

RUTEO DE INVENTARIOS CON VENTANAS DE TIEMPO FUERTES

Eliseo Pérez Kaligari

**Escuela Colombiana de Ingeniería Julio Garavito
Decanatura de Ingeniería Industrial
Maestría en Ingeniería Industrial
Bogotá D.C., Colombia
2015**

RUTEO DE INVENTARIOS CON VENTANAS DE TIEMPO FUERTES

Eliseo Pérez Kaligari

Trabajo de investigación para optar al título de
Magíster en Ingeniería Industrial

Director
William J. Guerrero Rueda
Ph.D.

**Escuela Colombiana de Ingeniería Julio Garavito
Decanatura de Ingeniería Industrial
Maestría en Ingeniería Industrial
Bogotá D.C., Colombia
2015**

© Únicamente se puede usar el contenido de las publicaciones para propósitos de información. No se debe copiar, enviar, recortar, transmitir o redistribuir este material para propósitos comerciales sin la autorización de la Escuela Colombiana de Ingeniería. Cuando se use el material de la Escuela se debe incluir la siguiente nota “Derechos reservados a Escuela Colombiana de Ingeniería” en cualquier copia en un lugar visible. Y el material no se debe notificar sin el permiso de la Escuela.

Publicado en 2015 por la Escuela Colombiana de Ingeniería Julio Garavito. Avenida 13 No 205-59 Bogotá. Colombia
TEL: +57 – 1 668 36 00

AGRADECIMIENTOS

Los autores de este trabajo agradecen a la Escuela Colombiana De Ingeniería Julio Garavito por el apoyo en esta investigación, al profesor Ph D. Ignacio Pérez por su colaboración y aporte al desarrollo de la misma. También al Ingeniero de Sistemas Ricardo Kaligari por su cooperación en el desarrollo de la heurística en el lenguaje de programación Visual Basic y a la profesora Ph D. Nubia M. Velasco por sus aportes y participación como jurado.

RESUMEN

El problema estudiado en esta tesis considera las decisiones de ruteo de inventarios con restricciones de ventanas fuertes de tiempo (IRPTW) en una cadena de abastecimiento compuesta por un depósito y múltiples minoristas. Se asume un horizonte de planeación discreto y finito en donde las decisiones por optimizar son: las cantidades a aprovisionar a los minoristas y, simultáneamente, la secuencia de minoristas a visitar en cada periodo de tiempo. El sistema cuenta con un sólo vehículo de capacidad limitada, restricciones de capacidad de almacenamiento de los minoristas, y tasa de producción limitada del depósito. Se requiere satisfacer la demanda que tienen los minoristas, prohibiendo faltantes o entregas tardías. Se asume que los minoristas tienen demanda determinística y no homogénea. Las ventanas de tiempo pueden ser diferentes para cada periodo de planeación. Las distancias son simétricas entre los nodos, la velocidad del vehículo y los costos de almacenamiento de inventarios son constantes. Se propone una formulación matemática basada en programación entera mixta y se estudia el impacto que tiene la configuración de las ventanas de tiempo sobre las decisiones de ruteo de inventarios, a través de dos métodos de optimización basados en programación entera mixta. El primer método optimiza las decisiones de forma simultánea, mientras que el segundo método propuesto optimiza las decisiones de forma secuencial. Se presenta un estudio computacional con 80 instancias adaptadas de la literatura. Los resultados muestran que en la mayoría de los casos, el método de optimización secuencial no es capaz de encontrar soluciones factibles al problema con ventanas duras y los ahorros potenciales de ejecutar el método de optimización simultaneo son de 98.6%.

ABSTRACT

The problem studied in this research considers the inventory routing problem with time windows (IRPTW) in a supply chain that consists of a single warehouse and multiple retailers. The considered planning horizon is discrete and finite and the decisions to optimize are: the quantities to supply to retailers, and simultaneously the sequence of retailers to visit in each time period. The system has a single vehicle with limited capacity. Storage capacity constraints at retailers and a limited production rate at the depot constraint is considered as well. It is required to meet the retailer's demand, without stock-out or backorders. It is assumed that retailers face deterministic and not homogeneous demand. Time windows can be different for each planning period. Distances are symmetrical between nodes and the vehicle speed and the storage costs are constant. It is proposed a mathematical formulation based on mixed integer programming, and it is studied the impact of the time windows configuration on inventory routing decisions

through two optimization methods based on mixed integer programming. The first method optimizes simultaneously the decisions, while the second proposed method optimizes decisions sequentially. A computational study of 80 instances adapted from the literature is presented. The results show that in most of the cases the sequential optimization method is not able to find feasible solutions for the problem with hard windows, and the potential savings to execute the simultaneous optimization method are about 98.6%.

TABLA DE CONTENIDO

LISTA DE GRÁFICAS	8
LISTA DE TABLAS	9
INTRODUCCIÓN	10
1 OBJETIVOS Y PREGUNTA DE INVESTIGACIÓN	11
1.1 <i>Preguntas de Investigación</i>	11
1.2 <i>Objetivos</i>	11
1.2.1 <i>Objetivo General</i>	11
1.2.2 <i>Objetivos Específicos</i>	11
2 ESTADO DEL ARTE	12
3 PROBLEMA Y FORMULACIÓN MATEMÁTICA	20
4 MÉTODOS DE SOLUCIÓN	26
4.1 <i>Métodos Basados en Programación Entera Mixta</i>	26
4.1.1 <i>Optimización Simultánea Basada en Programación Entera Mixta</i>	26
4.1.2 <i>Optimización Secuencial Basada en Programación Entera Mixta</i>	27
4.2 <i>Enfoque Basado en Heurísticas Constructivistas</i>	34
5 PRUEBAS COMPUTACIONALES Y ANÁLISIS	38
5.1 <i>Resultados de los Enfoques Basados en Programación Entera Mixta</i>	38
5.2 <i>Resultados de los Enfoques Basados en la Heurística Constructiva</i>	45
CONCLUSIONES	48
BIBLIOGRAFÍA	50
ANEXOS	53

LISTA DE GRÁFICAS

Gráfica 3 - 1 Ruteo obtenido para el IRPTW con una instancia de 3 periodos, costos altos de mantenimiento de inventario (High Cost) y ventanas amplias.....	23
Gráfica 3 - 2 Ruteo obtenido para el IRPTW con una instancia de 3 periodos, costos altos de mantenimiento de inventario (High Cost) y ventanas estrechas.....	24
Gráfica 3 - 3 Ruteo obtenido para el IRPTW con una instancia de 3 periodos, costos altos de mantenimiento de inventario (High Cost) y ventanas variadas.....	24
Gráfica 4 - 1 Ejemplo grafico de una solución inicial contemplando 5 clientes y 3 periodos.....	36
Gráfica 5 - 1 Comparacion de tiempos computacionales según el método de optimización para 3 periodos con costos bajos de mantenimiento de inventario (Low Cost)	43
Gráfica 5 - 2 Comparacion de tiempos computacionales según el método de optimización para 6 periodos con costos bajos de mantenimiento de inventario (Low Cost)	43
Gráfica 5 - 3 Comparacion de tiempos computacionales según el método de optimización para 3 periodos con costos altos de mantenimiento de inventario (High Cost).....	44
Gráfica 5 - 4 Comparacion de tiempos computacionales según el método de optimización para 6 periodos con costos altos de mantenimiento de inventario (High Cost).....	44

LISTA DE TABLAS

Tabla 2 - 1 Revisión Bibliográfica.....	17
Tabla 4 - 1 Estudio del impacto por incumplir las ventanas de tiempo para instancias aleatorias del problema de ruteo de inventarios. Low Cost.	29
Tabla 4 - 2 Estudio del impacto por incumplir las ventanas de tiempo para instancias aleatorias del problema de ruteo de inventarios. High Cost.	31
Tabla 4 - 3 Matriz de cantidades generada aleatoriamente para cada cliente y periodo.....	35
Tabla 4 - 4 Matriz de demanda para cada cliente y periodo	36
Tabla 4 - 5 Matriz de inventarios para cada cliente y periodo.....	36
Tabla 5 - 1 Estudio comparativo de los modelos y métodos de optimización para instancias aleatorias del problema de ruteo de inventarios. Low Cost.....	39
Tabla 5 - 2 Estudio comparativo de los modelos y métodos de optimización para instancias aleatorias del problema de ruteo de inventarios. High Cost.....	41
Tabla 5 - 3 Cuadro comparativo de los Tiempos de Computo según el método de optimización.....	42
Tabla 5 - 4 Estudio comparativo Heurística Constructiva (HC) contra modelo IRPTW y Heurística Secuencial TW para instancias aleatorias del problema de ruteo de inventarios. Low Cost.....	46
Tabla 5 - 5 Estudio comparativo Heurística Constructiva (HC) contra modelo IRPTW y Heurística Secuencial TW para instancias aleatorias del problema de ruteo de inventarios. High Cost.....	47

INTRODUCCIÓN

Los problemas de distribución y ruteo han sido ampliamente estudiados en la literatura científica. Estos problemas son comúnmente conocidos como problemas de ruteo de vehículos (Vehicle Routing Problems - VRP). El VRP consiste en el diseño de un conjunto de rutas de vehículos de mínimo costo, iniciando en un origen (depósito central) y terminando en el mismo, para una flota de vehículos que abastece a un conjunto de clientes geográficamente dispersos y que tienen demandas conocidas. Cada cliente es atendido exactamente una vez y todos los clientes deben ser asignados a los vehículos sin exceder la capacidad de transporte (Solomon, 1987).

Una variante del problema de VRP es contemplar las ventanas de tiempo como una restricción adicional. Este tipo de problemas de ruteo de vehículos con ventanas de tiempo son conocidos como Vehicle Routing Problem con Time Windows (VRPTW) y consisten en la asignación de una flota de vehículos de capacidad limitada para servir a un conjunto de clientes sin violar las restricciones de capacidad y tiempo de entrega. Este tema se ha discutido cada vez más en la práctica por ser una restricción que afecta de forma importante los costos de una compañía (Cordeau et al., 2001). Sin embargo, estos modelos sólo funcionan para optimizar las decisiones de ruteo con uno o varios vehículos, despreciando las decisiones de inventarios en los diferentes niveles de la cadena de suministros (clientes, bodegas, etc).

Por otro lado, los problemas de inventarios y de ruteo contemplan ambas decisiones de forma simultánea. Estos problemas son conocidos como Inventory Routing Problem (IRP) y nacen para dar solución a los problemas del Vendor Managed Inventory (VMI) el cual incluyen una variedad de diferentes problemas de optimización que, aunque a menudo muy diferentes unos de otros, todos consideran como factor común la secuencia de visita a los minoristas y el componente de gestión de inventarios. Cuando los costos de almacenamiento no están incluidos en la función objetivo, por lo general la capacidad disponible de inventario de los clientes es limitada y no se puede sobrepasar. El IRP ha recibido poca atención en comparación con los problemas de ruteo de vehículos (Archetti et al., 2012).

Las decisiones de inventarios y las decisiones de ruteo están fuertemente relacionadas puesto que el conjunto de rutas de mínimo costo se construye a partir de las cantidades a entregar a los clientes. Así mismo, estas cantidades se computan considerando los costos de hacer pedidos que incluyen, entre otros, los costos de transporte y distribución (Guerrero et al. 2013). En este trabajo abordaremos una nueva variante del IRP llamado el problema de ruteo de inventarios con ventanas de tiempo fuertes (Inventory Routing Problem with Time Windows IRPTW)

1 OBJETIVOS Y PREGUNTA DE INVESTIGACIÓN

1.1 Preguntas de Investigación

Las preguntas planteadas para el trabajo de investigación son:

1. ¿Cómo afectan las ventanas de tiempo en las decisiones de ruteo, la determinación de las cantidades y las fechas de entrega en los clientes?
2. ¿Cómo se puede modelar matemáticamente este problema?
3. ¿Cómo afectan factores como la amplitud de las ventanas de tiempo y la distribución geográfica de los clientes en la dificultad para resolver el problema?
4. ¿Cómo se puede resolver este problema de forma eficiente?
5. ¿Cuál es la relación entre las ventanas de tiempo y los costos de mantener inventario en una operación de distribución y abastecimiento de productos con un vehículo de capacidad limitada?

1.2 Objetivos

1.2.1 *Objetivo General*

Desarrollar un método eficiente que encuentre soluciones de alta calidad para el problema de ruteo de inventarios con ventanas de tiempo (IRPTW) y permita concluir sobre la importancia de optimizar simultáneamente las decisiones de distribución y de ruteo considerando las restricciones de ventanas de tiempo.

1.2.2 *Objetivos Específicos*

1. Proponer un modelo basado en programación matemática para el problema de ruteo de inventarios contemplando dentro de sus restricciones las ventanas de tiempo.
2. Realizar validaciones del método de optimización propuesto y efectividad usando instancias modificadas de la literatura.
3. Cuantificar el impacto de las ventanas de tiempo en los costos de mantener inventario y distribución de productos usando un vehículo de capacidad limitada.

2 ESTADO DEL ARTE

En los últimos años ha surgido una necesidad al interior de las compañías, que consiste en controlar y disminuir los costos asociados a la fabricación, manutención y distribución de un producto. Dentro de una cadena de abastecimiento existen costos asociados a las diferentes actividades que se ejecutan dentro de cada eslabón. Dos grandes grupos de costos encierran gran parte del costo logístico total de las operaciones: costos asociados a la administración de los inventarios y costos asociados a la logística de distribución o recolección de productos (Dondo et al., 2011). El enfoque tradicional de optimización de las decisiones asociadas es secuencial. Primero se optimizan las políticas de inventario y luego se optimiza la distribución. Sin embargo, al estudiar de forma integrada ambos problemas de optimización se esperan beneficios económicos y fortalecimiento en la cadena de abastecimiento, permitiendo tomar mejores decisiones tácticas y operativas. La presente revisión de la literatura se estructura de la siguiente manera: primero se describen sistemas de gestión de inventarios en cadenas de abastecimiento, luego se revisan modelos de optimización para los problemas de ruteo, y finalmente se discuten avances recientes de la literatura del problema de ruteo de inventarios.

