido al elemento aleatorio.

Paso 11. Se compara el valor de la función objetivo, obtenido en este intercambio, con respecto al mejor valor conocido. Si este valor es menor, entonces se almacenará en la *lista tabú*. En este caso, el mejor valor conocido era de 6 camiones, al llegar esta nueva solución, ésta reemplaza a la F (*) mejor conocida. Y su solución será el origen para repetir los pasos del 2 al 9.

Tabla 3.-15 Representación del movimiento en la Lista Tabú

ITERACION	CANDIDATO	DE			A			NUMERO DE ITERACIONES BLOQUEADAS	NUMERO DE ITERACIONES RESTANTES ANTES DE REGRESAR A UNA SOLUCION INICIAL	VALOR DE LA FUNCION OBJETIVO
		CAMION	PERIODO	CD	CAMION	PERIODO	CD			
1	1	3	-4	2	6	-2	2	3	0	5
		3	10	3	5	-10	3			
		3	42	8	4	48	8			
		3	54	12	6	33	12			
		5	-2	9	5	16	9			
		5	14	11	5	32	11			
		6	-1	9	6	12	9			
		6	15	9	6	45	9			
		4	-3	2	4	-3	2			
		4	11	4	4	11	4			
		4	31	11	4	31	11			
		6	-2	2	6	-2	2			
		6	12	9	6	12	9			
		6	45	9	6	45	9			

En esta tabla, pueden observarse dos columnas nuevas, la primera de ellas: *número de elemento en Lista Tabú*, que será un contador para la antigüedad de un elemento en dicha tabla. Al momento que el contador llegue al valor que se ingresó en la variable ENTRADA, en los datos solicitados por el algoritmo, el elemento será borrado de la lista. La segunda columna es la de *número de iteraciones bloqueadas* que indicará la cantidad de iteraciones en la que los movimientos, en la lista, no podrán realizarse, es decir, se tendrá que regresar a su posición original.

Paso 12.

Si en la siguiente iteración el valor de solución no es mejorado el *número de iteraciones*, antes de regresar a una solución glotona, incrementará en uno, y así sucesivamente hasta llegar al valor máximo solicitado. En ese momento, se seleccionará otra solución glotona al azar para iniciar nuevamente el proceso.

Paso 13.

Así, el algoritmo se repetirá hasta que llegue al límite de iteraciones indicadas en los valores iniciales.

CAPÍTULO 4. ANÁLISIS NUMÉRICO

4.1 Obtención de los datos de prueba y comparativos

Con el fin de evaluar los beneficios que el *algoritmo de búsqueda tabú* proporciona a la asignación y secuenciación de vehículos, se decidió validarlo a través de la creación de 64 problemas de prueba llamados: *instancias obtenidas de González*. Cada una de ellas representa un escenario de demanda diaria dada en número de viajes.

Los datos con los cuales se integró el programa de demandas para cada uno de los 12 centros de distribución se encuentran en el Anexo 7.

Para cada comparación se realizaron 3 corridas con características específicas, hacia cada una de las instancias presentadas en este trabajo y partir de una solución inicial, diferente, y llegar a la misma solución. Donde se nombra: **característica del algoritmo de búsqueda tabú** al número de iteraciones diferentes bajo el cual se ejecuta el algoritmo.

Los resultados obtenidos por el *algoritmo de búsqueda tabú*, corriendo bajo diferentes características, se muestran en la tabla (1). Donde se comparará cada característica entre sí, y con respecto a los resultados óptimos obtenidos por el algoritmo CPLEX, a fin de hallar la mejor solución para realizar la comparación con el algoritmo 1-RBH.

Para el diseño de cada característica del *algoritmo de búsqueda tabú* se hizo uso del generador de corridas ofrecido por el software **MINITAB 15** para un diseño de experimentos tipo Taguchi L18, sin ser considerado como tal, únicamente, se utilizó para obtener diferentes valores. Aquí se tomaron en cuenta las siguientes consideraciones:

Tabla 4-1 Definición del diseño experimental

Factores	Niveles	Variables de respuesta	Constantes	Equipo y/o instrumentos de medición	Unidad experimental
1.- NÚMERO DE ITERACIONES TABÚ	6	Número de vehículos	Rendimiento de la computadora	Tiempo de CPU y valores obtenidos por el algoritmo	Algoritmo
2.- NO. ITERACIONES PARA REGRESAR A SOLUCIÓN GLOTONA	2				
3.- NÚMERO CANDIDATOS	3				
4.- NÚMERO ITERACIONES BLOQUEADAS	3				
5.- NÚMERO DE ELEMENTOS EN LT	3				

Al aplicar la tabla 4 se obtiene la tabla 4.1, ésta se llena para cada una de las 47 instancias diferentes. Ofreciendo un total de 564 ocurrencias.

Cada instancia tendrá tres repeticiones, como se observa, para asegurar la consistencia del resultado. De esta manera, es posible obtener un total de 1833 veces que será ejecutado el algoritmo.

Tabla 4-2 Valores y replicas para ejecución del algoritmo

NÚMERO DE ITERACIONES TABÚ	NO. ITERACIONES PARA REGRESAR A SOLUCIÓN GLOTONA	NÚMERO CANDIDATOS	NÚMERO ITERACIONES BLOQUEADAS	NÚMERO DE ELEMENTOS EN LISTA TABÚ
100	4	5	3	3
100	12	10	7	7
250	4	5	7	7
250	12	10	10	10
700	4	10	3	10
700	12	15	7	3
1000	4	15	10	7
1000	12	5	3	10
1500	4	10	10	3
1500	12	15	3	7
2000	4	15	7	10
2000	12	5	10	3

Los resultados obtenidos para cada característica del algoritmo se muestran en el Anexo 5. De estos resultados, es posible obtener una tabla condensada de análisis como se muestra en la tabla 4.2; donde se compara el rendimiento de cada característica del *algoritmo de búsqueda tabú*, con respecto al algoritmo óptimo (C-Plex) y el 1-RBH. Los resultados del algoritmo óptimo, son obtenidos de González (2004), donde, de igual manera, pueden hallarse los valores del 1-RBH.

Tabla 4-3 Análisis de las corridas de cada algoritmo

TIPO DE ALGORITMO	VALOR DE R CUADRADA	VALOR DE R CUADRADA (AJUSTADA)	PROMEDIO DE ERROR DEL ALGORITMO CON RESPECTO AL C-PLEX
C-PLEX	93.10%	92.90%	
1-RBH	86.30%	86.00%	1.1875
CARACTERÍSTICA 1 TABÚ	92.30%	92.10%	1.395833333
CARACTERÍSTICA 2 TABÚ	78.70%	78.20%	0.916666667
CARACTERÍSTICA 3 TABÚ	92.60%	92.50%	0.875
CARACTERÍSTICA 4 TABÚ	93.30%	93.20%	1.166666667
CARACTERÍSTICA 5 TABÚ	92.70%	92.50%	1.041666667
CARACTERÍSTICA 6 TABÚ	93.50%	93.40%	1.125
CARACTERÍSTICA 7 TABÚ	94.10%	94.00%	0.979166667
CARACTERÍSTICA 8 TABÚ	91.60%	91.40%	1.125
CARACTERÍSTICA 9 TABÚ	92.80%	92.60%	0.979166667
CARACTERÍSTICA 10 TABÚ	93.20%	93.10%	0.979166667
CARACTERÍSTICA 11 TABÚ	94.00%	93.80%	0.979166667
CARACTERÍSTICA 12 TABÚ	93.00%	92.80%	0.979166667

La tabla de abajo, en sus primeras dos columnas, muestra el valor del coeficiente de regresión que permite conocer qué tanto el modelo es expresado en las variables

analizadas. En general, el desempeño del *algoritmo de búsqueda tabú* presentó un mejor desempeño, por encima del 90% de forma global. La siguiente columna es el Error Medio de cada una de las observaciones, de las características, con respecto a los valores del *algoritmo de búsqueda tabú*, es decir, es el valor que indica qué tanto se estará desviando el algoritmo, cada vez que se ejecute, bajo estos parámetros del *algoritmo de búsqueda tabú*. De esta columna, el mejor valor lo ofrece la característica 3. De igual manera, en general, todas las características se desvían menos de una unidad, es decir, ofrecerán un vehículo más, posiblemente, al ejecutarse una instancia. A contra parte, el algoritmo 1-RBH, tiene un valor de 2 unidades.

Tabla 4-4 Comparativos condensados de los tres algoritmos parte uno

DATOS		ALGORITMO 1-RBH		ALGORITMO DE BÚSQUEDA TABÚ		ALGORITMO OPTIMO		% DE DESVIACIÓN DEL ALGORITMO 1-RBH VS OPTIMO	% DE DESVIACIÓN TABÚ (CARACTERÍSTICA 3) VS OPTIMO	% DE DESVIACIÓN TABÚ VS 1=RBH
INSTANCIA	DEMANDA	ALGORITMO 1-RBH	TIEMPO DE CPU (SEG)	CARACTERÍSTICA MEJOR SOLUCIÓN 3	TIEMPO DE CPU SEG	ALGORITMO CPLEX	TIEMPO DE CPU (SEG)			
3	20	5	0.031	5	5.5	4	14.2	25%	25%	0%
49	20	5	0.016	5	5.7187	4	3.8	25%	25%	0%
5	23	5	0.047	5	5.9331	5	66.4	0%	0%	0%
53	24	5	0.031	6	6.1565	5	9	0%	20%	20%
9	26	6	0.047	6	6.3281	6	84.5	0%	0%	0%
12	29	6	0.031	6	7.98	5	9	20%	20%	0%
61	32	7	0.063	7	8.8284	7	653.8	0%	0%	0%
62	32	8	0.047	8	8.3148	7	16525.7	14%	14%	0%
97	33	8	0.078	7	8.504	7	240.7	14%	0%	-13%
64	34	8	0.047	8	9.22	7	180.1	14%	14%	0%
98	35	8	0.063	8	9.4476	8	954.9	0%	0%	0%
100	36	8	0.063	8	9.2496	7	17.5	14%	14%	0%
21	38	9	0.031	9	9.6096	9	5033	0%	0%	0%
67	38	8	0.047	8	9.355	8	2006	0%	0%	0%
103	39	8	0.063	8	9.4848	8	267.6	0%	0%	0%
104	39	9	0.078	9	9.7684	9	1170.5	0%	0%	0%
70	40	8	0.078	9	10.306	8	10138.4	0%	13%	13%
106	40	10	0.047	9	9.6854	9	4262.4	11%	0%	-10%
23	41	10	0.063	9	9.8124	9	7596.4	11%	0%	-10%

Tabla 4-5 Continuación de la tabla de comparativos

INSTANCIA	ALGORITMO 1-RBH			ALGORITMO DE BÚSQEDA TABÚ				% DE DESVIACIÓN OPTIMO VS 1-RBH	% DE DESVIACIÓN TABÚ (CARACTERÍSTICA 3) VS OPTIMO	% DE DESVIACIÓN TABÚ VS 1-RBH
	DEMANDA	ALGORITMO 1-RBH	TIEMPO DE CPU	CARACTERÍSTICA MEJOR SOLUCIÓN 3	TIEMPO DE CPU SEG	ALGORITMO CPLEX	TIEMPO DE CPU			
109	41	9	0.078	9		10.104		3036	13%	0%
111	41	8	0.063	8		9.7812		63078	14%	0%
72	42	11	0.063	11		9.7812		12040	10%	0%
25	44	11	0.078	10		10.296		44644	10%	-9%
117	44	10	0.063	10		10.405		7393	11%	0%
118	44	9	0.094	9		10.171		8195	0%	0%
119	44	10	0.094	10		10.405		6822	11%	0%
123	44	9	0.078	9		10.152		11055	13%	0%
127	45	11	0.078	10		10.873		15477	22%	-9%
76	46	12	0.047	11		11.794		32.8	20%	-8%
129	46	10	0.078	10		11.635		7554	11%	0%
130	46	10	0.063	10		10.421		694.7	11%	0%
134	47	9	0.078	9		10.539		1.00E+05	0%	0%
137	49	10	0.109	11		11.137		786.3	0%	10%
31	50	10	0.078	10		11.312		6093	0%	0%
139	51	10	0.109	11		11.544		12945	0%	10%
81	52	13	0.078	13		12.754		8114	8%	0%
82	52	12	0.078	12		12.516		4991	9%	0%
141	52	13	0.94	12		12.651		12655	18%	-8%
35	53	15	0.063	13		12.995		31195	25%	-13%
143	54	11	0.109	11		11.893		11788	10%	0%
38	56	14	0.078	14		13.9		28355	17%	0%
144	57	12	0.109	12		13.286		13820	9%	0%
89	60	18	0.109	15		15.234		52478	38%	-17%
90	60	17	0.109	15		14.972		3.00E+05	31%	-12%
45	62	16	0.109	15		15.803		6.00E+05	23%	-6%
92	62	18	0.125	15		15.66		88669	38%	-17%
94	64	18	0.125	17		22.45		35641	29%	-6%
47	65	18	0.109	18		19.944		47361	20%	0%

En un 31% de las instancias probadas el *algoritmo de búsqueda tabú* obtiene los valores óptimos. Para el 54.16% de las instancias, el algoritmo se aleja en una unidad del valor óptimo reportado por Cplex 8.1. En general, el *algoritmo de búsqueda tabú* se aleja en promedio una unidad del algoritmo óptimo, como se observa en la tabla 4.2 para la característica 3. Lo mismo sucede con el algoritmo 1-RBH que tiene una desviación de 11.9% con respecto al óptimo. A contra parte el *algoritmo de búsqueda tabú* tiene una desviación de 9% con respecto al óptimo. De esta forma, se concluye que el uso del *algoritmo de búsqueda tabú*, en este tipo problemas, ofrece una mejor solución que

aquellas obtenidas por el algoritmo 1-RBH. De forma análoga se señala que el desempeño del algoritmo, en este caso de investigación, arroja soluciones satisfactorias para la toma de decisiones de la empresa, mejorando la velocidad de decisión y disminuyendo el margen de error. Los estudios completos de regresión y del *Error Medio*, pueden hallarse en los Anexos 2 y 9.

CONCLUSIONES

Con base en el planteamiento hecho, en la introducción de esta investigación, se verifica que con la aplicación de los *algoritmos de búsqueda tabú* se pueden encontrar soluciones capaces de llevar a cabo la asignación y programación de viajes, con restricciones de ventanas de tiempo de atención, para cumplir con la demanda diaria de doce centros de distribución (CD's), localizados en la zona del Estado de México, utilizando un número mínimo de vehículos. Así como la de diseñar un sistema de soporte que permitiera, mediante la utilización de los algoritmos, minimizar el número de vehículos.

La metodología propuesta en este trabajo muestra una alternativa para ser implementada en cualquier compañía de distribución de bienes, cuando las cargas son estándar, es decir, que todas tengan las mismas características.

A partir de las pruebas realizadas en el último capítulo, de esta investigación, se confirma que los *algoritmos de búsqueda tabú* son una alternativa de optimización al momento de proponer una solución factible en un contexto incierto y variable. Esta incertidumbre y variabilidad, en el universo de soluciones posibles, permitió demostrar que los *algoritmos de búsqueda tabú* poseen una tendencia a la optimización en problemas que representan soluciones complejas.

Es importante aclarar que, en el *algoritmo de búsqueda tabú* propuesto, no se hace uso de todas sus herramientas, sino que solamente se acota la demostración a lo expresado en el capítulo 4. Por lo que se deja abierta la capacidad de aplicar herramientas de búsqueda más agresivas y mecanismos de penalización por movimiento. Este algoritmo es capaz de mejorar las operaciones de asignación realizadas por la compañía X de una manera más eficiente, además de realizar una selección adecuada, de soluciones factibles, utilizando las ventajas computacionales para una mejor toma de decisiones. El desempeño del algoritmo ofrece un rendimiento del 20.3% por encima de las soluciones obtenidas por González en 2004.

Después de evaluar cada una de las características se verifica que entre más iteraciones ejecute el algoritmo y más veces se regrese a una solución glotona, más rica será la búsqueda entre vecindades, permitiendo la capacidad de hallar soluciones más cercanas al valor óptimo. Sin embargo, esto demeritará su desempeño, el cual, por los tiempos mostrados en los anexos, es razonablemente rápido. El algoritmo tiene la particularidad de generar siempre soluciones factibles con base en las características de cada centro de distribución, su demanda y su ventana de atención.

RECOMENDACIONES

Una vez concluida la investigación se mencionan algunas líneas de trabajo para todo aquel interesado en continuar con este tema:

- Con la experiencia lograda en este trabajo de investigación se puede adaptar el algoritmo, para apoyar en el proceso de toma de decisiones cuando existan acontecimientos que alteren el programa de entregas tales como siniestros carreteros, demoras inevitables por condiciones climatológicas que impidan el cumplir con la demanda de un centro de distribución, etc.
- Adaptar el algoritmo evaluando la posibilidad de contar con vehículos de diversas capacidades de carga.
- Probar el algoritmo en problemas con aplicaciones que requieren entregas punto a punto como en el caso de la industria del transporte de concreto premezclado, por citar alguno.
- Crear una librería de problemas tipo, en la que los problemas se pudieran clasificar con base en el grado de dificultad para cumplir con las demandas de ciertos centros de distribución y con ellos probar el desempeño del algoritmo propuesto.
- Las soluciones generadas resuelven el problema satisfactoriamente, pero queda abierta a futuras investigaciones la utilización de otros métodos de creación de soluciones glotonas.

BIBLIOGRAFÍA

- Arunapuram S., Mathur K., Solow D. (2003). Vehicule Routing and Scheduling with Full Truckloads. *Transportation Science*. 170-182.
- Bodin L., Golden B., Assad A., Ball M. (1983). Routing and Scheduling of Vehicles and Crews: The State of Art. *Computation. and Operation Research* 10.2. 62-212.
- Borenstein, D., Jing-Quan, L., & Mirchandani. P.B (2005, diciembre 22). A decision support system for the single-depot vehicle. *Science Direct*. Recuperado en diciembre 2012 de www.sciencedirect.com
- Carraresi P., & Gallo, G. (1984), Networks Models for Vehicule and Crew Scheduling. *European Journal of Operational research*. 16. 139-151.
- Coello. C. (2010). Evolutionary Multiobjetive Optimization: Current and Future Challenges. Recuperado en febrero 2012 en <http://www.lania.mx/~ccoello/EMOO/EMOOconferences.html>.
- Crainic T. G., Laporte G. (1997). Planning Models for Freight Transportation. *European Journal of Operational research*. 409-438.
- Dantzig, G. B. & Ramser, J.H. (1988). The Truck Dispatching System. *Manage Sci*. 6. 80-91.
- Desrochers M. & Soumis F. (1988). A generalized permanent labeling algorithm for the shortest path problem with time Windows. *INFOR*. 26 (3). 191-211.
- Desrosiers J., DumasY. & Soumis M. M. (1995). Time Constrained Routing and Scheduling. *Magnanti, Monma, and Nemhouser* [1995a]. Capter 2. 35-139.
- Desrosiers, J., Pelletier, P. & F. Soumis. (1983). Plus court chemin avec contraintes d'horaires, r.a.i.r.o. *Recherche Operationnelle*. 17. 357-377.
- Fisher L.M. (1995) *Vehicle Routing.* " *Handbooks in Operations Research and Management Science*. Volumen 8. 1-33.
- Kerningham, L.S. & B. (1973). An effective Heuristic Algorithm for the Traveling Salesman Problem. *Operation Research*. 21. 498-516.
- Laporte, G. (1992). The Vehicule Routing Problem: an overview of exact and approximate algorithms. *European Journal of Operational Research*. 59. 345-358.
- Lim, A., Zhang, G.S., & Zuo, Y. (2005) A Tabu Search Algorithm for the Safe Transportation of Hazardous Materials. *ACM Digital Library*. Recuperado en enero 2012 <http://0-dl.acm.org.millenium.itesm.mx>

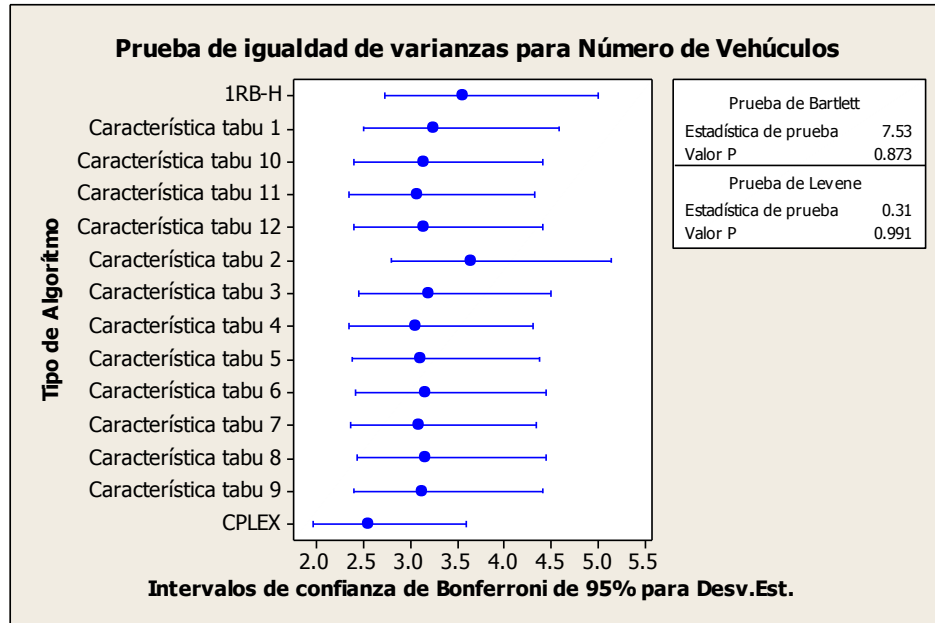
Lin, S. (1965) Computer Solution of the Traveling Salesman Problem. *Bell Systems*. 44. 22-69.

