

**Solución al problema de ruteo de vehículos Milk-run con ventanas de tiempo mediante la  
Metaheurística Búsqueda Tabú**

**Laura Ximena Sánchez Rincón**

**Trabajo de Grado para Optar al título de Ingeniería Industrial**

**Director**

**Ms. Carlos Eduardo Díaz Bohórquez**

**Codirector**

**Lina Mayerly Lozano Suarez  
Ingeniera Industrial**

**Universidad Industrial de Santander  
Facultad de Ingenierías Fisicomecánicas  
Escuela de Estudios Industriales y Empresariales  
Bucaramanga  
2020**

### **Agradecimiento**

Dios, por medio de la vida, nos brinda oportunidades, nos pone ángeles en el camino y finalmente, nos espera al final con los brazos abiertos en la paz de su amor y misericordia, por eso hoy le agradezco en particular por la bendición de poder graduarme como Ingeniera industrial, al cumplir con el requisito culmen de mi anhelado y luchado proyecto de grado, para el cual como siempre conté con la presencia de mis Padres y mis tres Hermanas, a quienes agradezco su apoyo, consejo y amor especialmente en este proceso. Dios, también dispuso dos personas especiales para que fueran mi compañía en esta travesía, mi director de tesis el profesor Carlos Díaz y mi apreciada codirectora Lina Lozano, a quienes agradezco el enriquecimiento intelectual que me aportaron en esta parte de mi vida, su paciencia, su ayuda y sus sugerencias invaluable y necesarias para que hoy esté dando gracias por mi proyecto terminado.

También Dios dispuso otras personas para que me apoyaran en este recorrido, amigos, compañeros del banco, jefes y familiares a quienes agradezco sus palabras de aliento y su presencia en esta etapa y finalmente a ti, que me has apoyado y acompañado haciéndome saber que estás para mí, de diferentes formas y en cualquier momento.

¡Gracias Dios, Gracias a la vida, Gracias a ustedes y gracias a mí!

**Tabla de Contenido**

|                                                                                                                                 | <b>Pág.</b> |
|---------------------------------------------------------------------------------------------------------------------------------|-------------|
| Introducción .....                                                                                                              | 13          |
| 1. Planteamiento del problema .....                                                                                             | 15          |
| 2. Justificación del proyecto .....                                                                                             | 16          |
| 3. Objetivos .....                                                                                                              | 18          |
| 3.1. Objetivo General .....                                                                                                     | 18          |
| 3.2. Objetivos Específicos .....                                                                                                | 18          |
| 4. Revisión de literatura .....                                                                                                 | 19          |
| 4.1. Análisis bibliométrico .....                                                                                               | 19          |
| 4.2. Análisis preliminar de la revisión literaria .....                                                                         | 24          |
| 4.2.1. Problema de ruteo de vehículo con ventana de tiempo .....                                                                | 25          |
| 4.2.2. Problema de ruteo de vehículo Milk-run con ventana de tiempo .....                                                       | 28          |
| 5. Marco teórico .....                                                                                                          | 33          |
| 5.1. Optimización Combinatoria .....                                                                                            | 33          |
| 5.2. Tipos de complejidad Computacional .....                                                                                   | 33          |
| 5.3. Cadena de suministro .....                                                                                                 | 35          |
| 5.4. Logística .....                                                                                                            | 35          |
| 5.5. Logística de aprovisionamiento .....                                                                                       | 36          |
| 5.6. Logística Milk-run .....                                                                                                   | 37          |
| 5.7. Problema de Ruteo de Vehículos (VRPTWs) vs Problema de Ruteo de Vehículos Milk-Run con ventanas de tiempo (MRVRPTWs) ..... | 40          |
| 5.8. Métodos de solución para el problema de ruteo de vehículos (VRP) .....                                                     | 41          |
| 5.8.1. Métodos de solución exactos .....                                                                                        | 42          |

|        |                                                                                                           |    |
|--------|-----------------------------------------------------------------------------------------------------------|----|
| 5.8.2. | Métodos de solución heurísticos .....                                                                     | 42 |
| 5.8.3. | Métodos de solución Metaheurísticos .....                                                                 | 43 |
| 5.8.1. | Métodos de solución para el problema de ruteo de vehículo Milk-run con ventana de tiempo (MRVRPTWs) ..... | 44 |
| 5.9.   | Metaheurística Búsqueda Tabú .....                                                                        | 44 |
| 5.9.1. | Conceptos importantes .....                                                                               | 44 |
| 5.9.2. | Características generales .....                                                                           | 45 |
| 5.9.3. | Componentes de la Metaheurística Búsqueda Tabú .....                                                      | 45 |
| 6.     | Problema Ruteo de vehículo Milk-run con ventana de tiempo (MRVRPTWs) .....                                | 47 |
| 6.1.   | Descripción del problema.....                                                                             | 47 |
| 6.2.   | Formulación del problema .....                                                                            | 47 |
| 6.2.1. | índices.....                                                                                              | 48 |
| 6.2.2. | Variables.....                                                                                            | 49 |
| 6.2.3. | Parámetros .....                                                                                          | 49 |
| 6.2.4. | Restricciones .....                                                                                       | 50 |
| 7.     | Algoritmo propuesto al VRPMR-TW .....                                                                     | 52 |
| 7.1.   | Vector inicial .....                                                                                      | 52 |
| 7.2.   | Función objetivo.....                                                                                     | 55 |
| 7.2.1. | Costo total asociado a distancia total recorrida.....                                                     | 55 |
| 7.2.2. | Costo total asociado por llegar fuera de la ventana de tiempo.....                                        | 56 |
| 7.3.   | Generación de vecindario.....                                                                             | 57 |
| 7.4.   | Lista Tabú.....                                                                                           | 59 |
| 8.     | Validación del algoritmo.....                                                                             | 62 |
| 8.1.   | Análisis de Varianza.....                                                                                 | 62 |
| 8.2.   | Validación de metaheurística Búsqueda Tabú con Lista Tabú de 3 .....                                      | 64 |
| 8.3.   | Validación de metaheurística Búsqueda Tabú modificado (Lista Tabú de 4) .....                             | 65 |

|     |                                  |    |
|-----|----------------------------------|----|
| 9.  | Resultados .....                 | 67 |
| 10. | Conclusiones .....               | 70 |
| 11. | Recomendaciones .....            | 71 |
|     | Referencias Bibliográficas ..... | 72 |

**Lista de Tablas**

|                                                                                             | <b>Pág.</b> |
|---------------------------------------------------------------------------------------------|-------------|
| Tabla 1. Cumplimiento de objetivos del proyecto .....                                       | 14          |
| Tabla 2 Ecuación de búsqueda utilizada en la base de datos Scopus, ISI Web of Science ..... | 19          |
| Tabla 3. Afiliación de los principales países. ....                                         | 23          |
| Tabla 4. Análisis comparativo entre ruteo de vehículo y ruteo de vehículo Milk run.....     | 41          |
| Tabla 5. Matriz distancia.....                                                              | 53          |
| Tabla 6. Ventana de tiempo .....                                                            | 54          |
| Tabla 7. Carga a recoger en cada nodo .....                                                 | 54          |
| Tabla 8. Matriz VI_T y Vector C_ALL .....                                                   | 59          |
| Tabla 9. Experimentos para el análisis factorial $2^k$ .....                                | 62          |
| Tabla 10. Análisis de varianza para el diseño experimental .....                            | 63          |
| Tabla 11. Resultados al modificar metaheurística búsqueda Tabú.....                         | 65          |
| Tabla 12. Resultado de búsqueda Tabú tradicional .....                                      | 67          |
| Tabla 13. Resultado de búsqueda Tabú con número Tabú de 4 .....                             | 68          |

**Lista de Figuras**

|                                                                                                                                                                              | <b>Pág.</b> |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| Figura 1. Protocolo de revisión.....                                                                                                                                         | 20          |
| Figura 2. Nube de palabras de revisión bibliométrica. ....                                                                                                                   | 21          |
| Figura 3 Cluster map de autores. ....                                                                                                                                        | 22          |
| Figura 4. Análisis de los países por publicación según análisis bibliométrico.....                                                                                           | 23          |
| Figura 5. Carga y transporte en el sistema Milk-run en logística de entrada.....                                                                                             | 30          |
| Figura 6. Cadena de suministro. ....                                                                                                                                         | 35          |
| Figura 7. Procesos y flujos logísticos ....                                                                                                                                  | 37          |
| Figura 8. Diferencia entre envío directo (Direct Shipment), Cross-dock (Shipment through CD) y Milk-run para una red que comprende proveedor de partes (part suppliers)..... | 38          |
| Figura 9. Sistema de Milk-run .....                                                                                                                                          | 39          |
| Figura 10. Funcionamiento de la Metaheurística Búsqueda Tabú.....                                                                                                            | 46          |
| Figura 11. Algoritmo de barrido .....                                                                                                                                        | 53          |
| Figura 12. Pseudocódigo del vector inicial.....                                                                                                                              | 55          |
| Figura 13. Determinación de la distancia total recorrida.....                                                                                                                | 56          |
| Figura 14. Algoritmo costo.....                                                                                                                                              | 57          |
| Figura 15. Intercambio de nodos para el algoritmo metaheurística búsqueda Tabú .....                                                                                         | 58          |
| Figura 16. Forma de guardar vectores en la Lista Tabú. ....                                                                                                                  | 59          |
| Figura 17. Búsqueda Tabú. ....                                                                                                                                               | 61          |
| Figura 18. Gráfico de Pareto, .....                                                                                                                                          | 63          |
| Figura 19. Grafica de efectos principales para tiempo de cómputo.....                                                                                                        | 64          |
| Figura 20. Proceso iterativo para Lista Tabú de 3 .....                                                                                                                      | 65          |
| Figura 21. Proceso iterativo para Lista Tabú de 4 .....                                                                                                                      | 66          |

**Lista de Apéndices**

Apéndice A. Matriz\_Distancia

Apéndice B. TABÚ\_INICIAL.m

Apéndice C. vectorIniciacion,m

Apéndice D. Artículo científico

## Resumen

**Título:** Solución al problema de ruteo de vehículos Milk-run con ventanas de tiempo mediante la Metaheurística Búsqueda Tabú\*

**Autora:** Laura Ximena Sánchez Rincón\*\*

**Palabras claves:** Búsqueda Tabú, Milk-run, ruteo de vehículo, ventana de tiempo.

### Descripción:

El problema de ruteo de vehículo Milk-run es aplicado a la logística de abastecimiento, o sea, puede ejecutarse en la recolección de mercancía, sin embargo, este busca mejorar el sistema de carga y descarga mediante el uso de “pallets”, ahorrando el tiempo de estadía de cada vehículo. Este problema es del tipo NP Hard, ya que al aumentar el número de actividades a procesar y estar limitado por la disponibilidad de recursos aumenta la complejidad computacional, por tal razón es necesario ser desarrollado mediante la implementación de métodos de optimización combinatoria que alcance una buena solución.

Para dar solución al problema de ruteo de vehículo Milk-run con ventana de tiempo se diseñó un algoritmo basado en metaheurística búsqueda Tabú, donde la ruta inicial se generó a partir del método del barrido, para ello se utiliza el concepto de Lista Tabú con una pequeña modificación, pues habitualmente la Lista Tabú es de 3, mientras en la presente investigación es un parámetro de entrada que puede tener cualquier valor numérico mientras sea entero, logrando un mejor resultado. Se comparó con una instancia de la literatura, logrando un mejor resultado con un valor de cuatro en la Lista Tabú.

Se realizó un diseño experimental  $2^k$  con cinco réplicas los factores son número de Lista Tabú (B) e iteraciones (A), obteniendo que B impacta mayormente en el tiempo de cómputo en comparación a A.

---

\* Trabajo de grado

\*\* Facultad de Ingeniería Fisicomecánicas. Escuela de Estudios Industriales y Empresariales. Director: MSc. Carlos Eduardo Díaz Bohórquez

### Abstract

**Title:** Solution to Milk-run vehicle routing problem with time windows using Metaheuristic Tabú Search \*

**Author:** Laura Ximena Sánchez Rincón \*\*

**Key word:** Tabú Search, Milk-run, routing vehicle, time window.

### Description:

The Milk-run vehicle routing problem is applied to supply logistics, that is, it can be carried out in the collection of merchandise, however, it seeks to improve the loading and unloading system through the use of "pallets", saving the length of stay of each vehicle. This problem is of the NP Hard type, since by increasing the number of activities to be processed and being limited by the availability of resources, computational complexity increases, for this reason it must be developed by implementing combinatorial optimization methods that reaches a good solution.

To solve the Milk-run vehicle routing problem with time window, an algorithm based on metaheuristic Tabú Search was designed, where the initial route was generated from the sweep method, for this the concept of Tabú list with a small modification, since usually the Tabú list is 3, while in the present investigation it is an input parameter that can have any numerical value while it is integer, achieving a better result. It was compared with an instance of the literature, achieving a better result with a value of four in the Tabú list.

An experimental 2k design was carried out with five replications, the factors are Tabú list number (B) and iterations (A), obtaining that B has a greater impact on the computation time compared to A.

---

\* Bachelor Thesis

\*\* Faculty of Physic-mechanical Engineering. School of Industrial and Business Studies. Director: MSc. Carlos Eduardo Díaz Bohórquez

## Introducción

La logística de aprovisionamiento es un factor clave para las organizaciones, pues este puede llegar hasta el 60% del capital suministrado a la empresa. En la logística de aprovisionamiento están incluido las políticas de compras, gestión de transporte y distribución, diseño de los almacenes, gestión de inventarios, entre otras (Servera-Francés, 2010).

Milk-run en logística se puede interpretar como visitar un punto una sola vez, lo cual no es necesariamente obligatorio en el problema de ruteo de vehículos con ventana de tiempo (Bianchessi, Drexl, & Irnich, 2019), además, la logística Milk-run busca optimizar los procesos de carga y descarga, lo que permite generar menos tiempo en dichas acciones, esto se realiza con el uso de “pallets”, los cuales, con ayuda de vehículos especiales se logra ahorrar tiempo en carga y descarga (De la Pava & Cardenas, 2016).

El problema de ruteo de vehículo Milk-run con ventana de tiempo se asemeja bastante a las situaciones reales, generando con ello un valor agregado para solucionarlo. La ventana de tiempo se traduce como el intervalo que el cliente dispone para ser visitado, ya sea recoger o dejar mercancía, ahora, si llega antes está obligado a esperar, no obstante, si es después deberá pagar una penalización (puede ser económica o con la inactividad normal en dicho punto). El problema de ruteo de vehículo busca disminuir costo de transporte como función objetivo y Milk-run maximizar la carga del camión.

Los problemas de ruteo de vehículos son de optimización combinatoria, el problema de ruteo de vehículo con ventana de tiempo está definido como un problema de gran complejidad computacional para su solución, Maldonado & Gómez Cruz, (2010) lo clasifica como NP-Hard.

De esta forma, para solucionarlo está: Colonia de hormiga (Monttez, 2017), Búsqueda Tabú de forma híbrida (Manisri, Mungwattana, & Janssens, 2011), Algoritmo Genético (C. Wang, Mu, Zhao, & Sutherland, 2015), Algoritmo *branch-price-and-cut* (BPC) (Pecin, Contardo, Desaulniers, & Uchoa, 2017) y el Algoritmo de ahorro (*C-W algorithm*) (Mei, Jingshuai, Teng, Xiuli, & Ting, 2017).

Dado que la Metaheurística Búsqueda Tabú ha reportado buenas soluciones al Problema de ruteo de vehículos con ventanas de tiempo (*Vehicle routing problema with time Windows*, VRPTW) (Azi, Gendreau, & Potvin, 2010), por lo tanto, se diseñará un Algoritmo de Búsqueda Tabú para dar solución al problema de ruteo de vehículos Milk-run con ventanas de tiempo.

Tabla 1.

Cumplimiento de objetivos del proyecto

| Objetivo                                                                                                                                                           | Cumplimiento |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|
| Realizar una revisión de literatura sobre el problema ruteo de vehículos Milk-run con ventanas de tiempo, así como los diferentes enfoques de solución propuestos. | Capítulo 4   |
| Definir el modelo matemático con los supuestos, variables y restricciones.                                                                                         | Capítulo 6   |
| Diseñar un Algoritmo de Búsqueda Tabú y programarlo en Matlab para dar solución al problema ruteo de vehículos Milk-run con ventanas de tiempo.                    | Capítulo 7   |
| Validar al algoritmo propuesto mediante instancias de la literatura.                                                                                               | Capítulo 8   |
| Elaborar un artículo de carácter publicable que contenga los resultados del proyecto de investigación.                                                             | Apéndice D   |

## 1. Planteamiento del problema

La distribución de mercancía en vehículos puede llegar a ser el 60% de los costos de suministro de la empresa (Ballou, 2004), por ende es importante manejar bien el ruteo de los vehículos. Esto juega un papel en la eficiencia operativa de la logística de abastecimiento, la cual depende de la red de transporte (directa, Milk-run o Cross dock) para lograr una programación de transporte idóneo. El transporte terrestre para Colombia en el 2018 fue del 6,30% en su PIB (Ministerio de transporte, 2019), del cual, el 37,1% está en el sector de operaciones logísticas, presentando acá el 52,1% de costos elevados e insuficiencia en los sistemas de información logística (Fajardo González, 2017), concluyendo en un buen plan de logística de aprovisionamiento como fundamental para disminuir costos.