La logística es ahora ampliamente reconocida como un valor agregado en las organizaciones puesto que se encarga de administrar los siguientes factores: disponibilidad del producto, consistencia de las entregas, precisión en el inventario, gestión de la demanda y facilidad de hacer los pedidos. El inventario gestionado por el proveedor (Vendor Managed Inventory - VMI) es uno de los ejemplos más representativos que existe hasta la fecha de valor agregado a través de la logística. Esta práctica de gestión consiste en controlar el abastecimiento de los clientes por parte del proveedor tomando decisiones de reposición con base en las políticas de inventario y la cadena de suministro. El inventario gestionado por el proveedor (VMI) es a menudo descrito como una situación gana – gana: los proveedores ahorrarán en los costos de distribución ya que pueden coordinar las demandas y los envíos para diferentes clientes, generando ahorros en los compradores al no tener que asignar recursos para el control y gestión del inventario. Por tal motivo, el proveedor tiene que tomar tres decisiones simultáneas: (1) cuándo debe abastecer a un cliente determinado, (2) la cantidad a entregar y (3) la forma de secuenciar los clientes para asignar las rutas. (Coelho et al., 2012)(1).

Un problema central e importante en la logística y gestión de cadena de abastecimiento son los problemas de ruteo de vehículos y sus variantes. La logística y la distribución de bienes, se encuentran en el corazón de la actividad comercial de cualquier compañía, ya que representan a menudo, junto con los inventarios y las decisiones de producción, el mayor porcentaje de participación en los costos logísticos totales. Las áreas encargadas de realizar la distribución de bienes necesitan tomar decisiones que le permitan mejorar el

diseño de sus sistemas logísticos de distribución, garantizando los tiempos de entrega al cliente de acuerdo con la flota de vehículos disponible (Baños et al., 2013).

Dondo et al. (2011) estudiaron el problema de ruteo de vehículos con cross-docking (Vehicle Routing Problem with Cross Docking - VRPCD). Desarrollaron un nuevo marco de optimización para la planificación de las operaciones a corto plazo en una red de distribución de varios artículos con N-escalones por medio de estrategias de almacenamiento y cross-docking. Revisaron simultáneamente el impacto que tenía en los costos cuando se realizaban las entregas de productos de los fabricantes a los clientes a través del envío directo y/o a través de los almacenes y puntos de cross-docking. La demanda, que podía incluir varios tipos de productos, las existencias iniciales de las fábricas y los almacenes eran datos conocidos en el inicio del horizonte de planificación.

El VRPTW es un problema de distribución que se enfoca en minimizar costos de transporte a través del diseño de rutas que permitan abastecer desde un depósito central a un conjunto de puntos (clientes) geográficamente dispersos. Las rutas deben estar diseñadas de tal forma que cada cliente es visitado sólo una vez por un vehículo en un intervalo de tiempo determinado; todas las rutas comienzan y terminan en el depósito y las demandas totales de todos los puntos de una ruta en particular no deben exceder la capacidad del vehículo. El modelo presentado por Bräysy et al., 2005a y 2005b consideran múltiples objetivos que consisten en minimizar no sólo el número de vehículos necesarios, sino también el tiempo total de viaje o distancia de recorrido total realizado por la flota de vehículos. La función objetivo es jerárquica, es decir, se minimiza primero el número de rutas y luego para el mismo número de rutas, la distancia o tiempo total recorrida. Existen en la literatura dos tipos de ventanas de tiempos que se pueden utilizar: las ventanas de tiempo suaves las cuales permiten la presencia de tiempos de espera y entregas tardías en las instalaciones del cliente, aunque se incurre en una penalidad en la función objetivo. Estos problemas han sido estudiados por Qureshi et al. (2009), Calvete et al. (2007) y Qureshi et al. (2012). El otro tipo de ventana de tiempo son las ventanas fuertes, que a diferencia de las ventanas suaves, no permiten tiempos de espera ni llegadas tardías (Bräysy et al., 2005a y 2005b). Este tipo de problemas con ventanas de tiempo fuertes han sido trabajadas por Baños et al. (2013), Chang et al. (2009) y Wang et al. (2002).

Vidal et al. (2013) proponen un método de solución llamado Hybrid Genetic Search with Advanced Diversity Control (HGSADC). Este método soluciona una amplia gama de problemas de enrutamiento de vehículos con ventanas de tiempo, restricciones con respecto al tiempo de recorrido por ruta y atributos adicionales que implican los requisitos para la asignación de clientes a todo tipo de vehículos o depósitos. Además, proponen

procedimientos de evaluación por movimiento que se adaptan a través de la penalización por inviabilidad respecto a las restricciones de las ventanas de tiempo.

Sin embargo, es interesante notar que los costos de transporte dependen de las decisiones de inventarios y de las frecuencias de visita a los clientes. En este sentido, al incrementar los niveles de inventario de los clientes, reduciendo la frecuencia de entrega, los costos de distribución tienden a ser modificados (Bertazzi et al., 2013). Por lo tanto, se han propuesto varios modelos de optimización para la integración de los diferentes tipos de costos logísticos. El objetivo es determinar las políticas de envío para minimizar la suma de los costes de transporte y gestión de inventario. Una revisión de la literatura es presentada por Andersson et al. (2010) haciendo énfasis en los aspectos industriales, y se concluye la importancia del diseño de métodos de optimización del IRP especialmente para la industria del transporte marítimo.

Consecuentemente, el IRP modela el reabastecimiento de un conjunto de minoristas en un corto o mediano plazo (horizonte de tiempo), con un único o varios productos provenientes de una planta de producción o depósito, conocida como proveedor. Durante el horizonte de planeación, el proveedor decide las cantidades a entregar y las rutas para hacer estas entregas. Estas situaciones se observan, por ejemplo, en la industria de los supermercados cuando un decisor centralizado toma las decisiones de inventario sobre los productos en el horizonte de planificación para una red de minoristas geográficamente dispersos. Esto ocurre cuando la capacidad de los vehículos es limitada y comparable con respecto al volumen total requerido de los minoristas (Bertazzi et al., 2013).

Para solucionar el IRP de múltiples productos y periodos, Moin et al. (2011) proponen usar un algoritmo genético para un problema con una estructura de la red de distribución many-to-one. La capacidad de los vehículos es limitada y transporta los productos desde los proveedores hasta la planta ensambladora (depósito). Una vez terminados los viajes el vehículo regresa al depósito y se asume que no existen tiempos de operación durante la recolección de los productos, es decir, el producto está disponible al instante cuando el vehículo arriba al cliente. Por otro lado, el depósito proyecta la demanda para varios periodos e incurrir en un costo de almacenamiento con el objetivo de evitar pedidos retrasados.

Archetti et al. (2012) sugiere una nueva heurística llamada Hybrid Approach to Inventory Routing (HAIR) para solucionar un IRP contemplando un vehículo, un producto, capacidad limitada de almacenamiento y transporte. Además, utilizan dos diferentes políticas de inventarios: "Order Up to (OU)" y "Maximun Level (ML)" que permiten fijar el tipo de abastecimiento deseado para cada cliente. La política OU establece que la capacidad de inventario de un cliente se llena cada vez que el cliente recibe la visita de un vehículo, a

diferencia de la política ML que permite decidir las cantidades a entregar al momento de realizar la visita a cada cliente sin estar obligado a utilizar la capacidad máxima de inventario en los clientes. Dentro de su método de solución, Archetti et al. (2012) aplican operadores de búsqueda de tabú y optimizadores comerciales para resolver una serie de sub-problemas usando métodos basados en programación entera mixta con un enfoque Ad Hoc. Coelho et al. (2012) resuelven un IRP con trasbordos (Inventory Routing Problem with Transshipment - IRPT), capacidad de vehículos limitada, no se permite los desabastecimientos, faltantes ni retrasos; aplica políticas de inventarios OU y ML. Se soluciona el problema aplicando una heurística basada en un Adaptive Large Neighborhood Search (ALNS) para 4 tipos de escenarios: (1) Inventory Routing Problem - Order Up to Level (IRP – OU), (2) Inventory Routing Problem – Maximum Level (IRP – ML), (3) Inventory Routing Problem with Transshipment – Order Up to Level (IRPT – OU) y (4) Inventory Routing Problem with Transshipment – Maximum Level (IRPT – ML).

Qin et al. (2014) formulan un nuevo modelo para un IRP periódico sin tener en cuenta los niveles de inventario de los minoristas, solo en los depósitos, y proponen el uso de un método de búsqueda local sin referirse a los niveles de inventarios y un algoritmo de búsqueda tabú para resolver el problema de ruteo. Bard y Nananukul (2009) desarrollan un nuevo modelo para dar solución al problema de integrado de producción y distribución (Production Inventory Distribution Routing Problem PIDRP). Realizan un análisis comparativo de una serie de heurísticas para un problema de enrutamiento de inventario (IRP), en donde formulan el IRP como un programa entero mixto con el objetivo de maximizar los beneficios netos asociados a las entregas en un período de tiempo específico para un conjunto muy disperso de clientes. Se supone que el inventario puede ser almacenado en las instalaciones de los clientes y que toda la demanda debe ser satisfecha sin retrasos. Encontrar la solución exacta es difícil al problema planteado, por tal motivo utilizan un procedimiento de dos etapas: (1) estiman las cantidades de entrega diarias y (2) resuelven el problema de ruteo de vehículos para cada horizonte de planificación. Finalmente, el modelo completo tiene en cuenta las decisiones de producción y el balance de flujo de inventario en cada período. Para los cálculos, un algoritmo Branch – and – Price es utilizado para la solución de múltiples IRP's (uno en cada periodo).

Coelho y Laporte (2013) proponen un algoritmo de Branch and Cut para la solución exacta de varias clases de IRP. Resuelven el problema de IRP para múltiples vehículos (MIRP) con una flota homogénea y heterogénea, el IRP con transbordo y el IRP considerando ciertas consistencias. Para solucionar el IRP con múltiples vehículos (MIRP) aplicaron el modelo de borde, ya que requiere considerablemente menos variables y esto se convierte en un tema relevante a la hora de realizar las pruebas computacionales en instancias relativamente grandes. Con el fin de resolver el problema LP relajado, en cada

nodo se utiliza el algoritmo simplex dual. Sin embargo, los autores no calculan una cota superior inicial, pero si aplican un algoritmo de mejora en cada iteración permitiendo mejorar las soluciones enteras encontradas durante la búsqueda, ayudando a identificar rápidamente soluciones factibles.

En el trabajo propuesto por Gronhaug et al. (2010) se considera un problema de enrutamiento inventario marítimo en el negocio de gas natural licuado (LNG), llamado el problema de enrutamiento inventario de LNG (LNG-IRP). En este problema, existe un responsable de establecer el ruteo de la flota de buques para fines especiales y de los inventarios, tanto en las plantas de licuefacción como para las terminales de regasificación. En comparación con muchos otros problemas de enrutamiento de inventario marítimas, el LNG-IRP incluye algunos aspectos que complican, tales como: una tasa constante de la carga se evapora cada día y se utiliza como combustible durante el transporte, variabilidad en la producción y el consumo de gas natural licuado, y el número de tanques de descargue es variable en los terminales de regasificación. Los autores proponen resolver el problema usando el método de Branch-and-Price.

Por otro lado, Guerrero et al. (2013) investigan un problema de IRP en donde se contempló dentro de su modelo matemático las decisiones de localización de las bodegas; este modelo fue llamado Inventory Location Routing Problem (ILRP) el cual considera que los vehículos pueden visitar a más de un minorista por ruta y las decisiones de gestión de inventario se incluyen para un sistema multi-minorista, multi-depósito con capacidad de almacenamiento limitada en un horizonte de planificación discreto. El método propuesto permite determinar dentro de un conjunto de posibles candidatos de depósitos cuales se deberían abrir, las cantidades a enviar desde los proveedores hasta los depósitos y de los depósitos a los minoristas por cada período y la secuencia en que los minoristas se abastecen por una flota de vehículos homogénea. Además, proponen un modelo de programación entera mixta para representar el problema y proporcionar cotas para un conjunto de instancias aleatorias. Dado que no es posible encontrar la solución óptima dentro de un tiempo de cálculo razonable, proponen usar un método de solución heurístico de tipo híbrido.

Finalmente, Liu y Lee (2011), proponen un método de solución heurístico de dos fases para solucionar el Inventory Routing Problem contemplando ventanas de tiempo (IRPTW). La primera fase es encontrar una mejor solución inicial basada en el enfoque constructivista. La segunda fase consiste en mejorar la solución inicial aplicando un Variable Neighborhood Tabu Search (VNTS). El trabajo de Liu y Lee (2011) difiere de la propuesta de Archetti et al. (2007) porque considera un problema de ruteo con un solo periodo, mientras que Archetti et al. (2007) considera múltiples periodos para las decisiones de ruteo. Considerar múltiples periodos es más realista, puesto que asumir

que en todos los periodos se visitarán a todos los minoristas no es posible en algunos casos industriales.

En la Tabla 2-1 se muestra un cuadro resumen de la revisión bibliográfica realizada.