Savelsbergh, M. W. P. (1985) Local search in routing problems with time windows. *Annals of Operations Research*. 4. 285-305.

Stamatis, D.H. (2002) *Six Sigma and Beyond*. Florida, US.

ANEXOS

Anexo 1. Prueba de varianzas iguales para los diferentes algoritmos



Resultados para resultados completos corregidos.MTW

Prueba de varianzas iguales: número de vehículos vs. tipo de algoritmo

Intervalos de confianza de Bonferroni de 95% para desviaciones estándares

Tipo de Algoritmo	N	Inferior	Desv.Est.	Superior
1RB-H	48	2.72105	3.55597	5.01154
Característica tabú 1	48	2.48910	3.25286	4.58436
Característica tabú 10	48	2.39889	3.13497	4.41820
Característica tabú 11	48	2.34787	3.06830	4.32425
Característica tabú 12	48	2.39889	3.13497	4.41820
Característica tabú 2	48	2.79037	3.64656	5.13922

Característica tabú 3	48	2.44279	3.19234	4.49907
Característica tabú 4	48	2.33857	3.05614	4.30711
Característica tabú 5	48	2.37628	3.10542	4.37656
Característica tabú 6	48	2.41673	3.15828	4.45106
Característica tabú 7	48	2.36226	3.08709	4.35073
Característica tabú 8	48	2.41957	3.16200	4.45630
Característica tabú 9	48	2.39515	3.13008	4.41132
CPLEX	48	1.95648	2.55680	3.60338

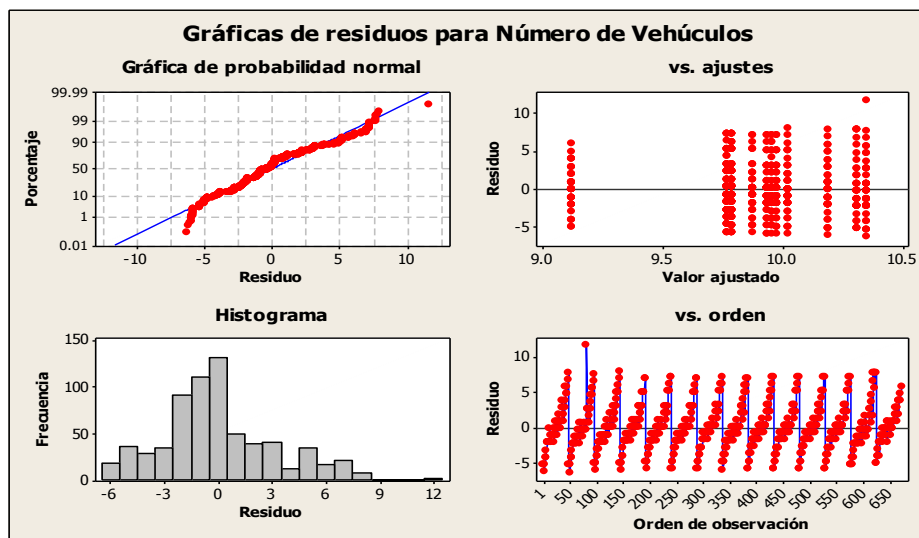
Prueba de Bartlett (distribución normal)

Estadística de prueba = 7.53, valor p = 0.873

Prueba de Levene (cualquier distribución continua)

Estadística de prueba = 0.31, valor p = 0.991

Prueba de varianzas iguales: número de vehículos vs. tipo de algoritmo



Anexo 2. Análisis de regresión el *algoritmo de búsqueda tabú*

CARACTERÍSTICA (Desrochers M, 1988) (L., 1995) (G., 1992) (Savelsbergh, 1985)
(Arunapuram S., 2003) (Stamatis, 2002) (Carraresi P., 1984)1

Análisis de regresión: característica tabú 1 vs. demanda

La ecuación de regresión es

Característica tabú 1 = - 1.68 + 0.272 Demanda

Predictor	Coef	Coef. de EE	T	P
Constante	-1.6785	0.5226	-3.21	0.002
Demanda	0.27226	0.01160	23.46	0.000

S = 0.913050 R-cuad. = 92.3% R-cuad.(ajustado) = 92.1%

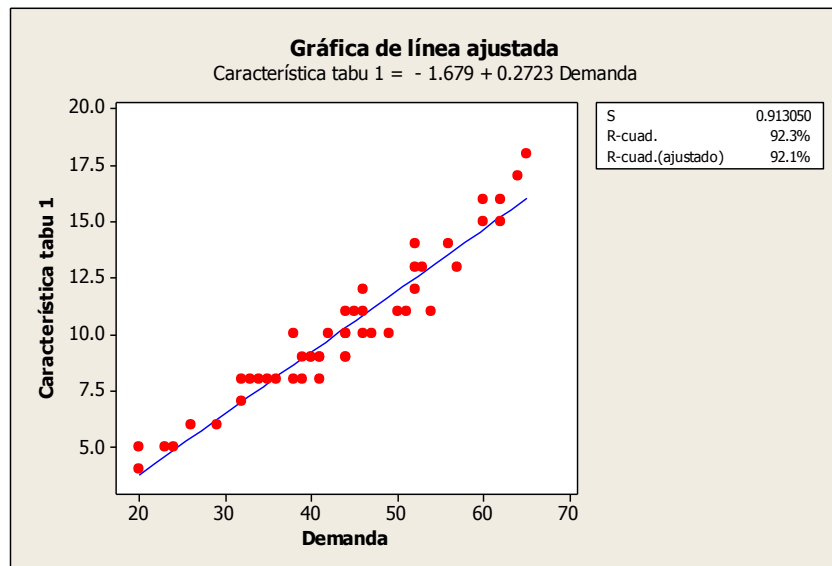
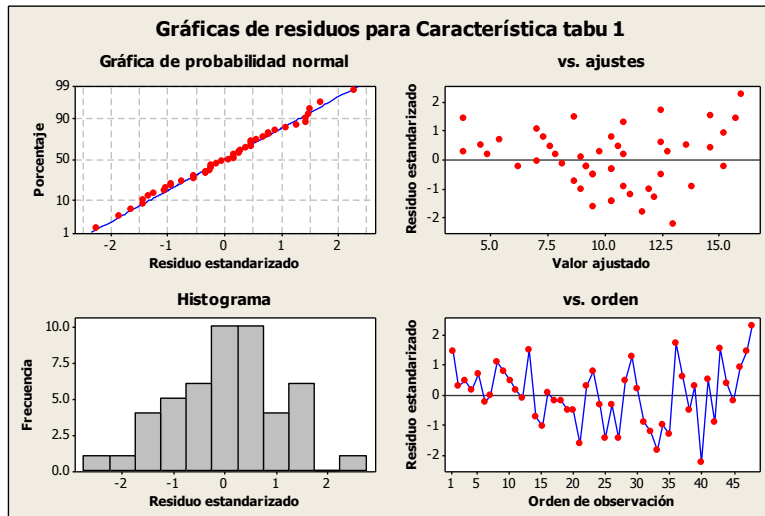
Análisis de varianza

Fuente	GL	SC	MC	F	P
Regresión	1	458.96	458.96	550.54	0.000
Error residual	46	38.35	0.83		
Total	47	497.31			

Observaciones poco comunes

Obs	Característica				Residuo	
	Demanda	tabú 1	Ajuste	Ajuste SE	Residuo	estándar
40	54.0	11.000	13.024	0.179	-2.024	-2.26R
48	65.0	18.000	16.018	0.281	1.982	2.28R

R denota una observación con un residuo estandarizado grande.



CARACTERÍSTICA 2

Análisis de regresión: característica tabú 2 vs. demanda

La ecuación de regresión es

$$\text{Característica tabú 2} = - 1.93 + 0.282 \text{ Demanda}$$

Predictor	Coef	Coef. de EE	T	P
Constante	-1.9295	0.9738	-1.98	0.054
Demanda	0.28184	0.02162	13.04	0.000

S = 1.70127 R-cuad. = 78.7% R-cuad.(ajustado) = 78.2%

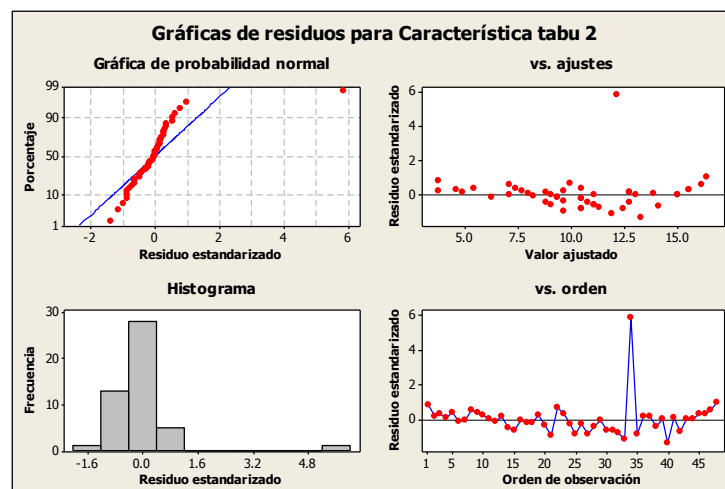
Análisis de varianza

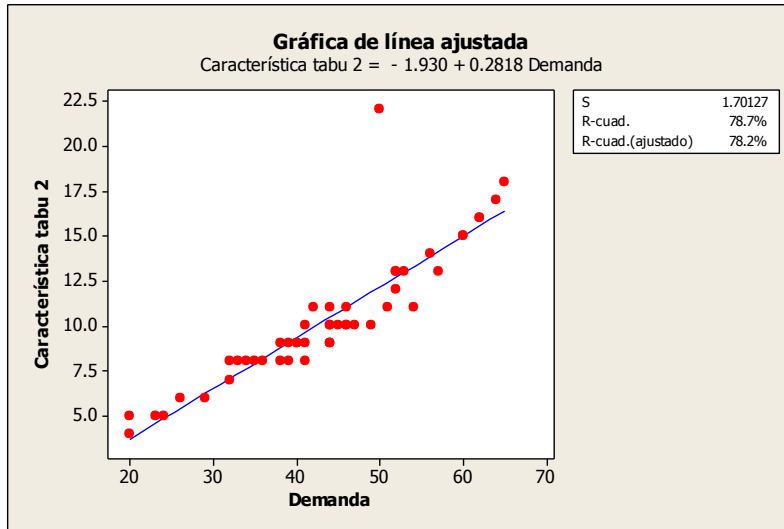
Fuente	GL	SC	MC	F	P
Regresión	1	491.84	491.84	169.93	0.000
Error residual	46	133.14	2.89		
Total	47	624.98			

Observaciones poco comunes

Obs	Demanda	Característica	Residuo
34	50.0	tabú 2 Ajuste	estándar
		Ajuste SE	Residuo estándar
		0.282	9.837 5.86R

R denota una observación con un residuo estandarizado grande.





CARACTERÍSTICA 3

Análisis de regresión: característica tabú 3 vs. demanda

La ecuación de regresión es

$$\text{Característica tabú 3} = - 1.65 + 0.268 \text{ Demanda}$$

Predictor	Coef	Coef. de EE	T	P
Constante	-1.6458	0.5015	-3.28	0.002
Demanda	0.26769	0.01113	24.04	0.000

$$S = 0.876176 \quad R\text{-cuad.} = 92.6\% \quad R\text{-cuad. (ajustado)} = 92.5\%$$

Análisis de varianza

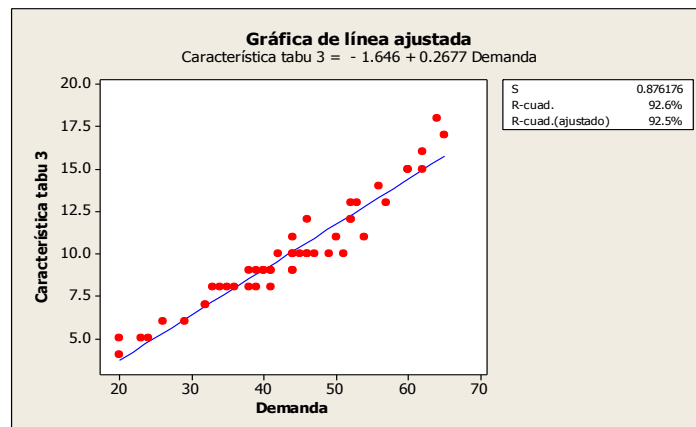
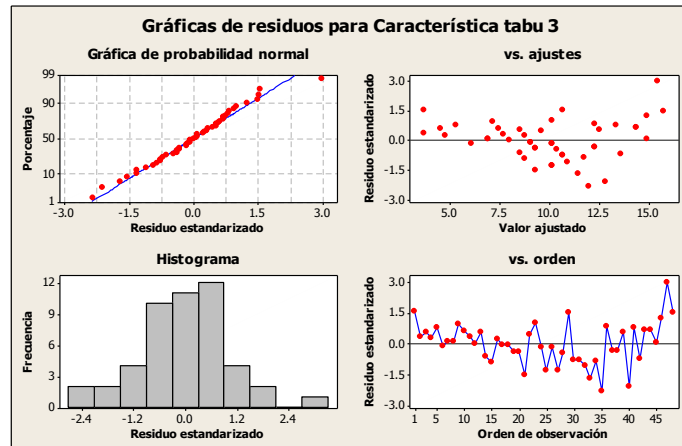
Fuente	GL	SC	MC	F	P
Regresión	1	443.67	443.67	577.93	0.000
Error residual	46	35.31	0.77		

Total 47 478.98

Observaciones poco comunes

Obs	Demanda	Característica	Residuo
		tabú 3 Ajuste Ajuste SE Residuo estándar	
35	51.0	10.000 12.006 0.151 -2.006	-2.32R
40	54.0	11.000 12.809 0.172 -1.809	-2.11R
47	64.0	18.000 15.486 0.260 2.514	3.00R

R denota una observación con un residuo estandarizado grande.



CARACTERÍSTICA 4

Análisis de regresión: característica tabú 4 vs. demanda

La ecuación de regresión es

Característica tabú 4. = - 1.23 + 0.257 Demanda

Predictor	Coef	Coef. de EE	T	P
Constante	-1.2311	0.4572	-2.69	0.010
Demanda	0.25721	0.01015	25.34	0.000

S = 0.798701 R-cuad. = 93.3% R-cuad. (ajustado) = 93.2%

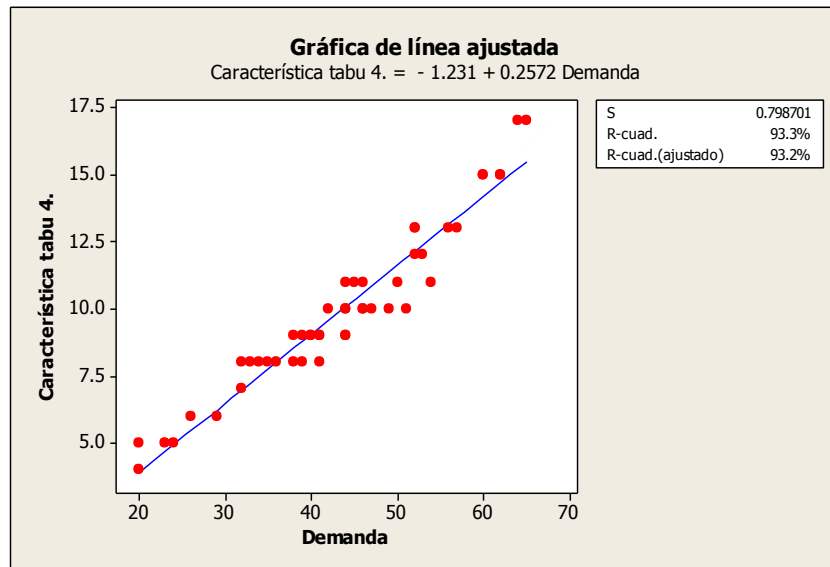
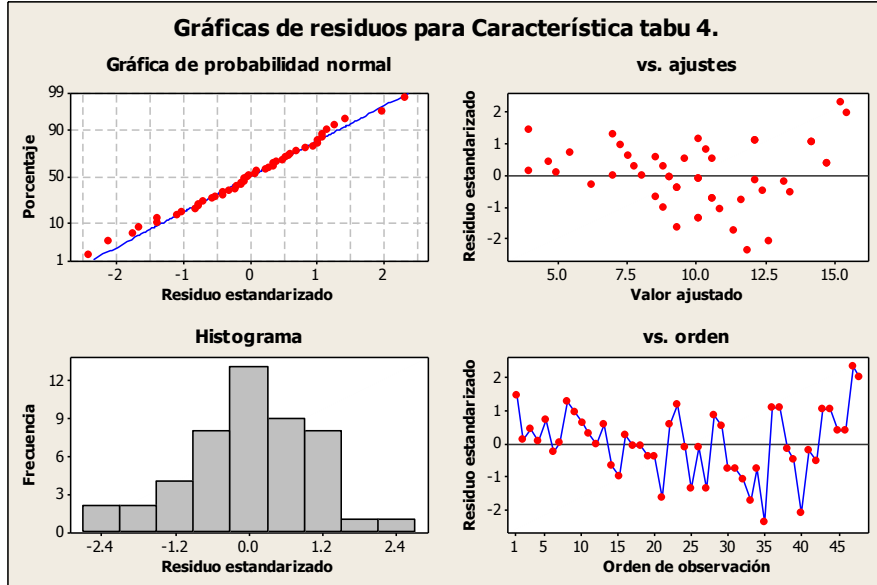
Análisis de varianza

Fuente	GL	SC	MC	F	P
Regresión	1	409.63	409.63	642.14	0.000
Error residual	46	29.34	0.64		
Total	47	438.98			

Observaciones poco comunes

Obs	Demanda	Característica			Residuo	
		tabú 4.	Ajuste	Ajuste SE	Residuo	estándar
35	51.0	10.000	11.887	0.138	-1.887	-2.40R
40	54.0	11.000	12.658	0.156	-1.658	-2.12R
47	64.0	17.000	15.231	0.237	1.769	2.32R

R denota una observación con un residuo estandarizado grande.