La Logística Milk-run aplicado a logística de aprovisionamiento busca alcanzar el menor gasto posible en el transporte obteniendo la mejor ruta posible, claro, sujeta a recibirla en “pallets”, en días específicos según la necesidad de la empresa y tratar que el vehículo esté lo más lleno posible, y si a esto se le suma la restricción de tiempo que posee cada proveedor (nodo), se obtiene el problema de ruteo de vehículos Milk-run con ventana de tiempo, este es un problema combinatorio tipo NP Hard (Azi et al., 2010), obligando a buscar una solución entre todas las posibles, requiriendo tiempo computacional para su solución, siendo la Búsqueda Tabú uno que consume poco tiempo y llega a muy buenas soluciones (Gonzales 2011).

## 2. Justificación del proyecto

El ruteo de vehículo Milk-run con ventana de tiempo es aplicable a la industria de autopartes o cualquiera que requiera recepción de mercancía y sea direccionada a un centro de acopio, sin embargo, este problema es del tipo NP-Hard, necesitando el uso de algoritmos de optimización combinatoria para solucionarlo.

Por tal razón surge la necesidad de investigar acerca del Problema de ruteo de vehículo Milk-run con ventana de tiempo, pues es altamente aplicable a la logística de entrada, independientemente del objetivo planteado por la empresa u organización.

Durante la revisión de literatura se confirmó aportes importantes y varios estudios sobre el tema en cuestión, esto se debe a su importancia en el ámbito práctico y teórico; inicialmente se aplicaron métodos heurísticos, posteriormente métodos metaheurísticos y se menciona que no es práctico aplicar métodos exactos, pues el tiempo de cómputo es extenso, concluyendo que los métodos metaheurísticos son los más eficientes si se analiza tiempo de cómputo y calidad de la respuesta.

Por tal motivo, es de gran interés el estudio de este problema para el grupo de investigación OPALO y así contribuir a los avances que puedan surgir.

La presente investigación utiliza metaheurística Búsqueda Tabú para solucionar el ruteo de vehículo Milk-run con ventana de tiempo, siendo este un método muy eficiente y eficaz, pues en pocas iteraciones llega a una mejor solución a al algoritmo de Optimización por enjambre de Partículas (PSO). La investigación finaliza realizando un análisis comparativo entre el resultado

de Ma, & Wei, (2013) y la Búsqueda Tabú propuesta, demostrando ser mejor en tiempo computacional y calidad de la solución.

### 3. Objetivos

#### 3.1. Objetivo General

Diseñar un algoritmo de Búsqueda Tabú para la solución del problema de ruteo de vehículos Milk-run con ventanas de tiempo.

#### 3.2. Objetivos Específicos

- Realizar una revisión de literatura sobre el problema ruteo de vehículos Milk-run con ventanas de tiempo, así como los diferentes enfoques de solución propuestos.
- Definir el modelo matemático con los supuestos, variables y restricciones.
- Diseñar un Algoritmo de Búsqueda Tabú y programarlo en Matlab para dar solución al problema ruteo de vehículos Milk-run con ventanas de tiempo.
- Validar al algoritmo propuesto mediante instancias de la literatura.
- Elaborar un artículo de carácter publicable que contenga los resultados del proyecto de investigación.

#### 4. Revisión de literatura

##### 4.1. Análisis bibliométrico

Objetivo de la revisión de literatura es conocer la forma de modelar la logística Milk-run como problema aplicado al ruteo de vehículo.

➤ Protocolo de la revisión.

Se realizó la búsqueda en la base de datos Scopus, ISI Web of Science (colección principal).

Tabla 2

Ecuación de búsqueda utilizada en la base de datos Scopus, ISI Web of Science

| Base de datos  | Ecuación de Búsqueda Inicial                                                                                | N° artículos encontrados inicialmente | Artículos después de aplicar los criterios de inclusión y exclusión |
|----------------|-------------------------------------------------------------------------------------------------------------|---------------------------------------|---------------------------------------------------------------------|
| Scopus         | TITLE-ABS-KEY ( ( "milk run" OR "Mik-Run" ) AND ( logistic* OR transportation* ) AND ( vehicle OR rout* ) ) | 77                                    | 36                                                                  |
| Web of Science | TS= ( ( Milk-run OR Milk-run) AND ( logistic* OR transportation* ) AND ( vehicle OR rout* ) )               | 37                                    | 11                                                                  |
| <b>Total</b>   |                                                                                                             | <b>114</b>                            | <b>47</b>                                                           |

Como se muestra en la anterior tabla, la búsqueda se caracteriza con Milk run, transportes (logística) y vehículo (ruta), mostrando para la parte (a) 77 resultados y en la parte (b) 37, lo cual requiere un proceso de refinación, o sea, un protocolo de revisión, tal cual se ilustra en la Figura 1.

➤ Protocolo de revisión para la base de datos Scopus.

Criterios de inclusión:

- a) Documentos en idioma inglés y español.
- b) Documentos relacionados con reducción de costo, optimización, producción justo a tiempo, manufacturación automovilística, costo de transporte y problema de transporte.

Criterios de exclusión:

- a) Se excluyeron artículos de las siguientes áreas: agricultura, biología, ingeniería química, física y astronomía, ciencia de los materiales, ciencia de la tierra, química y ciencias sociales y ciencia de las decisiones.

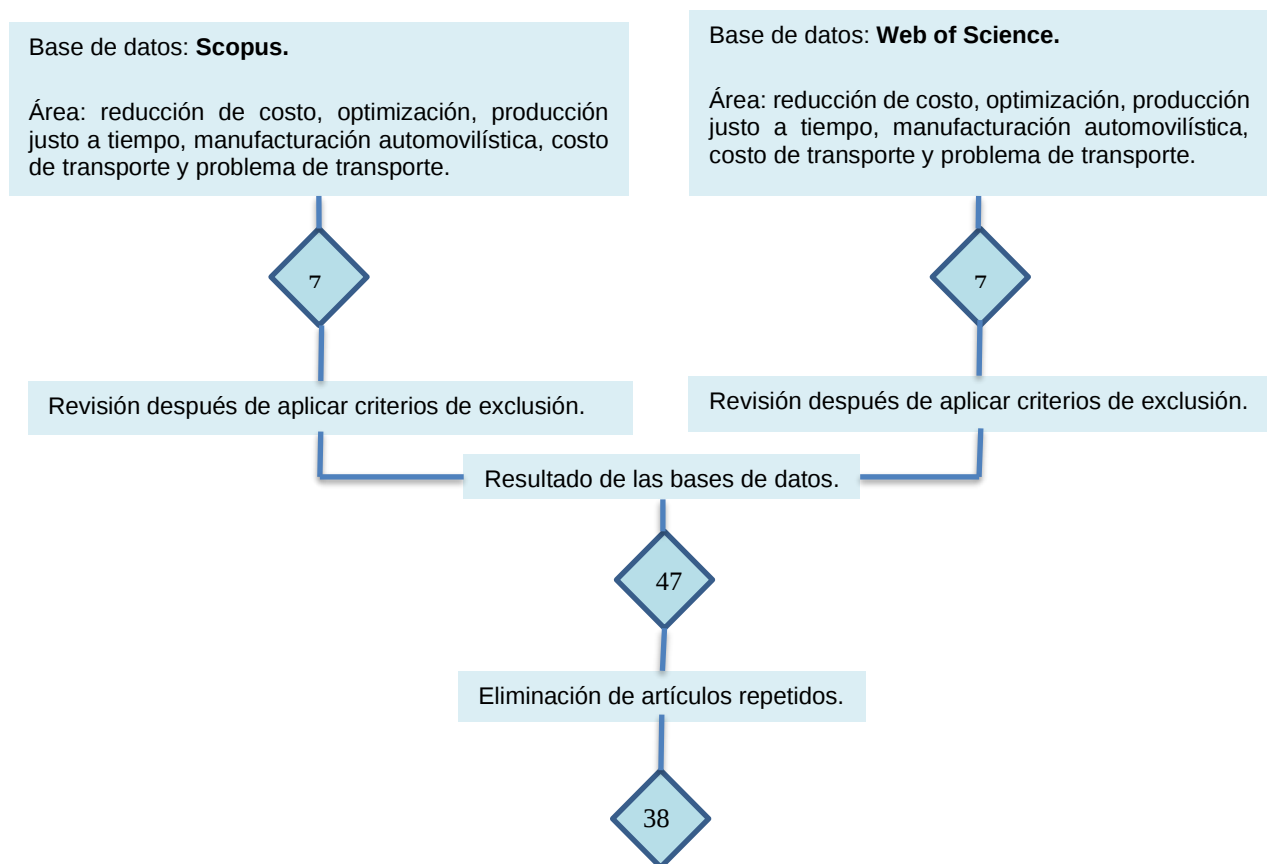


Figura 1. Protocolo de revisión.

- Protocolo de revisión para la base de datos web of science

Criterios de inclusión:

- a) Documentos en idioma inglés y español.
- b) Documentos relacionados con reducción de costo, optimización, producción justo a tiempo, manufacturación automovilística, costo de transporte y problema de transporte.

Criterios de exclusión:

- a) Se excluyeron artículos de las siguientes áreas: agronomía, economía, administración, ingeniería civil e ingeniería multidisciplinaria.
- Ejecución de la revisión

La búsqueda de las dos Base de datos tiene un total de 47 artículos, se realizó el análisis bibliométrico con el software VantagePoint, se combinaron las dos bases de datos, 11 documentos en común, quedando 38 artículos en total. La siguiente nube (Figura 2) de palabras muestra la frecuencia de palabras.

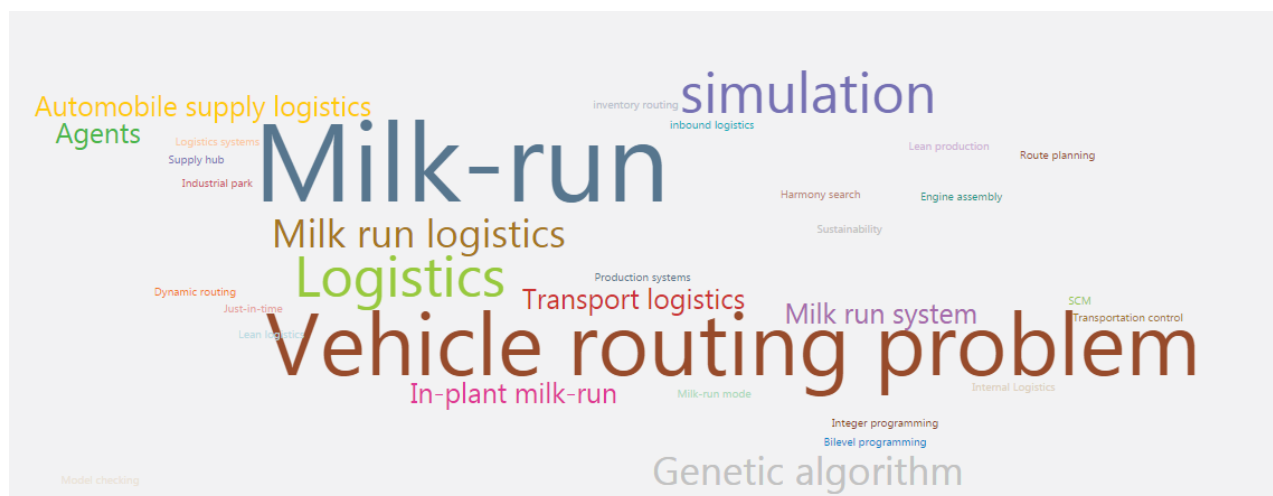


Figura 2. Nube de palabras de revisión bibliométrica.

Como se muestra en la anterior Figura, lo más importante es Milk-run y problema de ruteo de vehículo y logística Milk run, por lo cual ello va a ser lo tenido en cuenta en el presente trabajo. Luego se realiza la relación entre autores, identificando lo ilustrado en la Figura 3. Mostrando a Kilic, H, Toth, P y Costa, B como los máximos exponentes en el tema con 10, 9 y 9 artículos respectivamente.

En la Figura 4 se muestran los países con publicaciones, mostrando China, Alemania, Portugal y Turquía como los máximos exponentes.

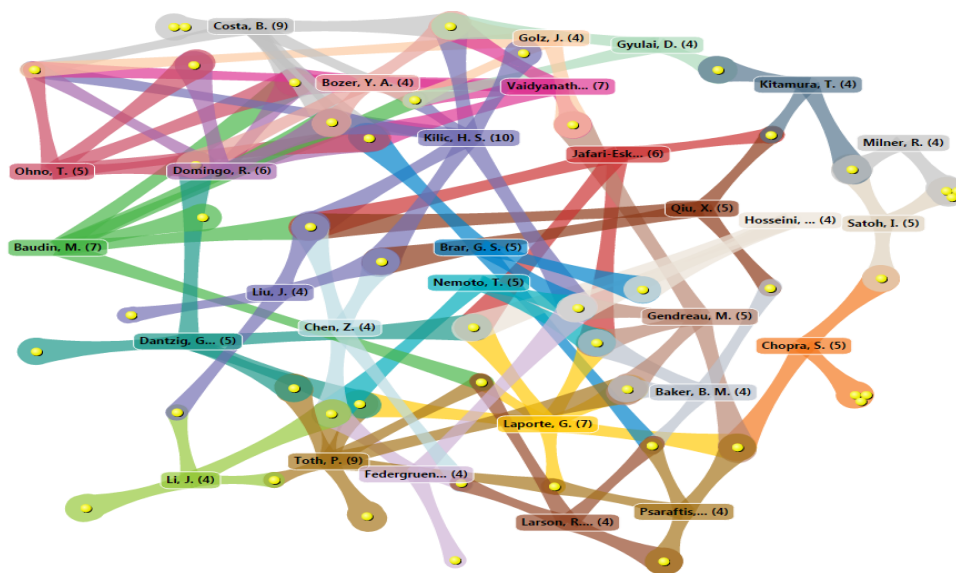


Figura 3 Cluster map de autores.

China es el máximo exponente en logística Milk-run ruteo de vehículo, lo cual no es de sorprender, ya que es una superpotencia, el cual se ha caracterizado por tener empresa de fabricación y tratamiento de productos y dada la extensión misma del país, es indispensable pensar en la logística de entrada y el ruteo de vehículos. Brasil es actualmente de los países de Latinoamérica que apunta a ser superpotencia, explicando también su aparición en la Figura 4.

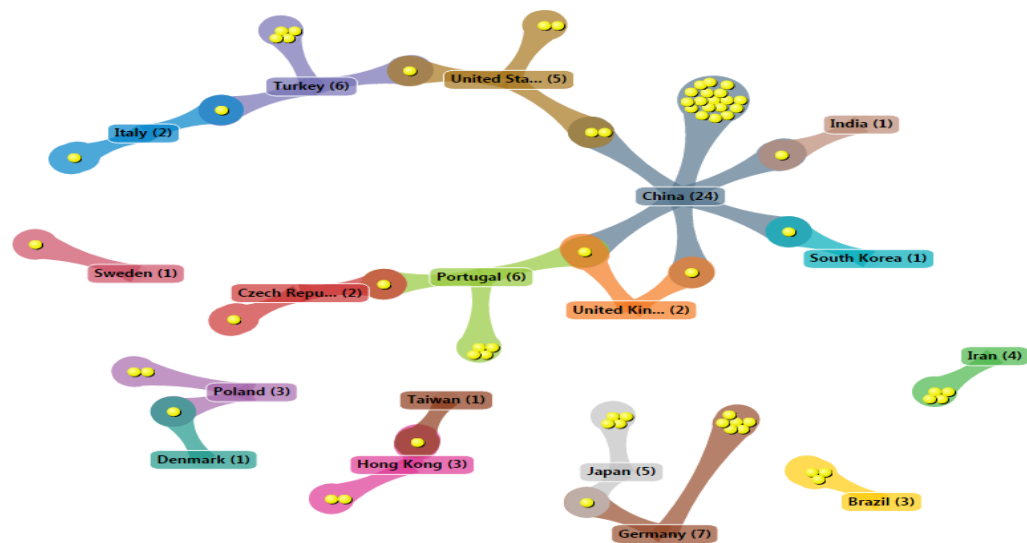


Figura 4. Análisis de los países por publicación según análisis bibliométrico.

Tabla 3.  
Afiliación de los principales países.

| País                      | Afiliación de autor                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>China [24]</b>         | Department of Industrial Engineering, Chung Yuan Christian University [7]<br>Department of Industrial Engineering and Management, St. John's and St. Mary's<br>Institute of Technology [3]<br>Department of Business Administration, Meiho Institute of Technology [2]<br>Department of Commerce Automation and Management, National Pingtung<br>Institute of Commerce [1]<br>Department of Distribution Management, National Taichung University of<br>Science and Technology [1]<br>Department of Industrial and Business Management, Chang Gung University,<br>[1] |
| <b>Alemania [7]</b>       | School of Management, University of Science and Technology of China, [2]<br>Department of Industrial and Manufacturing Systems Engineering, University<br>of Hong Kong, [1]<br>Department of Management Science, Sun Yat-Sen University, Guangzhou [1]<br>Department of Mathematics, Electronic Engineering Institute, PLA, Hefei [1]<br>Institute of Logistics and Supply Chain Management, Hefei University of<br>Technology [1]<br>Logistics Institute of Shandong University, Shandong University, Jinan,<br>Shandong Province, China [1]                         |
| <b>Portugal [6]</b>       | Department of Applied Mathematics, Vidyasagar University, [2]<br>Department of Management Science, Sun Yat-Sen University, Guangzhou [3]<br>Department of Mathematics, Electronic Engineering Institute, [1]                                                                                                                                                                                                                                                                                                                                                          |
| <b>Estados Unidos [5]</b> | Department of Mechanical and Industrial Engineering, Louisiana State<br>University, [5]                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

En la Tabla 3 se resume la afiliación de los artículos. Con la información del análisis bibliométrico se resalta la participación de Asia en el ruteo de vehículo Milk-run con ventana de tiempo, siendo 24 para china y 5 para Japón, luego aparece Portugal, Turquía y Estados Unidos, siendo estos países industrializados, mostrando un común denominador entre artículos publicados y la industrialización para el ruteo de vehículo, explicando ello también la aparición de Brasil, pues es el país latino con más participación industrial.