Tabla 2 - 1 Revisión Bibliográfica

AUTOR (RES)	PROBLEMA ESTUDIADO	AÑO	PAÍS	CANTIDAD DE VEHÍCULOS	CAPACIDAD DE LOS VEHÍCULOS	DEMANDA	VENTANAS DE TIEMPO
Angel A. Juan, Scott E. Grasman, Jose Caceres-Cruz , Tolga Bektas	IRP con un solo período y faltantes	2013	España / USA	1	Limitada	Estocástica	NO
Archetti, C., Bertazzi, L., Hertz, A., & Speranza, M. G.	IRP (Heurística HAIR)	2012	Italia	1	Limitada	Determinística	NO
Archetti, C., Bertazzi, L., Laporte, G., & Speranza, M. G.	IRP / VMIR- OU	2007	Italia	1	Limitada	Determinística	NO
Christiansen, M., Fagerholt, K., Flatberg, T., Haugen, Ø., Kloster, O., & Lund, E. H.	Marítimo IRP (Múltiples Productos)	2011	Noruega	Varios Vehículos (Buques Graneleros)	Limitada	Determinística	NO
Jonathan F. Barda, Narameth Nananukul	IRP (PIDRP)	2009	USA	1	Limitada	Determinística	NO
Leandro C. Coelho, Gilbert Laporte	IRP y MIRP (Vehículos homogéneos y heterogéneos)	2013	Canadá	Varios Vehículos	Limitada	Determinística	NO

AUTOR (RES)	PROBLEMA ESTUDIADO	AÑO	PAÍS	CANTIDAD DE VEHÍCULOS	CAPACIDAD DE LOS VEHÍCULOS	DEMANDA	VENTANAS DE TIEMPO
Leandro C. Coelho, Jean-François Cordeau, Gilbert Laporte	IRPT	2012	Canadá	1	Limitada	Determinística	NO
Leandro C. Coelho, Jean-François Cordeau, Gilbert Laporte	IRP de múltiples vehículos (MIRP)	2012	Canadá	Varios Vehículos	Limitada	Determinística	NO
Lei Qin, Lixin Miao, Qingfang Ruan, Ying Zhang Research	IRP	2014	China	1	Limitada	Estocástica	NO
Li, K., Chen, B., Sivakumar, A. I., & Wu, Y.	IRP	2014	China / Singapur / Australia	1	Ilimitada	Determinística	SI
Liu, S.-C., & Lee, W.-T.	IRPTW	2011	Taiwán	Varios Vehículos	Limitada	Estocástica	SI
Luca Bertazzi, Adamo Bosco, Francesca Guerriero, Demetrio Laganà	IRP (SIRP)	2013	Italia	1	Limitada	Estocástica	NO
N.C.P. Edirisinghe , R.J.W. James	IRP / VRP	2014	USA / New Zealand	Varios Vehículos	Limitada	Estocástica	SI
N.H Moin, S. Salhi, N.A.B Aziz	IRP	2011	Malasia	Varios Vehículos	Ilimitada	Determinística	NO
W.J. Guerrero, C. Prodhona, N. Velasco, C.A. Amaya	ILRP (Inventory-Location-Routing Problem)	2013	Francia / Colombia	Varios Vehículos	Limitada	Determinística	NO

De este ejercicio de revisión de literatura se concluye que el IRPTW es un problema de interés académico e industrial, puesto que existe una brecha a nivel teórico en este tema de ruteo de inventarios. Además, una revisión bibliográfica de (Coelho et al., 2012) muestra que existen más de 580 artículos científicos que proponen métodos para encontrar una solución óptima, o cercana al óptimo, al problema de ruteo para la distribución de un producto (VRP). Adicionalmente, las ventanas de tiempo han sido involucradas a los problemas de ruteo (Vehicle Routing Problems with Time Windows - VRPTW), pero estos modelos sólo funcionan para optimizar las decisiones de ruteo con uno o varios vehículos, despreciando las decisiones de inventarios en los diferentes niveles de la cadena de suministros (clientes, bodegas, etc). Por otro lado, el Inventory Routing Problem (IRP) es un tema recientemente estudiado y que toma importancia puesto que existen alrededor de 70 artículos científicos que han sido publicados sobre el tema.

3 PROBLEMA Y FORMULACIÓN MATEMÁTICA

Se introduce el IRPTW modificando el modelo de IRP presentado por Archetti et al. (2007). Para ello, se considera una red de distribución de un producto en el que se envía desde una bodega central “0” a un conjunto $\mathcal{M} = \{1, 2, \dots, n\}$ de clientes (minoristas) en un horizonte de tiempo H . En cada periodo discreto $t \in \mathcal{T} = \{1, 2, \dots, H\}$ el proveedor fabrica una cantidad de producto r_{0t} y el cliente $s \in \mathcal{M}$ consume r_{st} cantidades. El nivel de inventario del proveedor en el periodo t es una variable de decisión auxiliar y se denota como B_t . Adicionalmente, B_0 representa el inventario inicial del depósito, siendo este un parámetro conocido. Cada minorista s fija su nivel máximo de inventario U_s y establece un nivel de inventario inicial $I_{s0} \leq U_s$. Si el minorista s es visitado en el tiempo t , entonces la cantidad x_{st} que se entrega al minorista es tal que no puede sobrepasar su valor máximo. Es decir, se aplica políticas de inventario “Maximum Level” (ML) descritas por Archetti et al. (2007). Si denotamos I_{st} como el nivel de inventario del minorista s en el periodo t , entonces x_{st} es menor o igual a $U_s - I_{st}$ si un envío a s se lleva a cabo en el periodo t , o igual a cero en caso contrario.

Sea h_0 el costo unitario de mantener inventario por el proveedor en el tiempo t , donde $\mathcal{T}' = \mathcal{T} \cup \{H + 1\}$ permitirá tener en cuenta el costo de un periodo $H + 1$ dentro del análisis. Se expresa h_s como el costo de mantener inventario en los minoristas $s \in \mathcal{M}$. El nivel de inventario al final del horizonte de planeación puede ser diferente al nivel de inventario inicial, convirtiendo el problema en no periódico. Las entregas desde el proveedor hacia los minoristas en cada periodo $t \in \mathcal{T}$ serán realizadas por un vehículo de capacidad limitada C capaz de abastecer a varios clientes en una misma ruta. Los clientes que se encuentren dentro de la ruta para ser visitados en el periodo t , serán atendidos por el vehículo en ese periodo t .

Además, se establece C_{ij} como la distancia de ir del nodo i al nodo j y y_{ijt} como una variable binaria igual a 1 si y solo si el vehículo selecciona ir del arco i, j en el periodo t o 0 de lo contrario. Se definen la variable v_{it} como la hora de llegada al nodo i en el periodo t y el parámetro d_{ij} como el tiempo de viaje de ir desde el nodo i al nodo j , donde $i, j \in \mathcal{M}' = \mathcal{M} \cup \{0\}$. El parámetro e_{jt} indica el inicio de la ventana del minorista i en el periodo t y el parámetro l_{jt} como el fin de la venta del minorista j en el periodo t . Usaremos la variable auxiliar binaria z_{it} que toma el valor de 1 si el nodo i es visitado en el periodo t o 0 de lo contrario. Usando esta notación, el IRPTW se puede expresar de la siguiente manera:

$$\text{Min } Z = \sum_{t \in \mathcal{T}'} h_0 \cdot B_t + \sum_{s \in \mathcal{M}} \sum_{t \in \mathcal{T}'} h_s \cdot I_{st} + \sum_{i \in \mathcal{M}'} \sum_{j \in \mathcal{M}', j < i} \sum_{t \in \mathcal{T}} C_{ij} \cdot y_{ijt} \quad (1)$$

Sujeto a:

$$B_t = B_{t-1} + r_{0t-1} - \sum_{s \in \mathcal{M}} x_{st-1} \text{ donde } t \in \mathcal{T}' \quad (2)$$

$$B_t \geq \sum_{s \in \mathcal{M}} x_{st} \text{ donde } t \in \mathcal{T} \quad (3)$$

$$I_{st} = I_{st-1} + x_{st-1} - r_{st-1} \text{ donde } s \in \mathcal{M} \text{ y } t \in \mathcal{T}' \quad (4)$$

$$I_{st} \geq 0 \text{ donde } s \in \mathcal{M} \text{ y } t \in \mathcal{T}' \quad (5)$$

$$x_{st} \leq U_s - I_{st} \text{ donde } s \in \mathcal{M} \text{ y } t \in \mathcal{T} \quad (6)$$

$$x_{st} \leq U_s z_{st} \text{ donde } s \in \mathcal{M} \text{ y } t \in \mathcal{T} \quad (7)$$

$$\sum_{s \in \mathcal{M}} x_{st} \leq C z_{0t} \text{ donde } t \in \mathcal{T} \quad (8)$$

$$\sum_{j \in \mathcal{M}', j < i} y_{ijt} + \sum_{j \in \mathcal{M}', j > i} y_{ijt} = 2 \cdot z_{it} \text{ donde } i \in \mathcal{M}' \text{ y } t \in \mathcal{T} \quad (9)$$

$$v_{jt} \geq e_{jt} \cdot z_{jt} \quad (10)$$

$$v_{jt} \geq v_{jt} + d_{ji} - [(1 - y_{jit}) \cdot (l_{jt} + d_{ji})] \quad (11)$$

$$v_{jt} \leq l_{jt} \cdot z_{jt} \quad (12)$$

$$x_{st} \geq 0 \text{ donde } s \in \mathcal{M} \text{ y } t \in \mathcal{T} \quad (13)$$

$$v_{it} \geq 0 \text{ donde } i \in \mathcal{M}' \text{ y } j \in \mathcal{M}' \quad (14)$$

$$y_{ijt} \in \{0,1\} \text{ donde } i \in \mathcal{M}' ; j \in \mathcal{M}' \text{ y } t \in \mathcal{T} \quad (15)$$

$$z_{jt} \in \{0,1\} \text{ donde } j \in \mathcal{M}' \text{ y } t \in \mathcal{T} \quad (16)$$

El modelo de IRP presentado por Archetti et al. (2007) comprende las ecuaciones de la (1) a la (9) y el modelo propuesto para solucionar el IRPTW contempla adicionalmente las restricciones de la (10) a la (12).

El objetivo del modelo es minimizar la suma de los costos de gestión de inventarios en el depósito, en los minoristas y el de distribución, como se describe en la ecuación (1). Dentro de la minimización, se contempla el costo de mantener los inventarios del proveedor y minoristas.

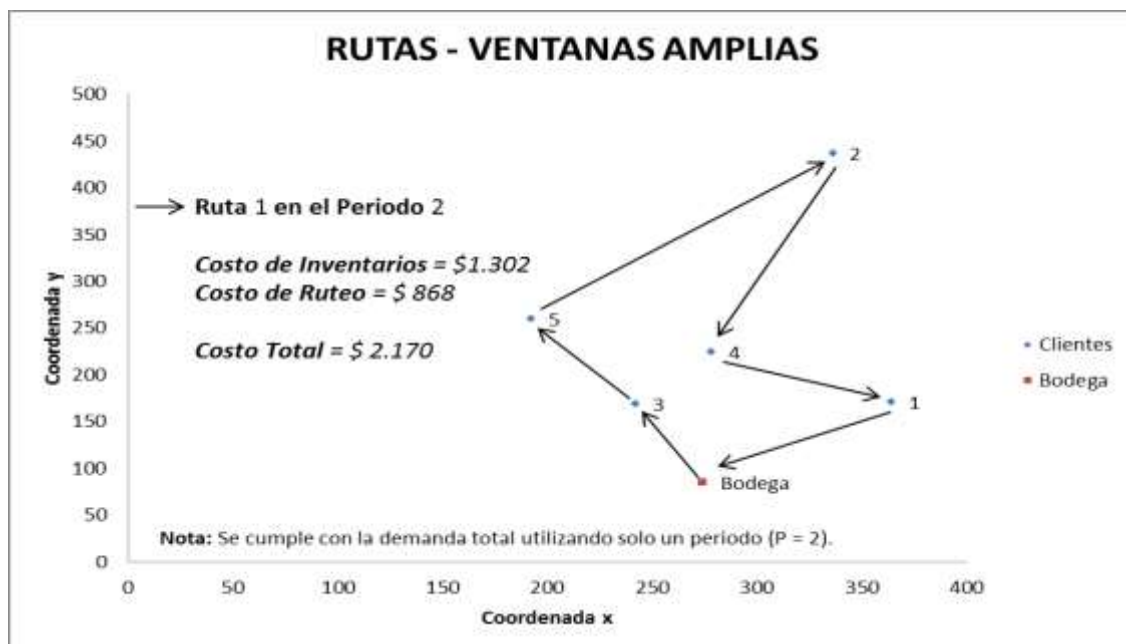
Las restricciones (2) y (3) están enfocadas a controlar los niveles de inventarios y evitar desabastecimiento en los proveedores. Las restricciones (4) y (5) controlan los niveles de inventarios y evitar los desabastecimientos de los minoristas. Las políticas de inventarios están definidas por las restricciones (6) y (7), en donde se garantiza la coordinación entre las cantidades a entregar y el inventario máximo si el nodo es visitado o no, el cumplimiento de la política seleccionada (Maximum Level (ML)) explicada en la sección 2 y el cumplimiento de la capacidad máxima, tanto de los minoristas como el de la bodega. La restricción (8) establece la capacidad del vehículo utilizado para realizar el ruteo. La restricción (9) obliga a que si se visita algún minorista en el tiempo t , esa ruta deba contener al proveedor en su ruta y evita que salgan o entren más de dos arcos al mismo nodo. Las restricciones (10), (11) y (12) garantizan el cumplimiento de las ventanas de tiempo establecidas para cada minorista en el periodo t . Las ventanas de tiempo son dinámicas y duras (hard), es decir, no se permite el incumplimiento a los horarios de atención establecidos por los minorista (si el vehículo llega antes o después de la ventana establecida, el minorista no recibe la entrega) y estos horarios pueden ser diferentes entre dos periodos dentro del horizonte de planeación. Se considera que sólo hay una ventana por minorista por periodo. Además, se considera despreciable el tiempo de descargue y de cargue. Finalmente, las restricciones (13) a (16) indican la naturaleza de las variables de decisión.