CARACTERÍSTICA 5

Análisis de regresión: característica tabú 5 vs. demanda

La ecuación de regresión es

$$\text{Característica tabú 5} = -1.48 + 0.260 \text{ Demanda}$$

Predictor	Coef	Coef. de EE	T	P
Constante	-1.4754	0.4870	-3.03	0.004
Demanda	0.26043	0.01081	24.08	0.000

S = 0.850901 R-cuad. = 92.7% R-cuad. (ajustado) = 92.5%

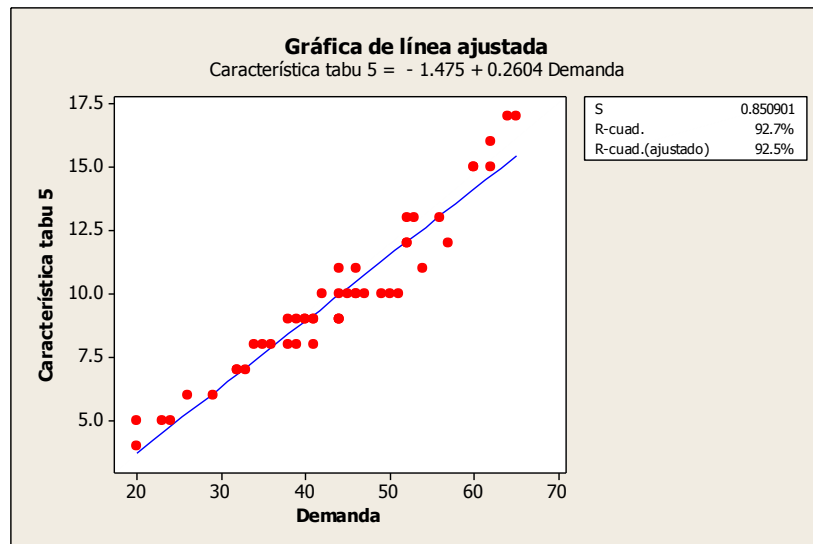
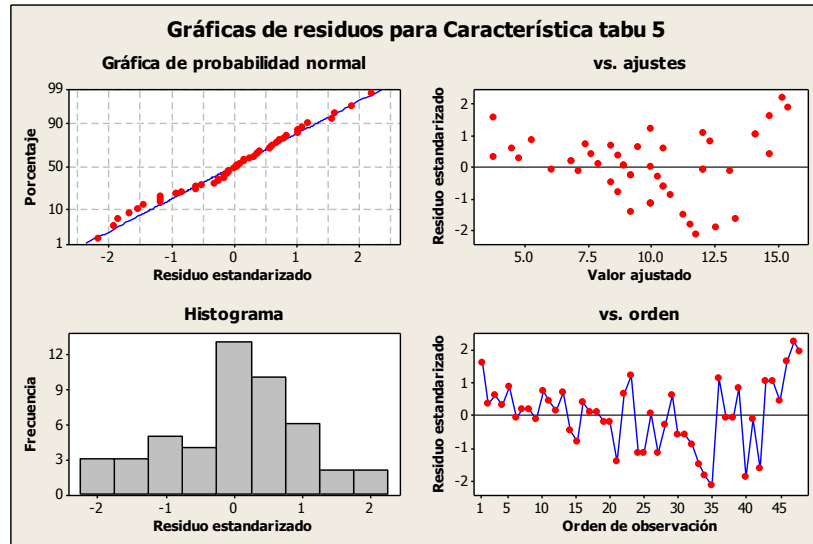
Análisis de varianza

Fuente	GL	SC	MC	F	P
Regresión	1	419.94	419.94	580.01	0.000
Error residual	46	33.31	0.72		
Total	47	453.25			

Observaciones poco comunes

Obs	Característica				Residuo	
	Demanda	tabú 5	Ajuste	Ajuste SE	Residuo	estándar
35	51.0	10.000	11.807	0.147	-1.807	-2.16R
47	64.0	17.000	15.192	0.253	1.808	2.22R

R denota una observación con un residuo estandarizado grande.



CARACTERÍSTICA 6

Análisis de regresión: característica tabú 6 vs. demanda

La ecuación de regresión es

Característica tabú 6 = - 1.66 + 0.266 Demanda

Predictor	Coef	Coef. de EE	T	P
Constante	-1.6611	0.4646	-3.58	0.001
Demanda	0.26612	0.01032	25.80	0.000

S = 0.811694 R-cuad. = 93.5% R-cuad. (ajustado) = 93.4%

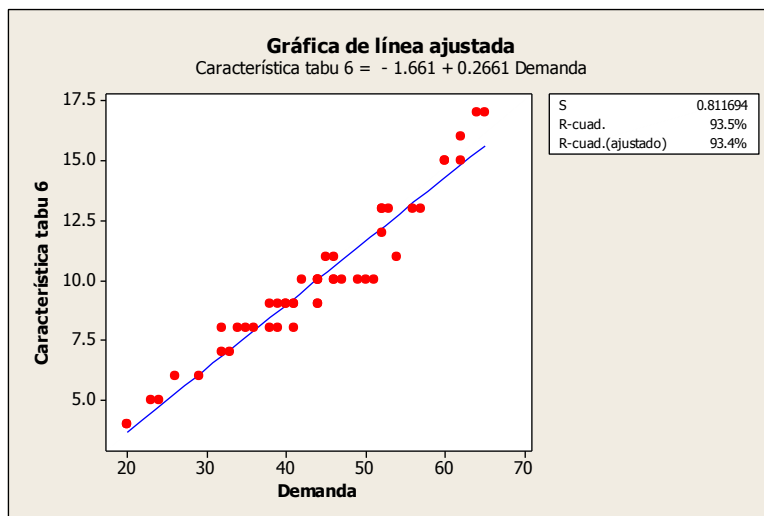
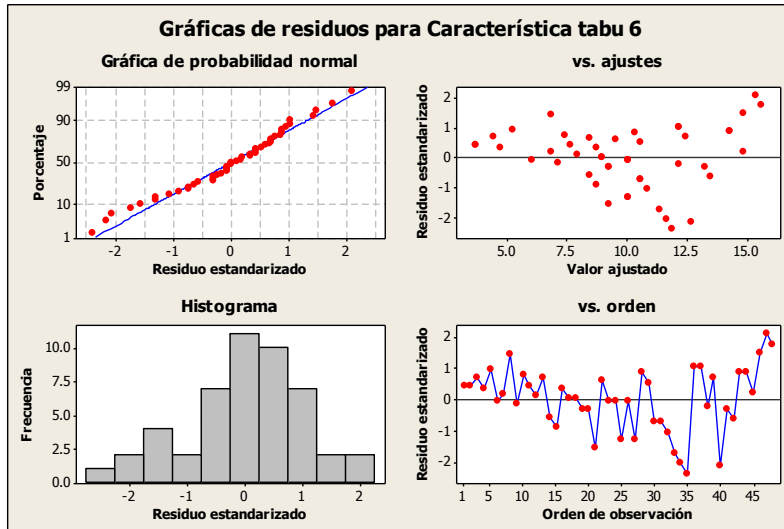
Análisis de varianza

Fuente	GL	SC	MC	F	P
Regresión	1	438.51	438.51	665.56	0.000
Error residual	46	30.31	0.66		
Total	47	468.81			

Observaciones poco comunes

Obs	Característica				Residuo	
	Demanda	tabú 6	Ajuste	Ajuste SE	Residuo	estándar
34	50.0	10.000	11.645	0.135	-1.645	-2.06R
35	51.0	10.000	11.911	0.140	-1.911	-2.39R
40	54.0	11.000	12.710	0.159	-1.710	-2.15R
47	64.0	17.000	15.371	0.241	1.629	2.10R

R denota una observación con un residuo estandarizado grande.



CARACTERISTICA 7

Análisis de regresión: característica tabú 7 vs. demanda

La ecuación de regresión es

$$\text{Característica tabú 7} = - 1.58 + 0.261 \text{ Demanda}$$

Predictor	Coef	Coef. de EE	T	P
-----------	------	-------------	---	---

Constante -1.5822 0.4323 -3.66 0.001
 Demanda 0.260969 0.009597 27.19 0.000

S = 0.755188 R-cuad. = 94.1% R-cuad.(ajustado) = 94.0%

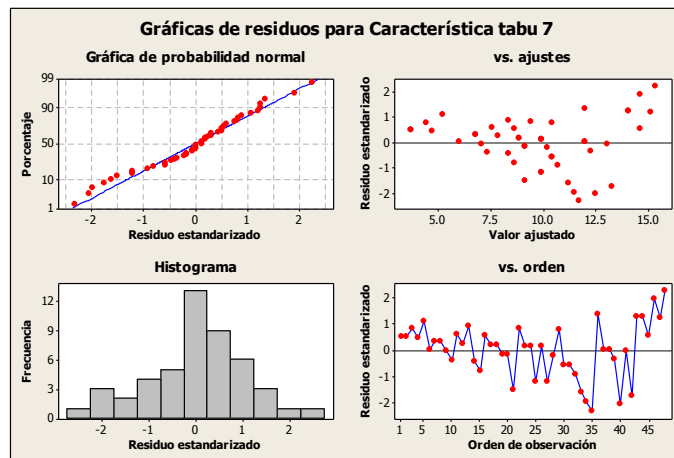
Análisis de varianza

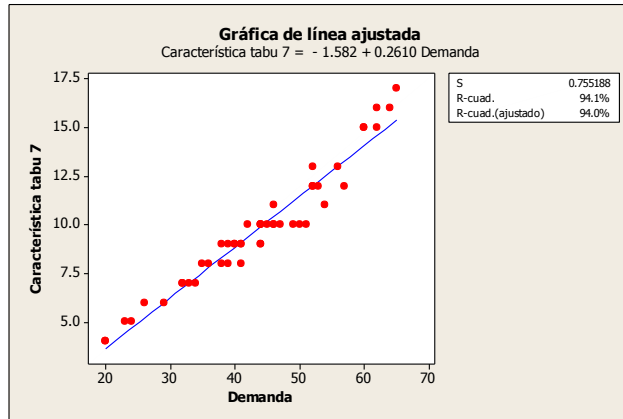
Fuente	GL	SC	MC	F	P
Regresión	1	421.68	421.68	739.39	0.000
Error residual	46	26.23	0.57		
Total	47	447.92			

Observaciones poco comunes

Obs	Demanda	Característica	tabú 7	Ajuste	Ajuste SE	Residuo	Residuo estándar
35	51.0		10.000	11.727	0.130	-1.727	-2.32R
40	54.0		11.000	12.510	0.148	-1.510	-2.04R
48	65.0		17.000	15.381	0.233	1.619	2.25R

R denota una observación con un residuo estandarizado grande.





CARACTERÍSTICA 8

Análisis de regresión: característica tabú 8 vs. demanda

La ecuación de regresión es

$$\text{Característica tabú 8} = - 1.53 + 0.264 \text{ Demanda}$$

Predictor	Coef	Coef. de EE	T	P
Constante	-1.5305	0.5315	-2.88	0.006
Demanda	0.26361	0.01180	22.34	0.000

$$S = 0.928605 \quad R\text{-cuad.} = 91.6\% \quad R\text{-cuad. (ajustado)} = 91.4\%$$

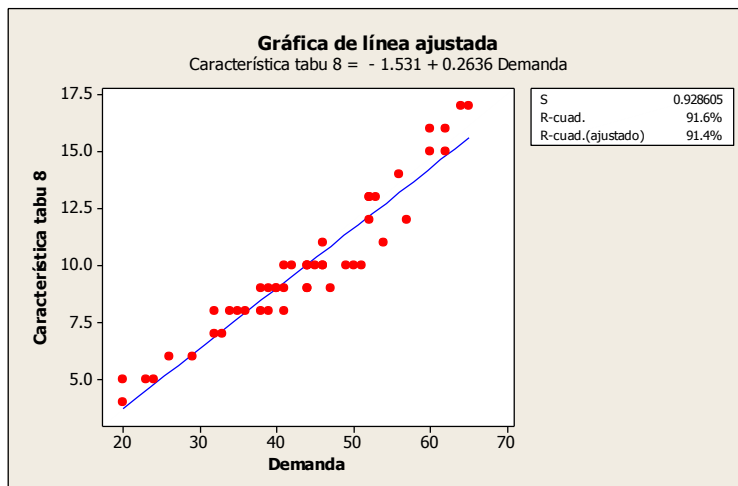
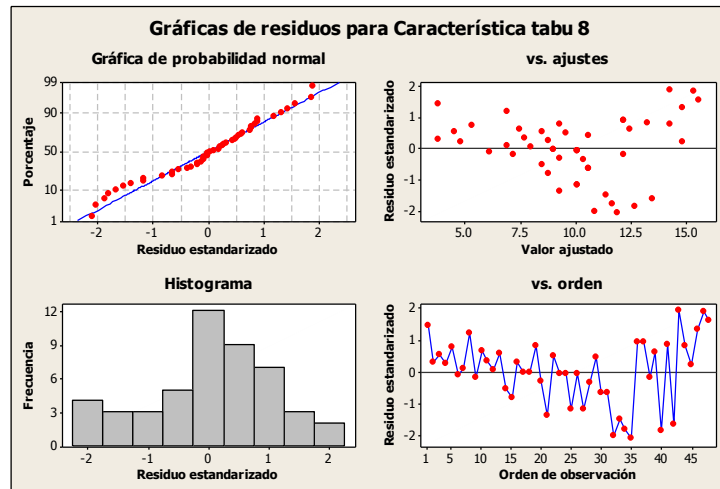
Análisis de varianza

Fuente	GL	SC	MC	F	P
Regresión	1	430.25	430.25	498.95	0.000
Error residual	46	39.67	0.86		
Total	47	469.92			

Observaciones poco comunes

	Característica			Residuo		
Obs	Demanda	tabú 8	Ajuste	Ajuste SE	Residuo	estándar
32	47.0	9.000	10.859	0.140	-1.859	-2.03R
35	51.0	10.000	11.913	0.160	-1.913	-2.09R

R denota una observación con un residuo estandarizado grande.



CARACTERÍSTICA 9

Análisis de regresión: característica tabú 9 vs. demanda

La ecuación de regresión es

$$\text{Característica tabú 9} = - 1.68 + 0.263 \text{ Demanda}$$

Predictor	Coef	Coef. de EE	T	P
Constante	-1.6776	0.4867	-3.45	0.001
Demanda	0.26268	0.01081	24.31	0.000

S = 0.850253 R-cuad. = 92.8% R-cuad. (ajustado) = 92.6%

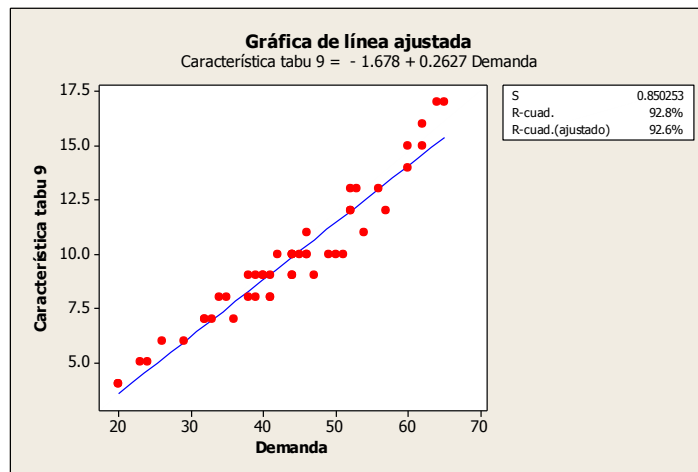
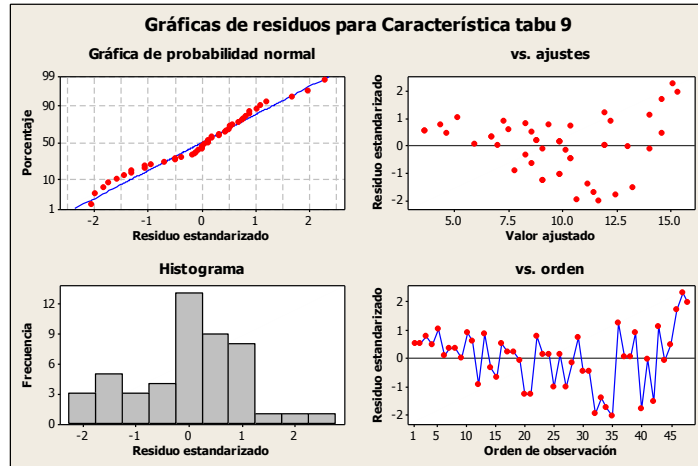
Análisis de varianza

Fuente	GL	SC	MC	F	P
Regresión	1	427.22	427.22	590.96	0.000
Error residual	46	33.25	0.72		
Total	47	460.48			

Observaciones poco comunes

Obs	Característica				Residuo	
	Demanda	tabú 9	Ajuste	Ajuste SE	Residuo	estándar
35	51.0	10.000	11.719	0.147	-1.719	-2.05R
47	64.0	17.000	15.134	0.252	1.866	2.30R

R denota una observación con un residuo estandarizado grande.



CARACTERÍSTICA 10

Análisis de regresión: característica tabú 10 vs. demanda

La ecuación de regresión es

$$\text{Característica tabú 10} = - 1.70 + 0.264 \text{ Demanda}$$

Predictor	Coef	Coef. de EE	T	P
Constante	-1.7019	0.4723	-3.60	0.001

Demanda 0.26371 0.01049 25.15 0.000

S = 0.825077 R-cuad. = 93.2% R-cuad.(ajustado) = 93.1%

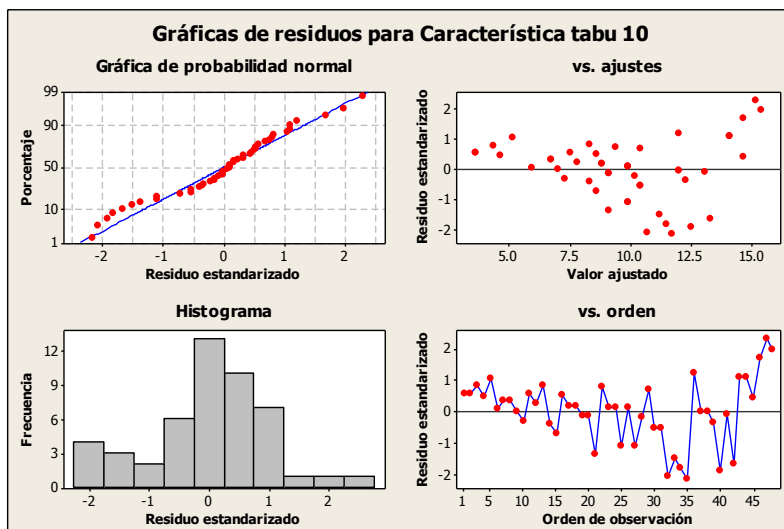
Análisis de varianza

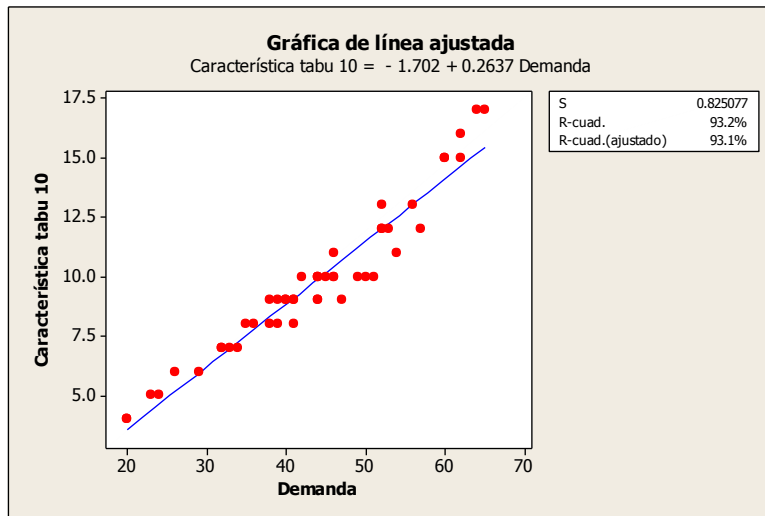
Fuente	GL	SC	MC	F	P
Regresión	1	430.60	430.60	632.54	0.000
Error residual	46	31.31	0.68		
Total	47	461.92			

Observaciones poco comunes

Obs	Demanda	Característica	Residuo
		tabú 10	estándar
32	47.0	9.000	-2.08R
35	51.0	10.000	-2.15R
47	64.0	17.000	2.32R

R denota una observación con un residuo estandarizado grande.