#### 4.2. Análisis preliminar de la revisión literaria

El problema de ruteo de vehículo con ventana de tiempo (*vehicle routing problem with time Windows*, VRPTW) es planteado por primera vez por Pullen & Webb, (1967), el cual busca la mejor ruta agregando la restricción de ventana de tiempo y es del tipo NP- Hard (Azi et al., 2010), y se parece al problema de ruteo de vehículo Milk-run con ventana de tiempo (*vehicle routing problema Milk-run with time Windows*, VRPMRTW) al tener ventana de tiempo y ser ambos del tipo del tipo NP- Hard (Sadjadi, Jafari, & Amini, 2009; L.-F. Wang, Liu, Huang, Wang, & Kong, 2013).

Históricamente hablando, fue en el año 1956 cuando se plantea el problema de ruteo de vehículos como generalidad, no obstante, se le conocía como el problema del agente viajero (Flood, 1956), posteriormente, en 1959 se describe el problema con el nombre que actualmente se maneja (problema de ruteo de vehículo) implementando una formulación matemática, dejando de ser un problema teórico para convertirse en numérico (Dantzig & Ramser, 1959) y fue hasta 1964 cuando se llegara a una solución, con el algoritmo de ahorros (Clarke & Wright, 1964), tres años más tardes, con Pullen & Webb, (1967) plantean el problema de ruteo de vehículos con ventana de tiempo, encontrando por otros autores distintas modificaciones como ventana de entrega con

fecha de entrega por parte de Swanson, (1986) y ventanas blandas de tiempo con Koskosidis, Powell, & Solomon, (1992). Lenstra & Kan, (1981) demostraron que el problema de ruteo de vehículo y sus variantes se encuentran en la clase de NP-Hard.

El ruteo de vehículo con ventana de tiempo solo busca la mejor ruta bajo ciertos parámetros, los cuales consolidaran la Logística Milk-run. El ruteo de vehículo Milk-run con ventana de tiempo busca adicionalmente reducir costos logísticos, disminuir los niveles de stock, estandarizar el volumen de recepción de materiales y aligerar el proceso de carga y descarga (Fracchia, 2011).

**4.2.1. Problema de ruteo de vehículo con ventana de tiempo.** Donati, Montemanni, Casagrande, Rizzoli, & Gambardella, (2008) solucionaron el problema de ruteo de vehículo con ventana de tiempo usando el Algoritmo de Colonia de Hormigas ((Ant Colony Optimization, ACO), toman restricciones en el tiempo basándose en información real de situaciones contaminantes, velocidad y tráfico vehicular, documentándola en un sistema llamado Cartesio.

El problema dispone de 1522 nodos y una ventana de tiempo de 30 minutos, también considera dos rutas diferentes, la primera son calles con velocidad vehicular constante y la otra depende del tiempo horario, dividiendo el tiempo de salida vehicular en 4 intervalos de 150 minutos cada uno, generando una solución para una masa muestral de 30 nodos, además muestra un error del 7,58% con vehículos a velocidad constante, no obstante, si la ruta depende del tiempo el error aumenta a 12%.

Manisri et al., (2011) utilizaron algoritmos de dos fases, primero la modificación inserción heurística hacia adelante (*Push-Forward Insertion Heuristic*, MPFIH) y luego la Búsqueda Tabú (*Tabú Search*, TS) para el problema de ruteo de vehículo con ventana de tiempo en 25 nodos,

tomando el (MPFIH) como algoritmo constructor inicial requerido, luego aplica (TS), disminuyendo el tiempo necesario para encontrar una solución adecuada.

Moccia, Cordeau, & Laporte, (2012) utilizaron la variante incremental Búsqueda Tabú (*Incremental Search Tabú*, IST), es similar a la Búsqueda Tabú tradicional, pero cambia la estructura del vecindario, o sea, en vez de cambiar un nodo entre rutas (vecindarios), permuta varios nodos simultáneamente, generando menos tiempo computacional, mostrando buenas soluciones de hasta 1198 nodos.

Kok, Hans, & Schutten, (2012) van un poco más allá con la solución al problema de ruteo de vehículos con ventana de tiempo, una heurística de programación dinámica estricta y el algoritmo Dijkstra modificado; este usa un ejemplo real en donde se tiene mucha penalización por llegadas tardes, logrando reducir hasta en un 70% la tardanza, aplicándolo para rutas con alta congestión vehicular entre nodos, para ello implementa 15, 50 y 100 nodos, logrando también reducir el número de vehículos necesarios.

Braekers, Caris, & Janssens, (2014) realizan una combinación híbrida con el algoritmo del método iterativo (*iterative method*, IM) y el recorrido determinista (*deterministic advance*, DA), formando el algoritmo (*iterative method- deterministic advance*, IM-DA). Se comparó entre el método iterativo y la Búsqueda Tabú y el recorrido determinista y Búsqueda Tabú, siendo estos últimos más exactos. En las distintas aplicaciones de la Búsqueda Tabú es indispensable un algoritmo constructor, si este dispone de hallar un óptimo local no tan alejado del objetivo, el tiempo computacional disminuye y es más probable mejorar la solución que si se realizara de otra manera.

C. Wang et al., (2015) realizaron una variante al problema de ruteo de vehículos con ventana de tiempo al agregar en cada nodo recogida y entrega, esto hace más extensa la ventana de tiempo y agrega en la formulación matemática el parámetro  $\alpha$ , el cual se añade en la ecuación de minimizar costos y representa la compensación de costos por recoger y entregar. Para solucionarlo usaron el “*Parallel Simulated Annealing*, (p-SA)” y comparándolo con un algoritmo genético (este último tiene por objetivo minimizar el número de vehículos), muestra buenos resultados para nodos inferiores a 50, soluciones aceptables para mayores de 50 e inferiores a 100.

Afifi, Dang, & Moukrim, (2016) agregaron al clásico problema de ruteo de vehículos con ventana de tiempo una sincronización de visita, esto obliga añadir un balance de carga en los camiones. La solución algorítmica denominada método de mejoramientos de conjuntos locales, muestra que la heurística Búsqueda local tiene una eficiencia comparada con métodos Metaheurísticos en la literatura, visualizando adecuación para la restricción mencionada (sincronización de visita).

Pecin, Contardo, Desaulniers, & Uchoa, (2017) solucionaron el problema de ruteo de vehículos con ventana de tiempo introduciendo dos mejoras al *branch-price-and-cut* (BPC), primero desarrollaron una forma más aguda para la memoria limitada, representándola como un subconjunto de arcos en vez de un subconjunto de nodos y segundo desarrollaron una familia de desigualdades, ello permite solucionar para más de 50 clientes o nodos (siendo el límite superior convencional para los métodos exactos), más precisamente hasta 200 puntos, mostrando buenas soluciones para encontrar la solución con alto número de nodos.

Bianchessi et al., (2019) resolvieron el clásico problema de ruteo de vehículos con ventana de tiempo con la suma de sus respectivas restricciones de entregas divididas y especificaciones del

cliente o nodo usando el diseño extendido del algoritmo “*branch-and-cut*”, mostrando que la división de entrega permite mayor grados de libertad, pues admite más opciones en las rutas, realizando las siguientes recomendaciones: (a) las entregas divididas dan buenos resultados, considerándola independiente del objetivo. (b) aunque un número de visitas mayor al clásico problema de ruteo de vehículo se cambie, esto puede repercutir económicamente al disminuir costo por penalización y reducción de rutas y (c) dividir las visitas también permite una mejor distribución de los bienes, logrando reducir pérdidas de tiempo también.

**4.2.2. Problema de ruteo de vehículo Milk-run con ventana de tiempo.** Sadjadi, Jafari, & Amini,(2008) propusieron un modelo matemático para el problema Milk-run en ruteo, este difiere del problema de ruteo de vehículos convencional desde la función objetivo: en el segundo caso busca disminuir costo de transporte habitualmente y Milk-run le añade costos relacionado con inventario, este modelo lo solucionan con un Algoritmo Genético.

L.-F. Wang, Liu, Huang, Wang, & Kong, (2013) muestran las ventajas de aplicar la Logística Milk-run a la logística de suministro y diferenciándolo con el clásico problema de ruteo de vehículo, pues la mercancía empacada de forma estandarizada disminuye el tiempo de carga y descarga, repercutiendo positivamente al momento de realizar una ruta, dicha modificación en la función objetivo busca maximizar la carga del vehículo.

Ma, & Wei, (2013) solucionaron el problema de ruteo de vehículos Milk-run con ventanas de tiempo mediante el método heurístico enjambre de partícula. Este problema en particular posee 15 nodos y mediante el ajuste de codificación de partículas se concluye el uso de 5 camiones. El método heurístico utilizado genera la respuesta final en 140 iteraciones.

Qiu, Xu, & Huang, (2014) utilizaron el sistema de ciclo de tiempo “Milk-run” para transporte de partes en horarios de entrega definidos, el objetivo acá es minimizar el costo de transporte, sin embargo, cuenta con más parámetros, como el ciclo de reposición de materia prima, costo fijado por despacho, tiempo de ciclo Milk-run, entre otras, mostrando resultados alentadores para el ruteo de vehículo de esta forma.

Hosseini, Shirazi, & Karimi, (2014) compararon el sistema de transporte de entrega directa, envío a través de Cross-dock (entrega indirecta) y Milk-run, con un algoritmo híbrido entre Búsqueda Armónica (*Harmony search*, HS) y Recocido Simulado (*simulated annealing*, SA) basado en heurística, cuyo objetivo es minimizar el costo de entrega.

Pei & Zhao, (2016) utilizaron el sistema “Milk-run” al ruteo de vehículos para partes de electrodoméstico, no obstante, ellos lo dividieron en dos *clústeres*, esto permite disminuir el tiempo computacional, dichos *clústeres* son conjunto de nodos que se encuentran muy cercanos entre sí y en donde un solo camión pueda pasar por todos los puntos sin que se sobrecargue en su capacidad; este se aplicó para un caso de estudio en donde hay 2 camiones (2 *clústeres*) y se comparó sin aplicar Milk-run, mostrando una disminución en 11,5% en ahorro de transporte.

Staab, Klenk, Galka, & Guenther, (2016) aplicaron el sistema “Milk-run” a una planta, o sea, en logística interna, la planta posee diez áreas de almacenamiento, hay siete trenes recolectores, dado la extensión de esto, se hace indispensable implementar un sistema que mejore el manejo de mercancía, por lo que Milk-run es buena opción, la conclusión después de aplicarlo es un menor tiempo en manejo, almacenamiento y disposición de mercancía, pues los trenes y la logística permite optimizar tiempos tanto de transporte como carga y descarga.

Bae, Evans, & Summers, (2016) aplicaron ruteo de vehículo “Milk-run” a una planta de autopartes localizada en Louisville, Kentucky, teniendo en cuenta las siguientes características: se considera tráfico, el material se capta de forma estándar, o sea, empacado volumétricamente y está dispuesto en pequeños contenedores (empaques), el objetivo es disminuir la distancia recorrida, aunque no tiene presente la ventana de tiempo y se usaron dos vehículos, mostró buenos resultados para la selección de rutas.

Campos et al., (2017) identificaron falencias en el movimiento de mercancía final para un almacén, por lo cual se utiliza el sistema Milk-run para mejorar el transporte de los productos, garantizando así una solución al proceso de parada y captación de mercancía, al menos de forma sistemática, generando menos pérdidas de tiempo y dinero, no obstante, el principal aporte es la aplicación para la logística interna, se simuló en el software ARENA, mostrando una mejora en el manejo de bienes, aun en logística interna.

Greenwood, Kluska, & Pawlewski, (2017) al igual que (Campos et al., 2017), implementaron el sistema Milk-run en logística interna, para ello los carros de cargas deben modificarse con la intención de manejar volumen variado según las cajas del empaque y la ruta las toma según una línea establecida en el suelo, tal cual se muestra en la Figura 5.



Figura 5. Carga y transporte en el sistema Milk-run en logística de entrada. Adaptado de Greenwood, A. G., Kluska, K., & Pawlewski, P. (2017). A Multi-level Framework for Simulating Mik-Run, In-plant Logistics Operations.

Este sistema les permite optimizar tiempo y distribuir de forma más eficientemente la mercancía, lo cual genera menor gasto.

Mei, Jingshuai, Teng, Xiuli, & Ting, (2017) solucionaron el problema de ruteo de vehículo Milk-run con ventana de tiempo mediante el Algoritmo de Ahorros de Clarke. & Wright minimizando la distancia recorrida y el tiempo de estadía en el punto o nodo. Entre los datos de entrada está el tiempo de carga, siendo esta la sección en donde entra Milk-run (aparte de visitar un punto una única vez).

Xiong et al., (2018) tiene muy presente la cadena de suministro al momento de generar un algoritmo en el ruteo Milk-run, pues involucra a los proveedores de primer y segundo nivel, dicho modelo lo denomina nuevo modelo de logística Milk-run (*new Milk-runlogistic model*), el objetivo es disminuir costo, el cual está en su función objetivo asociado al transporte de los proveedores de nivel uno y dos, entre sus restricciones se encuentra una distancia máxima para el recorrido del vehículo; para solucionarlo se emplea un algoritmo genético y se comparó con un modelo de ruteo Milk-run donde no se involucre proveedores de distintos niveles, encontrando un ahorro para 22 nodos de 1740 yuanes, o sea, 242 dólares. Hay que aclarar que el artículo hace un ejemplo en donde no hay más de 20 Km entre nodos, la demanda no supera las 1,9 toneladas, la capacidad máxima del camino es de 8 toneladas, no se supera los 50 km de recorrido por vehículo y son 4 vehículos.

Aragao et al., (2019) utilizaron Milk-run con el enfoque basado en el agente a la empresa de transporte y captación de mercancía *EOM Operation*, dicho enfoque permite un comportamiento dinámico para la cadena de suministro, la aplicación Milk-run se usa principalmente para captar la mercancía de los proveedores de forma estratégica, o sea, en componentes debidamente

empacados que permiten una fácil carga y descarga, generando consigo en la mayor rapidez en las paradas y por ende en el tiempo total.

Bocewicz, Nielsen, & Banaszak, (2019) para el problema de ruteo Milk-run, a parte de las consideraciones optadas por (Sadjadi et al., 2008), añade variables como número de trabajadores, final del tiempo de producción y transporte, lo cual complementa el modelo analítico.

## 5. Marco teórico

Para solucionar el problema en cuestión, es indispensable entenderlo desde un punto de vista matemático, para ello se empieza con optimización combinatoria y tipos de complejidad computacional, posterior a ello se entra en detalle con Milk-run y el problema de ruteo de vehículo con ventanas de tiempo, las soluciones que se le puede dar a este tipo de problemas, la solución seleccionada y el mecanismo de esta.

### 5.1. Optimización Combinatoria

Optimización combinatoria hace referencia a la solución de problemas en donde las variables de decisión está formada por números naturales enteros, con lo cual se trata de hallar el mejor ordenamiento de los números para encontrar la mejor solución según la función objetivo (Vélez & Montoya, 2007).

Esto se ha aplicado en distintos problema de economía como la planificación y gestión de las operaciones en el uso eficiente de los recursos, también se ha aplicado en problemas de optimización discreta como informática, gestión, telecomunicaciones, logística, etc. (Yepes Piqueras, 2008).

### 5.2. Tipos de complejidad Computacional

La complejidad computacional se puede clasificar según el tipo de problema computacional, modelo de cómputo o el recurso computacional que está consumiendo, siendo el último el más aplicado, dividiéndose en P y NP.

Los problemas P son aquellos en los que se puede encontrar una solución en un tiempo polinómico y los NP son aquellos en los que podemos comprobar, en un tiempo polinomial si la respuesta es correcta, como lo es el ruteo de vehículos (Maldonado & Gómez Cruz, 2010).

La complejidad algorítmica se rige principalmente por criterios de espacio (cuanta memoria se necesita para resolver un problema) y tiempo (cuantos pasos se requieren para resolverlo). Se han definido varias clases de problemas, destacándose los siguientes:

- Clase P: está formada por problemas de decisión que pueden ser resueltos mediante algunos algoritmos en un número de pasos acotado por un polinomio fijo en función del tamaño de la entrada.
- Clase NP: incluye a todos los problemas de decisión cuyas soluciones pueden ser verificadas en un tiempo polinómico dada la información correcta o lo que es lo mismo, pueden ser encontradas en tiempo polinómico haciendo uso de una máquina de Turing no determinista.
- Clase NP-completos: son los problemas NP más difíciles, debido a que es menos probable que pertenezcan a la clase P, debido a su dificultad por encontrar una forma de resolver el problema rápidamente (en un tiempo polinómico).
- Clase NP-hard: se caracterizan porque si se resuelve un problema en un tiempo polinómico permite resolver todos los problemas de la clase NP en un tiempo polinómico (Navarro, 2008).

### 5.3. Cadena de suministro

La cadena de suministro es: “El conjunto de empresas integradas por proveedores, fabricantes, distribuidores y vendedores (mayoristas o detallistas) coordinados eficientemente por medio de relaciones de colaboración en sus procesos clave para colocar los requerimientos de insumos o productos en cada eslabón de la cadena en el tiempo preciso al menor costo, buscando el mayor impacto en la cadena de valor de los integrantes con el propósito de satisfacer los requerimientos de los consumidores finales” (No, 2002), según lo representa la Figura 6.