Por otro lado, con el ánimo de estudiar el impacto que tienen las diferentes configuraciones de ventanas de tiempo fuertes dentro de las rutas de distribución de los

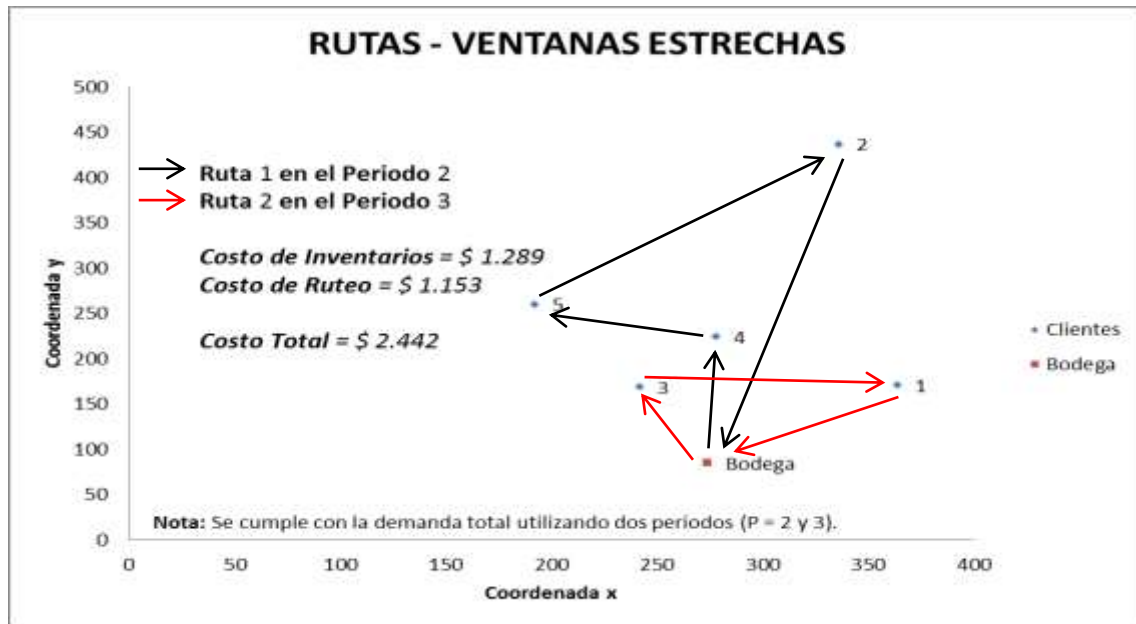
productos, sus inventarios y costos, se realizan tres experimentos con el objetivo de validar dicho supuesto usando un ejemplo numérico. Se selecciona una instancia de 3 periodos y 5 clientes disponible en <http://www.leandro-coelho.com/instances/inventory-routing>. Posteriormente se modifica la instancia adicionando ventanas de tiempo fuertes aleatoriamente usando una distribución uniforme. En total se construyen 3 instancias con la siguientes características: 1. *Ventanas amplias*, donde el rango de la ventana varía entre 0 y 2000 min; 2. *Ventanas estrechas*, donde el rango de la ventana varía entre 190 y 1700 min; 3. *Ventanas variadas*, donde el rango de la ventana varía entre 0 y 1859 min. Las 3 instancias se resolvieron a través del enfoque exacto. Los rangos generados para las ventanas de tiempo en cada una de las instancias descritas anteriormente, fueron seleccionados debido a que en pruebas preliminares, estos permitieron mostrar adecuadamente un impacto generado sobre los costos totales, sin quitarle factibilidad a la solución.

En las Gráficas 3-1, 3-2 y 3-3 se ilustran las diferentes configuraciones de rutas obtenidas con el costo total que se incurre por realizarlas.

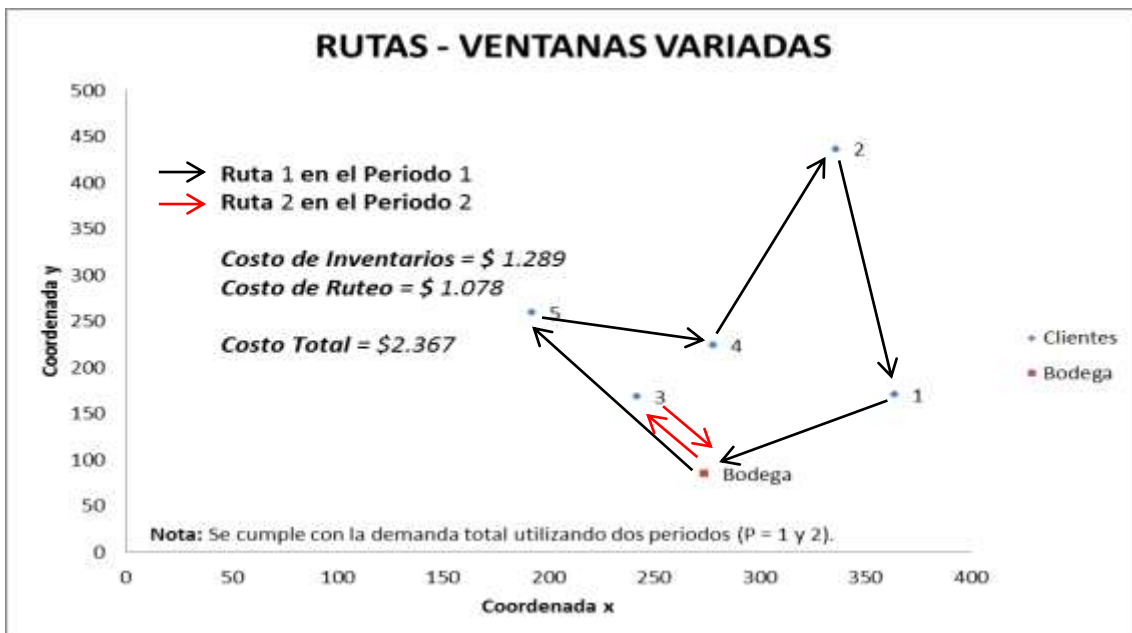
Gráfica 3 - 1 Ruteo obtenido para el IRPTW con una instancia de 3 periodos, costos altos de mantenimiento de inventario (High Cost) y ventanas amplias



Gráfica 3 - 2 Ruteo obtenido para el IRPTW con una instancia de 3 periodos, costos altos de mantenimiento de inventario (High Cost) y ventanas estrechas



Gráfica 3 - 3 Ruteo obtenido para el IRPTW con una instancia de 3 periodos, costos altos de mantenimiento de inventario (High Cost) y ventanas variadas



En la Gráfica 3-1 se concluye que el tener ventanas amplias permite visitar la totalidad de los clientes en un solo periodo con una sola ruta. Los costos de inventarios y ruteo son de \$ 1.302 y \$ 868 respectivamente; el costo total fue \$ 2.170.

En la Gráfica 3-2 se observa el efecto que tiene las ventanas estrechas en los costos, incrementando el de ruteo en 32.83%, reduciendo el de inventario en 0.99% e incrementando el costo total en 12.53% con respecto a la solución encontrada en la Gráfica 3-1. Los clientes se logran abastecer usando dos rutas en periodos diferentes.

En la Gráfica 3-3 se analizó el efecto de combinar los dos tipos de ventanas de tiempo (Amplias y Estrechas) en una misma instancia. El resultado fue un incremento en el costo de ruteo del 24.19% y una reducción del costo de inventario de 0.99%, comparado con la solución lograda para el problema de la Gráfica 3-1. Estas variaciones afectaron el costo total, haciéndolo decrecer en 9.08%. Con respecto al problema de la Gráfica 3-2, la solución obtenida arrojó una reducción en el costo de ruteo del 6.5% y no mostró cambios en el costo de inventarios. El costo final se redujo en 3.07% comparado con el logrado en la Gráfica 3-2.

Esto demuestra que los costos de las decisiones de ruteo y de inventarios se ven afectados por los diferentes rangos de ventanas de tiempo que se pueden presentar.

4 MÉTODOS DE SOLUCIÓN

Como lo indica el capítulo anterior, el IRPTW es una generalización del VRPTW incluyendo las decisiones de gestión de inventarios. Qin et al. (2014) y (1) Coelho et al. (2012) muestran que el VRPTW es un problema NP-completo. Proponemos a continuación tres métodos de solución para el IRPTW y en el capítulo 5 se presentan las pruebas computacionales que muestran el desempeño de cada uno de los métodos.

4.1 Métodos Basados en Programación Entera Mixta

4.1.1 Optimización Simultánea Basada en Programación Entera Mixta

Esta investigación se desarrolló usando el optimizador CPLEX y el lenguaje GAMS para encontrar la solución óptima a los problemas del IRPTW. Este solver utiliza el método de Branch-and-Bound para resolver el problema a optimalidad. Sin embargo, se requiere tiempos de cómputo importantes (de hasta 35.000 segundos con un procesador AMD Athlon® 64x2 Dual Core 2.7 GHz y 3 GB de RAM, ver tabla 5-3) y gran capacidad de almacenamiento de datos. Por lo tanto, el solver CPLEX intenta encontrar el óptimo o la mejor solución a cada una de las instancias dentro de un intervalo de tiempo definido por el usuario.

Coelho et al. (2013) logran determinar valores cercanos al óptimo para las instancias de prueba, que no cuentan con ventanas de tiempo, usando un Procesador Intel Xeon™ de 2.66 GHz, sistema operativo Linux 6.1 y con una capacidad de memoria RAM muy superior al equipo mencionado anteriormente (con más de 48 GB de RAM), obteniendo tiempos computacionales cortos (de hasta 76 segundos). Los resultados logrados por la implementación del enfoque exacto propuesto usando GAMS y con las mismas características de hardware expuestas en el párrafo anterior, se obtiene soluciones que en promedio no difieren estadísticamente¹ a las encontradas por Coelho et al. (2013).

Con respecto a los tiempos de cómputo no son comparables, puesto que los equipos utilizados para resolver las instancias difieren en sus características de hardware. En la sección 5 se presentan con mayor detalle los resultados obtenidos por cada uno de los métodos aplicados empleando el mismo equipo de cómputo para su solución.

¹ Se realizó una prueba t - Student de medias de dos muestras emparejadas con los resultados para las instancias de la literatura de Archetti et al. (2008) para la implementación del modelo propuesta por Coelho (2013) y la propuesta por esta tesis. Se concluye que la calidad de la solución promedio obtenida por la implementación de Coelho et al. (2013) y la propuesta en este documento, no son diferentes estadísticamente (p-value de 7.21%).

4.1.2 Optimización Secuencial Basada en Programación Entera Mixta

Se estudian dos versiones de un método heurístico propuesto en este trabajo para resolver el problema de ruteo de inventarios con ventanas de tiempo fuertes de forma secuencial. Las versiones planteadas son: 1) Heurística Secuencial (SEQ-H) y 2) Heurística Secuencial con Ventanas de Tiempo (SEQ-H (TW)).

El método SEQ-H consiste en encontrar una solución resolviendo a optimalidad dos sub-problemas de forma separada, teniendo en cuenta que el costo de la solución final se obtiene sumando los dos costos conseguidos en cada uno de los sub-problemas. Cada uno de los sub-problemas se resuelve mediante un modelo matemático basado en programación entera mixta, enfocado a la optimización de soluciones parciales de forma secuencial.

En esta versión (SEQ-H) se ejecutan dos modelos matemáticos en dos fases: El primer modelo soluciona un problema de inventarios, ignorando las restricciones de ruteo con el objetivo de encontrar el costo mínimo de gestión de inventarios y las cantidades a enviar a cada uno de los minoristas. Para este sub-modelo se consideran las restricciones de la (2) a la (9) y de la (13) a la (16) expuestas en la sección 3 usando el optimizador CPLEX para la experimentación.

En una segunda fase, se resuelve otro modelo basado en programación lineal entera mixta que optimiza las decisiones de ruteo fijando las variables de decisión asociadas al problema de gestión de inventarios encontradas en la primera fase. Para esta fase se aplican las restricciones de la (9) a la (16) expuestas en la sección 3. Finalmente se calcula el costo total de la solución como la suma del costo de inventario y ruteo encontrada por cada sub-problema. Nuevamente, se usa el optimizador CPLEX para la experimentación.

La segunda versión del método propuesto, llamada SEQ-H (TW), incluye las ventanas de tiempo fuertes dentro del sub-problema de ruteo de la fase 2 y se realiza el mismo procedimiento de ejecución que con el método SEQ-H para calcular los costos.

La ventaja de usar estos métodos secuenciales es la disminución de complejidad computacional y en consecuencia los tiempos requeridos para encontrar las soluciones se reducen en promedio en 74% (ver tabla 5-3). Sin embargo, para algunas instancias, la heurística SEQ-H (TW), presenta dificultad en encontrar soluciones factibles debido a las ventanas de tiempo fuertes que restringen el conjunto de posibles soluciones que no se tienen en cuenta en la fase 1 de la heurística. Esto ocurre al optimizar las decisiones de inventario sin considerar la restricción de las ventanas de tiempo. Además, el fijar las

decisiones de inventario no garantiza que logren ser distribuidas de forma factible. Para solucionar este problema, se realiza una relajación de las ventanas de tiempo, asignando una penalización elevada donde se afecta la función objetivo por el incumplimiento de las mismas. Esta penalización se aplica en la fase 2 y se conoce como ventanas de tiempo suaves, donde se permite al vehículo llegar antes o después de la ventana asignada para cada cliente, sin modificar las demás restricciones de ruteo.

A continuación en las Tablas 4-1 y 4-2 se muestra el impacto que tiene el incumplimiento de las ventanas de tiempo en los costos para las instancias con solución factible, aplicando la heurística secuencial con ventanas de tiempo fuertes (SEQ-H (TW)). La columna "Instancia" es el nombre del problema analizado. Las columnas "N" y "P" indican respectivamente la cantidad de minoristas y periodos contemplados en cada instancia. Las columnas "Tiempo de Espera Max" y "Tardanza Max" registran los tiempos de incumplimiento de las ventanas para cada cliente en un periodo específico. Las columnas C, P, H_0 y H_f son respectivamente el cliente, el periodo, la hora de inicio y la hora fin de la ventana incumplida. En la columna "Promedio - Rango Ventana" se muestra el promedio entre la hora de apertura y hora de cierre de la ventana para cada cliente C. La columna "Promedio Tardanza/Espera" indica la media entre el tiempo de espera máximo y la tardanza máxima para el cliente C. La columna "Error relativo de Incumpli." muestra el porcentaje de incumplimiento de la ventana de tiempo para el cliente C en el periodo P. La columna "Costo Penalidad" registra el producto entre la cantidad de minutos por fuera de la ventana y el costo por minuto incumplido, que en nuestro caso es de \$10 como se muestra en la columna "Costo Penalidad", y el costo total se calcula sumando el costo de ruteo y el costo de inventario. Dentro del análisis solo se contemplan las instancias con soluciones factibles y penalizadas por el incumplimiento de las ventanas de tiempo.

Tabla 4 - 1 Estudio del impacto por incumplir las ventanas de tiempo para instancias aleatorias del problema de ruteo de inventarios. Low Cost.