CARACTERÍSTICA 11

Análisis de regresión: característica tabú 11 vs. demanda

La ecuación de regresión es

$$\text{Característica tabú 11} = - 1.52 + 0.259 \text{ Demanda}$$

Predictor	Coef	Coef. de EE	T	P
Constante	-1.5227	0.4364	-3.49	0.001
Demanda	0.259125	0.009688	26.75	0.000

$$S = 0.762354 \quad R\text{-cuad.} = 94.0\% \quad R\text{-cuad. (ajustado)} = 93.8\%$$

Análisis de varianza

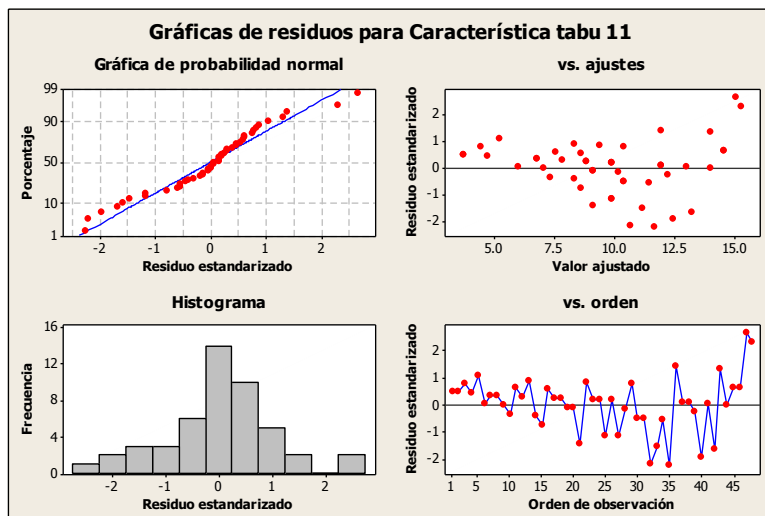
Fuente	GL	SC	MC	F	P
Regresión	1	415.74	415.74	715.34	0.000

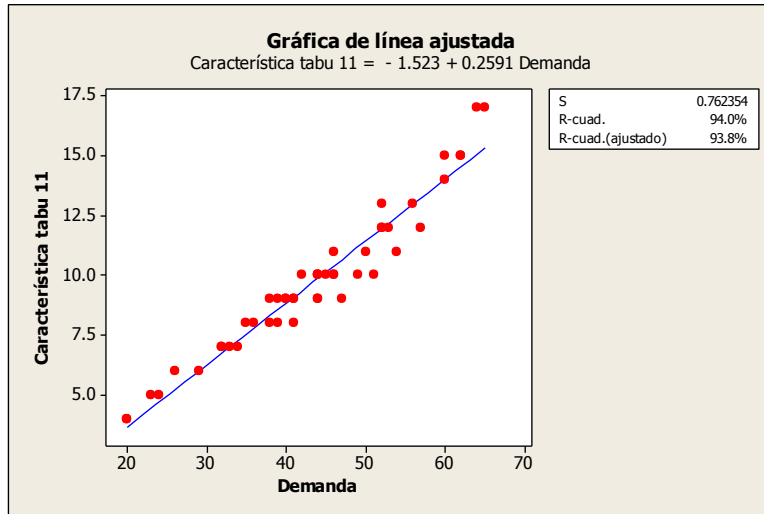
Error residual 46 26.73 0.58
 Total 47 442.48

Observaciones poco comunes

Obs	Demanda	Característica			Residuo	
		tabú 11	Ajuste	Ajuste SE	Residuo	estándar
32	47.0	9.000	10.656	0.115	-1.656	-2.20R
35	51.0	10.000	11.693	0.131	-1.693	-2.25R
47	64.0	17.000	15.061	0.226	1.939	2.66R
48	65.0	17.000	15.320	0.235	1.680	2.32R

R denota una observación con un residuo estandarizado grande.





CARACTERÍSTICA 12

Análisis de regresión: característica tabú 12 vs. demanda

La ecuación de regresión es

$$\text{Característica tabú 12} = - 1.69 + 0.263 \text{ Demanda}$$

Predictor	Coef	Coef. de EE	T	P
Constante	-1.6878	0.4801	-3.52	0.001
Demanda	0.26339	0.01066	24.71	0.000

S = 0.838850 R-cuad. = 93.0% R-cuad.(ajustado) = 92.8%

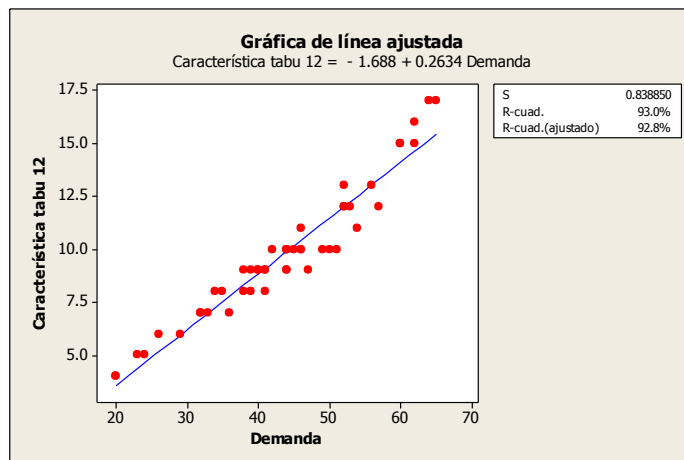
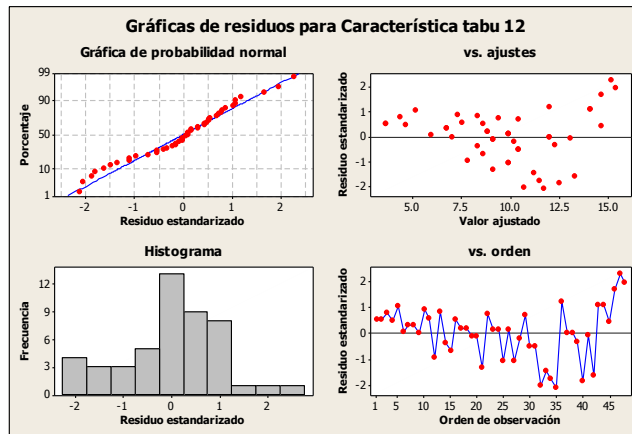
Análisis de varianza

Fuente	GL	SC	MC	F	P
Regresión	1	429.55	429.55	610.44	0.000
Error residual	46	32.37	0.70		
Total	47	461.92			

Observaciones poco comunes

Obs	Demanda	Característica			Residuo	
		tabú 12	Ajuste	Ajuste SE	Residuo	estándar
32	47.0	9.000	10.692	0.126	-1.692	-2.04R
35	51.0	10.000	11.745	0.145	-1.745	-2.11R
47	64.0	17.000	15.169	0.249	1.831	2.29R

R denota una observación con un residuo estandarizado grande.



Anexo 3. Análisis de regresión utilizando el algoritmo 1-rbh

Análisis de regresión: 1RB-H vs. demanda

La ecuación de regresión es

$$1RB-H = - 2.23 + 0.288 \text{ Demanda}$$

Predictor	Coef	Coef. de EE	T	P
Constante	-2.2328	0.7609	-2.93	0.005
Demanda	0.28785	0.01689	17.04	0.000

S = 1.32942 R-cuad. = 86.3% R-cuad.(ajustado) = 86.0%

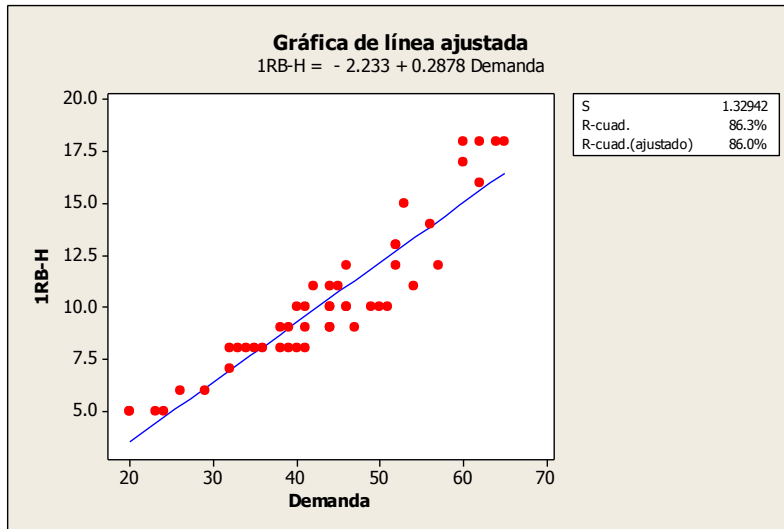
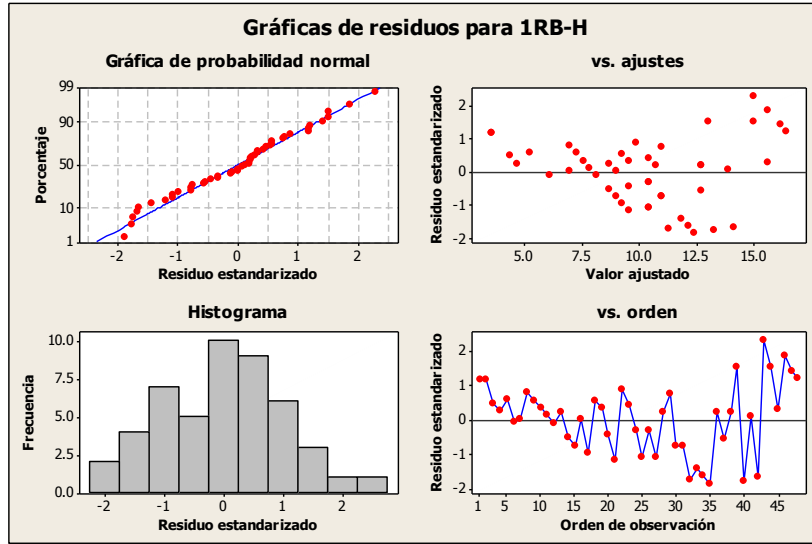
Análisis de varianza

Fuente	GL	SC	MC	F	P
Regresión	1	513.01	513.01	290.27	0.000
Error residual	46	81.30	1.77		
Total	47	594.31			

Observaciones poco comunes

Obs	Demanda	1RB-H	Ajuste	Ajuste SE	Residuo	Residuo estándar
43	60.0	18.000	15.038	0.337	2.962	2.30R

R denota una observación con un residuo estandarizado grande.



Anexo 4. Análisis de regresión utilizando el algoritmo cplex

Análisis de regresión: CPLEX vs. demanda

La ecuación de regresión es

$$\text{CPLEX} = - 0.240 + 0.215 \text{ Demanda}$$

Predictor	Coef	Coef. de EE	T	P
Constante	-0.2404	0.3899	-0.62	0.541
Demanda	0.214886	0.008657	24.82	0.000

S = 0.681187 R-cuad. = 93.1% R-cuad.(ajustado) = 92.9%

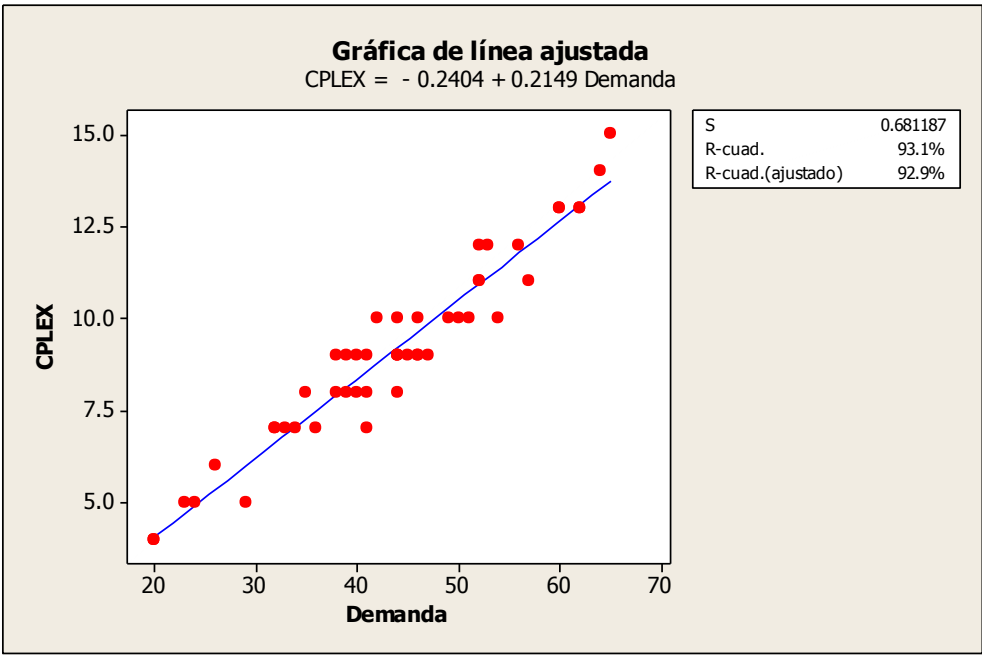
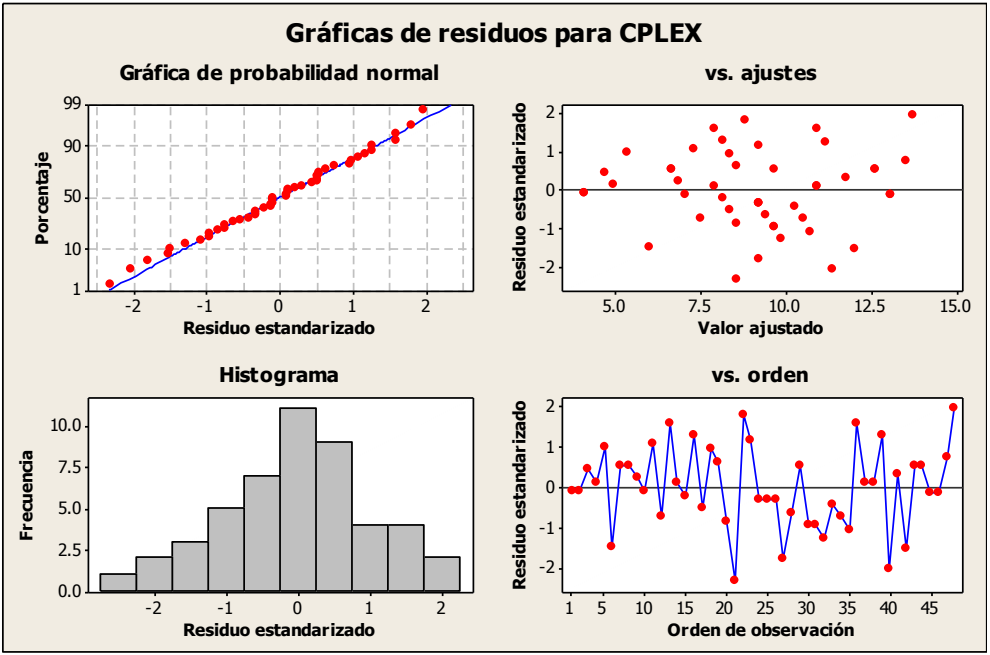
Análisis de varianza

Fuente	GL	SC	MC	F	P
Regresión	1	285.91	285.91	616.15	0.000
Error residual	46	21.34	0.46		
Total	47	307.25			

Observaciones poco comunes

Obs	Demanda	CPLEX	Ajuste	Ajuste SE	Residuo	Residuo estándar
21	41.0	7.0000	8.5699	0.1008	-1.5699	-2.33R
40	54.0	10.0000	11.3634	0.1334	-1.3634	-2.04R

R denota una observación con un residuo estandarizado grande.



Anexo 5. Resultados del *algoritmo de búsqueda tabú*

INSTANCI A	DEMAND A	CARACTERÍSTI CA MEJOR SOLUCIÓN 1	TIEMP O DE CPU SEG	CARACTERÍSTI CA MEJOR SOLUCIÓN 2	TIEMP O DE CPU SEG	CARACTERÍSTI CA MEJOR SOLUCIÓN 3	TIEMPO DE CPU SEG
3	20	5	0.2195	5	3.659	5	5.6096
49	20	4	2.1372	4	3.604	4	5.7012
5	23	5	2.1372	5	3.732	5	5.18.8
53	24	5	2.1608	5	3.76	5	5.223
9	26	6	2.2636	6	3.689	6	5.4132
12	29	6	2.3058	6	3.868	6	5.6
61	32	7	2.383	7	3.853	7	5.818
62	32	8	2.34	8	3.84	7	5.934
97	33	8	2.3942	8	4.722	8	4.3
64	34	8	2.449	8	4.562	8	5.8812
98	35	8	2.447	8	4.64	8	6.11
100	36	8	2.4754	8	4.272	8	6.148
21	38	10	2.574	9	4.114	9	6.246
67	38	8	2.5344	8	4.928	8	6.4104
103	39	8	2.589	8	4.142	8	6.45
104	39	9	2.588	9	4.194	9	6.4348
70	40	9	2.6094	9	4.112	9	8.679
106	40	9	2.5918	9	4.14	9	6.368
23	41	9	2.622	10	4.348	9	6.7214
109	41	9	2.678	9	4.18	9	6.437
111	41	8	2.582	8	4.134	8	7.2
72	42	10	2.647	11	4.16	10	6.52
25	44	11	2.709	11	4.255	11	7.4592
117	44	10	3.276	10	4.352	10	6.8286
118	44	9	2.82	9	4.306	9	6.815
119	44	10	2.7454	10	4.29	10	6.817
123	44	9	2.7308	9	4.346	9	6.6908
127	45	11	2.7924	10	4.327	10	6.7932
76	46	12	2.8392	11	4.332	12	7.622
129	46	11	2.8358	10	4.39	10	6.9108

INSTANCI A	DEMAND A	CARACTERÍSTICA MEJOR SOLUCIÓN 1	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 2	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 3	TIEMPO DE CPU SEG
130	46	10	2.9078	10	5.7	10	7.224
134	47	10	2.9438	10	4.688	10	7.2958
137	49	10	3.34	10	4.658	10	7.2652
31	50	11	3.5412	22	4.59	11	7.3018
139	51	11	2.9484	11	4.477	10	7.2196
81	52	14	3.131	13	4.602	13	7.5786
82	52	13	3.582	13	4.668	12	7.783
141	52	12	3.652	12	4.875	12	7.43
35	53	13	3.2254	13	4.571	13	7.7626
143	54	11	3.1828	11	5.144	11	7.7877
38	56	14	3.3228	14	4.867	14	8.4244
144	57	13	3.288	13	4.814	13	8.184
89	60	16	3.493	15	4.679	15	8.6268
90	60	15	3.4944	15	5.7	15	8.6016
45	62	15	3.5412	16	5.94	15	8.8608
92	62	16	3.6132	16	5.12	16	9.206
94	64	17	4.244	17	5.184	18	9.5512
47	65	18	5.1636	18	5.307	17	12.588