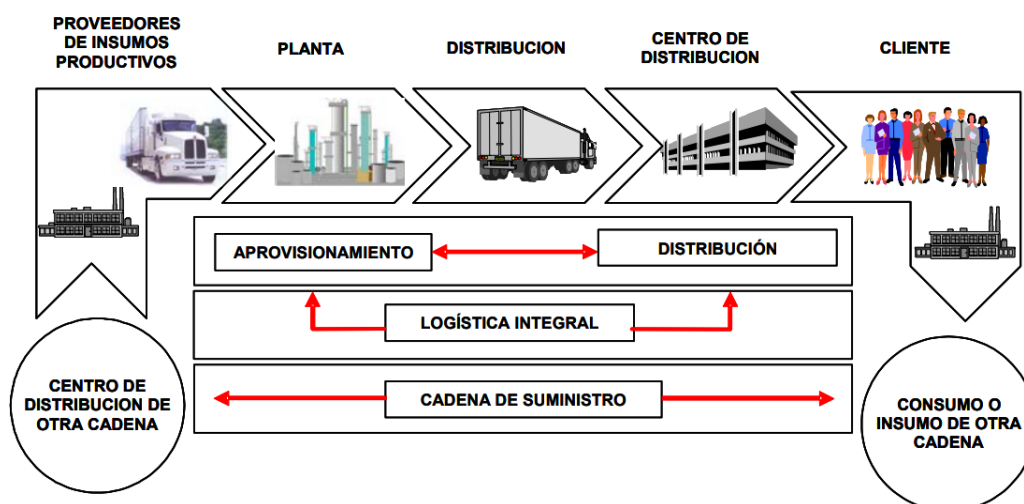


Figura 6. Cadena de suministro. Adaptado de No, P. (2002). *Marco conceptual de la cadena de suministro: Un nuevo enfoque logístico*

### 5.4. Logística

Según Riveros & Silva, (2004), la logística es la capacidad por parte de un ente para entregar los bienes o servicios a tiempo. De acuerdo a Mazo, Montoya, & Henao, (2014), “La logística es aquella parte del proceso de la cadena de suministro que planea, implementa y controla el flujo y almacenamiento de productos y servicios, y su información relacionada, desde el punto donde se originan hasta el punto donde se consumen, en forma eficiente y al menor costo posible, para

satisfacer los requerimientos de los clientes”; en otras palabras, es la buena administración de los productos o servicios (ello incluye información, dinero y equipos necesario para lograrlo) para satisfacer adecuadamente el cliente según la necesidad de este.

No se debe confundir logística con la Gestión de la Cadena de Suministro (*supply chain management*), pues la segunda es una metodología de trabajo integrada para gestionar los distintos flujos en los canales de distribución; proveedor, cliente y consumidor final, mientras la logística consiste en entrega o distribución del producto o servicio en el lugar, hora, cantidad, precio y calidad correcto de acuerdo con los requerimientos de este, o sea, la logística hace parte de la cadena de suministro (Riveros & Silva, 2004).

### **5.5. Logística de aprovisionamiento**

Como se muestra en la Figura 7, el flujo de los materiales sigue el siguiente orden: aprovisionamiento, fabricación y distribución. La logística de aprovisionamiento hace referencia al manejo o gestión de materia prima, piezas y demás elementos necesario para lograr la producción óptima de la empresa (Puente, García, Fernández, & Parreño, 2008), en este ciclo se centran las políticas de compras, todo lo relacionado a la gestión de transporte (manejo de la mercancía desde el proveedor hasta la central), diseño de los almacenes y la gestión de inventarios (Servera-Francés, 2010).

La logística Milk-run ignora las políticas de compra y gestión de inventario, resultando en lo relacionado a transporte y diseño de almacenamiento.



Figura 7. Procesos y flujos logísticos. Adaptado de Orrego, J. J. M. (2014). *logística de aprovisionamiento*. Ediciones Paraninfo, S.A

### 5.6. Logística Milk-run

Existen diferentes redes de transporte: envío directo, “*cross-docking*”, Milk-run y redes a medida (Du, Wang, & Lu, 2007). Si el envío consiste en “*full truckload*” (TL), es económico enviar directamente del vendedor al comprador. Sin embargo, cuando las órdenes son menos que un “*truckload*” (LTL), los otros enfoques se consideran. La red a medida, combina TL y LTL al permitir que los pedidos de gran volumen se envíen directamente de los proveedores a los clientes, y los pedidos de bajo volumen se consoliden

El envío directo es el más arcaico de los sistemas logísticos de recepción y entrega de mercancía y básicamente es traer la mercancía de un punto a otro punto. “*Cross-docking*” consiste en captar la mercancía de distintos sectores en un centro de acopio, luego se distribuye a su respectivo lugar, este sistema es muy aplicable para empresas como Servientrega, pues se debe combinar mercancía de recepción, imposibilitando la entrega con un solo camión. Por último, el sistema Milk -Run hace referencia a la visita de varios puntos antes de llegar al centro de recepción.

Como se muestra en la Figura 8 Milk-run es, en apariencia, el que menos recorrido hace, para ello, siendo Milk-run el que genera menor gasto, lo que demuestra su implementación adecuada para esta clase de problemas.

La red de transporte Milk-run consiste en compartir un vehículo en la máxima división de clientes y/o proveedores, en contraposición a lo exigido en la red de entrega directa. (Christopher, 2005).

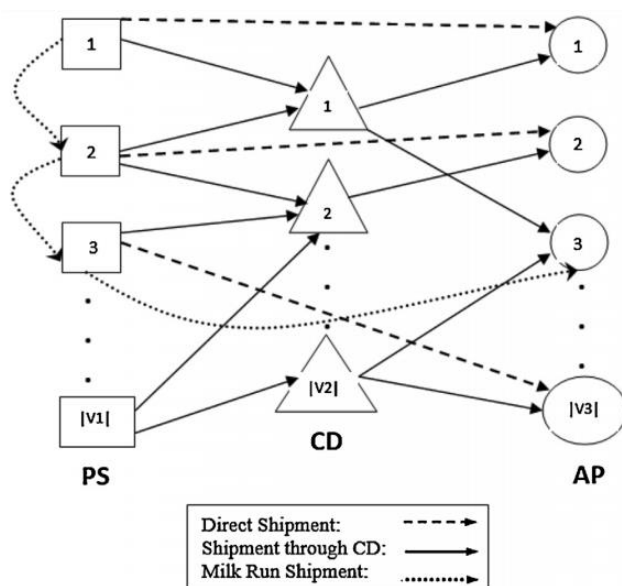


Figura 8. Diferencia entre envío directo (Direct Shipment), Cross-dock (Shipment through CD) y Milk-run para una red que comprende proveedor de partes (part suppliers, Adaptado de Hosseini, S. D., Shirazi, M. A., & Karimi, B. (2014). Cross-docking and Milk-run logistics in a consolidation network: A hybrid of harmony search and simulated annealing approach.

Milk-run se debe aplicar cuando hay más clientes que vehículos y se le denomina Logística Milk-run. El Logística Milk-run consiste en la recolección programada de materiales. Utiliza un único equipamiento de transporte para realizar las recolecciones o entrega de uno o más proveedores para luego entregar los materiales en el destino final, siempre con horarios preestablecidos (Fracchia, 2011). En Figura 9, los nodos grises no son pertenecientes a los nodos de la planta, también se puede interpretar de dos formas:

- Utilizó dos camiones de carga para repartir o recolectar mercancía en un mismo día.
- Es el mismo camión de carga, pero en días diferentes.

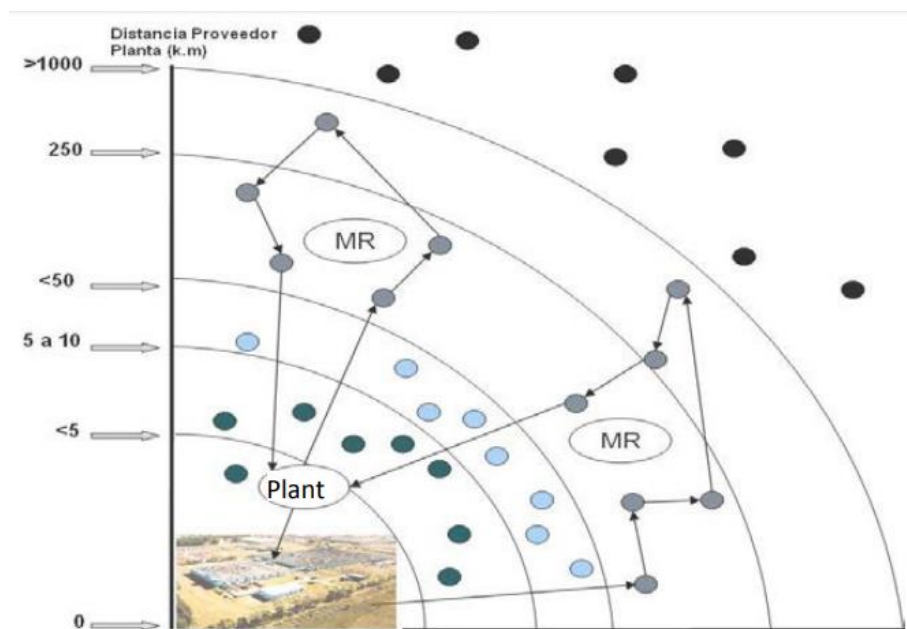


Figura 9. Sistema de Milk-run. Adaptado de Christopher, M. (2005). Logistics and Supply Chain Management: Creating Value-Adding Networks.

La logística Milk-run determina: (a) la ruta a adoptar, (b) el horario y (c) el tipo y número de partes de la mercancía que debe tomar cada camión, comprendiendo que todos deben retornar a la central en cuanto se vacíen o se llenen (según si entrega o recoge mercancía con cada cliente), otra ventaja es reducir los niveles de stock, ello se logra utilizando “*pallets*” y aumentando la frecuencia de despacho, agilizando la tarea de cargar y descargar, tanto en el camión como en la central, acortando el tiempo de estadía en la ventana de tiempo (Martínez, 2015).

El sistema Milk-run ha sido usado en varias industrias y las empresas de fabricación de automóviles en todo el mundo han sido (y son) los clientes más importantes de este sistema.

Para la logística Milk-run en la fabricación de autopartes se tiene muy en cuenta la logística de aprovisionamiento, también se denomina logística de entrada (Montoya & Andrés, 2011).

### 5.7. Problema de Ruteo de Vehículos (VRPTWs) vs Problema de Ruteo de Vehículos Milk-Run con ventanas de tiempo (MRVRPTWs)

El problema de ruteo de vehículo (*vehicle routing problem*, VRP) es usado para diseñar la mejor ruta para servir a unos clientes, dado un conjunto de restricciones. El principal objetivo es minimizar el costo de transporte.

En la revisión se encontraron diferentes funciones objetivo para el contexto del problema de ruteo de vehículos Milk-run por ejemplo L.-F. Wang, Liu, Huang, Wang, & Kong, (2013) y Sadjadi, Jafari, & Amini, (2008), en el modelo matemático para el ruteo de vehículo Milk-Run tiene como función objetivo maximizar la carga que puede cargar el camión, como se muestra en la ecuación (1) (maximizar el volumen del camión), el cual está restringido a la ecuación (2) y (3) (solo puede llegar una vez al punto o nodo que esté en la ruta), y la ecuación (4) (el volumen nunca superará el volumen límite del vehículo).

$$Max = \sum_{i=0}^n \sum_{j=0}^n \sum_{v=1}^m C_{ij} x_{ij}^v \quad (1)$$

$$\sum_{j=0}^n \sum_{v=1}^m ij x_{ij}^v = A; \quad \forall i, j = 0, 1, 2, \dots, n \quad (2)$$

$$A = (i, j) : i, j \in Cl, i \neq j; \quad (3)$$

$$\sum_{v=1}^n Y_{iv} \leq Q, \forall v = 1, 2, \dots, k, \quad \cap, \quad i = 0, 1, 2, \dots, n \quad (4)$$

$C_{ij}$  es el volumen asociado a la carga,  $X_v$  es la permisión del camión a continuar,  $A$  es el número de nodo,  $Q$  es la carga máxima.

En la Tabla 4 se muestra la diferencia entre el ruteo de vehículo con ventana de tiempo (*Vehicle Routing Problem with Time Windows*, VRPTWs ) y ruteo de vehículo Milk-run con ventana de tiempo (*Milk-run vehicle routing problem with time windows*, MRVRPTWs) con base a lo investigado en el análisis de antecedentes y marco teórico.

Tabla 4.

Análisis comparativo entre ruteo de vehículo y ruteo de vehículo Milk run.

| VRPTWs                                                                             | MRVRPTWs                                                                                                 |
|------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------|
| Busca la ruta más corta únicamente                                                 | Busca disminuir tiempo con la ruta más corta y mejorando el sistema de carga-descarga de la mercancía.   |
| En ventana de tiempo suele tardar más.                                             | Dado el sistema de carga-descarga es más eficiente, se evita con mayor facilidad la penalización.        |
| El tiempo total del camión, aunque corto es mayor que el ruteo de vehículo Mik-run | En comparación al ruteo de vehículo clásico, tiene menor tiempo el vehículo en carretera.                |
| Es un problema netamente matemático.                                               | Es un problema matemático y de logística, aplicado no solo en aprovisionamiento, sino en almacenamiento. |
| No optimiza carga en el camión.                                                    | Optimiza carga en el camión al buscar la mayor cantidad de mercancía.                                    |

## 5.8. Métodos de solución para el problema de ruteo de vehículos (VRP)

Dado el ruteo de vehículo es un problema NP-HARD (Monttez, 2017), se han implementado métodos exactos, heurísticas y Metaheurísticas.

**5.8.1. Métodos de solución exactos.** Se dividen en cuatro grandes grupos: (a) Técnicas de relajación, (b) Búsqueda directa de árbol, (c) Programación dinámica y (d) Programación lineal entera (Monttez, 2017).

- (a) Técnica de relajación: se encuentra relajación lagrangiana, descomposición de Dantzig-Wolfe y generación de Planos de Corte.
- (b) Búsqueda directa de árbol: acá se encuentra el algoritmo de Ramificación y Corte (Branch & Cut), algoritmo de Búsqueda de Árbol, algoritmo de Ramificación y Acotamiento (*Branch & Bound*) y asignación de cota inferior.
- (c) Programación lineal entera: Conjunto de particiones y generaciones de columnas, formulación de flujo de vehículos de dos índices, formulación de flujo de vehículos de tres índices y el Programa Lineal Entero-Mixto.

**5.8.2. Métodos de solución heurísticos.** Estos se dividen en métodos constructivos, métodos de dos fases y heurísticas de mejora (Monttez, 2017).

- (a) Métodos constructivos: Algoritmo de Ahorros (Algoritmo *Clark y Wright*), mejoras del Algoritmo de Ahorros, algoritmo de Ahorros basado en acoplamientos, heurística de construcción paralela-adaptativa, heurísticas de inserción secuencial, heurísticas de inserción en paralelo y heurística de inserción guiada.
- (b) Métodos de dos fases: Algoritmo de barrido, algoritmo basado en asignación generalizada, heurística basada en localización, algoritmo de ramificación y acotamiento truncados, métodos de asignar primero y rutear después, métodos de rutear primero y asignar después y algoritmo de los pétalos.

- (c) Heurísticas de mejora: El operador de intercambio, el algoritmo de Lin-Kernighan, el operador Or-opt, GENI y GENIUS, algoritmos de transferencias cíclicas y operadores de Van Breedam.

**5.8.3. Métodos de solución Metaheurísticos.** En estos se encuentran la Búsqueda de Vecindario Variable, Algoritmos Genéticos (Genetic Algorithm, GA), Recocido Simulado (Simulated Annealing, SA), Redes Neuronales (Neural Networks, NN), Búsqueda Tabú (Tabú Search, TS), y Algoritmo de Colonia de Hormigas (Ant Colony Optimization Algorithm, ACO) (Monttez, 2017).

El algoritmo de ahorros es del tipo heurístico, se basa en el principio de pasar desde una red de entrega directa a una red de distribución más eficaz, en donde el objetivo es cubrir la mayor cantidad de puntos posibles por un mismo vehículo (De la Pava & Cardenas, 2016). El algoritmo de colonia de hormigas es del tipo Metaheurística y se basa en la disposición que estos invertebrados presentan para adoptar una ruta con respecto a otra, ocasionado por las feromonas que ellos segregan, dicha ruta es la más corta, y bajo ese principio se puede obtener una ruta con menor recorrido (Monttez, 2017).

En su tesis doctoral, Gonzales (2011), analiza distintos métodos Metaheurístico para problemas de optimización combinatoria, algoritmos genéticos y de búsquedas locales, reportando una muy buena solución con Búsqueda Tabú, superando en disminución de tiempo computacional al algoritmo genético, sin embargo, también muestra que combinarlos puede mejorar la búsqueda de óptimos no locales, ya que puede solucionar aún con gran número de vecinos (nodos), sin que baje la calidad de la solución.

**5.8.1. Métodos de solución para el problema de ruteo de vehículo Milk-run con ventana de tiempo (MRVRPTWs).** Los principales métodos de solución encontrados para el problema de ruteo de vehículo Milk-run con ventana de tiempo (MRVRPTWs) encontrados fueron: Algoritmo de Colonia de Hormigas (Monttez, 2017), Búsqueda Tabú de forma híbrida (Manisri et al., 2011), Algoritmo Genético (C. Wang et al., 2015), el Algoritmo “branch-price-and-cut” (BPC) (Pecin, Contardo, Desaulniers, & Uchoa, 2017) y el Algoritmo de Ahorro (C-W algorithm) (Mei, Jingshuai, Teng, Xiuli, & Ting, 2017).