Instancia	N	P	Tiempo de Espera Max (min)	Rango de la Ventana					Rango de la Ventana					Promedio - Rango Ventana (min)	Promedio Tardanza/Espera (min)	Error Relativo de Incumpli.	Costo Inventario (\$)	Costo Ruteo (\$)	Penalización x Tardanza/Espera (\$)	Costo Penalidad (\$)	Costo Total (\$)	
				C	P	Ho (min)	Hf (min)	Total (min)	Tardanza Max (min)	C	P	Ho (min)	Hf (min)									Total (min)
abs2n5	5	3	253	1	2	672	1.349	677	178	2	2	815	1.423	608	643	216	34%	86,7	5.389	10	4.310	5.475,7
abs4n5	5	3	208	4	2	487	1.115	628	0	0	0	0	0	0	314	104	33%	64,4	3.888	10	2.080	3.952,4
abs5n5	5	3	174	2	2	910	1.563	653	0	0	0	0	0	0	327	87	27%	131,4	2.720	10	1.740	2.851,4
abs2n10	10	3	329	1	2	437	1.343	906	0	0	0	0	0	0	760	252	33%	254,5	15.163	10	3.290	15.417,5
			402	5	2	611	1.602	991	0	0	0	0	0	0						10	4.020	
			450	7	2	527	1.474	947	0	0	0	0	0	0						10	4.500	
			79	8	2	500	1.457	957	0	0	0	0	0	0						10	790	
abs4n10	10	3	236	1	2	1.044	2.039	995	0	0	0	0	0	0	716	427	60%	242,0	19.030	10	2.360	19.272,0
			428	3	2	1.040	1.965	925	0	0	0	0	0	0						10	4.280	
			1.044	7	2	1.080	2.022	942	0	0	0	0	0	0						10	10.440	
abs1n15	15	3	10	13	2	332	2.281	1.949	0	0	0	0	0	0	1.285	68	5%	378,8	4.737	10	100	5.115,8
			195	15	2	329	2.235	1.906	0	0	0	0	0	0						10	1.950	
abs2n15	15	3	136	1	2	535	2.524	1.989	0	0	0	0	0	0	1.317	97	7%	365,2	5.631	10	1.360	5.996,2
			156	8	2	563	2.525	1.962	0	0	0	0	0	0						10	1.560	
abs1n20	20	3	151	18	2	1.193	4.189	2.996	59	10	2	1.017	3.863	2.846	2.921	105	4%	507,3	3.860	10	2.100	4.367,3
abs2n20	20	3	75	4	2	653	3.219	2.566	70	1	2	880	3.638	2.758	2.744	230	8%	514,9	9.254	10	1.450	9.768,9
			545	5	2	969	3.876	2.907												10	5.450	
abs4n20	20	3	483	6	2	995	3.928	2.933	0	0	0	0	0	0	1.950	188	10%	423,9	9.429	10	4.830	9.852,9
			81	15	2	1.081	3.997	2.916	0	0	0	0	0	10						810		
abs2n5	5	6	96	5	5	531	1.192	661	0	0	0	0	0	0	331	48	15%	257,3	3.763	10	960	4.020,3

Instancia	N	P	Tiempo de Espera Max (min)	Rango de la Ventana			Rango de la Ventana			Tardanza Max (min)	C	P	Ho (min)	Hf (min)	Total (min)	Promedio - Rango Ventana (min)	Promedio Tardanza/Espera (min)	Error Relativo de Incumpli.	Costo Inventario (\$)	Costo Ruteo (\$)	Penalización x Tardanza/Espera (\$)	Costo Penalidad (\$)	Costo Total (\$)
				C	P	Ho (min)	Hf (min)	Total (min)															
abs5n5	5	6	93	2	2	765	1.419	654	0	0	0	0	0	0	436	32	7%	236,1	3.272,0	10	930	3.508,1	
			4	2	4	278	932	654	0	0	0	0	0	0						10	40		
abs1n10	10	6	247	4	5	694	2.398	1.704	0	0	0	0	0	0	852	124	14%	471,9	6.960,0	10	2.470	7.431,9	
abs3n15	15	6	17	4	4	154	2.993	2.839	354	8	2	1.387	4.333	2.946	2.868	188	7%	822,4	13.419,0	10	3.710	14.241,4	
			15	13	2	1.322	4.203	2.881	364	15	2	1.648	4.455	2.807						10	3.790		

Tabla 4 - 2 Estudio del impacto por incumplir las ventanas de tiempo para instancias aleatorias del problema de ruteo de inventarios. High Cost.

Instancia	N	P	Tiempo de Espera Max (min)	Rango de la Ventana			Rango de la Ventana			Tardanza Max (min)	C	P	Rango de la Ventana			Promedio - Rango Ventana (min)	Promedio Tardanza/Espera (min)	Error Relativo de Incumpli.	Costo Inventario (\$)	Costo Ruteo (\$)	Penalización x Tardanza/Esp era (\$)	Costo Penalidad (\$)	Costo Total (\$)
				C	P	Ho (min)	Hf (min)	Total (min)	Ho (min)				Hf (min)	Total (min)									
abs1n5	5	3	556	4	2	939	1.608	669	242	2	2	1.028	1.687	659	664	399	60%	967,3	9.305	10	7.980	10.272,3	
abs2n5	5	3	455	1	2	883	1.498	615	0	0	0	0	0	0	308	228	74%	865,1	5.629	10	4.550	6.494,1	
abs4n5	5	3	195	2	2	465	1.111	646	0	0	0	0	0	0	323	98	30%	648,1	3.758	10	1.950	4.406,1	
abs5n5	5	3	165	2	2	796	1.483	687	26	3	2	1.008	1.693	685	686	96	14%	1.302,0	2.901	10	1.910	4.203,0	
abs1n15	15	3	251	6	2	284	2.203	1.919	0	0	0	0	0	0	960	126	13%	3.812,8	5.263	10	2.510	9.075,8	
abs2n15	15	3	38	1	2	917	2.865	1.948	0	0	0	0	0	0	974	19	2%	3.647,5	2.966	10	380	6.613,5	
abs4n15	15	3	60	13	2	326	2.229	1.903	0	0	0	0	0	0	952	30	3%	3.117,7	3.773	10	600	6.890,7	
abs2n20	20	3	196	5	2	743	3.565	2.822	0	0	0	0	0	0	1.411	98	7%	5.035,0	5.396	10	1.960	10.431,0	
abs4n20	20	3	280	7	2	881	3.841	2.960	0	0	0	0	0	0	2.057	173	8%	5.035	5.396	10	2.800	10.431,0	
			99	9	2	1.052	3.782	2.730	0	0	0	0	0	0						10	990		
			313	14	2	888	3.427	2.539	0	0	0	0	0	0						10	3.130		
abs3n5	5	6	327	1	5	1.047	2.068	1.021	0	0	0	0	0	0	679	161	24%	2.381,8	9.593	10	3.270	11.974,8	
			157	2	4	885	1.901	1.016	0	0	0	0	0	0						10	1.570		
abs5n5	5	6	41	2	2	757	1.390	633	26	1	2	1.014	1.688	674	639	66	10%	2.346,7	4.175	10	670	6.521,7	
			131	3	4	315	924	609												10	1.310		
abs1n10	10	6	56	4	2	553	2.024	1.471	0	0	0	0	0	0	1.096	97	9%	4.757,1	8.450	10	560	13.206,9	
			197	4	5	573	2.001	1.428	0	0	0	0	0	10						1.970			
			135	10	5	807	2.290	1.483	0	0	0	0	0	10						1.350			

En la Tablas 4-1 y 4-2 se observa que en promedio el incumplimiento de las ventanas es de 19% para Low Cost y 21% para High Cost. Además, se evidencia que entre más amplia sea la ventana menor es el error de incumplimiento. Como se explicó en la sección 3, el impacto en los costos también se ve afectado por la amplitud de las ventanas.

4.2 Enfoque Basado en Heurísticas Constructivistas

Los enfoques exactos pueden encontrar soluciones a problemas pequeños del IRPTW (hasta 20 clientes con 3 y 6 periodos) con tiempos computacionales en promedio de 15.000s (ver tablas 5-1 y 5-2). Sin embargo, para problemas de talla mayor (más de 20 clientes) los tiempos computacionales son elevados y no se garantizan la optimalidad de las soluciones, puesto que en algunas instancias se requiere almacenar una cantidad de datos en memoria que supera la capacidad de almacenamiento de datos del computador utilizado². Por otro lado, para problemas grandes (más de 25 clientes) no se logran encontrar soluciones y emplean tiempos de cómputo bastante elevados (más de 35.000s). Por lo tanto, los métodos heurísticos se convierten en una buena alternativa para encontrar soluciones a instancias grandes. El algoritmo propuesto tiene dos componentes que al combinarse, logran encontrar soluciones factibles con tiempos de cómputo relativamente cortos (hasta 30.6 s). Las fases del algoritmo son: 1. Generación de una solución inicial; 2. Mejora de la solución inicial.

La generación de una solución inicial factible (fase 1) se describe en el algoritmo 1. Este consiste en armar en primera instancia la matriz de cantidades a entregar a cada cliente en cada uno de los periodos considerados, garantizando el cumplimiento de las restricciones de satisfacción de la demanda, capacidad del vehículo, capacidad máxima de almacenamiento de los clientes y de producción de la bodega. Posteriormente a fijar estas cantidades a entregar para cada cliente y periodo, se elabora una ruta maestra aplicando el método de Clark Wright - Método de ahorros (Clarke y Wright, 1964). Esta es una ruta que visita a todos los clientes y es independiente de las cantidades a entregar.

Una vez armada la ruta maestra, se procede a completar la solución al problema. Es decir, las rutas de entrega por periodo resultan de modificar la ruta maestra, teniendo en cuenta si el cliente es visitado o no de acuerdo a las cantidades fijadas anteriormente, respetando la precedencia fijada por la ruta maestra.

Por otro lado, la fase 2 del algoritmo tiene como objetivo calcular la función objetivo y mejorar el resultado arrojado por la fase 1 a través de una búsqueda local, contemplando las diferentes posibilidades de movimientos entre clientes, sin modificar la matriz de cantidades.

² El computador usado en los experimentos de este trabajo de investigación tiene una memoria RAM de 3Gb.

Para mostrar el funcionamiento del algoritmo, se definen las siguientes variables: 1) matrizCantidades = Matriz que fija las cantidades a entregar a cada uno de los clientes en cada periodo, 2) numclientes = Número de Clientes a abastecer, 3) numperiodos = Número de periodos disponibles dentro del horizonte de planeación, 4) Vcap = Capacidad del vehículo, 5) InventariolInicial = Inventario disponible en el periodo 0, 6) demanda = Cantidad de productos requeridos por el cliente, 7) Capcalmacenamiento = Espacio disponible para almacenar productos en los clientes y bodega.

Fase 1: Construcción de una solución inicial.

- 1: For i = 0 To numperiodos
- 2: For j = 0 To numclientes
- 3: //Generar las cantidades aleatoriamente de forma uniforme
- 4: Next j
- 5: Next i
- 6: if (Solución excede capacidad vehículo en algún periodo (\sum matrizCantidades > Vcap), no satisface con la demanda en el periodo “numperiodos” o excede la capacidad de almacenamiento cliente y bodega ($Capcalmacenamiento \geq X_{st}$) then
- 7: corregir matriz de cantidades para satisfacer capacidad del vehículo, demanda y capacidad de almacenamiento en los clientes y bodega.
- 8: End if
- 9: Genera Ruta Maestra - Hacer Clark and Wright para todos los clientes sin restricción de capacidad. Día por día aquellos clientes cuya cantidad a entregar sea igual a cero, son eliminados.
- 10: For i As Integer = 0 To numperiodos - 1
- 11: Ruteo para el periodo i modificando la ruta maestra
- 12: Next i
- 13: End

A continuación se muestra un ejemplo numérico de las diferentes rutas establecidas para visitar a cada uno de los clientes por periodo contemplando la ruta maestra y la matriz de cantidades fijada en la fase 1:

Tabla 4 - 3 Matriz de cantidades generada aleatoriamente para cada cliente y periodo

	Periodo 1	Periodo 2	Periodo 3
Cliente 1	0	0	40
Cliente 2	20	30	10
Cliente 3	100	20	0
Cliente 4	5	0	0
Cliente 5	40	0	0

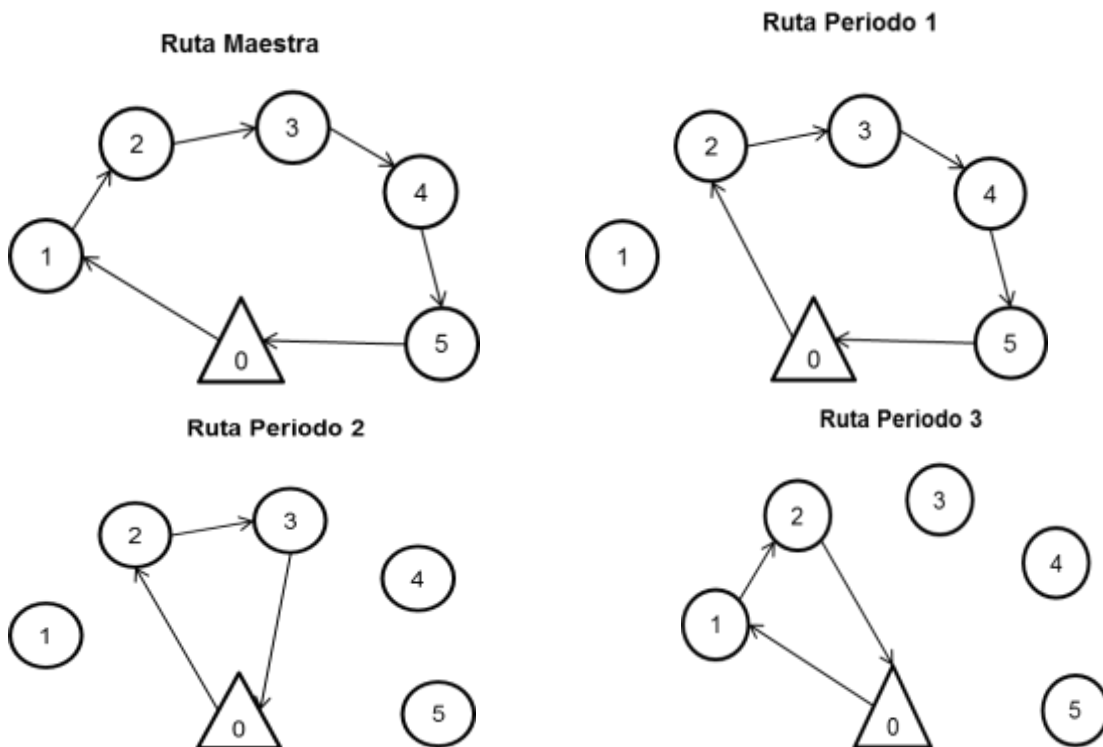
Tabla 4 - 4 Matriz de demanda para cada cliente y periodo