INSTANCIA	DEMANDA	CARACTERÍSTICA MEJOR SOLUCIÓN 4	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 5	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 6	TIEMPO DE CPU SEG
3	20	5	8.9856	5	26.48	4	38.674
49	20	4	9.29.76	4	26.36	4	38.3445
5	23	5	9.662	5	26.94	5	38.05
53	24	5	10.141	5	26.25	5	36.672
9	26	6	9.375	6	26.39	6	36.41
12	29	6	9.462	6	27.51	6	38.155
61	32	7	9.6166	7	28.26	7	38.266
62	32	8	9.676	7	28.15	8	37.932
97	33	8	10.629	7	27.92	7	39.759
64	34	8	9.9285	8	28.69	8	38.512
98	35	8	10.05	8	29.4	8	39.3032
100	36	8	10.73	8	29.69	8	39.4122
21	38	9	10.858	9	30.52	9	40.3906

INSTANCIA	DEMANDA	CARACTERÍSTICA MEJOR SOLUCIÓN 4	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 5	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 6	TIEMPO DE CPU SEG
67	38	8	7.5508	8	29.71	8	40.5106
103	39	8	10.436	8	30.21	8	40.45
104	39	9	10.329	9	30.25	9	40.9438
70	40	9	11.734	9	31.3	9	40.849
106	40	9	10.288	9	30.97	9	40.659
23	41	9	10.49	9	30.57	9	40.9933
109	41	9	10.463	9	30.61	9	41.135
111	41	8	10.626	8	30.55	8	41.4812
72	42	10	10.515	10	30.69	10	40.9128
25	44	11	10.697	11	31.42	10	41.4842
117	44	10	10.742	9	32.11	10	42.1832
118	44	9	10.907	9	31.92	9	42.9388
119	44	10	10.67	10	31.93	10	42.17
123	44	9	10.748	9	31.93	9	42.3852
127	45	11	11.84	10	33.65	11	43.2922
76	46	11	11.342	11	33.84	11	42.999
129	46	10	10.873	10	32.5	10	43.1942
130	46	10	11.748	10	34.85	10	44.132
134	47	10	11.136	10	44.58	10	43.9064
137	49	10	11.252	10	33.74	10	44.5066
31	50	11	11.241	10	33.48	10	44.8599
139	51	10	11.902	10	33.93	10	44.9687
81	52	13	11.528	13	34.98	13	44.6108
82	52	13	11.658	12	34.64	13	45.315
141	52	12	11.48	12	34.63	12	45.144
35	53	12	11.566	13	35.27	13	45.37
143	54	11	11.876	11	35.74	11	46.8734
38	56	13	12.67	13	36.47	13	47.2072
144	57	13	12.203	12	37.88	13	47.8644
89	60	15	12.34	15	37.75	15	46.8474
90	60	15	12.19	15	38.38	15	47.7662
45	62	15	12.605	15	38.9	15	48.875
92	62	15	12.6	16	39.17	16	49.24
94	64	17	13.354	17	42.59	17	50.809
47	65	17	15.943	17	48.91	17	51.228

INSTANCIA	DEMANDA	CARACTERÍSTICA MEJOR SOLUCIÓN 7	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 8	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 9	TIEMPO DE CPU SEG
-----------	---------	---------------------------------------	-------------------------	---------------------------------------	-------------------------	---------------------------------------	----------------------

3	20	4	57.22	5	20.87	4	57.5796
49	20	4	57.408	4	21.26	4	58.5
5	23	5	52.515	5	19.7	5	56.9632
53	24	5	53.206	5	20.13	5	55.779
9	26	6	53.83	6	20.41	6	56.68
12	29	6	56.192	6	21.9	6	59.215
61	32	7	56.893	7	21.96	7	01:00.8
62	32	7	57.552	8	21.68	7	01:00.5
97	33	7	54.865	7	21.62	7	01:00.7
64	34	7	57.675	8	21.86	8	01:05.1
98	35	8	58.28	8	22.2	8	01:02.3
100	36	8	59.185	8	22.47	7	01:03.4
21	38	9	59.7	9	22.87	9	01:03.9
67	38	8	1:0041	8	22.66	8	01:03.6
103	39	8	1:0071	8	23.33	8	01:04.5
104	39	9	01:00.1	9	22.7	9	01:04.2
70	40	9	01:07.4	9	23.83	9	01:04.9
106	40	9	01:01.2	9	23.26	9	01:05.4
23	41	9	01:01.3	10	23.65	9	01:05.0
109	41	9	01:08.5	9	23.46	8	01:05.3
111	41	8	01:03.0	8	23.77	8	01:05.9
72	42	10	01:02.0	10	23.29	10	01:06.2
25	44	10	01:02.7	10	23.81	10	01:07.1
117	44	10	01:04.2	10	24.59	10	01:08.5
118	44	9	01:03.7	9	24.73	9	01:08.3
119	44	10	01:03.4	10	24.13	10	01:07.3
123	44	9	01:03.6	9	24.15	9	01:07.9
127	45	10	01:04.9	10	24.51	10	01:09.3
76	46	11	01:05.5	11	25.69	11	01:09.8
129	46	10	01:05.3	10	24.74	10	01:09.3
130	46	10	01:06.2	10	37.26	10	01:09.4
134	47	10	01:05.4	9	24.69	9	01:09.6
137	49	10	01:07.2	10	25.43	10	01:12.8
31	50	10	01:06.5	10	25.71	10	01:11.6

INSTANCIA	DEMANDA	CARACTERÍSTICA MEJOR SOLUCIÓN 7	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 8	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 9	TIEMPO DE CPU SEG
-----------	---------	---------------------------------	-------------------	---------------------------------	-------------------	---------------------------------	-------------------

139	51	10	01:08.2	10	26.18	10	01:12.0
81	52	13	01:08.9	13	26.53	13	01:14.7
82	52	12	01:08.9	13	26.14	12	01:14.1
141	52	12	01:08.6	12	26.31	12	01:14.7
35	53	12	01:09.4	13	26.8	13	01:16.4
143	54	11	01:11.2	11	27.22	11	01:15.9
38	56	13	01:13.2	14	27.5	13	01:19.2
144	57	12	01:12.6	12	27.8	12	01:18.8
89	60	15	01:14.9	16	28.37	15	01:22.3
90	60	15	01:13.7	15	28.32	14	01:21.7
45	62	15	01:16.0	15	29.47	15	01:23.3
92	62	16	01:16.6	16	29.46	16	01:24.4
94	64	16	01:21.9	17	31.76	17	01:30.8
47	65	17	01:32.2	17	34.39	17	01:50.7

INSTANCIA	DEMANDA	CARACTERÍSTICA MEJOR SOLUCIÓN 10	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 11	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 12	TIEMPO DE CPU SEG
-----------	---------	----------------------------------	-------------------	----------------------------------	-------------------	----------------------------------	-------------------

3	20	4	19.451	4	01:52.4	4	40.848
49	20	4	01:17.3	4	01:53.5	4	39.358
5	23	5	01:17.0	5	01:45.4	5	38.95
53	24	5	01:20.9	5	01:46.2	5	39.7466
9	26	6	01:18.1	6	01:48.6	6	40.357
12	29	6	01:20.8	6	01:51.7	6	41.512
61	32	7	01:21.7	7	01:54.1	7	42.387
62	32	7	01:21.4	7	01:52.4	7	42.4942
97	33	7	01:21.9	7	01:54.3	7	43.256
64	34	7	01:22.7	7	01:56.7	8	43.459
98	35	8	01:23.9	8	01:56.5	8	43.705
100	36	8	01:24.5	8	01:58.0	7	44.733
21	38	9	01:25.7	9	02:00.1	9	45.152
67	38	8	01:26.2	8	02:00.4	8	45.706
103	39	8	01:26.5	8	02:02.6	8	45.778
104	39	9	01:25.8	9	02:02.4	9	45.159

INSTANCIA	DEMANDA	CARACTERÍSTICA MEJOR SOLUCIÓN 10	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 11	TIEMPO DE CPU SEG	CARACTERÍSTICA MEJOR SOLUCIÓN 12	TIEMPO DE CPU SEG
70	40	9	01:27.4	9	02:02.6	9	45.7624
106	40	9	01:26.5	9	02:02.4	9	46.357
23	41	9	01:28.8	9	02:03.3	9	46.3478
109	41	9	01:28.0	9	02:03.5	9	46.08.38
111	41	8	01:28.6	8	01:03.9	8	46.862
72	42	10	01:27.5	10	02:03.0	10	46.726
25	44	10	01:28.9	10	02:06.7	10	47.978
117	44	10	01:30.7	10	02:07.9	10	48.322
118	44	9	01:30.7	9	02:07.8	9	48.529
119	44	10	01:30.7	10	02:07.2	10	47.8442
123	44	9	01:30.8	9	02:08.4	9	47.687
127	45	10	01:31.2	10	02:08.4	10	48.7526
76	46	11	01:32.0	11	02:10.2	11	49.4482
129	46	10	01:31.8	10	02:10.6	10	49.5256
130	46	10	01:36.2	10	02:11.2	10	48.921
134	47	9	01:33.6	9	02:10.5	9	49.728
137	49	10	01:35.9	10	02:14.6	10	50.9384
31	50	10	1.11.65	11	02:53.3	10	51.774
139	51	10	01:37.0	10	02:14.8	10	51.537
81	52	13	01:36.9	13	02:18.3	13	52.3058
82	52	12	01:35.3	12	02:17.3	12	52.4402
141	52	12	01:36.5	12	02:17.3	12	52.479
35	53	12	01:22.3	12	01:49.5	12	52.776
143	54	11	01:40.8	11	2.21.95	11	53.334
38	56	13	01:41.2	13	02:25.6	13	55.70.60
144	57	12	01:42.4	12	2.25.54	12	55.2588
89	60	15	01:42.2	15	02:29.5	15	56.7544
90	60	15	01:43.0	14	02:28.5	15	56.6764
45	62	15	01:44.2	15	02:34.0	15	57.7046
92	62	16	01:44.8	15	02:33.6	16	58.5368
94	64	17	01:49.6	17	02:47.4	17	01:02.4
47	65	17	01:58.4	17	03:36.3	17	01:07.4

Anexo 6. Diferentes corridas del algoritmo de búsqueda tabú

INSTANCIA 3

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	5	5	5
100	12	10	7	7	5	5	5
250	4	5	7	7	5	5	5
250	12	10	10	10	5	5	5
700	4	10	3	10	5	5	5
700	12	15	7	3	4	4	4
1000	4	15	10	7	4	4	4
1000	12	5	3	10	5	5	5
1500	4	10	10	3	4	4	4
1500	12	15	3	7	4	4	4
2000	4	15	7	10	4	4	4
2000	12	5	10	3	4	4	4

INSTANCIA 5

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	5	5	5
100	12	10	7	7	5	5	5
250	4	5	7	7	5	5	5
250	12	10	10	10	5	5	5
700	4	10	3	10	5	5	5
700	12	15	7	3	5	5	5
1000	4	15	10	7	5	5	5
1000	12	5	3	10	5	5	5
1500	4	10	10	3	5	5	5
1500	12	15	3	7	5	5	5
2000	4	15	7	10	5	5	5
2000	12	5	10	3	5	5	5

INSTANCIA 9

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LISTA TABU	1	2	3
100	4	5	3	3	6	6	6
100	12	10	7	7	6	6	6
250	4	5	7	7	6	6	6
250	12	10	10	10	6	6	6
700	4	10	3	10	6	6	6
700	12	15	7	3	6	6	6
1000	4	15	10	7	6	6	6
1000	12	5	3	10	6	6	6
1500	4	10	10	3	6	6	6
1500	12	15	3	7	6	6	6
2000	4	15	7	10	6	6	6
2000	12	5	10	3	6	6	6

INSTANCIA 12

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	6	6	6
100	12	10	7	7	6	6	6
250	4	5	7	7	6	6	6
250	12	10	10	10	6	6	6
700	4	10	3	10	6	6	6
700	12	15	7	3	6	6	6
1000	4	15	10	7	6	6	6
1000	12	5	3	10	6	6	6
1500	4	10	10	3	6	6	6
1500	12	15	3	7	6	6	6
2000	4	15	7	10	6	6	6
2000	12	5	10	3	6	6	6

INSTANCIA 53

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	5	5	5
100	12	10	7	7	5	5	5
250	4	5	7	7	5	5	5
250	12	10	10	10	5	5	5
700	4	10	3	10	5	5	5
700	12	15	7	3	5	5	5
1000	4	15	10	7	5	5	5
1000	12	5	3	10	5	5	5
1500	4	10	10	3	5	5	5
1500	12	15	3	7	5	5	5
2000	4	15	7	10	5	5	5
2000	12	5	10	3	5	5	5

INSTANCIA 97

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	8	8	8
100	12	10	7	7	8	8	8
250	4	5	7	7	8	8	8
250	12	10	10	10	8	8	8
700	4	10	3	10	7	7	7
700	12	15	7	3	7	7	7
1000	4	15	10	7	8	8	8
1000	12	5	3	10	7	7	7
1500	4	10	10	3	7	7	7
1500	12	15	3	7	7	7	7
2000	4	15	7	10	7	7	7
2000	12	5	10	3	7	7	7

INSTANCIA 98

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	8	8	8
100	12	10	7	7	8	8	8
250	4	5	7	7	8	8	8
250	12	10	10	10	8	8	8
700	4	10	3	10	8	8	8
700	12	15	7	3	8	8	8
1000	4	15	10	7	8	8	8
1000	12	5	3	10	8	8	8
1500	4	10	10	3	8	8	8
1500	12	15	3	7	8	8	8
2000	4	15	7	10	8	8	8
2000	12	5	10	3	8	8	8

INSTANCIA 100

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	8	8	8
100	12	10	7	7	8	8	8
250	4	5	7	7	8	8	8
250	12	10	10	10	8	8	8
700	4	10	3	10	8	8	8
700	12	15	7	3	8	8	8
1000	4	15	10	7	8	8	8
1000	12	5	3	10	8	8	8
1500	4	10	10	3	7	7	7
1500	12	15	3	7	8	8	8
2000	4	15	7	10	8	8	8
2000	12	5	10	3	7	7	7

INSTANCIA 21

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	10	10	10
100	12	10	7	7	9	9	9
250	4	5	7	7	9	9	9
250	12	10	10	10	9	9	9
700	4	10	3	10	9	9	9
700	12	15	7	3	9	9	9
1000	4	15	10	7	9	9	9
1000	12	5	3	10	9	9	9
1500	4	10	10	3	9	9	9
1500	12	15	3	7	9	9	9
2000	4	15	7	10	9	9	9
2000	12	5	10	3	9	9	9

INSTANCIA 67

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	8	8	8
100	12	10	7	7	8	8	8
250	4	5	7	7	8	8	8
250	12	10	10	10	8	8	8
700	4	10	3	10	8	8	8
700	12	15	7	3	8	8	8
1000	4	15	10	7	8	8	8
1000	12	5	3	10	8	8	8
1500	4	10	10	3	8	8	8
1500	12	15	3	7	8	8	8
2000	4	15	7	10	8	8	8
2000	12	5	10	3	8	8	8

INSTANCIA 103

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	8	8	8
100	12	10	7	7	8	8	8
250	4	5	7	7	8	8	8
250	12	10	10	10	8	8	8
700	4	10	3	10	8	8	8
700	12	15	7	3	8	8	8
1000	4	15	10	7	8	8	8
1000	12	5	3	10	8	8	8
1500	4	10	10	3	8	8	8
1500	12	15	3	7	8	8	8
2000	4	15	7	10	8	8	8
2000	12	5	10	3	8	8	8

INSTANCIA 104

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	9	9	9
100	12	10	7	7	9	9	9
250	4	5	7	7	9	9	9
250	12	10	10	10	9	9	9
700	4	10	3	10	9	9	9
700	12	15	7	3	9	9	9
1000	4	15	10	7	9	9	9
1000	12	5	3	10	9	9	9
1500	4	10	10	3	9	9	9
1500	12	15	3	7	9	9	9
2000	4	15	7	10	9	9	9
2000	12	5	10	3	9	9	9

INSTANCIA 70

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	9	9	9
100	12	10	7	7	9	9	9
250	4	5	7	7	9	9	9
250	12	10	10	10	9	9	9
700	4	10	3	10	9	9	9
700	12	15	7	3	9	9	9
1000	4	15	10	7	9	9	9
1000	12	5	3	10	9	9	9
1500	4	10	10	3	9	9	9
1500	12	15	3	7	9	9	9
2000	4	15	7	10	9	9	9
2000	12	5	10	3	9	9	9

INSTANCIA 106

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	9	9	9
100	12	10	7	7	9	9	9
250	4	5	7	7	9	9	9
250	12	10	10	10	9	9	9
700	4	10	3	10	9	9	9
700	12	15	7	3	9	9	9
1000	4	15	10	7	9	9	9
1000	12	5	3	10	9	9	9
1500	4	10	10	3	9	9	9
1500	12	15	3	7	9	9	9
2000	4	15	7	10	9	9	9
2000	12	5	10	3	9	9	9

INSTANCIA 23

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	9	9	9
100	12	10	7	7	10	10	10
250	4	5	7	7	9	9	9
250	12	10	10	10	9	9	9
700	4	10	3	10	9	9	9
700	12	15	7	3	9	9	9
1000	4	15	10	7	9	9	9
1000	12	5	3	10	10	10	10
1500	4	10	10	3	9	9	9
1500	12	15	3	7	9	9	9
2000	4	15	7	10	9	9	9
2000	12	5	10	3	9	9	9

INSTANCIA 109

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	9	9	9
100	12	10	7	7	9	9	9
250	4	5	7	7	9	9	9
250	12	10	10	10	9	9	9
700	4	10	3	10	9	9	9
700	12	15	7	3	9	9	9
1000	4	15	10	7	9	9	9
1000	12	5	3	10	9	9	9
1500	4	10	10	3	8	8	8
1500	12	15	3	7	9	9	9
2000	4	15	7	10	8	8	8
2000	12	5	10	3	9	9	9

INSTANCIA 11

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	8	8	8
100	12	10	7	7	8	8	8
250	4	5	7	7	8	8	8
250	12	10	10	10	8	8	8
700	4	10	3	10	8	8	8
700	12	15	7	3	8	8	8
1000	4	15	10	7	8	8	8
1000	12	5	3	10	8	8	8
1500	4	10	10	3	8	8	8
1500	12	15	3	7	8	8	8
2000	4	15	7	10	8	8	8
2000	12	5	10	3	8	8	8

INSTANCIA 72

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	10	10	10
100	12	10	7	7	11	11	11
250	4	5	7	7	10	10	10
250	12	10	10	10	10	10	10
700	4	10	3	10	10	10	10
700	12	15	7	3	10	10	10
1000	4	15	10	7	10	10	10
1000	12	5	3	10	10	10	10
1500	4	10	10	3	10	10	10
1500	12	15	3	7	10	10	10
2000	4	15	7	10	10	10	10
2000	12	5	10	3	10	10	10

INSTANCIA 25

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	11	11	11
100	12	10	7	7	11	11	11
250	4	5	7	7	11	11	11
250	12	10	10	10	11	11	11
700	4	10	3	10	11	11	11
700	12	15	7	3	10	10	10
1000	4	15	10	7	10	10	10
1000	12	5	3	10	10	10	10
1500	4	10	10	3	10	10	10
1500	12	15	3	7	10	10	10
2000	4	15	7	10	10	10	10
2000	12	5	10	3	10	10	10

INSTANCIA 117

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	10	10	10
100	12	10	7	7	10	10	10
250	4	5	7	7	10	10	10
250	12	10	10	10	10	10	10
700	4	10	3	10	9	9	9
700	12	15	7	3	10	10	10
1000	4	15	10	7	10	10	10
1000	12	5	3	10	10	10	10
1500	4	10	10	3	10	10	10
1500	12	15	3	7	10	10	10
2000	4	15	7	10	10	10	10
2000	12	5	10	3	10	10	10