## 5.9. Metaheurística Búsqueda Tabú

A Glover, (1986) se le atribuye el concepto metodológico de Búsqueda Tabú, cuya finalidad era de salir de los procesos heurísticos, escapando de los óptimos locales, encontrando soluciones más exactas. Este algoritmo posee memoria y la implementa mediante estructuras simples, dirigiendo una búsqueda más objetiva, pues la tiene en cuenta para no repetir caminos, o sea, hacerlos Tabú. En ese sentido, la Búsqueda Tabú es un buscador inteligente, haciendo uso de la Lista Tabú.

**5.9.1. Conceptos importantes.** En el siguiente listado se encuentra los conceptos importantes:

- Entorno o vecindario: Conjunto de posibles soluciones asociadas a una solución.
- Lista Tabú: Es una lista donde se registran aquellas soluciones que no deben ser elegidas
- Criterio de aspiración: Excepción para usar un movimiento de la Lista Tabú, si, y solo si dicho vecindario es la mejor solución obtenida hasta el momento.
- Función objetivo: Es la ecuación que se busca optimizar.

**5.9.2. Características generales.** El termino Tabú hace referencia a movimientos no permitidos, o sea, aquellos que se sabe no generan mejor resultado que el óptimo local, ellos son guardados en la memoria misma del algoritmo. Su estructura básica o elementos de la Búsqueda Tabú son el denominado vecindario, definiendo un movimiento como una operación de intercambio de dos vecinos (integrantes del vecindario) que genera una solución, la cual, al ser evaluada, se mantendrá este como nueva respuesta (óptimo local), siendo dicho intercambio un Tabú, los intercambios generados se guardan en la memoria y se denominan Lista Tabú (Número Tabú) o simplemente Tabú. En la Búsqueda Tabú se encuentra también el criterio de aspiración, el cual, al realizar un cambio en el vecindario y este llega a estar en la Lista Tabú se omite, si, y solo si la solución es mejor que la obtenida hasta el momento.

**5.9.3. Componentes de la Metaheurística Búsqueda Tabú .** La Metaheurística Búsqueda Tabú se ilustra en la Figura 10, y como se muestra es necesario suministrarle una solución inicial, siendo esta la “solución actual” que aparece en la parte derecha de los movimientos luego se generan  $n$  movimientos.

Cada uno es evaluado, eligiendo el mejor, se analiza si está en la Lista Tabú, llegado el caso no está se procede a repetir el proceso, ahora, si está en la Lista Tabú se revisa el criterio de aspiración y si se encuentra, se toma dicho movimiento como solución actual, ahora, si no se encuentra se genera nuevos movimientos de vecinos.

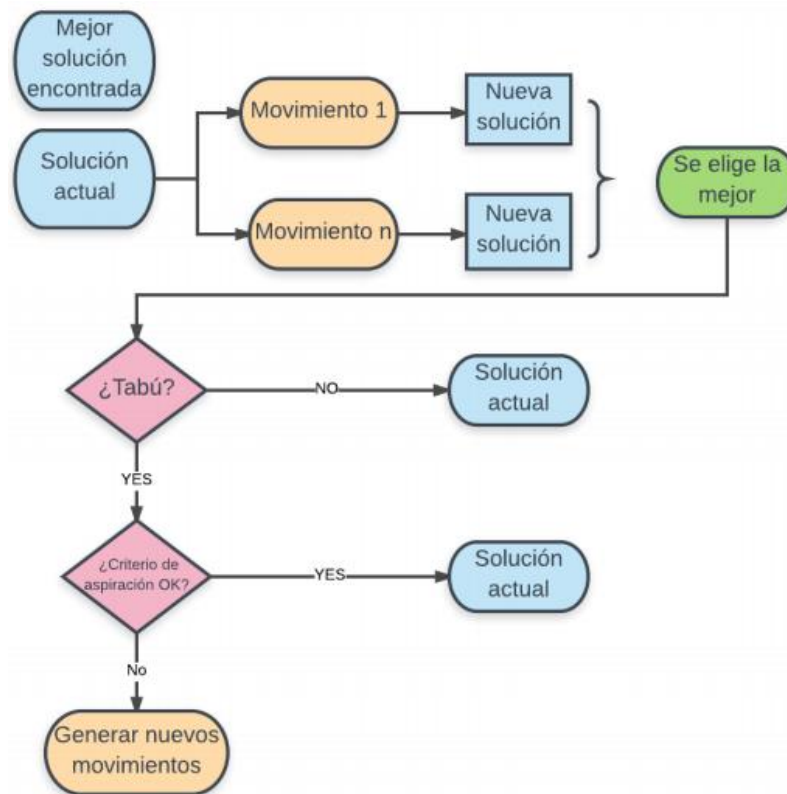


Figura 10. Funcionamiento de la Metaheurística Búsqueda Tabú. Adaptado de Bodas López, R. (2017). La metaheurística de Búsqueda Tabú aplicada al problema de Enrutamiento de Vehículos.

La razón computacional del éxito para la Metaheurística búsqueda Tabú (Gonzales 2011), es impedir que se repita malas soluciones, en otras palabras, la Lista Tabú, pues ello genera menos tiempo computacional para encontrar otras soluciones (dado tiene menos combinaciones posibles), permitiendo salir de óptimos locales (soluciones de baja calidad), esto, teóricamente hablando, hace que la Búsqueda Tabú sea un algoritmo de optimización combinatoria para aplicado al ruteo de vehículos (Bodas López, 2017: Gonzales 2011).

## 6. Problema Ruteo de vehículo Milk-run con ventana de tiempo (MRVRPTWs)

Se realizó una revisión de los modelos propuestos en la literatura para representar el problema y el que más se ajusta es el propuesto por Ma, & Wei, (2013).

### 6.1. Descripción del problema

El problema de Ruteo de vehículo Milk-run con ventana de tiempo es una extensión del problema del ruteo de vehículo con ventana de tiempo. Este busca la mejor ruta para recibir mercancía de proveedores, agregando que se optimice la recepción y despacho de dicha mercancía, pues se usa “pallets” para que la estadía en los proveedores sea lo menor posible, adicionalmente optimiza el espacio de los camiones, pues se cargan lo máximo según permita la ruta seleccionada.

### 6.2. Formulación del problema

El siguiente modelo es propuesto por Ma, & Wei, (2013). Antes de entrar en detalle al problema matemático, es necesario aclarar ciertos términos:

$$G = (Cl, A); \quad (5)$$

$$Cl = \{cl_0, cl_1, cl_2, \dots, cl_n\}; \quad (6)$$

$$A = (i, j) : i, j \in Cl, i \neq j; \quad (7)$$

$$C = (C_{ij}); \quad (8)$$

$$d_i \in \mathbb{Z}^+; \quad (9)$$

$$k; \quad (10)$$

$$Q = \epsilon \mathbb{Z}^+; \quad (11)$$

$$Y = Y_{ij}; \quad (12)$$

Sea (5) una grafo no dirigido, se tiene los nodos (6), entendiendo que  $cl_0$  es el nodo depósito, el conjunto de aristas que unen a todos los proveedores en (7), el costo de movilización desde el punto  $i$  hasta el punto  $j$  se representa con  $C$ , o sea, la ecuación (8). La demanda en (9) para el nodo  $i$ , (10) representa el número de vehículos o flota, (11) la capacidad de los vehículos, (12) representa la carga que lleva el vehículo del punto  $i$  al  $j$ ,  $Y_{iv}$  (cantidad interceptada por el vehículo  $v$  al cliente  $i$ ) y  $b_i^v$  es la hora de llegada del vehículo  $v$  al cliente  $i$ . Función objetivo

$$\text{Min } z = \sum_{i=0}^n \sum_{j=0}^n \sum_{k=1}^m C_{ij} x_{ijk} + M1 \sum_{i=1}^n \text{Max}(ET_i - S_i, 0) + M2 \sum_{i=1}^n \text{Max}(S_i - LT_i, 0) \quad (13)$$

El problema de ruteo de vehículos Milk-run con ventana de tiempo busca minimizar el valor de la ecuación (13) el costo relacionado con el transporte y tiempo de espera por llegar fuera de la ventana de tiempo, es decir, el segundo y tercer término son los costos de penalización por llegar antes y después de la ventana de tiempo respectivamente.

### 6.2.1. índices

- $i$ : nodo inicial
- $j$ : nodo final

- $k$ : vehículo

### 6.2.2. Variables

- $x_{ijk} \begin{cases} 1, \text{ si el vehículo } k \text{ viaja del nodo } i \text{ al nodo } j \\ 0, \text{ lo contrario} \end{cases}$
- $y_{ik} \begin{cases} 1, \text{ si el vehículo } k \text{ recoge en el punto } i \\ 0, \text{ lo contrario} \end{cases}$

### 6.2.3. Parámetros

- $C$ : costo de transporte de ir desde el nodo  $i$  al nodo  $j$ .
- $M1$ : Costo por llegar antes de la ventana de tiempo.
- $M2$ : Costo por llegar después de la ventana de tiempo
- $Q$ : Carga máxima del camión
- $q_i$ : Carga del nodo  $i$ .
- $LT_i$ : tiempo máximo de llegada al nodo  $i$ .
- $S_i$ : Tiempo de llegada al nodo  $i$
- $ET_i$ : tiempo más temprano para llegar al nodo  $i$ .
- $m$ : representa el número de vehículo.
- $t_{ij}$ : es tiempo de viaje del nodo  $i$  al nodo  $j$ .
- $T_i$ : indica el tiempo de carga de mercancía en el nodo  $i$
- $n$ : es el número del nodo.

#### 6.2.4. Restricciones

$$\sum_i q_i y_{ki} \leq Q, k = 1, 2, \dots, m \quad (14)$$

$$\sum_{k=1}^m y_{ki} = 1, i = 1, 2, \dots, n \quad (15)$$

$$\sum_{i=0}^n x_{ki} = y_{ki}, j = 0, 1, 2, \dots, n, k = 1, 2, \dots, m \quad (16)$$

$$\sum_{j=0}^n x_{ijk} = y_{ki}, i, j = 0, 1, 2, \dots, n, k = 1, 2, \dots, m \quad (17)$$

$$x_{ijk} = 0 \cap 1, i, j = 0, 1, 2, \dots, n, k = 1, 2, \dots, m \quad (18)$$

$$y_{ki} = 0 \cap 1, i, j = 0, 1, 2, \dots, n, k = 1, 2, \dots, m \quad (19)$$

$$s_i + T_i - t_{ij} - K(1 - x_{ijk}) \leq s_j, i, j = 0, 1, 2, \dots, n, k = 1, 2, \dots, m \quad (20)$$

$$X = \{x_{ijk} \in S\} \quad (21)$$

La ecuación (14) garantiza que el camión no exceda el peso, la ecuación (15) indica que la carga solo puede ser tomada por un camión, la ecuación (16) y ecuación (17) Indica que el vehículo solo puede visitar y partir una sola vez en cada nodo, la ecuación (18) y la ecuación (19) muestra el rango de variables, la ecuación (20) muestra que el camión k no puede salirse de la ventana de

tiempo y la ecuación (21) indica que la ruta debe estar conectada, o sea, tener continuidad (Cordeau, Desaulniers, Desrosiers, Solomon, & Soumis, 2002) y (L.-F. Wang et al., 2013).

Para implementar satisfactoriamente el problema de ruteo de vehículos Milk-Run con ventana de tiempo es indispensable entender que la mercancía lleva “pallets” y con ayuda de un solo trabajador en un vehículo especial para ello, esto permite disminuir el tiempo de estadía en el punto.

## 7. Algoritmo propuesto al MRVRPTWs

El algoritmo empieza hallando el vector inicial (VI), luego determina la función objetivo y finalmente, generación de vecindario y lista Tabú.

### 7.1. Vector inicial

En la generación del vector, se debe tener en cuenta que todo vector empieza en el nodo cero y termina es ese mismo, por lo cual, la longitud del vector a analizar es de 19, pues posee 5 vehículos y 15 nodos.

El vector inicial se genera a partir de una modificación al método del barrido, pues este empieza en la fábrica (nodo cero), luego busca el más próximo a este, analizando la carga, posteriormente se dirige al nodo más cercano de este último con su respectivo análisis de carga y se devuelve a la fábrica cuando no pueda captar más carga (Daza, Montoya & Narducci 2009). En la Figura 11 está la ruta 0-12-4-2-13-0-15-14-5-0-10-11-7-6-0-1-9-3-0-8-0. La modificación busca que la ruta esté lo más cerca a la hora de llegada de cada nodo, por lo cual no se elige el más cercano, sino el que genere menos tiempo de pérdida.

Los parámetros de entrada del vector inicial son: Matriz distancias (DIS), parte inferior de la ventana de tiempo (HLL), parte superior de la ventana de tiempo (HS), vector carga o carga en cada nodo (CARGA), velocidad promedio de los camiones (VP) y carga máxima del camión (CG).

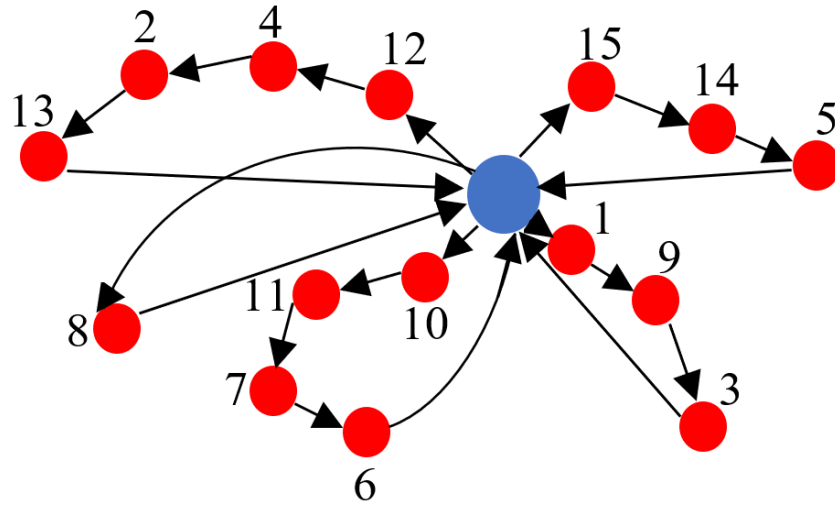


Figura 11. Algoritmo de barrido

La matriz de distancia se muestra en la Tabla 5, contiene las distancias entre cada par de nodos, nodos desde 0 hasta n nodos (proveedores), donde (:) significa que toma todos los elementos de la fila a la cual está asociada, para (1,:) significa que todos los elementos de la primera fila de su respectiva matriz. Se comienza determinado el proveedor más cercano al nodo cero (N1), siendo este el primero que visita el camión. Una vez establecido el primer proveedor a visitar, se procede a determinar el vector T\_D (ecuación (22)), donde T es el tiempo que lleva el camión hasta la hora de salida del nodo anterior.

$$T\_D = \left( \frac{DIS(N1,:)}{VP} \right) + T \quad (22)$$

Tabla 5.  
Matriz distancia

|     | 0          | 1               | 2               | ...               | n               |
|-----|------------|-----------------|-----------------|-------------------|-----------------|
| 0   | <b>0</b>   | <b>DIS(0,1)</b> | <b>DIS(0,2)</b> | <b>DIS(0,...)</b> | <b>DIS(0,n)</b> |
| 1   | DIS(1,0)   | 0               | DIS(1,2)        | DIS(1,...)        | DIS(1,n)        |
| 2   | DIS(2,0)   | DIS(2,1)        | 0               | DIS(2,...)        | DIS(2,n)        |
| ... | DIS(...,0) | DIS(...,1)      | DIS(...,2)      | 0                 | DIS(...,n)      |
| n   | DIS(n,0)   | DIS(n,1)        | DIS(n,2)        | DIS(n,...)        | 0               |

Tabla 6.  
Ventana de tiempo

| NODO | 0        | 1             | 2             | ...             | n             |
|------|----------|---------------|---------------|-----------------|---------------|
| HORA | <b>0</b> | <b>HLL(1)</b> | <b>HLL(2)</b> | <b>HLL(...)</b> | <b>HLL(n)</b> |
| HORA | 0        | HS(1)         | HS(2)         | HS(...)         | HS(n)         |

Tabla 7.  
Carga a recoger en cada nodo

| NODO  | 0 | 1        | 2        | ...        | n        |
|-------|---|----------|----------|------------|----------|
| CARGA | 0 | CARGA(1) | CARGA(2) | CARGA(...) | CARGA(n) |

Luego de determinar el  $T\_D$  de la ecuación (22), se procede a restarlo con el vector HLL, o sea, la segunda fila de la Tabla 6 (fila resaltada), generando un nuevo vector llamado  $T\_1$ , luego se halla la posición del escalar más cercano a cero en el vector  $T\_1$ , obteniendo el proveedor  $N2$ , siendo este el nodo siguiente a visitar. Los nodos visitados van siendo eliminados en la matriz distancia, ventana de tiempo y vector carga para evitar que se llegue a repetir en la fase del vector inicial.

Una vez determinado el segundo proveedor, se analiza la carga que está llevando el camión en ( $N2$ ), y si esta es menor a la carga máxima permitida ( $CG$ ) se procede a repetir el proceso mencionado en el párrafo anterior.

Cuando  $CG$  llega a ser mayor en el nodo  $Nn$ , el vector inicial ( $VI$ ) se escribe de la siguiente manera:  $0-NI-N2-***-N_{(n-1)}-0-Nn$ . Una vez determinado  $Nn$ , se procede a repetir todo el proceso mencionado hasta acá, el cual finaliza cuando se han eliminado todos los nodos, anexando un cero al final para que el ultimo camión pueda retornar a la fábrica. El pseudocódigo correspondiente a la generación del vector inicial se muestra en la Figura 12.