	Periodo 1	Periodo 2	Periodo 3
Cliente 1	20	20	20
Cliente 2	15	15	15
Cliente 3	35	35	35
Cliente 4	10	10	10
Cliente 5	25	25	25

Tabla 4 - 5 Matriz de inventarios para cada cliente y periodo

	Periodo 0	Periodo 1	Periodo 2	Periodo 3
Cliente 1	40	20	0	20
Cliente 2	15	20	35	30
Cliente 3	5	70	55	20
Cliente 4	80	75	65	55
Cliente 5	35	50	25	0

Gráfica 4 - 1 Ejemplo grafico de una solución inicial contemplando 5 clientes y 3 periodos



En la Gráfica 4-1 se observa que la ruta maestra incluye a todos los retailers del problema, minimizando la distancia total de la ruta. En este caso, se utilizó el algoritmo de Clarke and Wright para el computo de la misma. Luego, se modificó la ruta maestra con el fin de adaptar las rutas del periodo del 1 al 3. En el periodo 1 con el cliente 1, en el periodo 2 con los clientes 1, 4 y 5 y en el periodo 3 con los clientes 3, 4 y 5 no son visitados, puesto quedan excluidos de la ruta maestra al no tener cantidades a entregar en esos periodos.

Posteriormente, en la fase 2 se calcula el costo de la función objetivo de la solución inicial y se procede a mejorarla realizando una búsqueda local intercambiando clientes sin modificar la matriz de cantidades realizada en fase 1. Si las soluciones empeoran, esas soluciones son eliminadas contemplando únicamente las mejores soluciones y se detiene hasta realizar todos los movimientos de clientes.

Fase 2: Mejora de la solución inicial.

```
1: Realizar movimientos de cada cliente
2:  $\Delta^* = 0$ 
3:   For x = 1 To numclientes - 2
4:     For t = 0 To numperiodos - 1
5:       For y = 0 To numclientes - 1
6:         Intercambiar clientes Y y X entre sí para la ruta del periodo t
7:         Evaluar nueva Función Objetivo
8:         If (Hay mejora ) then
9:           If ( El ahorro encontrado es mayor a  $\Delta^*$  ) then
10:            Guardar el movimiento de intercambio de Y y X.
11:             $\Delta^* =$  ahorro encontrado
12:          End if
13:        End if
14:      Next y
15:    Next t
16:  Next x
17: If (Hay mejora ) then
18:   Ejecuta el mejor movimiento encontrado
19: End If
20: End
```

5 PRUEBAS COMPUTACIONALES Y ANÁLISIS

Para evaluar el desempeño de los métodos de solución expuestos en la sección anterior, se consideraron 80 instancias de la literatura compuestas por 3 y 6 periodos con 5, 10, 15 y 20 clientes y se encuentran disponibles públicamente en <http://www.leandro-coelho.com/instances/inventory-routing>. Estas instancias han sido estudiadas por Archetti et al. (2007), Bertazzi et al. (2012), Guerrero et al. (2013) y Coelho et al. (2012). Por otro lado, las instancias consideran una demanda constante para cada periodo pero diferente para cada minorista, una matriz de distancia simétrica y costo de almacenamiento de inventario constante. Adicionalmente, las instancias son clasificadas en dos tipos: “Low Cost” (costo bajo de mantener inventario en los minoristas y bodega) y “High Cost” (costo alto de mantener inventario en los minoristas y bodega).

Estas instancias no contemplaban ventanas de tiempo. Por lo tanto, fueron modificadas para el problema en estudio. Se agregan las ventanas de tiempo con una distribución aleatoria discreta uniforme entre 0 y 4.000 minutos, con rangos específicos de acuerdo al tipo problema (e.g. Problemas con 6 periodos y hasta 20 clientes el rango aplicado es de 3.500 min a 4.000 min). Además, se calculó la matriz de tiempos de viaje asumiendo una velocidad del vehículo constante e igual a 30 Km/h; los tiempos de los recorridos y ventanas están dados en minutos (min).

Los experimentos presentados a continuación fueron desarrollados usando un procesador AMD Athlon® 64x2 Dual Core 2.7 GHz con 3 GB de RAM. El solver de programación matemática empleado para encontrar las soluciones fue CPLEX versión 23.5.

5.1 Resultados de los Enfoques Basados en Programación Entera Mixta

A continuación se muestra en las tablas 5-1 y 5-2 los resultados promedios obtenidos para cada una de las instancias estudiadas según su tipo de costo de inventario. Se presentan los tiempos de cómputo y las diferencias porcentuales entre la mejor solución conocida para la versión sin ventanas de tiempo (IRP) y la solución resultante de cada modelo. La columna “N” indica la cantidad de minoristas, la columna “P” indica la cantidad de periodos, la columna “BKS” (Best Known Solution) muestra las mejores soluciones conocidas y la columna “CPU” muestra los tiempos de cómputo para cada uno de los problemas en segundos.

Para el modelo que incluye las ventanas de tiempo (IRPTW), en las tablas 5-1 y 5-2 en la columna 6, se presentan las mejores soluciones encontradas por el método de optimización exacto presentado en la sección 4.1.1. En la columna 8 se presenta el

tiempo de cálculo (CPU - segundos) y en la columna 9 el incremento porcentual (Gap - %) en el costo de la solución por el hecho de incluir la restricción de ventanas de tiempos fuertes. El Gap en esta columna se calcula como el error relativo entre la mejor solución encontrada por el IRP y la mejor solución encontrada por el IRPTW.

Para la heurística SEQ-H se presenta en las tablas 5-1 y 5-2 en la columna 9 y 11, el costo promedio de la mejor solución o una cota inferior y su tiempo computacional asociado. Además, en la columna 10 se presenta la diferencia porcentual (Gap) entre el costo del método SEQ-H y aquella obtenida por el método IRP que ignora las ventanas de tiempo mostrada en la columna 4. El Gap de esta columna se calcula como el error relativo entre la mejor solución encontrada por el SEQ-H y la mejor solución encontrada por el IRP.

Para la heurística SEQ-H (TW) se presenta en las tablas 5-1 y 5-2 en la columna 12, el costo promedio de la mejor solución o una cota inferior con un límite máximo de tiempo de ejecución de 35.000 s. La columna 13 presenta la diferencia porcentual entre el costo del método SEQ-H (TW) y aquella obtenida por el método IRPTW que contempla las ventanas de tiempo fuertes mostrada en la columna 6. El Gap se calcula como el error relativo entre la mejor solución encontrada por el SEQ-H (TW) y la mejor solución encontrada por el IRPTW.

Tabla 5 - 1 Estudio comparativo de los modelos y métodos de optimización para instancias aleatorias del problema de ruteo de inventarios. Low Cost

Tipo de Problema	N	P	IRP		IRPTW			SEQ-H			SEQ-H(TW)		
			BKS [\$]	CPU[s]	BKS [\$]	gap[%]	CPU[s]	SOL [\$]	gap[%]	CPU[s]	SOL [\$]	gap[%]	CPU[s]
1	5	3	1,275.9**	3.5	1,804.3**	36.8	1.0	1,482.1	12.9	6.6	4,093.2	201.4	6.9
2	10	3	1,910.9**	10.4	2,746.9**	44.8	1,369.4	2,345.4	22.8	6.3	17,344.7	469.4	4,594.8
3	15	3	2,207.8**	16.0	2,704.0	22.4	14,089.2	2,796.4	27.5	17.0	4,601.1	79.9	1,206.4
4	20	3	2,665.6	14,728.7	3,392.3	25.9	15,382.1	3,004.1	11.5	4,621.6	10,362.5	232.6	2,596.2
5	5	6	3,136.9**	10.1	3,564.4**	13.3	27.5	3,433.2	8.9	9.6	3,764.2	40.3	5.6
6	10	6	4,612.5**	3,273.3	5,470.9	17.7	28,032.0	5,168.2	12.1	10.7	9,014.6	67.1	3,021.9
7	15	6	5,477.2	12,947.5	5,625.9	2.5	24,936.9	6,225.0	13.9	3,728.7	8,109.6	40.2	1,627.2
8	20	6	6,949.5	16,087.3	6,900.2	-0.7	24,213.8	7,789.4	12.7	7,699.3	8,744.3	30.3	6,377.0
Promedio			3,529.5	5,884.6	4,026.1	20.4	13,506.5	4,030.5	15.3	2,012.5	8,254.3	145.1	2,429.5

Notas:

* Cada tipo de problema está compuesto por 5 instancias con diferentes configuraciones de costos.

** Los resultados que se encuentran resaltados en negrita son los valores óptimos demostrados.

*** Los resultados que se encuentran en cursiva son mejores soluciones encontradas con penalización, es decir, aplicando ventanas de tiempo débiles.

**** NS - No Solución hallada con un límite máximo de tiempo de ejecución de 35.000 s.

En la tabla 5-1 se observa lo siguiente:

Primero, el método de optimización simultáneo basado en programación lineal entera mixta, sin considerar ventanas de tiempo (IRP), logró encontrar soluciones óptimas para la mayoría de instancias, excepto para los tipos de problemas 4, 7 y 8. El tiempo promedio computacional empleado para solucionar todos los problemas fue de 5,884.6 s.

Segundo, el método de optimización simultáneo basado en programación lineal entera mixta, considerando ventanas de tiempo (IRPTW), logró encontrar soluciones óptimas a los problemas de tipo 1, 2 y 5; para los demás tipos de problemas encontró una buena solución con respecto a una cota inferior. El tiempo promedio computacional empleado para solucionar todos los problemas fue de 13,506.5 s con un sobrecosto en promedio del 20.4% comparado con las soluciones óptimas para el problema sin ventanas de tiempo.

Tercero, la heurística SEQ-H logró encontrar soluciones factibles a una gran cantidad de problemas, excepto los problemas de tipo 4, 7 y 8. El tiempo promedio computacional empleado para solucionar todos los problemas fue de 2,012.5 s con un sobrecosto en promedio del 15.3% comparadas con las soluciones encontradas por el IRP.

Cuarto, la heurística SEQ-H (TW) logró encontrar únicamente dos soluciones factibles (problema 1 y 5) pero penalizando la función objetivo por el incumplimiento de las ventanas de tiempo. El tiempo promedio computacional empleado para solucionar todos los problemas fue de 2,429.5 s con un sobrecosto en promedio del 145.1% comparado con las soluciones encontradas por el IRPTW.

Quinto, los problemas 4, 7 y 8 ningún método logró encontrar una solución factible luego de 25.000 s.

Sexto, en el problema 8, el modelo IRPTW encontró una mejor solución con respecto a la solución encontrada por el método IRP, donde redujo el costo en 0.7% debido a que no se logró encontrar una solución óptima para el IRP.

Tabla 5 - 2 Estudio comparativo de los modelos y métodos de optimización para instancias aleatorias del problema de ruteo de inventarios. High Cost

Tipo de Problema	N	P	IRP		IRPTW			SEQ-H			SEQ-H(TW)		
			BKS [\$]	CPU[s]	BKS [\$]	gap[%]	CPU[s]	SOL [\$]	gap[%]	CPU[s]	SOL [\$]	gap[%]	CPU[s]
1	5	3	2,199.9**	1.2	2,584.6**	16.3	4.3	2,360.1	6.7	9.4	6,343.9	181.4	18.1
2	10	3	4,338.0**	3.3	5,000.4**	15.6	629.4	4,817.8	10.8	9.7	NS	NS	9,290.7
3	15	3	5,435.8**	12.2	5,939.4	9.2	24,906.6	6,008.6	11.0	28.6	7,029.5	22.7	2,268.7
4	20	3	7,235.5**	5,473.2	7,625.3	5.3	20,493.4	8,001.7	10.6	4,564.4	10,973.5	45.5	3,473.6
5	5	6	5,354.2**	24.9	5,738.6**	7.3	25.9	5,632.5	4.9	9.3	9,248.3	56.7	8.1
6	10	6	8,607.3**	2,654.9	9,362.2	8.8	28,491.5	9,183.7	6.7	11.0	13,279.1	46.5	4,471.7
7	15	6	11,676.0	12,229.5	11,717.6	0.3	17,079.4	12,374.7	6.0	3,205.5	12,663.2	5.7	1,325.5
8	20	6	15,791.0	15,022.2	14,726.3	-5.5	34,794.2	15,594.6	0.1	3,264.4	15,659.8	6.0	4,367.0
Promedio			7,579.7	4,427.7	7,836.8	7.2	15,803.1	7,996.7	7.1	1,387.8	10,742.5	52.1	3,152.9

Notas:

* Cada tipo de problema está compuesto por 5 instancias con diferentes configuraciones de costos.

** Los resultados que se encuentran resaltados en negrita son los valores óptimos demostrados.

*** Los resultados que se encuentran en cursiva son mejores soluciones encontradas con penalización, es decir, aplicando ventanas de tiempo débiles.

**** NS - No Solución hallada con un límite máximo de tiempo de ejecución de 35.000 s.

Por otro lado, en la tabla 5-2 se observa lo siguiente:

Primero, el enfoque exacto para el modelo IRP logró encontrar soluciones óptimas para la gran mayoría de instancias, excepto para los problemas 7 y 8. El tiempo promedio computacional empleado para solucionar todos los problemas fue de 4,427.7 s.