INSTANCIA 118

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	10	9	9
100	12	10	7	7	9	9	9
250	4	5	7	7	9	9	9
250	12	10	10	10	9	9	9
700	4	10	3	10	9	9	9
700	12	15	7	3	9	9	9
1000	4	15	10	7	9	9	9
1000	12	5	3	10	9	9	9
1500	4	10	10	3	9	9	9
1500	12	15	3	7	9	9	9
2000	4	15	7	10	9	9	9
2000	12	5	10	3	9	9	9

INSTANCIA 119

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	10	10	10
100	12	10	7	7	10	10	10
250	4	5	7	7	10	10	10
250	12	10	10	10	10	10	10
700	4	10	3	10	10	10	10
700	12	15	7	3	10	10	10
1000	4	15	10	7	10	10	10
1000	12	5	3	10	10	10	10
1500	4	10	10	3	10	10	10
1500	12	15	3	7	10	10	10
2000	4	15	7	10	10	10	10
2000	12	5	10	3	10	10	10

INSTANCIA 123

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	9	9	9
100	12	10	7	7	9	9	9
250	4	5	7	7	9	9	9
250	12	10	10	10	9	9	9
700	4	10	3	10	9	9	9
700	12	15	7	3	9	9	9
1000	4	15	10	7	9	9	9
1000	12	5	3	10	9	9	9
1500	4	10	10	3	9	9	9
1500	12	15	3	7	9	9	9
2000	4	15	7	10	9	9	9
2000	12	5	10	3	9	9	9

INSTANCIA 127

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	11	11	11
100	12	10	7	7	10	11	10
250	4	5	7	7	11	10	10
250	12	10	10	10	11	11	11
700	4	10	3	10	10	10	10
700	12	15	7	3	11	11	11
1000	4	15	10	7	10	10	10
1000	12	5	3	10	10	10	10
1500	4	10	10	3	10	10	10
1500	12	15	3	7	10	10	10
2000	4	15	7	10	10	10	10
2000	12	5	10	3	10	10	10

INSTANCIA 76

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	12	12	12
100	12	10	7	7	11	11	11
250	4	5	7	7	12	11	11
250	12	10	10	10	11	11	11
700	4	10	3	10	11	11	11
700	12	15	7	3	11	11	11
1000	4	15	10	7	11	11	11
1000	12	5	3	10	11	11	11
1500	4	10	10	3	11	11	11
1500	12	15	3	7	11	11	11
2000	4	15	7	10	11	11	11
2000	12	5	10	3	11	11	11

INSTANCIA 129

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	11	11	11
100	12	10	7	7	10	11	10
250	4	5	7	7	10	10	10
250	12	10	10	10	10	10	10
700	4	10	3	10	10	10	10
700	12	15	7	3	10	10	10
1000	4	15	10	7	10	10	10
1000	12	5	3	10	10	10	10
1500	4	10	10	3	10	10	10
1500	12	15	3	7	10	10	10
2000	4	15	7	10	10	10	10
2000	12	5	10	3	10	10	10

INSTANCIA 130

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	10	10	10
100	12	10	7	7	10	10	10
250	4	5	7	7	10	10	10
250	12	10	10	10	10	10	10
700	4	10	3	10	10	10	10
700	12	15	7	3	10	10	10
1000	4	15	10	7	10	10	10
1000	12	5	3	10	10	10	10
1500	4	10	10	3	10	10	10
1500	12	15	3	7	10	10	10
2000	4	15	7	10	10	10	10
2000	12	5	10	3	10	10	10

INSTANCIA 134

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	10	10	10
100	12	10	7	7	10	10	10
250	4	5	7	7	10	10	10
250	12	10	10	10	10	10	10
700	4	10	3	10	10	10	10
700	12	15	7	3	10	10	10
1000	4	15	10	7	10	10	10
1000	12	5	3	10	9	9	9
1500	4	10	10	3	9	9	9
1500	12	15	3	7	9	9	9
2000	4	15	7	10	9	9	9
2000	12	5	10	3	9	9	9

INSTANCIA 137

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	10	10	10
100	12	10	7	7	10	10	10
250	4	5	7	7	10	10	10
250	12	10	10	10	10	10	10
700	4	10	3	10	10	10	10
700	12	15	7	3	10	10	10
1000	4	15	10	7	10	10	10
1000	12	5	3	10	10	10	10
1500	4	10	10	3	10	10	10
1500	12	15	3	7	10	10	10
2000	4	15	7	10	10	10	10
2000	12	5	10	3	10	10	10

INSTANCIA 31

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	11	11	11
100	12	10	7	7	11	11	11
250	4	5	7	7	11	11	11
250	12	10	10	10	11	11	11
700	4	10	3	10	10	10	10
700	12	15	7	3	10	10	10
1000	4	15	10	7	11	10	10
1000	12	5	3	10	11	11	10
1500	4	10	10	3	11	10	10
1500	12	15	3	7	10	10	10
2000	4	15	7	10	11	11	11
2000	12	5	10	3	11	10	10

INSTANCIA 139

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	11	11	11
100	12	10	7	7	11	11	11
250	4	5	7	7	10	10	10
250	12	10	10	10	10	10	10
700	4	10	3	10	10	10	10
700	12	15	7	3	10	10	10
1000	4	15	10	7	10	10	10
1000	12	5	3	10	10	10	10
1500	4	10	10	3	10	10	10
1500	12	15	3	7	10	10	10
2000	4	15	7	10	10	10	10
2000	12	5	10	3	10	10	10

INSTANCIA 81

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	14	14	14
100	12	10	7	7	13	13	13
250	4	5	7	7	13	13	13
250	12	10	10	10	13	13	13
700	4	10	3	10	13	13	13
700	12	15	7	3	13	13	13
1000	4	15	10	7	13	13	13
1000	12	5	3	10	13	13	13
1500	4	10	10	3	13	13	13
1500	12	15	3	7	13	13	13
2000	4	15	7	10	13	13	13
2000	12	5	10	3	13	13	13

INSTANCIA 82

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	12	12	12
100	12	10	7	7	12	12	12
250	4	5	7	7	13	12	12
250	12	10	10	10	12	12	12
700	4	10	3	10	12	12	12
700	12	15	7	3	12	12	12
1000	4	15	10	7	12	12	12
1000	12	5	3	10	12	12	12
1500	4	10	10	3	12	12	12
1500	12	15	3	7	12	12	12
2000	4	15	7	10	12	12	12
2000	12	5	10	3	12	12	12

INSTANCIA 141

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	12	12	12
100	12	10	7	7	12	12	12
250	4	5	7	7	13	12	12
250	12	10	10	10	12	12	12
700	4	10	3	10	12	12	12
700	12	15	7	3	12	12	12
1000	4	15	10	7	12	12	12
1000	12	5	3	10	12	12	12
1500	4	10	10	3	12	12	12
1500	12	15	3	7	12	12	12
2000	4	15	7	10	12	12	12
2000	12	5	10	3	12	12	12

INSTANCIA 35

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	14	13	13
100	12	10	7	7	13	13	13
250	4	5	7	7	13	13	13
250	12	10	10	10	12	12	12
700	4	10	3	10	13	13	13
700	12	15	7	3	13	13	13
1000	4	15	10	7	13	12	12
1000	12	5	3	10	13	13	13
1500	4	10	10	3	12	12	12
1500	12	15	3	7	12	12	12
2000	4	15	7	10	12	12	12
2000	12	5	10	3	12	12	12

INSTANCIA 143

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	11	12	11
100	12	10	7	7	11	11	12
250	4	5	7	7	12	11	11
250	12	10	10	10	11	11	11
700	4	10	3	10	11	11	11
700	12	15	7	3	11	11	11
1000	4	15	10	7	11	11	11
1000	12	5	3	10	11	11	11
1500	4	10	10	3	11	11	11
1500	12	15	3	7	11	11	11
2000	4	15	7	10	11	11	11
2000	12	5	10	3	11	11	11

INSTANCIA 38

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	14	14	14
100	12	10	7	7	14	14	14
250	4	5	7	7	14	14	14
250	12	10	10	10	14	13	13
700	4	10	3	10	13	13	13
700	12	15	7	3	14	13	13
1000	4	15	10	7	14	14	13
1000	12	5	3	10	14	14	14
1500	4	10	10	3	13	13	13
1500	12	15	3	7	13	13	13
2000	4	15	7	10	13	13	13
2000	12	5	10	3	14	13	13

INSTANCIA 144

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	13	13	13
100	12	10	7	7	13	13	13
250	4	5	7	7	13	13	13
250	12	10	10	10	13	13	13
700	4	10	3	10	12	12	12
700	12	15	7	3	13	13	13
1000	4	15	10	7	12	12	12
1000	12	5	3	10	12	12	12
1500	4	10	10	3	12	12	12
1500	12	15	3	7	12	12	12
2000	4	15	7	10	12	12	12
2000	12	5	10	3	12	12	12

INSTANCIA 89

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	16	16	16
100	12	10	7	7	15	16	15
250	4	5	7	7	16	16	15
250	12	10	10	10	16	15	15
700	4	10	3	10	15	15	16
700	12	15	7	3	15	15	15
1000	4	15	10	7	15	15	15
1000	12	5	3	10	16	16	16
1500	4	10	10	3	15	15	15
1500	12	15	3	7	15	15	15
2000	4	15	7	10	15	15	15
2000	12	5	10	3	15	15	15

INSTANCIA 90

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	15	15	15
100	12	10	7	7	16	15	16
250	4	5	7	7	15	15	16
250	12	10	10	10	15	15	15
700	4	10	3	10	15	15	15
700	12	15	7	3	15	15	15
1000	4	15	10	7	15	15	15
1000	12	5	3	10	15	15	15
1500	4	10	10	3	14	15	15
1500	12	15	3	7	15	15	15
2000	4	15	7	10	14	15	14
2000	12	5	10	3	15	15	15

INSTANCIA 45

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	16	15	15
100	12	10	7	7	16	16	16
250	4	5	7	7	15	15	15
250	12	10	10	10	16	15	15
700	4	10	3	10	16	15	16
700	12	15	7	3	15	15	15
1000	4	15	10	7	15	15	15
1000	12	5	3	10	15	15	15
1500	4	10	10	3	15	15	15
1500	12	15	3	7	15	15	15
2000	4	15	7	10	15	15	15
2000	12	5	10	3	15	15	15

INSTANCIA 92

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	16	16	16
100	12	10	7	7	16	16	16
250	4	5	7	7	16	16	16
250	12	10	10	10	16	15	15
700	4	10	3	10	16	15	15
700	12	15	7	3	16	16	16
1000	4	15	10	7	16	16	16
1000	12	5	3	10	16	16	16
1500	4	10	10	3	16	16	16
1500	12	15	3	7	16	16	16
2000	4	15	7	10	15	15	15
2000	12	5	10	3	16	16	16

INSTANCIA 94

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	17	17	17
100	12	10	7	7	17	17	17
250	4	5	7	7	18	18	18
250	12	10	10	10	17	17	17
700	4	10	3	10	17	17	17
700	12	15	7	3	17	17	17
1000	4	15	10	7	16	16	16
1000	12	5	3	10	17	17	17
1500	4	10	10	3	17	17	17
1500	12	15	3	7	17	17	17
2000	4	15	7	10	17	17	17
2000	12	5	10	3	17	17	17

INSTANCIA 47

Número de iteraciones Tabú	No. Iteraciones para regresar a solución glotona	Número Candidatos	Número Iteraciones Bloqueadas	Número de elementos en LT	1	2	3
100	4	5	3	3	18	18	18
100	12	10	7	7	18	18	18
250	4	5	7	7	17	17	17
250	12	10	10	10	17	17	17
700	4	10	3	10	17	17	17
700	12	15	7	3	17	17	17
1000	4	15	10	7	17	17	17
1000	12	5	3	10	17	17	17
1500	4	10	10	3	17	17	17
1500	12	15	3	7	17	17	17
2000	4	15	7	10	17	17	17
2000	12	5	10	3	17	17	17

Anexo 7. Instancias para ejecutar los algoritmos

Instancia Numero	3	49	5	53	9	12	61	62	97	64	98	100	21	67	103
Demanda total	20	20	23	24	26	29	32	32	33	34	35	36	38	38	39
A	1	1	1	1	1	2	1	2	1	2	1	1	4	4	1
B	2	2	2	2	2	2	4	3	2	2	2	3	4	3	2
C	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
D	1	1	1	1	2	2	1	4	3	1	3	2	3	1	3
E	1	2	1	1	1	1	2	3	1	3	1	3	2	4	2
F	1	1	1	1	2	1	2	1	2	3	2	4	3	4	3
G	1	1	1	1	2	1	2	3	3	2	3	1	3	1	3
H	1	1	2	4	1	1	1	3	1	3	1	3	2	3	2
I	3	2	2	2	4	2	4	2	5	4	6	2	2	3	4
J	1	2	2	3	1	2	3	1	4	3	4	9	4	1	7
K	4	3	3	3	4	3	6	5	6	6	5	3	6	3	3
L	3	3	6	4	5	11	5	4	4	4	6	4	4	10	8

Instancia Numero	104	70	106	23	109	111	72	25	117	118	119	123	127	76	129
Demanda total	39	40	40	41	41	41	42	44	44	44	44	44	45	46	46
A	1	1	2	3	1	1	5	3	3	1	1	1	2	3	3
B	2	2	2	2	4	3	2	5	3	2	4	4	2	5	3
C	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
D	3	3	4	3	1	2	3	3	2	3	3	3	4	6	2
E	4	5	2	3	2	3	3	2	2	2	3	3	1	2	3
F	3	1	2	2	4	5	4	4	4	4	4	4	5	3	1
G	3	4	2	2	2	1	4	4	2	1	3	1	3	3	2
H	1	3	3	4	3	2	4	5	4	2	2	2	1	3	2
I	6	3	6	4	3	2	2	5	4	5	6	3	6	4	5
J	3	6	1	1	8	5	3	2	5	9	6	4	9	8	6
K	7	4	10	11	8	5	7	4	8	7	5	7	5	3	9
L	5	7	5	5	4	11	4	6	6	7	6	11	6	5	9

Instancia Numero	130	134	137	31	139	81	82	141	35	143	38	144	89	90	45
Demanda total	46	47	49	50	51	52	52	52	53	54	56	57	60	60	62
A	3	1	2	1	3	6	6	2	2	2	3	3	6	5	1
B	3	3	4	7	2	6	5	4	8	4	5	3	2	3	2
C	1	1	1	2	1	1	1	1	2	1	1	1	1	1	1
D	2	3	4	3	2	4	2	3	4	2	6	3	5	2	5
E	4	3	2	1	2	3	4	3	4	2	7	2	5	7	6
F	2	4	5	1	4	2	1	5	6	4	1	3	3	5	5
G	1	1	1	3	1	6	5	2	4	2	5	2	5	4	4
H	4	4	1	5	3	3	4	4	6	2	2	3	5	4	1
I	6	5	2	5	3	4	6	5	3	6	3	6	4	7	10
J	2	8	7	6	9	2	1	10	4	7	8	10	4	8	12
K	7	3	10	4	9	5	5	8	5	8	7	8	10	5	3
L	11	11	10	12	12	10	12	5	5	14	8	13	10	9	12

Instancia Numero	92	94	47
Demanda total	62	64	65
A	5	4	6
B	2	6	6
C	1	1	1
D	4	4	4
E	5	4	8
F	1	6	6
G	3	6	5
H	4	2	1
I	9	8	2
J	7	6	7
K	10	7	6
L	11	10	13

Anexo 8. Horarios de ventana de atención

CD	1	2	3	4	5	6	7	8	9	10	11	12	13
A			04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
			-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
					1	1	1	1	1	1	1	1	1
B			04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
			-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
						1	1	1	1	1	1	1	1
C	03:20	03:40	04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
	1	1	1	1	1	1	1	1	1	1	1	1	1
D			04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
			-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
					1	1	1	1	1	1	1	1	1
E			04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
			-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
						1	1	1	1	1	1	1	1
F	03:20	03:40	04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
	1	1									1	1	1
G			04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
			-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
									1	1	1	1	1
H	03:20	03:40	04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
	1	1	1								1	1	1
I			04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
			-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
					1	1	1	1	1	1	1	1	1
J	03:20	03:40	04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
	1	1	1	1	1						1	1	1
K			04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
			-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
										1	1	1	1
L	03:20	03:40	04:00	04:20	04:40	05:00	05:20	05:40	06:00	06:20	06:40	07:00	07:20
	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
	1	1											1

CD	14	15	16	17	18	19	20	21	22	23	24	25	26
A	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1
B	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1
C	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1
D	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1
E	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1
F	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1
G	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1
H	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1
I	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1
J	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1
K	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1
L	07:40	08:00	08:20	08:40	09:00	09:20	09:40	10:00	10:20	10:40	11:00	11:20	11:40
	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1	1	1	1	1	1	1	1

CD	27	28	29	30	31	32	33	34	35	36	37	38	39
A	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
B	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
	1	1	1	1									
C	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
D	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
	1	1	1	1									
E	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
	1	1	1	1									
F	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
	1	1	1	1	1	1							
G	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
	1	1	1	1	1	1	1	1	1	1			
H	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
	1	1	1	1	1	1							
I	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
	1	1	1										
J	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
						1	1	1	1	1	1	1	1
K	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
	1	1	1	1	1	1	1	1	1	1	1	1	1
L	12:00	12:20	12:40	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
	16	17	18	19	20	21	22	23	24	25	26	27	28
	1	1	1	1	1						1	1	1

CD	40	41	42	43	44	45	46	47	48	49	50	51	52
A	16:20	16:40	17:00	17:20	17:40	18:00	18:20	18:40	19:00	19:20	19:40	20:00	20:20
	29	30	31	32	33	34	35	36	37	38	39	40	41
B	16:20	16:40	17:00	17:20	17:40	18:00	18:20	18:40	19:00	19:20	19:40	20:00	20:20
	29	30	31	32	33	34	35	36	37	38	39	40	41
C	16:20	16:40	17:00	17:20	17:40	18:00	18:20	18:40	19:00	19:20	19:40	20:00	20:20
	29	30	31	32	33	34	35	36	37	38	39	40	41
D	16:20	16:40	17:00	17:20	17:40	18:00	18:20	18:40	19:00	19:20	19:40	20:00	20:20
	29	30	31	32	33	34	35	36	37	38	39	40	41
E	16:20	16:40	17:00	17:20	17:40	18:00	18:20	18:40	19:00	19:20	19:40	20:00	20:20
	29	30	31	32	33	34	35	36	37	38	39	40	41
F	16:20	16:40	17:00	17:20	17:40	18:00	18:20	18:40	19:00	19:20	19:40	20:00	20:20
	29	30	31	32	33	34	35	36	37	38	39	40	41
G				1									
	16:20	16:40	17:00	17:20	17:40	18:00	18:20	18:40	19:00	19:20	19:40	20:00	20:20
H	29	30	31	32	33	34	35	36	37	38	39	40	41
	16:20	16:40	17:00	17:20	17:40	18:00	18:20	18:40	19:00	19:20	19:40	20:00	20:20
I	29	30	31	32	33	34	35	36	37	38	39	40	41
	16:20	16:40	17:00	17:20	17:40	18:00	18:20	18:40	19:00	19:20	19:40	20:00	20:20
J	29	30	31	32	33	34	35	36	37	38	39	40	41
	1	1											
K	16:20	16:40	17:00	17:20	17:40	18:00	18:20	18:40	19:00	19:20	19:40	20:00	20:20
	29	30	31	32	33	34	35	36	37	38	39	40	41
L	1	1	1	1									
	16:20	16:40	17:00	17:20	17:40	18:00	18:20	18:40	19:00	19:20	19:40	20:00	20:20
	29	30	31	32	33	34	35	36	37	38	39	40	41
	1	1	1	1	1	1	1	1	1	1	1		