**Algoritmo:** Vector inicial

**Entrada:** Matriz distancias (DIS), parte inferior de la ventana de tiempo (HLL), parte superior de la ventana de tiempo (HS), vector carga o carga en cada nodo (CARGA) y carga máxima del camión (CG).

**Salida:** Vector inicial (VI)

**Etapas 1.**  $i$ =Número de nodos sin incluir el centro de acopio;

$J=0$ ;

$VI=0$ ;

**Etapas 2.** Repetir hasta  $i$ ;

{

**Etapas 3.** Se determina el nodo más cercano al nodo  $N_{i-1}$ , obteniendo el nodo  $N_i$ ;

**Etapas 4.** Guardar carga como  $J=J+CARGA(N_i)$ ;

**Etapas 5.**  $VI=[VI, N_i]$

**Etapas 6.** Si  $CARGA \geq CG$ , entonces  
 $VI=[0, N_1, \dots, 0, N_i]$

}

**Etapas 7.** Parar y reportar  $VI=[VI, 0]$

Figura 12. Pseudocódigo del vector inicial.

## 7.2. Función objetivo

La función objetivo es la ecuación (13), la cual busca minimizar costo relacionado con transporte y tiempo de espera por llegar fuera de la ventana de tiempo. Está en función del costo de transporte ( $C\_TRANS$ ) y el tiempo que se encuentra por fuera de la ventana de tiempo ( $C\_EARLY-M1$ ) costo por llegar antes de la ventana de tiempo y ( $C\_LATER-M2$ ) costo por llegar después de la ventana de tiempo, todo esto aplicado a la ruta obtenida en la fase anterior (vector inicial,  $VI$ ). En las secciones siguientes se describirán los costos de la función objetivo.

**7.2.1. Costo total asociado a distancia total recorrida.** Primero se empieza determinando la distancia total recorrida, la cual se muestra en la Figura 13.

Dado el  $VI$ , se halla  $DIS(1)+DIS(2)\dots DIS(n)$ , donde 1 es  $(0, N_1)$ , 2 es  $(N_1, N_2)$ , ...,  $n$  es  $(N_n, 0)$ .

En  $DIS(x,y)$ ,  $DIS$  es la matriz distancia y  $(x,y)$  es la posición de la matriz. En la

Tabla 5 se halla la distancia. El costo se determina multiplicando el costo de transporte por la distancia total ( $C\_TRANS$  por  $DIS(1)+DIS(2)+\dots+DIS(n)$ ).

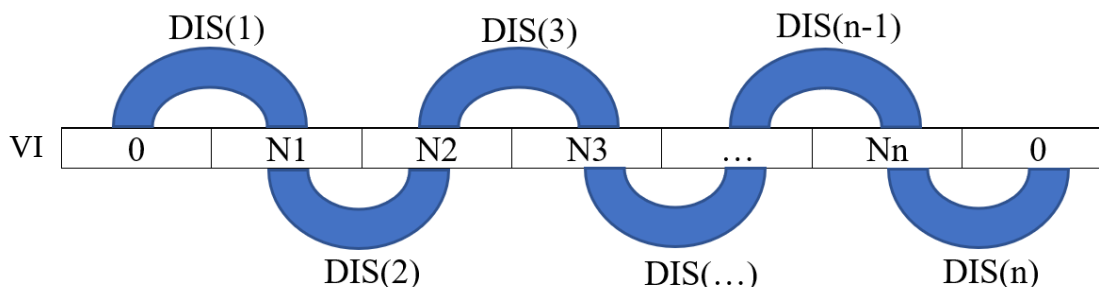


Figura 13. Determinación de la distancia total recorrida.

**7.2.2. Costo total asociado por llegar fuera de la ventana de tiempo .** En cada nodo, el camión tarda 0.25 horas, este es el tiempo de carga ( $C$ ).

Se comienza determinando la distancia acumulada  $DIS\_T=DIS(1)+C+DIS(2)+C$ , luego se analiza en el nodo  $N2$  si la suma anterior está dentro de la ventana de tiempo, si se encuentra ahí, no hay costo asociado por llegar a deshora, no obstante, si esta fuera de la ventana de tiempo, se analiza si está por debajo o por encima de la ventana de tiempo, si está en el primer caso se guarda como  $VT\_EARLY$ , si es el segundo caso, se guarda como  $VT\_LATER$ , se repite el proceso para el nodo  $N3$ , comprendiendo que la distancia acumulada general se expresa de la siguiente manera:  $DIS\_T=DIS(1)+DIS(2)+\dots+DIS(n)+(n*C)$ , donde  $n$  es el nodo que se está analizando sin contar la fábrica (nodo 0).

El en nodo  $N3$ , se vuelve a analizar si esta por dentro o fuera de la ventana de tiempo, si está antes de la venta de tiempo, se suma con el valor anterior  $VT\_EARLY$ , guardando este nuevo escalar como  $VT\_EARLY$ , si está después de la venta de tiempo, se suma con el valor anterior

VT\_LATER, guardando este nuevo escalar como VT\_LATER. Una vez recorrido el VI, se obtiene VT\_EARLY y VT\_LATER y se aplica la ecuación (13), en pseudocódigo se muestra en la Figura 14.

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p><b>Algoritmo:</b> Costo.</p> <p><b>Entrada:</b> Matriz distancias (DIS), parte inferior de la ventana de tiempo (HLL), parte superior de la ventana de tiempo (HS), velocidad promedio (VP), costo asociado al transporte (M), costo asociado por llegar antes de la ventana de tiempo (M1) y costo asociado por llegar después de la ventana de tiempo (M2).</p> <p><b>Salida:</b> COSTO.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <p><b>Etapas:</b></p> <p><b>Etapas 1.</b> <math>i = \text{diámetro}(\text{VI}) - 2</math>;</p> <p style="padding-left: 40px;"><math>J = 0</math>;</p> <p style="padding-left: 40px;"><math>H = 0</math>;</p> <p style="padding-left: 40px;"><math>H1 = 0</math>;</p> <p style="padding-left: 40px;"><math>H2 = 0</math>;</p> <p><b>Etapas 2.</b> Repetir hasta <math>i</math>;</p> <p>{</p> <p style="padding-left: 40px;"><b>Etapas 3.</b> Determinar <math>J = J + \text{DIS}_{(i,i-1)}</math>;</p> <p style="padding-left: 40px;"><b>Etapas 4.</b> Hallar <math>H = H + (J/VP)</math>;</p> <p style="padding-left: 40px;"><b>Etapas 5.</b> Si <math>H &lt; \text{HLL}_{(i)}</math>, entonces <math>H1 = H1 + H</math>;</p> <p style="padding-left: 80px;">Else if Si <math>H &lt; \text{HS}_{(i)}</math>, entonces <math>H2 = H2 + H</math>;</p> <p>}</p> <p><b>Etapas 7.</b> Parar y reportar <math>\text{COSTO} = (M * J) + (H1 * M1) + (H2 * M2)</math>;</p> |

Figura 14. Algoritmo costo.

La razón por la cual se empieza desde el nodo N2 y no N1 es que para este último (N1) no se llega por fuera de la ventana de tiempo, sino en el valor inferior de la ventana de tiempo (HLL), pues es absurdo que en el primer nodo se llegue tarde matemáticamente hablando, explicando así  $\text{DIS}(1) = \text{HLL}(N1)$ .

### 7.3. Generación de vecindario

En el vector no se modifica el primer y último nodo. Los movimientos se generan por intercambio de un nodo con otro, o sea, el número de intercambio es una combinatoria dado por la ecuación (23).

$$\binom{m}{n} = \frac{m!}{n!(m-n)!} \quad (23)$$

En donde  $m$  es el número de elementos del vector a analizar y  $n$  es el número de nodos intercambiados en una iteración, siendo para el presente caso 2, los intercambios se muestran, a modo de ejemplo en la Figura 15.

Estos intercambios generan nuevos vectores (Figura 15), los cuales se guardarán en una matriz denominada VI\_T con su respectivo costo asociado (vector C\_ALL), el costo asociado se determina según proceso de la página 55. La Tabla 8 muestra la forma como se asocia C\_ALL y VI\_T.

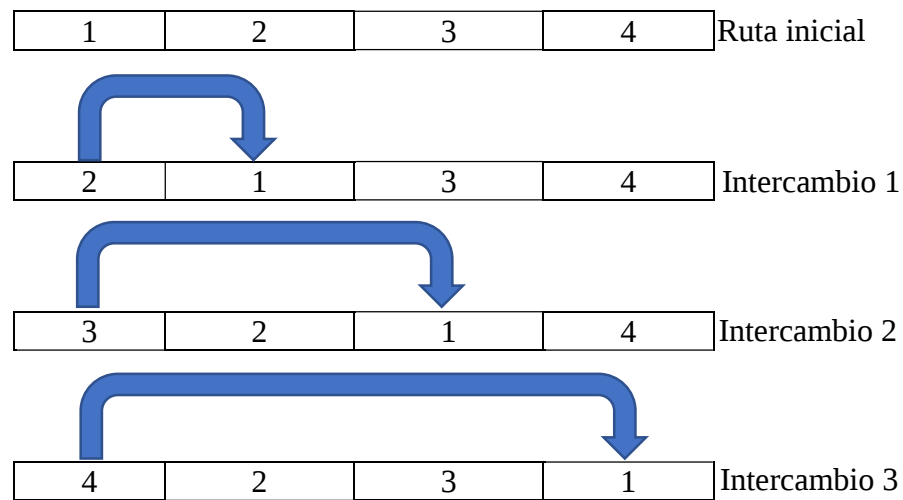


Figura 15. Intercambio de nodos para el algoritmo metaheurística búsqueda Tabú

Sin embargo, no todas las rutas son permisibles dentro de la restricción de carga, por lo cual, una vez determinada la Tabla 8, se procede a analizar cada ruta, y aquellas que no cumplan la restricción de la carga máxima, se transforma en 2 veces el valor máximo del C\_ALL inicial en el punto C\_ALL(i), garantizando así que no se pueda seleccionar como la solución de la iteración.

Tabla 8.

Matriz VI\_T y Vector C\_ALL

| Vector C_ALL |          |                 |          |     |                |
|--------------|----------|-----------------|----------|-----|----------------|
| C_ALL(1)     | C_ALL(2) | <b>C_ALL(3)</b> | C_ALL(4) | ... | C_ALL( $m_2$ ) |
| Matriz VI_T  |          |                 |          |     |                |
| N1           | N1       | <b>N1</b>       | N1       | ... | N1             |
| N2           | N3       | <b>N4</b>       | N5       | ... | N2             |
| N3           | N2       | <b>N3</b>       | N3       | ... | N3             |
| N4           | N4       | <b>N2</b>       | N4       | ... | N4             |
| N5           | N5       | <b>N5</b>       | N2       | ... | N5             |
| N6           | N6       | <b>N6</b>       | N6       | ... | N6             |
| (...)        | (...)    | <b>(...)</b>    | (...)    | ... | (...)          |
| N(n-2)       | N(n-2)   | <b>N(n-2)</b>   | N(n-2)   | ... | N(n-1)         |
| N(n-1)       | N(n-1)   | <b>N(n-1)</b>   | N(n-1)   | ... | N(n-2)         |

Después se selecciona y localiza el menor elemento (m) en el vector C\_ALL sin incluir al primer elemento (pues está asociado al vector inicial y no a una nueva ruta), dicha localización permite encontrar su respectiva ruta VI\_T(:,m), a modo de ejemplo, si el menor es C\_ALL(3), el nuevo VI es Matriz(:,3).

#### 7.4. Lista Tabú

La Lista Tabú impide que se repita soluciones que permitirían salir de un óptimo local. Después de obtener la ruta que tenga el menor costo asociado, se guarda dicho vector VI\_T(:,m) en la Lista Tabú, la forma de adjuntarlo se muestra en la Figura 16, este vector obtenido se transforma en VI para la nueva iteración, evitando, de esta manera, se repita la respuesta anterior.

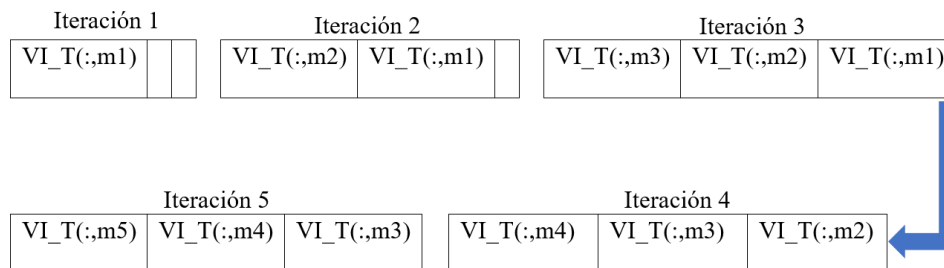


Figura 16. Forma de guardar vectores en la Lista Tabú.

Como se muestra en la anterior Figura, la lista Tabú es de 3 (número Tabú de 3), sin embargo, la presente investigación lo sube a 3 según se explicará en el capítulo 9. Los vectores que están en la Lista Tabú quedan prohibidos, sin embargo, para saber si el nuevo vector determinado está en la Lista Tabú, se procede a restar VI con cada vector de la Lista Tabú, si, y solo si el resultado es un vector de ceros, se puede asegurar que es el mismo, convirtiendo su costo asociado en 2 veces el valor máximo del C\_ALL actual, garantizando así que no se pueda seleccionar ningún vector de la Lista Tabú, posterior a esto se procede a determinar el costo y localización del menor elemento en el vector C\_ALL resultante (nuevo m) sin incluir al primer elemento.

Dicha localización permite encontrar su respectiva ruta VI\_T(:,(nuevo m)), siendo este el nuevo VI, este proceso se repite el número de iteraciones que el usuario desee, ya que este es un parámetro de entrada. Cada ruta se va almacenando en una matriz llamada RUTA y el costo asociado en un vector llamado GRAF, similar a la Tabla 8. Cuando termina las iteraciones, se localiza el escalar más pequeño en COSTO3 y su vector asociado en la matriz RUTA, siendo esta la mejor solución. El pseudocódigo se muestra en la Figura 21.

**Algoritmo:** Búsqueda Tabú

**Entrada:** Matriz distancias (DIS), parte inferior de la ventana de tiempo (HLL), parte superior de la ventana de tiempo (HS), velocidad promedio (VP), costo asociado al transporte (M), costo asociado por llegar antes de la ventana de tiempo (M1) y costo asociado por llegar después de la ventana de tiempo (M2), carga en cada nodo (CARGA), carga máxima del camión (CG), vector inicial (VI) y costo (COSTO).

**Salida:** RUTA y CostoFinal.

**Etapas 1.**  $i$ =Número de iteraciones que el usuario quiera;

$m$ =Diámetro de VI;

$n$ =Número Tabú;

VIT=VI;

VIT1=VI;

GRAF=COSTO;

COSTO1=COSTO;

$\max = 2 * \text{COSTO}$ ;

**Etapas 2.** Repetir hasta  $i$ ;

{

**Etapas 3.** Repetir hasta  $\binom{m}{2}$ ;

{

**Etapas 4.** Intercambiar elementos del vector VI y guardarlo como VI1;

**Etapas 5.** Aplicar el Algoritmo Costo a VI1 y guardarlo como COSTO2;

**Etapas 6.** Agregar VI1 a ListaTabú;

**Etapas 7.** Repetir hasta  $n$ ;

{

**Etapas 8.** Si  $(VI1_{(m)} - \text{ListaTabú}_{(n)}) = [0, 0, \dots, 0]$ , entonces  $\text{COSTO2} = \max$ ;

}

**Etapas 9.** Repetir hasta  $(m)$

{

**Etapas 10.** Si  $VI1_{(m)} = 0$ , entonces  $CG1 = 0$ ;

**Etapas 11.**  $CG1 = CG1 + CARGA_{(m)}$ ;

**Etapas 12.** Si  $CG1 > CG$ , entonces  $\text{COSTO2} = \max$ ;

}

**Etapas 13.**  $VIT = [VIT; VI1]$ ;

**Etapas 14.**  $\text{COSTO1} = [\text{COSTO1}, \text{COSTO2}]$ ;

**Etapas 15.**  $A = \min(\text{COSTO1})$ ;

**Etapas 16.**  $j$  es la posición de  $A$  en  $\text{COSTO1}$ ;

}

**Etapas 17.** Parar y reportar  $VI = VIT(j)$  y  $\text{COSTO1} = A$  ;

**Etapas 18.**  $VIT1 = [VIT1; VI]$  y  $GRAF = [GRAF, \text{COSTO1}]$ ;

}

**Etapas 19.**  $B = \min(\text{COSTO3})$ ;

**Etapas 20.**  $k$  es la posición de  $B$  en  $\text{COSTO3}$ ;

**Etapas 21.** Parar y reportar  $RUTA = VIT1(k, :)$  y  $\text{CostoFinal} = B$

Figura 17. Búsqueda Tabú.

## 8. Validación del algoritmo

Para analizar el comportamiento del Algoritmo Búsqueda Tabú en términos de tiempo computacional en el problema de ruteo Milk Run se realizó un diseño de experimentos  $2^k$ . El algoritmo se desarrolló en Matlab® versión 2020 y el diseño factorial se ejecutó en el software estadístico Minitab versión 2020. La instancia utilizada fue la propuesta por Ma, & Wei, (2013), con 15 proveedores, la carga máxima de cada camión es de  $30m^3$ , velocidad promedio de 40 km/h, costo de transporte de 15 yuanes/hora, costo por llegar antes de la ventana de tiempo de 12 yuanes/hora y costo por llegar después de la ventana de tiempo de 13 yuanes/hora. El diseño experimental es  $2^5$ , o sea, con 5 réplicas, los factores son (Lista Tabú) y B (Número de iteraciones), el nivel bajo para A es de 3 y su nivel alto es de 4, para B el nivel alto es de 100 y su nivel bajo es de 50. En este apartado se va a analizar que factor (A, B o AB) inciden mayormente en el tiempo de cómputo. Para ello se llevó a cabo 5 réplicas según se muestra en la Tabla 9, obteniendo el gráfico de Pareto (Figura 18) con un intervalo de confianza es del 95%.