Segundo, el enfoque exacto para el modelo IRPTW logró encontrar soluciones óptimas a los problemas 1, 2 y 5; el resto de problemas encontró una buena solución con respecto a una cota inferior. El tiempo promedio computacional empleado para solucionar todos los problemas fue de 15,803.1 s con un sobrecosto en promedio del 7.2% comparado con las soluciones encontradas por el IRP.

Tercero, la heurística SEQ-H logró encontrar soluciones factibles a una gran cantidad de problemas, excepto los problemas 4, 7 y 8. El tiempo promedio computacional empleado para solucionar todos los problemas fue de 1,387.8 s con un sobrecosto en promedio del 7.1% comparado con las soluciones encontradas por el IRP.

Cuarto, la heurística SEQ-H (TW) logró encontrar únicamente dos soluciones factibles (problema 1 y 7). La solución encontrada para el problema 1 se logra a través de la

penalización de la función objetivo por el incumplimiento de las ventanas de tiempo; para el problema 7 no se penaliza la función objetivo. El tiempo promedio computacional empleado para solucionar todos los problemas fue de 3,152.9 s convirtiéndose en una gran ventaja frente a los demás métodos que invierten mucho más tiempo computacional para encontrar soluciones factibles. Sin embargo, se debe contemplar que existe un sobrecosto en promedio del 52.1% comparado con las soluciones encontradas por el IRPTW.

Quinto, el problema 2 con la heurística SEQ-H (TW) no logró encontrar ninguna solución factible luego de 9,290.7 segundos.

Sexto, para el problema 8 ningún método logró encontrar una solución factible luego de 34,794.2 s. Sin embargo, el método IRPTW encontró una mejor solución en comparación al IRP, donde redujo el costo en 5.5% debido a que no se logró encontrar una solución óptima para el IRP.

Otro factor importante a contemplar dentro del análisis son los tiempos computacionales requeridos por cada método para encontrar las diferentes soluciones a cada una de las instancias, puesto que es relevante conocer dentro de un ambiente real el tiempo requerido para configurar la red de distribución. En la tabla 5-3 se muestra un cuadro comparativo de los tiempos de cómputo empleados por cada uno de los métodos.

Tabla 5 - 3 Cuadro comparativo de los Tiempos de Computo según el método de optimización

	Promedio - CPU [s]		Promedio [s]	Diferencia Relativa (%)
	Low Cost	High Cost		
IRP	5.884,6	4.427,7	5.156,2	-
IRPTW	15.803,1	13.506,5	14.654,8	184,2%
SEQ-H	1.387,8	2.012,5	1.700,1	-67,0%
SEQ-H(TW)	3.152,9	2.429,5	2.791,2	-81,0%

A partir de estos resultados, se puede analizar que cuando se incluye ventanas de tiempo al problema de ruteo de inventarios (IRPTW) se obtiene un incremento promedio en los costos del 13.8% (Low Cost y High Cost) con respecto al costo promedio de las soluciones encontradas para el IRP y un incremento promedio en los tiempos de ejecución del modelo IRPTW del 184.2% con respecto a los tiempos computacionales obtenidos para el IRP como se observa en la tabla 5-3. Este incremento es razonable, puesto al incluir este tipo de restricciones de ventanas de tiempo fuertes restringe que el

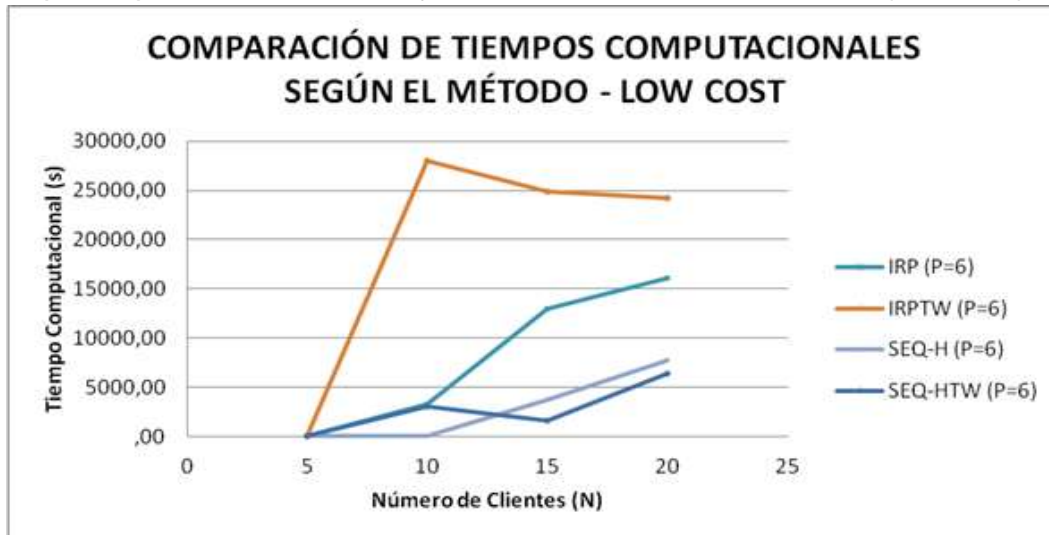
vehículo entregue pedidos a los clientes a cualquier hora, generando más rutas de abastecimiento con el objetivo de cumplir con las entregas dentro de los tiempos de atención establecidos.

En las Gráficas 5-1, 5-2, 5-3 y 5-4 se ilustran el comportamiento de los tiempos computacionales para cada uno de los métodos analizados.

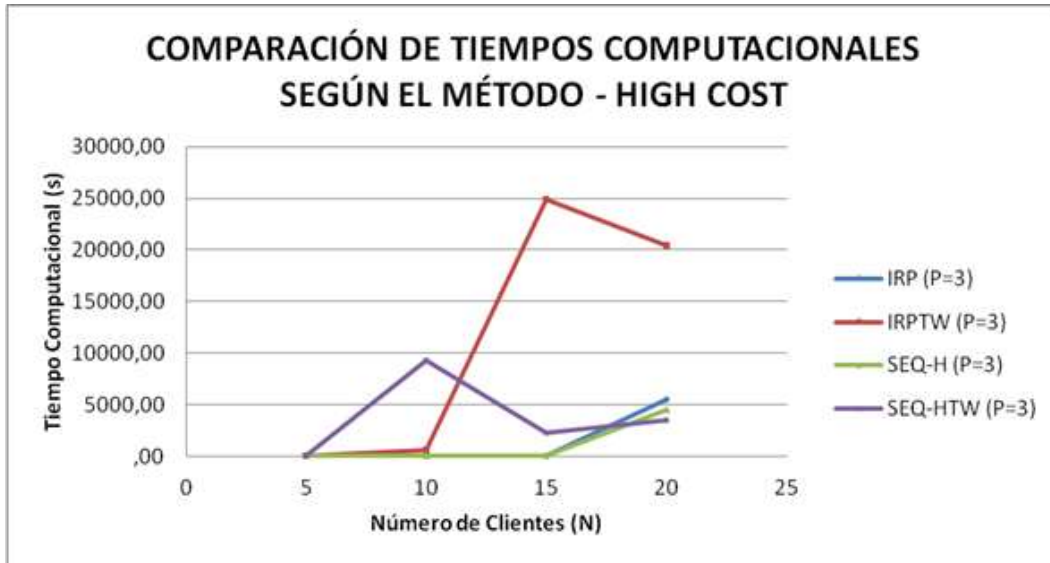
Gráfica 5 - 1 Comparación de tiempos computacionales según el método de optimización para 3 periodos con costos bajos de mantenimiento de inventario (Low Cost)



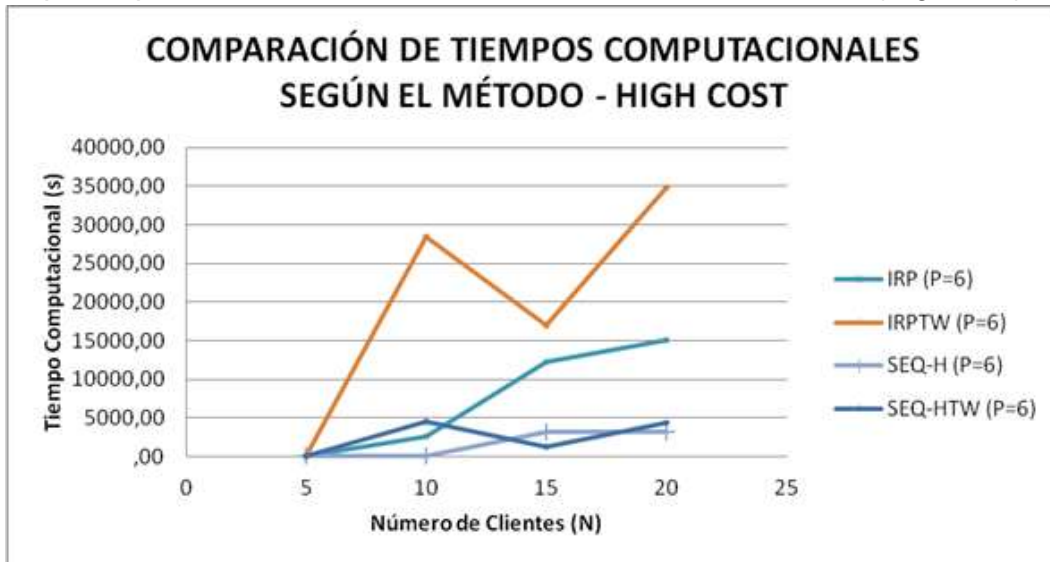
Gráfica 5 - 2 Comparación de tiempos computacionales según el método de optimización para 6 periodos con costos bajos de mantenimiento de inventario (Low Cost)



Gráfica 5 - 3 Comparación de tiempos computacionales según el método de optimización para 3 periodos con costos altos de mantenimiento de inventario (High Cost)



Gráfica 5 - 4 Comparación de tiempos computacionales según el método de optimización para 6 periodos con costos altos de mantenimiento de inventario (High Cost)



De acuerdo a las Gráficas 5-1, 5-2, 5-3 y 5-4 existe una tendencia creciente en los tiempos computacionales para todos los métodos cuando se incrementa el tamaño de la instancia. A mayor cantidad de periodos y minoristas, el tiempo de ejecución del software (GAMS - CPLEX) incrementa para encontrar una solución. Sin embargo, para el método IRPTW cuando soluciona instancias de bajo costo, 6 periodos y más de 10 clientes los

tiempos computacionales decrecen, como lo muestra la Gráfica 5-2, debido a que no se logra encontrar soluciones óptimas y se presenta el tiempo de cálculo de una cota superior. Además, se observa el mismo comportamiento con el mismo método para las instancias de alto costo, con 3 periodos y más de 15 clientes (ver Gráfica 5-3).

Analizando la heurística de ruteo de inventarios secuencial sin ventanas de tiempo (SEQ-H) muestra un incremento promedio (Low cost y High cost) en los costos del 11.2% con respecto a la soluciones encontradas por el modelo IRP. Por otro lado, se observa en promedio que los tiempos de cómputo del modelo SEQ-H son menores en 67.0 % con respecto a los tiempos de computo del modelo IRP como se observa en la tabla 5-3.

Es importante resaltar que el modelo de ruteo secuencial con ventanas de tiempo (SEQ-H (TW)) muestra en promedio un aumento en el costo del 98.6% comparado con el IRPTW. Este resultado se debe a las penalizaciones en la función objetivo aplicadas por la violación de las restricciones de ventanas de tiempo, cuando el método SEQ-H (TW) no logra encontrar soluciones factibles al problema original, y por lo tanto, se propone relajar la restricción como fue explicado en la sección 4.1.2. La penalización propuesta es de \$10 por cada minuto de tiempo que incumpla las ventanas asignadas a cada cliente. En total se resolvieron 80 instancias, de las cuales 65 no cumplieron con las ventanas y fueron penalizadas.

Las ventanas de tiempo fuertes dentro de los métodos que contemplan estas restricciones (IRPTW y SEQ-H (TW)) tienen un gran impacto en los tiempos de ejecución, es decir, entre más estrecha sea la ventana mayor es el tiempo requerido para el software encontrar una solución; los tiempos computacionales del modelo IRPTW son considerablemente elevados en comparación a la heurística secuencial con ventanas de tiempo. Sin embargo se debe considerar que el método SEQ-H (TW) fue relajado aplicando penalizaciones por el incumplimiento de las ventanas para lograr encontrar una solución factible. Finalmente, dentro de las soluciones encontradas, algunas no fueron óptimas debido a la complejidad del problema y a la capacidad del equipo en donde se ejecutaron las corridas.

5.2 Resultados de los Enfoques Basados en la Heurística Constructiva

A continuación se muestra en las tablas 5-4 y 5-5 los resultados promedios de los costos y tiempos computacionales obtenidos para cada una de las instancias estudiadas según su tipo de costo de inventario. La columna "N" indica la cantidad de minoristas, la columna "P" indica la cantidad de periodos, la columna "BKS" (*Best Known Solution*) muestra las mejores soluciones conocidas, la columna "SOL" muestra las soluciones factibles y la

columna "CPU" muestra los tiempos de cómputo empleados para cada uno de los problemas en segundos. Para el modelo que incluye las ventanas de tiempo (IRPTW) y la heurística SEQ-H (TW), en las tablas 5-4 y 5-5 se presenta la misma información expuesta en la sección 5.1.

Para la heurística constructiva con ventanas de tiempo - HC (TW) - se presenta en las tablas 5-4 y 5-5 en la columna 8, el costo promedio de la mejor solución o una cota inferior y en la columna 9, el tiempo computacional promedio invertido para encontrar las diferentes soluciones factibles.