CD	53	54	55	56	57	58	59	60	61	62	63	64	65
A	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
B	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
C	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
D	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
E	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
F	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
G	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
H	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
I	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
J	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
K	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
L	20:40	21:00	21:20	21:40	22:00	22:20	22:40	23:00	23:20	23:40	00:00	00:20	00:40
	42	43	44	45	46	47	48	49	50	51	52	53	54
									1	1	1	1	1

CD	66	67	68	69	70	71	72							
A	01:00	01:20	01:40	02:00	02:20	02:40	03:00	03:20	03:40	04:00	04:20	04:40	05:00	
	55	56	57	58	59	60	61	62	63	64	65	66	67	
B	01:00	01:20	01:40	02:00	02:20	02:40	03:00	03:20	03:40	04:00	04:20	04:40	05:00	
	55	56	57	58	59	60	61	62	63	64	65	66	67	
C	01:00	01:20	01:40	02:00	02:20	02:40	03:00	03:20	03:40	04:00	04:20	04:40	05:00	
	55	56	57	58	59	60	61	62	63	64	65	66	67	
D	01:00	01:20	01:40	02:00	02:20	02:40	03:00	03:20	03:40	04:00	04:20	04:40	05:00	
	55	56	57	58	59	60	61	62	63	64	65	66	67	
E	01:00	01:20	01:40	02:00	02:20	02:40	03:00	03:20	03:40	04:00	04:20	04:40	05:00	
	55	56	57	58	59	60	61	62	63	64	65	66	67	
F	01:00	01:20	01:40	02:00	02:20	02:40	03:00	03:20	03:40	04:00	04:20	04:40	05:00	
	55	56	57	58	59	60	61	62	63	64	65	66	67	
G	1	1	1	1	1	1	1							
	01:00	01:20	01:40	02:00	02:20	02:40	03:00	03:20	03:40	04:00	04:20	04:40	05:00	
H	55	56	57	58	59	60	61	62	63	64	65	66	67	
	1	1	1	1	1	1	1							
I	01:00	01:20	01:40	02:00	02:20	02:40	03:00	03:20	03:40	04:00	04:20	04:40	05:00	
	55	56	57	58	59	60	61	62	63	64	65	66	67	
J	1	1	1	1	1	1								
	01:00	01:20	01:40	02:00	02:20	02:40	03:00	03:20	03:40	04:00	04:20	04:40	05:00	
K	55	56	57	58	59	60	61	62	63	64	65	66	67	
	1	1	1	1	1	1								
L	01:00	01:20	01:40	02:00	02:20	02:40	03:00	03:20	03:40	04:00	04:20	04:40	05:00	
	55	56	57	58	59	60	61	62	63	64	65	66	67	
	1	1	1	1	1	1	1							

CD			
A	05:20	05:40	06:00
	68	69	70
B	05:20	05:40	06:00
	68	69	70
C	05:20	05:40	06:00
	68	69	70
D	05:20	05:40	06:00
	68	69	70
E	05:20	05:40	06:00
	68	69	70
F	05:20	05:40	06:00
	68	69	70
G	05:00	05:40	06:00
	68	69	70
H	05:20	05:40	06:00
	68	69	70
I	05:20	05:40	06:00
	68	69	70
J	05:20	05:40	06:00
	68	69	70
K	05:20	05:40	06:00
	68	69	70
L	05:20	05:40	06:00
	68	69	70

Anexo 9. Análisis de pronósticos de cada uno de los algoritmos

CARACTERÍSTICA TABÚ 7

ANÁLISIS ESTADÍSTICO PARA PRONOSTICOS EN BASE A ECUACION DE REGRESIÓN CON CARACTERÍSTICA TABU 1						
ECUACION	Vehículos = - 1.68 + 0.272*Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.76	4	0	0	0
20	4	3.76	4	0	0	0
23	5	4.576	5	0	0	0
24	5	4.848	5	0	0	0
26	6	5.392	6	0	0	0
29	5	6.208	7	-2	2	4
32	7	7.024	8	-1	1	1
32	7	7.024	8	-1	1	1
33	7	7.296	8	-1	1	1
34	7	7.568	8	-1	1	1
35	8	7.84	8	0	0	0
36	7	8.112	9	-2	2	4
38	9	8.656	9	0	0	0
38	8	8.656	9	-1	1	1
39	8	8.928	9	-1	1	1
39	9	8.928	9	0	0	0
40	8	9.2	10	-2	2	4
40	9	9.2	10	-1	1	1
41	9	9.472	10	-1	1	1
41	8	9.472	10	-2	2	4
41	7	9.472	10	-3	3	9
42	10	9.744	10	0	0	0
44	10	10.288	11	-1	1	1
44	9	10.288	11	-2	2	4
44	9	10.288	11	-2	2	4
44	9	10.288	11	-2	2	4
44	8	10.288	11	-3	3	9
45	9	10.56	11	-2	2	4
46	10	10.832	11	-1	1	1
46	9	10.832	11	-2	2	4
46	9	10.832	11	-2	2	4
47	9	11.104	12	-3	3	9
49	10	11.648	12	-2	2	4
50	10	11.92	12	-2	2	4
51	10	12.192	13	-3	3	9
52	12	12.464	13	-1	1	1
52	11	12.464	13	-2	2	4
52	11	12.464	13	-2	2	4
53	12	12.736	13	-1	1	1
54	10	13.008	14	-4	4	16
56	12	13.552	14	-2	2	4
57	11	13.824	14	-3	3	9
60	13	14.64	15	-2	2	4

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
60	13	14.64	15	-2	2	4
62	13	15.184	16	-3	3	9
62	13	15.184	16	-3	3	9
64	14	15.728	16	-2	2	4
65	15	16	16	-1	1	1
66		16.272	17			
67		16.544	17			
68		16.816	17			
69		17.088	18			
70		17.36	18			
MEDIA DE ERROR					1.54166667	

CARACTERÍSTICA 2

ANÁLISIS ESTADÍSTICO PARA PRONÓSTICOS EN BASE A ECUACIÓN DE REGRESIÓN CON CARACTERÍSTICA TABÚ 2						
ECUACIÓN	Vehículos = - 1.93 + 0.282 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.71	4	0	0	0
20	4	3.71	4	0	0	0
23	5	4.556	5	0	0	0
24	5	4.838	5	0	0	0
26	6	5.402	6	0	0	0
29	5	6.248	7	-2	2	4
32	7	7.094	8	-1	1	1
32	7	7.094	8	-1	1	1
33	7	7.376	8	-1	1	1
34	7	7.658	8	-1	1	1
35	8	7.94	8	0	0	0
36	7	8.222	9	-2	2	4
38	9	8.786	9	0	0	0
38	8	8.786	9	-1	1	1
39	8	9.068	10	-2	2	4
39	9	9.068	10	-1	1	1
40	8	9.35	10	-2	2	4
40	9	9.35	10	-1	1	1
41	9	9.632	10	-1	1	1
41	8	9.632	10	-2	2	4
41	7	9.632	10	-3	3	9
42	10	9.914	10	0	0	0
44	10	10.478	11	-1	1	1
44	9	10.478	11	-2	2	4
44	9	10.478	11	-2	2	4
44	9	10.478	11	-2	2	4
44	8	10.478	11	-3	3	9

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
45	9	10.76	11	-2	2	4
46	10	11.042	12	-2	2	4
46	9	11.042	12	-3	3	9
46	9	11.042	12	-3	3	9
47	9	11.324	12	-3	3	9
49	10	11.888	12	-2	2	4
50	10	12.17	13	-3	3	9
51	10	12.452	13	-3	3	9
52	12	12.734	13	-1	1	1
52	11	12.734	13	-2	2	4
52	11	12.734	13	-2	2	4
53	12	13.016	14	-2	2	4
54	10	13.298	14	-4	4	16
56	12	13.862	14	-2	2	4
57	11	14.144	15	-4	4	16
60	13	14.99	15	-2	2	4
60	13	14.99	15	-2	2	4
62	13	15.554	16	-3	3	9
62	13	15.554	16	-3	3	9
64	14	16.118	17	-3	3	9
65	15	16.4	17	-2	2	4
66		16.682	17			
67		16.964	17			
68		17.246	18			
69		17.528	18			
70		17.81	18			
MEDIA DE ERROR					1.75	

CARACTERÍSTICA 3

ANÁLISIS ESTADÍSTICO PARA PRONÓSTICOS EN BASE A ECUACIÓN DE REGRESIÓN CON CARACTERÍSTICA TABÚ 3						
ECUACIÓN	Vehículos =- 1.65 + 0.268 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.71	4	0	0	0
20	4	3.71	4	0	0	0
23	5	4.514	5	0	0	0
24	5	4.782	5	0	0	0
26	6	5.318	6	0	0	0
29	5	6.122	7	-2	2	4
32	7	6.926	7	0	0	0
32	7	6.926	7	0	0	0
33	7	7.194	8	-1	1	1
34	7	7.462	8	-1	1	1
35	8	7.73	8	0	0	0
36	7	7.998	8	-1	1	1
38	9	8.534	9	0	0	0
38	8	8.534	9	-1	1	1
39	8	8.802	9	-1	1	1
39	9	8.802	9	0	0	0
40	8	9.07	10	-2	2	4
40	9	9.07	10	-1	1	1
41	9	9.338	10	-1	1	1
41	8	9.338	10	-2	2	4

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
41	7	9.338	10	-3	3	9
42	10	9.606	10	0	0	0
44	10	10.142	11	-1	1	1
44	9	10.142	11	-2	2	4
44	9	10.142	11	-2	2	4
44	9	10.142	11	-2	2	4
44	8	10.142	11	-3	3	9
45	9	10.41	11	-2	2	4
46	10	10.678	11	-1	1	1
46	9	10.678	11	-2	2	4
46	9	10.678	11	-2	2	4
47	9	10.946	11	-2	2	4
49	10	11.482	12	-2	2	4
50	10	11.75	12	-2	2	4
51	10	12.018	13	-3	3	9
52	12	12.286	13	-1	1	1
52	11	12.286	13	-2	2	4
52	11	12.286	13	-2	2	4
53	12	12.554	13	-1	1	1
54	10	12.822	13	-3	3	9
56	12	13.358	14	-2	2	4
57	11	13.626	14	-3	3	9
60	13	14.43	15	-2	2	4
60	13	14.43	15	-2	2	4
62	13	14.966	16	-3	3	9
62	13	14.966	16	-3	3	9
64	14	15.502	16	-2	2	4
65	15	15.77	16	-1	1	1
66		16.038	17			
67		16.306	17			
68		16.574	17			
69		16.842	17			
70		17.11	18			
MEDIA DE ERROR					1.4375	

CARACTERÍSTICA 4

ANÁLISIS ESTADÍSTICO PARA PRONOSTICOS EN BASE A ECUACION DE REGRESIÓN CON CARACTERÍSTICA TABU 4						
ECUACION	Vehículos = - 1.23 + 0.257 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.91	4	0	0	0
20	4	3.91	4	0	0	0
23	5	4.681	5	0	0	0
24	5	4.938	5	0	0	0
26	6	5.452	6	0	0	0
29	5	6.223	7	-2	2	4
32	7	6.994	7	0	0	0
32	7	6.994	7	0	0	0
33	7	7.251	8	-1	1	1

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
34	7	7.508	8	-1	1	1
35	8	7.765	8	0	0	0
36	7	8.022	9	-2	2	4
38	9	8.536	9	0	0	0
38	8	8.536	9	-1	1	1
39	8	8.793	9	-1	1	1
39	9	8.793	9	0	0	0
40	8	9.05	10	-2	2	4
40	9	9.05	10	-1	1	1
41	9	9.307	10	-1	1	1
41	8	9.307	10	-2	2	4
41	7	9.307	10	-3	3	9
42	10	9.564	10	0	0	0
44	10	10.078	11	-1	1	1
44	9	10.078	11	-2	2	4
44	9	10.078	11	-2	2	4
44	9	10.078	11	-2	2	4
44	8	10.078	11	-3	3	9
45	9	10.335	11	-2	2	4
46	10	10.592	11	-1	1	1
46	9	10.592	11	-2	2	4
46	9	10.592	11	-2	2	4
47	9	10.849	11	-2	2	4
49	10	11.363	12	-2	2	4
50	10	11.62	12	-2	2	4
51	10	11.877	12	-2	2	4
52	12	12.134	13	-1	1	1
52	11	12.134	13	-2	2	4
52	11	12.134	13	-2	2	4
53	12	12.391	13	-1	1	1
54	10	12.648	13	-3	3	9
56	12	13.162	14	-2	2	4
57	11	13.419	14	-3	3	9
60	13	14.19	15	-2	2	4
60	13	14.19	15	-2	2	4
62	13	14.704	15	-2	2	4
62	13	14.704	15	-2	2	4
64	14	15.218	16	-2	2	4
65	15	15.475	16	-1	1	1
66		15.732	16			
67		15.989	16			
68		16.246	17			
69		16.503	17			
70		16.76	17			
MEDIA DE ERROR					1.395833333	

CARACTERÍSTICA 5

ANÁLISIS ESTADÍSTICO PARA PRONÓSTICOS EN BASE A ECUACION DE REGRESIÓN CON CARACTERÍSTICA TABU 5						
ECUACION	Vehículos = - 1.48 + 0.260 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.72	4	0	0	0
20	4	3.72	4	0	0	0
23	5	4.5	5	0	0	0
24	5	4.76	5	0	0	0
26	6	5.28	6	0	0	0
29	5	6.06	7	-2	2	4
32	7	6.84	7	0	0	0
32	7	6.84	7	0	0	0
33	7	7.1	8	-1	1	1
34	7	7.36	8	-1	1	1
35	8	7.62	8	0	0	0
36	7	7.88	8	-1	1	1
38	9	8.4	9	0	0	0
38	8	8.4	9	-1	1	1
39	8	8.66	9	-1	1	1
39	9	8.66	9	0	0	0
40	8	8.92	9	-1	1	1
40	9	8.92	9	0	0	0
41	9	9.18	10	-1	1	1
41	8	9.18	10	-2	2	4
41	7	9.18	10	-3	3	9
42	10	9.44	10	0	0	0
44	10	9.96	10	0	0	0
44	9	9.96	10	-1	1	1
44	9	9.96	10	-1	1	1
44	9	9.96	10	-1	1	1
44	8	9.96	10	-2	2	4
45	9	10.22	11	-2	2	4
46	10	10.48	11	-1	1	1
46	9	10.48	11	-2	2	4
46	9	10.48	11	-2	2	4
47	9	10.74	11	-2	2	4
49	10	11.26	12	-2	2	4
50	10	11.52	12	-2	2	4
51	10	11.78	12	-2	2	4
52	12	12.04	13	-1	1	1
52	11	12.04	13	-2	2	4
52	11	12.04	13	-2	2	4

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
53	12	12.3	13	-1	1	1
54	10	12.56	13	-3	3	9
56	12	13.08	14	-2	2	4
57	11	13.34	14	-3	3	9
60	13	14.12	15	-2	2	4
60	13	14.12	15	-2	2	4
62	13	14.64	15	-2	2	4
62	13	14.64	15	-2	2	4
64	14	15.16	16	-2	2	4
65	15	15.42	16	-1	1	1
66		15.68	16			
67		15.94	16			
68		16.2	17			
69		16.46	17			
70		16.72	17			
				MEDIA DE ERROR	1.229166667	

CARACTERÍSTICA 6

ANÁLISIS ESTADÍSTICO PARA PRONOSTICOS EN BASE A ECUACION DE REGRESIÓN CON CARACTERÍSTICA TABU 6						
ECUACION	Vehículos = - 1.66 + 0.266 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.66	4	0	0	0
20	4	3.66	4	0	0	0
23	5	4.458	5	0	0	0
24	5	4.724	5	0	0	0
26	6	5.256	6	0	0	0
29	5	6.054	7	-2	2	4
32	7	6.852	7	0	0	0
32	7	6.852	7	0	0	0
33	7	7.118	8	-1	1	1
34	7	7.384	8	-1	1	1
35	8	7.65	8	0	0	0
36	7	7.916	8	-1	1	1
38	9	8.448	9	0	0	0
38	8	8.448	9	-1	1	1
39	8	8.714	9	-1	1	1
39	9	8.714	9	0	0	0
40	8	8.98	9	-1	1	1
40	9	8.98	9	0	0	0
41	9	9.246	10	-1	1	1

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
41	8	9.246	10	-2	2	4
41	7	9.246	10	-3	3	9
42	10	9.512	10	0	0	0
44	10	10.044	11	-1	1	1
44	9	10.044	11	-2	2	4
44	9	10.044	11	-2	2	4
44	9	10.044	11	-2	2	4
44	8	10.044	11	-3	3	9
45	9	10.31	11	-2	2	4
46	10	10.576	11	-1	1	1
46	9	10.576	11	-2	2	4
46	9	10.576	11	-2	2	4
47	9	10.842	11	-2	2	4
49	10	11.374	12	-2	2	4
50	10	11.64	12	-2	2	4
51	10	11.906	12	-2	2	4
52	12	12.172	13	-1	1	1
52	11	12.172	13	-2	2	4
52	11	12.172	13	-2	2	4
53	12	12.438	13	-1	1	1
54	10	12.704	13	-3	3	9
56	12	13.236	14	-2	2	4
57	11	13.502	14	-3	3	9
60	13	14.3	15	-2	2	4
60	13	14.3	15	-2	2	4
62	13	14.832	15	-2	2	4
62	13	14.832	15	-2	2	4
64	14	15.364	16	-2	2	4
65	15	15.63	16	-1	1	1
66		15.896	16			
67		16.162	17			
68		16.428	17			
69		16.694	17			
70		16.96	17			
				MEDIA DE ERROR	1.33333333	

CARACTERÍSTICA 7

ANÁLISIS ESTADÍSTICO PARA PRONOSTICOS EN BASE A ECUACION DE REGRESIÓN CON CARACTERÍSTICA TABU 7						
ECUACION	Vehículos = - 1.58 + 0.261 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	0	4	0	0	0
20	4	3.64	4	0	0	0
23	5	4.423	5	0	1	0
24	5	4.684	5	0	0	0
26	6	5.206	6	0	1	0
29	5	5.989	6	-1	1	1
32	7	6.772	7	0	1	0
32	7	6.772	7	0	1	0
33	7	7.033	8	-1	1	1

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
34	7	7.294	8	-1	1	1
35	8	7.555	8	0	1	0
36	7	7.816	8	-1	1	1
38	9	8.338	9	0	1	0
38	8	8.338	9	-1	1	1
39	8	8.599	9	-1	1	1
39	9	8.599	9	0	1	0
40	8	8.86	9	-1	1	1
40	9	8.86	9	0	1	0
41	9	9.121	10	-1	1	1
41	8	9.121	10	-2	1	4
41	7	9.121	10	-3	3	9
42	10	9.382	10	0	1	0
44	10	9.904	10	0	1	0
44	9	9.904	10	-1	1	1
44	9	9.904	10	-1	1	1
44	9	9.904	10	-1	1	1
44	8	9.904	10	-2	2	4
45	9	10.165	11	-2	2	4
46	10	10.426	11	-1	1	1
46	9	10.426	11	-2	2	4
46	9	10.426	11	-2	2	4
47	9	10.687	11	-2	2	4
49	10	11.209	12	-2	2	4
50	10	11.47	12	-2	2	4
51	10	11.731	12	-2	2	4
52	12	11.992	12	0	1	0
52	11	11.992	12	-1	1	1
52	11	11.992	12	-1	1	1
53	12	12.253	13	-1	1	1
54	10	12.514	13	-3	3	9
56	12	13.036	14	-2	2	4
57	11	13.297	14	-3	3	9
60	13	14.08	15	-2	2	4
60	13	14.08	15	-2	2	4
62	13	14.602	15	-2	2	4
62	13	14.602	15	-2	2	4
64	14	15.124	16	-2	2	4
65	15	15.385	16	-1	1	1
66		15.646	16			
67		15.907	16			
68		16.168	17			
69		16.429	17			
70		16.69	17			
MEDIA DE ERROR					1.354166667	