Tabla 9.

Experimentos para el análisis factorial  $2^k$

| A | B   | T. computo_2 | T. computo_2 | T. computo_3 | T. computo_3 | T. computo_3 |
|---|-----|--------------|--------------|--------------|--------------|--------------|
| 3 | 50  | 5,246        | 6,582        | 4,782        | 5,76         | 5,85         |
| 4 | 50  | 5,6399       | 6,962        | 5,1934       | 6,715        | 6,014        |
| 3 | 100 | 5,77         | 6,47         | 5,75         | 5,95         | 6,152        |
| 4 | 100 | 6,18         | 6,66         | 6,19         | 6,25         | 6,564        |

### 8.1. Análisis de Varianza

A continuación, se presenta el Análisis de Varianza para el tiempo computacional en la instancia propuesta por Ma, & Wei, (2013).

Tabla 10.

Análisis de varianza para el diseño experimental

| Fuente                           | GL | SC Ajust. | MC Ajust. | Valor F | Valor p |
|----------------------------------|----|-----------|-----------|---------|---------|
| Modelo                           | 3  | 1,34728   | 0,44909   | 1,57    | 0,235   |
| Lineal                           | 2  | 1,33203   | 0,66601   | 2,33    | 0,129   |
| Lista Tabú                       | 1  | 0,82268   | 0,82268   | 2,88    | 0,109   |
| Número de iteraciones            | 1  | 0,50935   | 0,50935   | 1,79    | 0,200   |
| Interacciones de 2 términos      | 1  | 0,01525   | 0,01525   | 0,05    | 0,820   |
| Lista Tabú*Número de iteraciones | 1  | 0,01525   | 0,01525   | 0,05    | 0,820   |
| Error                            | 16 | 4,56419   | 0,28526   |         |         |
| Total                            | 19 | 5,91146   |           |         |         |

En el Diagrama de Pareto (Ver Figura 18) se muestra que ninguna de las variables es significativa con respecto al tiempo de cómputo, sin embargo, el número de iteraciones tiene mayor incidencia, no obstante, el número Tabú solo vario una unidad y las iteraciones variaron 50, pero analizando el gráfico de efectos principales (ver Figura 19 ) se observa que el número Tabú posee mayor pendiente, lo que se traduce, que si los factores dispusieran de la misma magnitud (igual cantidad numérica), sería el número Tabú más importante para el tiempo de cómputo.

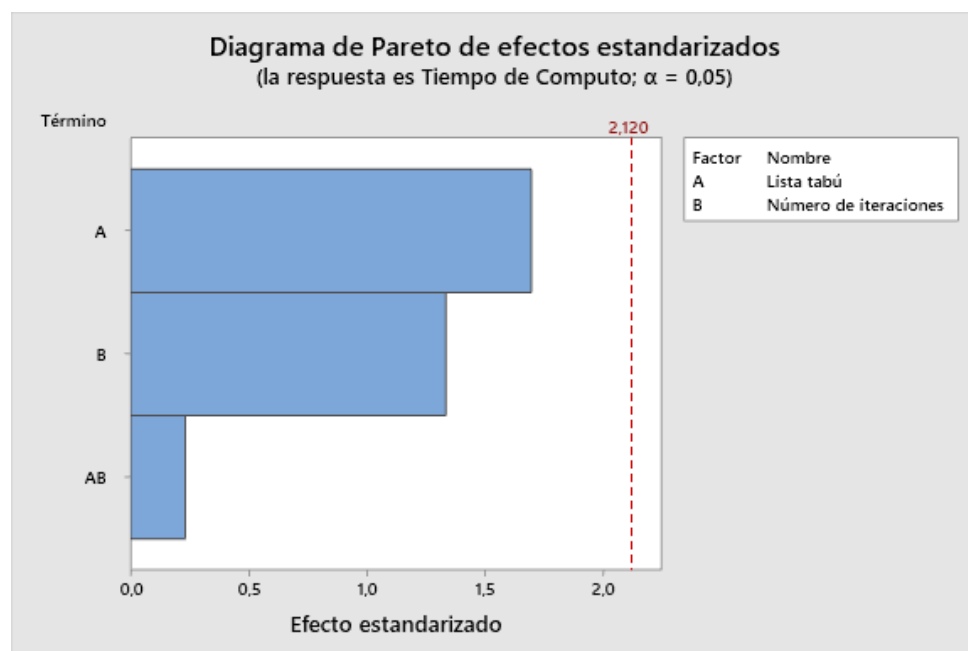


Figura 18. Gráfico de Pareto,

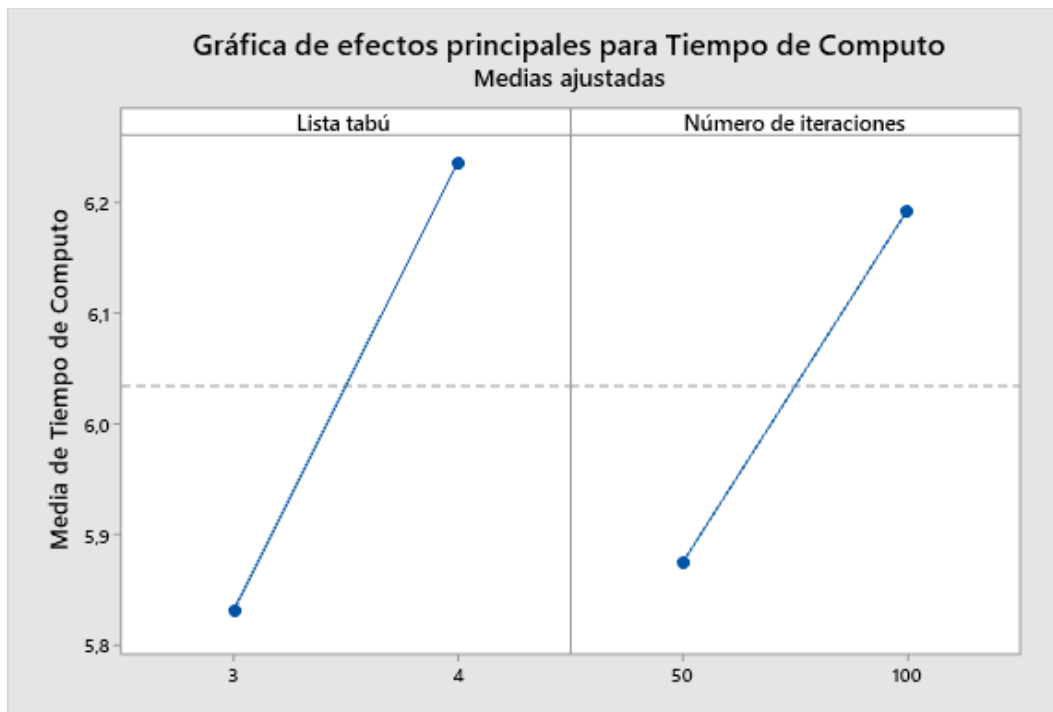


Figura 19. Grafica de efectos principales para tiempo de cómputo,

## 8.2. Validación de metaheurística Búsqueda Tabú con Lista Tabú de 3

Los parámetros de entrada como matriz distancia, ventana de tiempo y carga en cada nodo se muestra en el Apéndice A, la carga máxima de cada camión es de  $30\text{m}^3$ , velocidad promedio de 40 km/h, costo de transporte de 15 yuanes/hora, costo por llegar antes de la ventana de tiempo de 12 yuanes/hora y costo por llegar después de la ventana de tiempo de 13 yuanes/hora. Para realizar el proceso se procede a realizar 200 iteraciones como lo realizan Ma, & Wei, (2013), el resultado se muestra en la Figura 20, El vector inicial (VI) es 0-12-4-2-13-0-15-14-5-0-10-11-7-6-0-1-9-3-0-8-0 con un costo de 9346,9 yuanes y la mejor ruta es 0-5-6-13-2-0-4-15-14-0-10-7-1-11-0-12-9-3-0-8-0 con un costo de 6825 yuanes, el tiempo de cómputo es de 6,7 segundos. La respuesta obtenida hasta acá es 43 yuanes mayor que la obtenida por Ma, & Wei, (2013), sin embargo, como

se muestra en la Figura 20, a partir de la iteración 20 se llega a un ciclo, esto debido al número de restricciones de intercambios propios de este método (número Tabú de 3), surgiendo la necesidad de aumentar el número Tabú y mirar como impacta en la respuesta, siendo este la modificación.

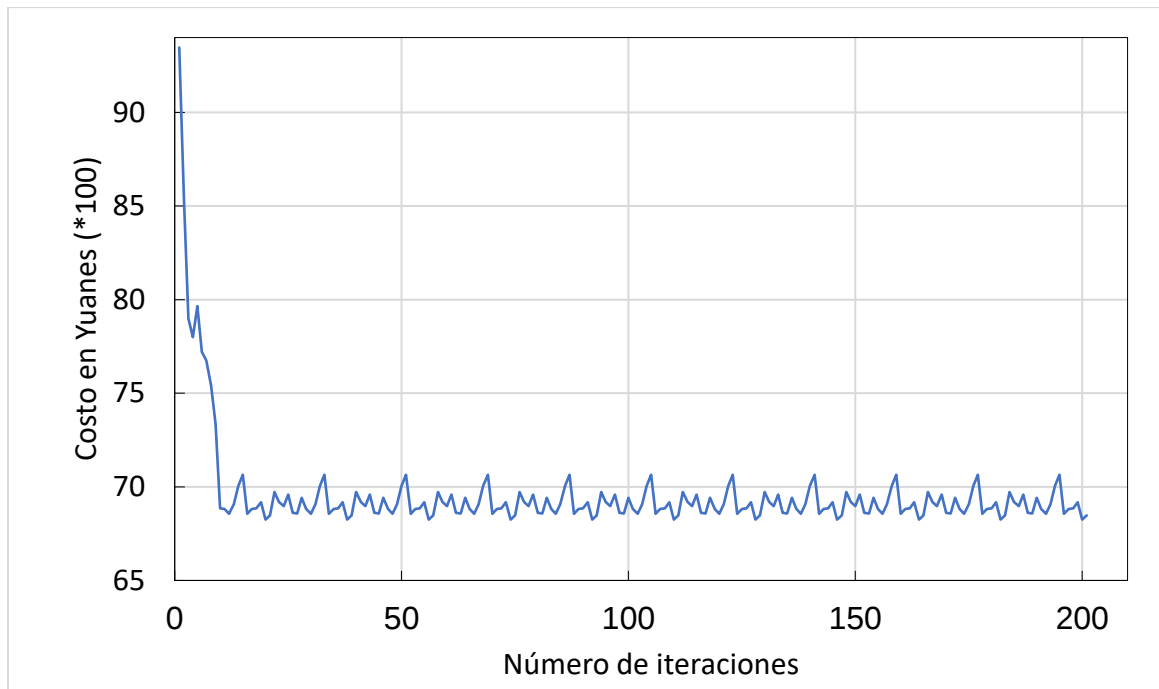


Figura 20. Proceso iterativo para Lista Tabú de 3

### 8.3. Validación de metaheurística Búsqueda Tabú modificado (Lista Tabú de 4)

La modificación es aplicable al número Tabú, para ello se deja dicha restricción como variable de entrada, obteniendo las respuestas en la Tabla 11.

Tabla 11.  
Resultados al modificar metaheurística Búsqueda Tabú

| Número Tabú | Ruta                                            | Costo (yuanes) | Tiempo (seg) |
|-------------|-------------------------------------------------|----------------|--------------|
| 4           | 0-4-2-13-6-0-5-14-15-0-10-7-1-11-0-12-9-3-0-8-0 | 6710,3         | 8,29         |
| 5           | 0-4-2-13-6-0-5-14-15-0-10-7-1-11-0-12-9-3-0-8-0 | 6710,3         | 8,33         |
| 8           | 0-4-2-13-6-0-5-14-15-0-10-7-1-11-0-12-9-3-0-8-0 | 6710,3         | 8,49         |
| 10          | 0-4-2-13-6-0-5-14-15-0-10-7-1-11-0-12-9-3-0-8-0 | 6710,3         | 8,99         |

Como se observa en Tabla 11, el tiempo de cómputo aumenta a medida que incrementa el número de los elementos Tabú, sin embargo, a partir de 4 se obtiene un costo constante, o sea, no mejora la respuesta de la función objetivo. El resultado obtenido en la Tabla 11 es 68,8 yuanes mejor que el obtenido por Ma, & Wei, (2013). En la Figura 21 se muestra que, a partir de la iteración 20 se llega a un ciclo. Este proceso es más eficiente que el realizado por Ma, & Wei, (2013), pues muestra convergencia más rápido, ya que la respuesta final se obtiene en la iteración 19 con un tiempo de cómputo de 6,1 segundos. El programa de Matlab se encuentra en el Apéndice B.

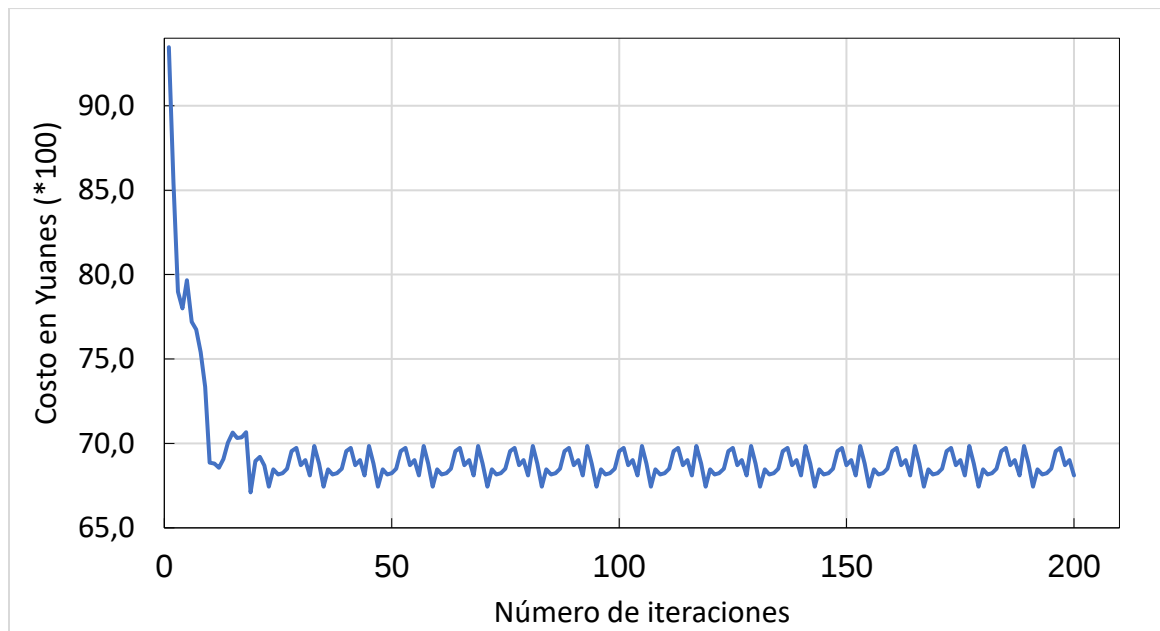


Figura 21. Proceso iterativo para Lista Tabú de 4

## 9. Resultados

El computador donde se correrá el programa es Intel-Corei7 de séptima generación, con 4 núcleos de 3,6 GHz cada uno, memoria RAM de 16 GB, sistema operativo de 64 bits, procesador x64 y memoria interna de 1 tera, perteneciente a la autora del presente proyecto y la versión de Matlab es R2020a. El resultado para el algoritmo clásico de Búsqueda Tabú se muestra en la Tabla 12 (hasta la iteración 29) y Búsqueda Tabú modificado (Tabla 13).

Tabla 12.  
Resultado de Búsqueda Tabú tradicional

| Costo       | Ruta                                                         |
|-------------|--------------------------------------------------------------|
| 9346,9      | 0-2--13-5--10-0-4--11--14-0--12-7-1-6-0--15-9-3-0-8-0        |
| 8542,1      | 0--11--13-5--10-0-4-2--14-0--12-7-1-6-0--15-9-3-0-8-0        |
| 7898,4      | 0--11--10-5--13-0-4-2--14-0--12-7-1-6-0--15-9-3-0-8-0        |
| 7799        | 0--10--11-5--13-0-4-2--14-0--12-7-1-6-0--15-9-3-0-8-0        |
| 7966        | 0--12--11-5--13-0-4-2--14-0--10-7-1-6-0--15-9-3-0-8-0        |
| 7720,6      | 0--12-6-5--13-0-4-2--14-0--10-7-1--11-0--15-9-3-0-8-0        |
| 7674,6      | 0--12--13-5-6-0-4-2--14-0--10-7-1--11-0--15-9-3-0-8-0        |
| 7540        | 0--12--15-5-6-0-4-2--14-0--10-7-1--11-0--13-9-3-0-8-0        |
| 7335,2      | 0--12-2-5-6-0-4--15--14-0--10-7-1--11-0--13-9-3-0-8-0        |
| 6885,6      | 0--13-2-5-6-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0        |
| 6881,7      | 0-2--13-5-6-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0        |
| 6856,8      | 0-2--14-5-6-0-4--15--13-0--10-7-1--11-0--12-9-3-0-8-0        |
| 6906,7      | 0-4--14-5-6-0-2--15--13-0--10-7-1--11-0--12-9-3-0-8-0        |
| 7004        | 0-4--15-5-6-0-2--14--13-0--10-7-1--11-0--12-9-3-0-8-0        |
| 7064,9      | 0-2--15-5-6-0-4--14--13-0--10-7-1--11-0--12-9-3-0-8-0        |
| 6856,8      | 0-2--14-5-6-0-4--15--13-0--10-7-1--11-0--12-9-3-0-8-0        |
| 6881,7      | 0-2--13-5-6-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0        |
| 6885,6      | 0--13-2-5-6-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0        |
| 6917,3      | 0-5-2--13-6-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0        |
| <b>6825</b> | <b>0-5-6--13-2-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0</b> |
| 6847,7      | 0-6-5--13-2-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0        |
| 6972,4      | 0--13-5-6-2-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0        |
| 6919,8      | 0--13-2-6-5-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0        |
| 6897,3      | 0-2--13-6-5-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0        |
| 6958,2      | 0-2-5-6--13-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0        |

| Costo  | Ruta                                                  |
|--------|-------------------------------------------------------|
| 6861,2 | 0-6-5-2--13-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0 |
| 6857   | 0-5-6-2--13-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0 |
| 6941,1 | 0-5--13-2-6-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0 |
| 6881,7 | 0-2--13-5-6-0-4--15--14-0--10-7-1--11-0--12-9-3-0-8-0 |

Tabla 13.