Tabla 5-4 Estudio comparativo Heurística Constructiva (HC) contra modelo IRPTW y Heurística Secuencial TW para instancias aleatorias del problema de ruteo de inventarios. Low Cost

Problema	N	P	IRPTW		SEQ-H(TW)		HC(TW)	
			BKS [\$]	CPU[s]	SOL [\$]	CPU[s]	SOL [\$]	CPU[s]
1	5	3	2.584,6**	4,3	6.343,9	18,1	<i>444.648,2</i>	0,9
2	10	3	5.000,4**	629,4	NS	9.290,7	<i>1.343.929,1</i>	2,9
3	15	3	5.939,4	24.906,6	<i>7.029,5</i>	2.268,7	<i>2.283.805,5</i>	6,3
4	20	3	7.625,3	20.493,4	<i>10.973,5</i>	3.473,6	<i>3.964.296,6</i>	12,3
5	5	6	5.738,6**	25,9	9.248,3	8,1	<i>513.913,4</i>	1,7
6	10	6	9.362,2	28.491,5	<i>13.279,1</i>	4.471,7	<i>1.476.672,2</i>	6,1
7	15	6	11.717,6	17.079,4	12.663,2	1.325,5	<i>2.451.465,4</i>	14,8
8	20	6	14.726,3	34.794,2	<i>15.659,8</i>	4.367,0	<i>3.865.520,1</i>	29,9
Promedio			<i>7.836,8</i>	<i>15.803,1</i>	<i>10.742,5</i>	<i>3.152,9</i>	<i>2.043.031,3</i>	<i>9,4</i>

Notas:

* Cada tipo de problema está compuesto por 5 instancias con diferentes configuraciones de costos.

** Los resultados que se encuentran resaltados en negrita son los valores óptimos demostrados.

*** Los resultados que se encuentran en cursiva son mejores soluciones encontradas con penalización, es decir, aplicando ventanas de tiempo débiles.

**** NS - No Solución hallada con un límite máximo de tiempo de ejecución de 35.000 s.

Tabla 5-5 Estudio comparativo Heurística Constructiva (HC) contra modelo IRPTW y Heurística Secuencial TW para instancias aleatorias del problema de ruteo de inventarios. High Cost

Problema	N	P	IRPTW		SEQ-H(TW)		HC(TW)	
			BKS [\$]	CPU[s]	SOL [\$]	CPU[s]	SOL [\$]	CPU[s]
1	5	3	2.584,6**	4,3	6.343,9	18,1	449.556,1	0,7
2	10	3	5.000,4**	629,4	NS	9.290,7	1.280.365,9	3,0
3	15	3	5.939,4	24.906,6	<i>7.029,5</i>	2.268,7	<i>2.296.095,0</i>	6,7
4	20	3	7.625,3	20.493,4	<i>10.973,5</i>	3.473,6	<i>3.899.753,0</i>	12,9
5	5	6	5.738,6**	25,9	9.248,3	8,1	569.598,1	1,7
6	10	6	9.362,2	28.491,5	<i>13.279,1</i>	4.471,7	<i>1.565.596,0</i>	6,3
7	15	6	11.717,6	17.079,4	<i>12.663,2</i>	1.325,5	<i>2.536.431,3</i>	15,3
8	20	6	14.726,3	34.794,2	<i>15.659,8</i>	4.367,0	<i>4.004.631,4</i>	30,6
Promedio			7.836,8	15.803,1	10.742,5	3.152,9	2.075.253,3	9,6

Notas:

* Cada tipo de problema está compuesto por 5 instancias con diferentes configuraciones de costos.

** Los resultados que se encuentran resaltados en negrita son los valores óptimos demostrados.

*** Los resultados que se encuentran en cursiva son mejores soluciones encontradas con penalización, es decir, aplicando ventanas de tiempo débiles.

**** NS - No Solución hallada con un límite máximo de tiempo de ejecución de 35.000 s.

Analizando los resultados de la heurística constructiva con ventanas de tiempo registradas en las tablas 5-4 y 5-5, se observa que las soluciones encontradas por el método no son de buena calidad al tener un costo elevado con respecto a las soluciones encontradas por el IRPTW y SEQ-H (TW). Sin embargo, los tiempos computacionales son mejores, puesto que, logra encontrar en promedio soluciones factibles a todos los problemas en 9.6 s a diferencia del IRPTW y SEQ-H (TW) que en promedio logran encontrar las diferentes soluciones en 7.836,8 s y 10.742,5 s respectivamente, convirtiéndose en una herramienta para encontrar soluciones iniciales rápidamente que podrían ser mejoradas por otros métodos heurísticos. Para mejorar las diferentes soluciones encontradas por la heurística constructiva, se propondrá realizar movimientos de los inventarios por cliente y periodo, explorando hacer mejoras en las decisiones de inventarios, lo cual puede tener un impacto considerable en el desempeño de la heurística.

CONCLUSIONES

Tradicionalmente, la investigación en logística interna y de transporte se ha concentrado en optimizar los modelos de inventarios y ruteo de vehículos (VRP) de forma separada o secuencial. Recientemente se han estudiado los problemas de ruteo de inventarios (Inventory Routing Problems - IRP). Estos optimizan conjuntamente el transporte y las decisiones de gestión de inventarios con el objetivo de reducir los costos logísticos asociados y se ha demostrado que integrar estas dos decisiones potencialmente reduce de manera importante los costos comparado con el enfoque tradicional. En este trabajo de investigación se ha presentado una formulación matemática para el problema de ruteo de inventarios con fuertes ventanas de tiempo, demanda determinística, y un solo vehículo. Se presentan dos métodos de optimización basados en programación entera mixta: El primero considera las decisiones de forma simultánea, y el segundo es una heurística que optimiza las decisiones de forma secuencial. Ambos son capaces de optimizar el ruteo de inventarios con y sin ventanas de tiempo fuertes.

El algoritmo de solución exacto desarrollado logra encontrar soluciones óptimas para instancias de hasta 10 clientes con 3 periodos de tiempo y de hasta 5 clientes con 6 periodos de tiempo. Esto evidencia de forma empírica la complejidad del problema y la necesidad del uso de métodos heurísticos para instancias de mayor tamaño.

Adicionalmente se presenta un estudio sobre el impacto que tiene las diferentes configuraciones de ventanas de tiempo en el problema IRPTW. Los resultados muestran que no solamente los costos de las decisiones de ruteo se ven afectados, sino que adicionalmente los costos de gestión de inventarios sufren variaciones debido a los rangos de ventanas de tiempo que se pueden presentar. El costo total se incrementa en promedio 12.53% para las instancias estudiadas cuando se tienen ventanas estrechas y disminuye en promedio 6,08% cuando se presentan ventanas amplias.

Como investigación futura se plantea el desarrollo de algoritmos capaces de calcular cotas inferiores para instancias de mayor tamaño y el desarrollo de heurísticas para obtener mejores cotas superiores. Adicionalmente se puede plantear un problema con ventanas de tiempo débiles (Soft Time Windows), múltiples vehículos, o una flota heterogénea de vehículos, o demanda estocástica.

Productos de investigación

Resultados preliminares de esta tesis fueron presentados en las siguientes conferencias científicas:

- E. Pérez, W.J. Guerrero (2014) Heurística para el problema de ruteo de inventarios con ventanas de tiempo. Primer Congreso Internacional de Logística & Supply Chain Management: Alianza del Pacífico, Un Mundo de Oportunidades. Octubre 6 - 7. Bogotá, Colombia.
- E. Pérez, W.J. Guerrero (2015). Ruteo de inventarios con ventanas de tiempo fuertes. Congreso de la Asociación Colombiana de Investigación Operativa – ASOCIO. Julio 15 – 16. Chía, Colombia.

BIBLIOGRAFÍA

- Andersson, H., Hoff, A., Christiansen, M., Hasle, G., & Løkketangen, A. (2010). Industrial aspects and literature survey: Combined inventory management and routing. *Computers & Operations Research*, 37(9), 1515–1536. doi:10.1016/j.cor.2009.11.009
- Archetti, C., Bertazzi, L., Hertz, A., & Speranza, M. G. (2012). A Hybrid Heuristic for an Inventory Routing Problem. *INFORMS Journal on Computing*, 24(1), 101–116. doi:10.1287/ijoc.1100.0439.
- Archetti, C., Bertazzi, L., Laporte, G., & Speranza, M. G. (2007). A Branch-and-Cut Algorithm for a Vendor-Managed Inventory-Routing Problem. *Transportation Science*, 41(3), 382–391. doi:10.1287/trsc.1060.0188.
- Baños, R., Ortega, J., Gil, C., Márquez, A. L., & de Toro, F. (2013). A hybrid meta-heuristic for multi-objective vehicle routing problems with time windows. *Computers & Industrial Engineering*, 65(2), 286–296.
- Bard, J. F., & Nananukul, N. (2009). Heuristics for a multiperiod inventory routing problem with production decisions. *Computers & Industrial Engineering*, 57(3), 713–723. doi:10.1016/j.cie.2009.01.020.
- Bertazzi, L., Bosco, A., Guerriero, F., & Laganà, D. (2013). A stochastic inventory routing problem with stock-out. *Transportation Research Part C: Emerging Technologies*, 27, 89–107. doi:10.1016/j.trc.2011.06.003.
- Bräysy, O., & Gendreau, M. (2005a). Vehicle Routing Problem with Time Windows, Part I: Route construction and local search algorithms. *Transportation Science*, 39(1), 104–118. doi:10.1287/trsc.1030.0056.
- Bräysy, O., & Gendreau, M. (2005b). Vehicle Routing Problem with Time Windows, Part II: Metaheuristics. *Transportation Science*, 39(1), 119–139. doi:10.1287/trsc.1030.0057.
- Calvete, H. I., Galé, C., Oliveros, M.-J., & Sánchez-Valverde, B. (2007). A goal programming approach to vehicle routing problems with soft time windows. *European Journal of Operational Research*, 177(3), 1720–1733. doi:10.1016/j.ejor.2005.10.010.
- Chang, T.-S., Wan, Y., & Ooi, W. T. (2009). A stochastic dynamic traveling salesman problem with hard time windows. *European Journal of Operational Research*, 198(3), 748–759. doi:10.1016/j.ejor.2008.10.012.

- Clarke, G. U., & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations research*, 12(4), 568-581. doi.org/10.1287/opre.12.4.568
- Coelho, L. C., & Laporte, G. (2013). The exact solution of several classes of inventory-routing problems. *Computers & Operations Research*, 40(2), 558-565. doi:10.1016/j.cor.2012.08.012.
- Coelho, L. C., Cordeau, J.-F., & Laporte, G. (2012). Consistency in multi-vehicle inventory-routing. *Transportation Research Part C: Emerging Technologies*, 24, 270-287. doi:10.1016/j.trc.2012.03.007.
- (1) Coelho, L. C., Cordeau, J.-F., & Laporte, G. (2012). The inventory-routing problem with transshipment. *Computers & Operations Research*, 39(11), 2537-2548. doi:10.1016/j.cor.2011.12.020.
- Cordeau, J. F., Laporte, G., & Mercier, A. (2001). A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational research society*, 52(8), 928-936. doi:10.1057/palgrave.jors.2601163.
- Dondo, R., Méndez, C. a., & Cerdá, J. (2011). The multi-echelon vehicle routing problem with cross docking in supply chain management. *Computers & Chemical Engineering*, 35(12), 3002-3024. doi:10.1016/j.compchemeng.2011.03.028.
- Gronhaug, R., Christiansen, M., Desaulniers, G., & Desrosiers, J. (2010). A Branch-and-Price Method for a Liquefied Natural Gas Inventory Routing Problem. *Transportation Science*, 44(January 2015), 400-415. doi:10.1287/trsc.1100.0317.
- Guerrero, W. J., Prodhon, C., Velasco, N., & Amaya, C. a. (2013). Hybrid heuristic for the inventory location-routing problem with deterministic demand. *International Journal of Production Economics*, 146(1), 359-370. doi:10.1016/j.ijpe.2013.07.025.
- Liu, S.-C., & Lee, W.-T. (2011). A heuristic method for the inventory routing problem with time windows. *Expert Systems with Applications*, 38(10), 13223-13231. doi:10.1016/j.eswa.2011.04.138.
- Moin, N. H., Salhi, S., & Aziz, N. A. B. (2011). An efficient hybrid genetic algorithm for the multi-product multi-period inventory routing problem. *International Journal of Production Economics*, 133(1), 334-343. doi: 10.1016/j.ijpe.2010.06.012.

- Qin, L., Miao, L., Ruan, Q., & Zhang, Y. (2014). A local search method for periodic inventory routing problem. *Expert Systems with Applications*, 41(2), 765–778. doi:10.1016/j.eswa.2013.07.100.
- Qureshi, a. G., Taniguchi, E., & Yamada, T. (2009). An exact solution approach for vehicle routing and scheduling problems with soft time windows. *Transportation Research Part E: Logistics and Transportation Review*, 45(6), 960–977. doi:10.1016/j.tre.2009.04.007.
- Qureshi, A. G., Taniguchi, E., & Yamada, T. (2012). A Microsimulation Based Analysis of Exact Solution of Dynamic Vehicle Routing with Soft Time Windows. *Procedia - Social and Behavioral Sciences*, 39, 205–216. doi:10.1016/j.sbspro.2012.03.102.
- Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35 (2), 254–265. doi:10.1287/opre.35.2.254.
- Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2013). A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. *Computers & Operations Research*, 40(1), 475–489. doi:10.1016/j.cor.2012.07.018.
- Wang, X., & Regan, A. C. (2002). Local truckload pickup and delivery with hard time window constraints. *Transportation Research Part B: Methodological*, 36(2), 97–112. doi:10.1016/S0965-8564(00)00037-9.

ANEXOS

Anexo 1. Tabla de estudio comparativo de los modelos y métodos de optimización para instancias aleatorias del problema de ruteo de inventarios. High Cost – Detalle de las 80 instancias.

Anexo 2. Tabla de estudio comparativo de los modelos y métodos de optimización para instancias aleatorias del problema de ruteo de inventarios. Low Cost – Detalle de las 80 instancias.

Anexo 3 Estudio comparativo Heurística Constructiva (HC) contra modelo IRPTW y Heurística Secuencial TW para instancias aleatorias del problema de ruteo de inventarios. Low Cost - Detalle de las 80 instancias.

Anexo 4 Estudio comparativo Heurística Constructiva (HC) contra modelo IRPTW y Heurística Secuencial TW para instancias aleatorias del problema de ruteo de inventarios. High Cost - Detalle de las 80 instancias.