CARACTERISTICA TABU 8

ANALISIS ESTADISTICO PARA PRONOSTICOS EN BASE A ECUACION DE REGRESIÓN CON CARACTERISTICA TABU 8						
ECUACION	Vehículos =- 1.53 + 0.264 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.75	4	0	0	0
20	4	3.75	4	0	0	0
23	5	4.542	5	0	0	0
24	5	4.806	5	0	0	0
26	6	5.334	6	0	0	0
29	5	6.126	7	-2	2	4
32	7	6.918	7	0	0	0
32	7	6.918	7	0	0	0
33	7	7.182	8	-1	1	1
34	7	7.446	8	-1	1	1
35	8	7.71	8	0	0	0
36	7	7.974	8	-1	1	1
38	9	8.502	9	0	0	0
38	8	8.502	9	-1	1	1
39	8	8.766	9	-1	1	1
39	9	8.766	9	0	0	0
40	8	9.03	10	-2	2	4
40	9	9.03	10	-1	1	1
41	9	9.294	10	-1	1	1
41	8	9.294	10	-2	2	4
41	7	9.294	10	-3	3	9
42	10	9.558	10	0	0	0
44	10	10.086	11	-1	1	1
44	9	10.086	11	-2	2	4
44	9	10.086	11	-2	2	4
44	9	10.086	11	-2	2	4
44	8	10.086	11	-3	3	9
45	9	10.35	11	-2	2	4
46	10	10.614	11	-1	1	1
46	9	10.614	11	-2	2	4
46	9	10.614	11	-2	2	4
47	9	10.878	11	-2	2	4
49	10	11.406	12	-2	2	4
50	10	11.67	12	-2	2	4
51	10	11.934	12	-2	2	4
52	12	12.198	13	-1	1	1
52	11	12.198	13	-2	2	4
52	11	12.198	13	-2	2	4
53	12	12.462	13	-1	1	1
54	10	12.726	13	-3	3	9
56	12	13.254	14	-2	2	4
57	11	13.518	14	-3	3	9
60	13	14.31	15	-2	2	4
60	13	14.31	15	-2	2	4
62	13	14.838	15	-2	2	4
62	13	14.838	15	-2	2	4
64	14	15.366	16	-2	2	4

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO	MSE
---------	--------------	------------	------------	-------	----------------	-----

DE ERROR						
65	15	15.63	16	-1	1	1
66		15.894	16			
67		16.158	17			
68		16.422	17			
69		16.686	17			
70		16.95	17			
					MEDIA DE ERROR	1.375

CARACTERISTICA TABU 9

ANALISIS ESTADISTICO PARA PRONOSTICOS EN BASE A ECUACION DE REGRESIÓN CON CARACTERISTICA TABU 9						
ECUACION	Vehículos =- 1.68 + 0.263 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.58	4	0	0	0
20	4	3.58	4	0	0	0
23	5	4.369	5	0	0	0
24	5	4.632	5	0	0	0
26	6	5.158	6	0	0	0
29	5	5.947	6	-1	1	1
32	7	6.736	7	0	0	0
32	7	6.736	7	0	0	0
33	7	6.999	7	0	0	0
34	7	7.262	8	-1	1	1
35	8	7.525	8	0	0	0
36	7	7.788	8	-1	1	1
38	9	8.314	9	0	0	0
38	8	8.314	9	-1	1	1
39	8	8.577	9	-1	1	1
39	9	8.577	9	0	0	0
40	8	8.84	9	-1	1	1
40	9	8.84	9	0	0	0
41	9	9.103	10	-1	1	1
41	8	9.103	10	-2	2	4
41	7	9.103	10	-3	3	9
42	10	9.366	10	0	0	0
44	10	9.892	10	0	0	0
44	9	9.892	10	-1	1	1
44	9	9.892	10	-1	1	1
44	9	9.892	10	-1	1	1
44	8	9.892	10	-2	2	4
45	9	10.155	11	-2	2	4
46	10	10.418	11	-1	1	1
46	9	10.418	11	-2	2	4
46	9	10.418	11	-2	2	4
47	9	10.681	11	-2	2	4
49	10	11.207	12	-2	2	4
50	10	11.47	12	-2	2	4

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
51	10	11.733	12	-2	2	4
52	12	11.996	12	0	0	0
52	11	11.996	12	-1	1	1
52	11	11.996	12	-1	1	1
53	12	12.259	13	-1	1	1
54	10	12.522	13	-3	3	9
56	12	13.048	14	-2	2	4
57	11	13.311	14	-3	3	9
60	13	14.1	15	-2	2	4
60	13	14.1	15	-2	2	4
62	13	14.626	15	-2	2	4
62	13	14.626	15	-2	2	4
64	14	15.152	16	-2	2	4
65	15	15.415	16	-1	1	1
66		15.678	16			
67		15.941	16			
68		16.204	17			
69		16.467	17			
70		16.73	17			
MEDIA DE ERROR					1.125	

CARACTERISTICA TABU 10

ANALISIS ESTADISTICO PARA PRONOSTICOS EN BASE A ECUACION DE REGRESIÓN CON CARACTERISTICA TABU 10						
ECUACION	Vehículos = - 1.70 + 0.264 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.58	4	0	0	0
20	4	3.58	4	0	0	0
23	5	4.372	5	0	0	0
24	5	4.636	5	0	0	0
26	6	5.164	6	0	0	0
29	5	5.956	6	-1	1	1
32	7	6.748	7	0	0	0
32	7	6.748	7	0	0	0
33	7	7.012	8	-1	1	1
34	7	7.276	8	-1	1	1
35	8	7.54	8	0	0	0
36	7	7.804	8	-1	1	1
38	9	8.332	9	0	0	0
38	8	8.332	9	-1	1	1
39	8	8.596	9	-1	1	1
39	9	8.596	9	0	0	0
40	8	8.86	9	-1	1	1
40	9	8.86	9	0	0	0
41	9	9.124	10	-1	1	1
41	8	9.124	10	-2	2	4
41	7	9.124	10	-3	3	9

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
42	10	9.388	10	0	0	0
44	10	9.916	10	0	0	0
44	9	9.916	10	-1	1	1
44	9	9.916	10	-1	1	1
44	9	9.916	10	-1	1	1
44	8	9.916	10	-2	2	4
45	9	10.18	11	-2	2	4
46	10	10.444	11	-1	1	1
46	9	10.444	11	-2	2	4
46	9	10.444	11	-2	2	4
47	9	10.708	11	-2	2	4
49	10	11.236	12	-2	2	4
50	10	11.5	12	-2	2	4
51	10	11.764	12	-2	2	4
52	12	12.028	13	-1	1	1
52	11	12.028	13	-2	2	4
52	11	12.028	13	-2	2	4
53	12	12.292	13	-1	1	1
54	10	12.556	13	-3	3	9
56	12	13.084	14	-2	2	4
57	11	13.348	14	-3	3	9
60	13	14.14	15	-2	2	4
60	13	14.14	15	-2	2	4
62	13	14.668	15	-2	2	4
62	13	14.668	15	-2	2	4
64	14	15.196	16	-2	2	4
65	15	15.46	16	-1	1	1
66		15.724	16			
67		15.988	16			
68		16.252	17			
69		16.516	17			
70		16.78	17			
MEDIA DE ERROR					1.208333333	

CARACTERISTICA TABU 11

ANALISIS ESTADISTICO PARA PRONOSTICOS EN BASE A ECUACION DE REGRESIÓN CON CARACTERISTICA TABU 11						
ECUACION	Vehículos =- 1.52 + 0.259 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.66	4	0	0	0
20	4	3.66	4	0	0	0
23	5	4.437	5	0	0	0
24	5	4.696	5	0	0	0

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
26	6	5.214	6	0	0	0
29	5	5.991	6	-1	1	1
32	7	6.768	7	0	0	0
32	7	6.768	7	0	0	0
33	7	7.027	8	-1	1	1
34	7	7.286	8	-1	1	1
35	8	7.545	8	0	0	0
36	7	7.804	8	-1	1	1
38	9	8.322	9	0	0	0
38	8	8.322	9	-1	1	1
39	8	8.581	9	-1	1	1
39	9	8.581	9	0	0	0
40	8	8.84	9	-1	1	1
40	9	8.84	9	0	0	0
41	9	9.099	10	-1	1	1
41	8	9.099	10	-2	2	4
41	7	9.099	10	-3	3	9
42	10	9.358	10	0	0	0
44	10	9.876	10	0	0	0
44	9	9.876	10	-1	1	1
44	9	9.876	10	-1	1	1
44	9	9.876	10	-1	1	1
44	8	9.876	10	-2	2	4
45	9	10.135	11	-2	2	4
46	10	10.394	11	-1	1	1
46	9	10.394	11	-2	2	4
46	9	10.394	11	-2	2	4
47	9	10.653	11	-2	2	4
49	10	11.171	12	-2	2	4
50	10	11.43	12	-2	2	4
51	10	11.689	12	-2	2	4
52	12	11.948	12	0	0	0
52	11	11.948	12	-1	1	1
52	11	11.948	12	-1	1	1
53	12	12.207	13	-1	1	1
54	10	12.466	13	-3	3	9
56	12	12.984	13	-1	1	1
57	11	13.243	14	-3	3	9
60	13	14.02	15	-2	2	4
60	13	14.02	15	-2	2	4
62	13	14.538	15	-2	2	4
62	13	14.538	15	-2	2	4
64	14	15.056	16	-2	2	4
65	15	15.315	16	-1	1	1
66		15.574	16			
67		15.833	16			
68		16.092	17			
69		16.351	17			
70		16.61	17			
MEDIA DE ERROR					1.125	

CARACTERISTICA 12

ANALISIS ESTADISTICO PARA PRONOSTICOS EN BASE A ECUACION DE REGRESIÓN CON CARACTERISTICA TABU 12						
ECUACION	Vehículos =- 1.69 + 0.263 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.57	4	0	0	0
20	4	3.57	4	0	0	0
23	5	4.359	5	0	0	0
24	5	4.622	5	0	0	0
26	6	5.148	6	0	0	0
29	5	5.937	6	-1	1	1
32	7	6.726	7	0	0	0
32	7	6.726	7	0	0	0
33	7	6.989	7	0	0	0
34	7	7.252	8	-1	1	1
35	8	7.515	8	0	0	0
36	7	7.778	8	-1	1	1
38	9	8.304	9	0	0	0
38	8	8.304	9	-1	1	1
39	8	8.567	9	-1	1	1
39	9	8.567	9	0	0	0
40	8	8.83	9	-1	1	1
40	9	8.83	9	0	0	0
41	9	9.093	10	-1	1	1
41	8	9.093	10	-2	2	4
41	7	9.093	10	-3	3	9
42	10	9.356	10	0	0	0
44	10	9.882	10	0	0	0
44	9	9.882	10	-1	1	1
44	9	9.882	10	-1	1	1
44	9	9.882	10	-1	1	1
44	8	9.882	10	-2	2	4
45	9	10.145	11	-2	2	4
46	10	10.408	11	-1	1	1
46	9	10.408	11	-2	2	4
46	9	10.408	11	-2	2	4
47	9	10.671	11	-2	2	4
49	10	11.197	12	-2	2	4
50	10	11.46	12	-2	2	4
51	10	11.723	12	-2	2	4
52	12	11.986	12	0	0	0
52	11	11.986	12	-1	1	1
52	11	11.986	12	-1	1	1
53	12	12.249	13	-1	1	1
54	10	12.512	13	-3	3	9
56	12	13.038	14	-2	2	4
57	11	13.301	14	-3	3	9
60	13	14.09	15	-2	2	4
60	13	14.09	15	-2	2	4

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
62	13	14.616	15	-2	2	4
62	13	14.616	15	-2	2	4
64	14	15.142	16	-2	2	4
65	15	15.405	16	-1	1	1
66		15.668	16			
67		15.931	16			
68		16.194	17			
69		16.457	17			
70		16.72	17			
MEDIA DE ERROR					1.125	

ALGORITMO 1-RBH

ANALISIS ESTADISTICO PARA PRONOSTICOS EN BASE A ECUACION DE REGRESIÓN CON EL ALGORITMO 1-RBH						
ECUACION	Vehículos =- 2.23 + 0.288 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	3.53	4	0	0	0
20	4	3.53	4	0	0	0
23	5	4.394	5	0	0	0
24	5	4.682	5	0	0	0
26	6	5.258	6	0	0	0
29	5	6.122	7	-2	2	4
32	7	6.986	7	0	0	0
32	7	6.986	7	0	0	0
33	7	7.274	8	-1	1	1
34	7	7.562	8	-1	1	1
35	8	7.85	8	0	0	0
36	7	8.138	9	-2	2	4
38	9	8.714	9	0	0	0
38	8	8.714	9	-1	1	1
39	8	9.002	10	-2	2	4
39	9	9.002	10	-1	1	1
40	8	9.29	10	-2	2	4
40	9	9.29	10	-1	1	1
41	9	9.578	10	-1	1	1
41	8	9.578	10	-2	2	4
41	7	9.578	10	-3	3	9
42	10	9.866	10	0	0	0
44	10	10.442	11	-1	1	1
44	9	10.442	11	-2	2	4
44	9	10.442	11	-2	2	4
44	9	10.442	11	-2	2	4
44	8	10.442	11	-3	3	9
45	9	10.73	11	-2	2	4
46	10	11.018	12	-2	2	4
46	9	11.018	12	-3	3	9
46	9	11.018	12	-3	3	9

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
47	9	11.306	12	-3	3	9
49	10	11.882	12	-2	2	4
50	10	12.17	13	-3	3	9
51	10	12.458	13	-3	3	9
52	12	12.746	13	-1	1	1
52	11	12.746	13	-2	2	4
52	11	12.746	13	-2	2	4
53	12	13.034	14	-2	2	4
54	10	13.322	14	-4	4	16
56	12	13.898	14	-2	2	4
57	11	14.186	15	-4	4	16
60	13	15.05	16	-3	3	9
60	13	15.05	16	-3	3	9
62	13	15.626	16	-3	3	9
62	13	15.626	16	-3	3	9
64	14	16.202	17	-3	3	9
65	15	16.49	17	-2	2	4
66		16.778	17			
67		17.066	18			
68		17.354	18			
69		17.642	18			
70		17.93	18			
				MEDIA DE ERROR	1.75	

ALGORITMO ÓPTIMO (C-PLEX)

ANÁLISIS ESTADÍSTICO PARA PRONOSTICOS EN BASE A ECUACION DE REGRESIÓN CON EL ALGORITMO OPTIM (C-PLEX)						
ECUACION	Vehiculos = 0.215 Demanda					
DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
20	4	4.3	5	-1	1	1
20	4	4.3	5	-1	1	1
23	5	4.945	5	0	0	0
24	5	5.16	6	-1	1	1
26	6	5.59	6	0	0	0
29	5	6.235	7	-2	2	4
32	7	6.88	7	0	0	0
32	7	6.88	7	0	0	0
33	7	7.095	8	-1	1	1
34	7	7.31	8	-1	1	1
35	8	7.525	8	0	0	0
36	7	7.74	8	-1	1	1
38	9	8.17	9	0	0	0
38	8	8.17	9	-1	1	1
39	8	8.385	9	-1	1	1
39	9	8.385	9	0	0	0

DEMANDA	VALOR OPTIMO	PRONOSTICO	PRONOSTICO	ERROR	VALOR ABSOLUTO DE ERROR	MSE
40	8	8.6	9	-1	1	1
40	9	8.6	9	0	0	0
41	9	8.815	9	0	0	0
41	8	8.815	9	-1	1	1
41	7	8.815	9	-2	2	4
42	10	9.03	10	0	0	0
44	10	9.46	10	0	0	0
44	9	9.46	10	-1	1	1
44	9	9.46	10	-1	1	1
44	9	9.46	10	-1	1	1
44	8	9.46	10	-2	2	4
45	9	9.675	10	-1	1	1
46	10	9.89	10	0	0	0
46	9	9.89	10	-1	1	1
46	9	9.89	10	-1	1	1
47	9	10.105	11	-2	2	4
49	10	10.535	11	-1	1	1
50	10	10.75	11	-1	1	1
51	10	10.965	11	-1	1	1
52	12	11.18	12	0	0	0
52	11	11.18	12	-1	1	1
52	11	11.18	12	-1	1	1
53	12	11.395	12	0	0	0
54	10	11.61	12	-2	2	4
56	12	12.04	13	-1	1	1
57	11	12.255	13	-2	2	4
60	13	12.9	13	0	0	0
60	13	12.9	13	0	0	0
62	13	13.33	14	-1	1	1
62	13	13.33	14	-1	1	1
64	14	13.76	14	0	0	0
65	15	13.975	14	1	1	1
66		14.19	15			
67		14.405	15			
68		14.62	15			
69		14.835	15			
70		15.05	16			
MEDIA DE ERROR					0.770833333	

Anexo 10. Pronósticos para el algoritmo de búsqueda tabú.

DEMANDA	ALGORITMO	ECUACION	PRONOSTICOS	VALOR DE R CUADRADA
66	CARACTERISTICA ALGORITMO TABU 1	Vehículos =- 1.68 + 0.272*Demanda	17	92.30%
67			17	
68			17	
69			18	
70			18	
66	CARACTERISTICA ALGORITMO TABU 2	Vehículos =- 1.93 + 0.282 Demanda	17	78.70%
67			17	
68			18	
69			18	
70			18	
66	CARACTERISTICA ALGORITMO TABU 3	Vehículos =- 1.65 + 0.268 Demanda	17	92.60%
67			17	
68			17	
69			17	
70			18	
66	CARACTERISTICA ALGORITMO TABU 4	Vehículos =- 1.23 + 0.257 Demanda	16	93.30%
67			16	
68			17	
69			17	
70			17	
66	CARACTERISTICA ALGORITMO TABU 5	Vehículos =- 1.48 + 0.260 Demanda	16	92.70%
67			16	
68			17	
69			17	
70			17	
66	CARACTERISTICA ALGORITMO TABU 5	Vehículos =- 1.66 + 0.266 Demanda	16	93.50%
67			17	
68			17	
69			17	
70			17	
66	CARACTERISTICA ALGORITMO TABU 6	Vehículos =- 1.66 + 0.266 Demanda	16	93.50%
67			17	
68			17	
69			17	
70			17	
66	CARACTERISTICA ALGORITMO TABU 7	Vehículos = - 1.58 + 0.261 Demanda	16	94.10%
67			16	
68			17	
69			17	
70			17	
66	CARACTERISTICA ALGORITMO TABU 8	Vehículos =- 1.53 + 0.264 Demanda	16	91.60%
67			17	
68			17	
69			17	
70			17	
66	CARACTERISTICA ALGORITMO TABU 9	Vehículos =- 1.68 + 0.263 Demanda	16	92.80%
67			16	
68			17	
69			17	
70			17	

DEMANDA	ALGORITMO	ECUACION	PRONOSTICOS	VALOR DE R CUADRADA
66			16	
67			16	
68	CARACTERISTICA ALGORITMO TABU 10	Vehículos = - 1.70 + 0.264 Demanda	17	93.20%
69			17	
70			17	
66			16	
67			16	
68	CARACTERISTICA ALGORITMO TABU 11	Vehículos = - 1.52 + 0.259 Demanda	17	94.00%
69			17	
70			17	
66			16	
67			16	
68	CARACTERISTICA ALGORITMO TABU 12	Vehículos = - 1.69 + 0.263 Demanda	17	93.00%
69			17	
70			17	
66			17	
67			18	
68	1-RBH	Vehículos = - 2.23 + 0.288 Demanda	18	86.30%
69			18	
70			18	
66			15	
67			15	
68	ALGORITMO OPTIMO (C-PLEX)	Vehículos = 0.215 Demanda	15	93.10%
69			15	
70			16	

Anexo 11. Diagrama de algoritmo de búsqueda tabú