Resultado de búsqueda Tabú con número tabú de 4

| Costo         | Ruta                                                                   |
|---------------|------------------------------------------------------------------------|
| 9346,9        | -0- 2-13- 5-10- 0- 4-11-14- 0-12- 7- 1- 6- 0-15- 9- 3- 0- 8- 0         |
| 8542,1        | - 0-11-13- 5-10- 0- 4- 2-14- 0-12- 7- 1- 6- 0-15- 9- 3- 0- 8- 0        |
| 7898,4        | - 0-11-10- 5-13- 0- 4- 2-14- 0-12- 7- 1- 6- 0-15- 9- 3- 0- 8- 0        |
| 7799          | - 0-10-11- 5-13- 0- 4- 2-14- 0-12- 7- 1- 6- 0-15- 9- 3- 0- 8- 0        |
| 7966          | - 0-12-11- 5-13- 0- 4- 2-14- 0-10- 7- 1- 6- 0-15- 9- 3- 0- 8- 0        |
| 7720,6        | - 0-12- 6- 5-13- 0- 4- 2-14- 0-10- 7- 1-11- 0-15- 9- 3- 0- 8- 0        |
| 7674,6        | - 0-12-13- 5- 6- 0- 4- 2-14- 0-10- 7- 1-11- 0-15- 9- 3- 0- 8- 0        |
| 7540          | - 0-12-15- 5- 6- 0- 4- 2-14- 0-10- 7- 1-11- 0-13- 9- 3- 0- 8- 0        |
| 7335,2        | - 0-12- 2- 5- 6- 0- 4-15-14- 0-10- 7- 1-11- 0-13- 9- 3- 0- 8- 0        |
| 6885,6        | - 0-13- 2- 5- 6- 0- 4-15-14- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6881,7        | - 0- 2-13- 5- 6- 0- 4-15-14- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6856,8        | - 0- 2-14- 5- 6- 0- 4-15-13- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6906,7        | - 0- 4-14- 5- 6- 0- 2-15-13- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 7004          | - 0- 4-15- 5- 6- 0- 2-14-13- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 7064,9        | - 0- 2-15- 5- 6- 0- 4-14-13- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 7031,5        | - 0- 2-13- 5- 6- 0- 4-14-15- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 7035,4        | - 0-13- 2- 5- 6- 0- 4-14-15- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 7067,2        | - 0- 5- 2-13- 6- 0- 4-14-15- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| <b>6710,3</b> | <b>- 0- 4- 2-13- 6- 0- 5-14-15- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0</b> |
| 6896,6        | - 0- 2- 4-13- 6- 0- 5-14-15- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6920,2        | - 0-13- 4- 2- 6- 0- 5-14-15- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6868,9        | - 0-13- 6- 2- 4- 0- 5-14-15- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6744,6        | - 0- 6-13- 2- 4- 0- 5-14-15- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6846,6        | - 0- 6-15- 2- 4- 0- 5-14-13- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6815,6        | - 0- 4-15- 2- 6- 0- 5-14-13- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6823,4        | - 0- 4- 2-15- 6- 0- 5-14-13- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6849,7        | - 0- 6- 2-15- 4- 0- 5-14-13- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6954,4        | - 0- 6- 4-15- 2- 0- 5-14-13- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6974,2        | - 0- 6-15- 4- 2- 0- 5-14-13- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |
| 6870,8        | - 0- 6-13- 4- 2- 0- 5-14-15- 0-10- 7- 1-11- 0-12- 9- 3- 0- 8- 0        |

Se muestra que para busque Tabú tradicional converge en 20 iteraciones a la respuesta, igual que la modificación, sin embargo, esta última es más eficiente, pues pasa de 6825 (búsqueda Tabú tradicional) a 6710,3 (búsqueda Tabú modificada), esto muestra que el número Tabú es de gran importancia para mejorar la respuesta de un determinado problema, por lo que la modificación real es encontrar un equilibrio entre número Tabú y tiempo de cómputo.

## 10. Conclusiones

Según la revisión de literatura, en las últimas dos décadas han surgido distintas investigaciones con la finalidad de solucionar el Problema de Ruteo de Vehículo Milk-run con Ventana de Tiempo (MRVRPTWs), los principales enfoques para solucionarlo son con métodos heurísticos, como el algoritmo de ahorros y optimización de enjambre de partículas.

De acuerdo con el resultado del diseño de experimento aplicado a metaheurística búsqueda tabú, se encontró que la lista tabú impacta más en el tiempo de cómputo en comparación al número de iteraciones, esto se debe a que cada vector en la lista tabú se compara con los vectores generados en la iteración, el cual es una combinatoria, siendo este mayor que las etapas de las iteraciones.

El presente algoritmo genera un costo de transporte de 6710,3 yuanes, al compararlo con el algoritmo heurístico de optimización de enjambre de partículas, se obtiene en este último un costo superior, ya que es 1,04% mayor. Además, el presente algoritmo llega al resultado final en 19 iteraciones, mientras el método heurístico tarda 140 iteraciones.

Para MRVRPTWs no se tenía presente la magnitud de la lista tabú en la literatura, siendo el presente trabajo de investigación una forma de determinarlo, pues se concluye que con un valor de 4 se llega a una solución mejor que si se usara enjambre de partículas, este proceso no supera las 20 iteraciones para lograr el mejor óptimo de la búsqueda tabú y genera un tiempo de cómputo no superior a 10 segundos.

## 11. Recomendaciones

El numero Tabú permite que una ruta pueda salir de un óptimo local, no obstante, se llega a un ciclo (repetición de óptimos locales o de rutas), por tal motivo se recomienda analizar el resultado de números Tabús grandes, con la intención de mejorar la respuesta y evitar caer en ciclos de manera temprana (en pocas iteraciones).

La solución inicial (SI) se realiza por el método del barrido en la presente investigación, no obstante, se podría llegar a una mejor solución con otra metodología para determinar la (SI), por tal motivo se recomienda utilizar distintas metodologías.

En el presente problema se comparan con optimización de enjambre de partículas, metaheurística búsqueda tabú y la modificación del anterior método, por tal motivo se recomienda utilizar otros métodos metaheurísticos con la intención de concluir cual es más eficiente.

**Referencias Bibliográficas**

- Afifi, S., Dang, D.-C., & Moukrim, A, (2016), Heuristic solutions for the vehicle routing problem with time windows and synchronized visits, *Optimization Letters*, 10(3), 511-525, <https://doi.org/10.1007/s11590-015-0878-3>
- Aragao, D, P., Jr., Naclerio Novaes, A, G., & Mendes Luna, M, M, (2019), An agent-based approach to evaluate collaborative strategies in Milk-runOEM operations, *COMPUTERS & INDUSTRIAL ENGINEERING*, 129, 545-555, <https://doi.org/10.1016/j.cie.2019.01.026>
- Avellaneda, J., & Estévez, R, (2019), Un algoritmo colonia de hormigas para el problema de ruteo de vehículos eléctricos con funcion de carga pararcial y ventana de tiempo, (Trabajo de grado para optar por el título de Ingeniero Industrial), Universidad Industrial de Santander, Bucaramanga,
- Azi, N., Gendreau, M., & Potvin, J,-Y, (2010), An exact algorithm for a vehicle routing problem with time windows and multiple use of vehicles, *European Journal of Operational Research*, 202(3), 756-763, <https://doi.org/10.1016/j.ejor.2009.06.034>
- Bae, K, G., Evans, L, A., & Summers, A, (2016), Lean design and analysis of a Milk-rundelivery system: Case study, 2016 Winter Simulation Conference (WSC), 2855-2866, <https://doi.org/10.1109/WSC.2016.7822321>
- Ballou, R, H, (2004), *Logística: Administración de la cadena de suministro*, Pearson Educación,

- Bianchessi, N., Drexler, M., & Irnich, S. (2019), The Split Delivery Vehicle Routing Problem with Time Windows and Customer Inconvenience Constraints, *Transportation Science*, <https://doi.org/10.1287/trsc.2018.0862>
- Bodas López, R. (2017), La metaheurística de Búsqueda Tabú aplicada al problema de Enrutamiento de Vehículos, Recuperado de <http://uvadoc.uva.es/handle/10324/26070>
- Braekers, K., Caris, A., & Janssens, G. K. (2014), Bi-objective optimization of drayage operations in the service area of intermodal terminals, *Transportation Research Part E: Logistics and Transportation Review*, 65(1), 50-69, <https://doi.org/10.1016/j.tre.2013.12.012>
- Campos, T. M. C., Do Sameiro Carvalho, M., Oliveira, J. A., Silva, P. V., & Machado, T. (2017), Using discrete simulation to support internal logistics process design, *Proceedings of International Conference on Computers and Industrial Engineering, CIE*, Recuperado de <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85040936057&partnerID=40&md5=fefa6050e4942aa13043dd3cce4059a1>
- Christopher, M. (2005), *Logistics and Supply Chain Management: Creating Value-Adding Networks*, Recuperado de <http://digilib.umpalopo.ac.id:8080/jspui/handle/123456789/85>
- Clarke, G., & Wright, J. W. (1964), Scheduling of Vehicles from a Central Depot to a Number of Delivery Points, *Operations Research*, 12(4), 568-581, <https://doi.org/10.1287/opre.12.4.568>

- Cordeau, J., Desaulniers, G., Desrosiers, J., Solomon, M., & Soumis, F, (2002), 7, VRP with Time Windows, En Discrete Mathematics and Applications, The Vehicle Routing Problem (Vols, 1–0, pp, 157-193), <https://doi.org/10.1137/1.9780898718515.ch7>
- Dantzig, G, B., & Ramser, J, H, (1959), The Truck Dispatching Problem, Management Science, 6(1), 80-91, <https://doi.org/10.1287/mnsc.6.1.80>
- Daza, J, M., Montoya, J, R., & Narducci, F, (2009), Solving the capacitated vehicle routing problem using a two-phase metaheuristic procedure, Revista EIA, (12), 23-38,
- De la Pava, W., & Cardenas, J, (2016), Diseño de una red de distribución para un producto de consumo masivo, con enfoque a redes de valor, (Documento como prerrequisito para optar al título de Magister en Gestión de Redes de Valor y Logística), Universidad Piloto de Colombia, Bogotá,
- Du, T., Wang, F, K., & Lu, P,-Y, (2007), A real-time vehicle-dispatching system for consolidating milk runs, Transportation Research Part E: Logistics and Transportation Review, 43(5), 565-577, <https://doi.org/10.1016/j.tre.2006.03.001>
- Fajardo González, H, M, (2017), Análisis del sector de los operadores logísticos en Colombia, para la creación de un modelo de selección de servicios logísticos utilizando la metodología AHP (Masters, Universidad Nacional de Colombia-Sede Bogotá), Recuperado de <http://www.bdigital.unal.edu.co/57949/>
- Fracchia, F, (2011), Implementación de un sistema de aprovisionamiento programado a proveedores que tenga por finalidad mejorar el nivel de servicio post-venta de la compañía,

(Tesis de grado para optar por el título de Ingeniero Industrial), Instituto Tecnológico de Buenos Aires, Buenos Aires,

Glover, F, (1986), Future paths for integer programming and links to artificial intelligence, Computers & Operations Research, 13(5), 533-549, [https://doi.org/10.1016/0305-0548\(86\)90048-1](https://doi.org/10.1016/0305-0548(86)90048-1)

Greenwood, A, G., Kluska, K., & Pawlewski, P, (2017), A Multi-level Framework for Simulating Mik-Run, In-plant Logistics Operations, En J, Bajo, Z, Vale, K, Hallenborg, A, P, Rocha, P, Mathieu, P, Pawlewski, ... J, Holmgren (Eds.), Highlights of Practical Applications of Cyber-Physical Multi-Agent Systems (pp, 209-220), Springer International Publishing,

Hosseini, S, D., Shirazi, M, A., & Karimi, B, (2014), Cross-docking and Milk-run logistics in a consolidation network: A hybrid of harmony search and simulated annealing approach, JOURNAL OF MANUFACTURING SYSTEMS, 33(4), 567-577, <https://doi.org/10.1016/j.jmsy.2014.05.004>

Kok, A, L., Hans, E, W., & Schutten, J, M, J, (2012), Vehicle routing under time-dependent travel times: The impact of congestion avoidance, Computers & Operations Research, 39(5), 910-918, <https://doi.org/10.1016/j.cor.2011.05.027>

Koskosidis, Y, A., Powell, W, B., & Solomon, M, M, (1992), An Optimization-Based Heuristic for Vehicle Routing and Scheduling with Soft Time Window Constraints, Transportation Science, 26(2), 69-85, <https://doi.org/10.1287/trsc.26.2.69>

- Lenstra, J, K., & Kan, A, H, G, R, (1981), Complexity of vehicle routing and scheduling problems, *Networks*, 11(2), 221-227, <https://doi.org/10.1002/net.3230110211>
- Ma, H, J., & Wei, J, (2013), Milk-Run Vehicle Routing Optimization Model and Algorithm of Automobile Parts, In *Applied Mechanics and Materials* (Vol, 253, pp, 1463-1467), Trans Tech Publications Ltd,
- Maldonado, C, E., & Gómez Cruz, N, A, (2010), El mundo de las ciencias de la complejidad [WorkingPaper], Recuperado de Editorial Universidad del Rosario website: <http://repository,urosario,edu,co/handle/10336/3301>
- Manisri, T., Mungwattana, A., & Janssens, G, K, (2011), Minimax optimisation approach for the Robust Vehicle Routing Problem with Time Windows and uncertain travel times, *International Journal of Logistics Systems and Management*, 10(4), 461-477, <https://doi.org/10.1504/IJLSM,2011,043105>
- Martínez, L, (2015), Optimización de la cadena de abastecimiento de materia prima en una empresa del sector metalmecánico (Trabajo fin de grado en ingeniería en tecnología industriales), Universidad Politécnica de València, Madrid,
- Mazo, A, Z., Montoya, R, A, G., & Henao, S, A, F, (2014), Indicadores logísticos en la cadena de suministro como apoyo al modelo scor, *CLIO América*, 8(15), 90-110,
- Mei, H., Jingshuai, Y., Teng, M., Xiuli, L., & Ting, W, (2017), The Modeling of Milk-run Vehicle Routing Problem Based on Improved C-W Algorithm that Joined Time Window, *Transportation Research Procedia*, 25, 716-728, <https://doi.org/10.1016/j.trpro,2017,05,453>

- Ministerio de transporte, (2019), Recuperado 3 de septiembre de 2019, de [continuidad-mintrans,nexura.com/](http://continuidad-mintrans.nexura.com/)
- Moccia, L., Cordeau, J.-F., & Laporte, G, (2012), An incremental tabú search heuristic for the generalized vehicle routing problem with time windows, *Journal of the Operational Research Society*, 63(2), 232-244, <https://doi.org/10.1057/jors.2011.25>
- Montoya, G., & Andrés, R, (2011), Logística inversa un proceso de impacto ambiental y productividad, Recuperado de <http://repository.lasallista.edu.co/dspace/handle/10567/93>
- Montez, E, (2017), Metaheurísticas para el problema de ruteo de vehículos con ventanas de tiempo (VRP-TW), (Postgrado de Optimización), Universidad Autónoma Metropolitana Azcapotzalco, México,
- No, P, (2002), Marco conceptual de la cadena de suministro: Un nuevo enfoque logístico,
- Orrego, J, J, M, (2014), logística de aprovisionamiento, Ediciones Paraninfo, S,A,
- Pecin, D., Contardo, C., Desaulniers, G., & Uchoa, E, (2017), New Enhancements for the Exact Solution of the Vehicle Routing Problem with Time Windows, *INFORMS Journal on Computing*, 29(3), 489-502, <https://doi.org/10.1287/ijoc.2016.0744>
- Pei, S., & Zhao, J, (2016), A scalable approach support for external Milk-runplanning and optimization in home appliances enterprises, *ACM International Conference Proceeding Series*, 67-70, <https://doi.org/10.1145/3012258.3012277>

- Puente, J., García, N., Fernández, I., & Parreño, J, (2008), Modelo borroso para la evaluación y selección de proveedores, II International Conference on Industrial Engineering and Industrial Management, 1523-1532,
- Pullen, H, G, M., & Webb, M, H, J, (1967), A computer application to a transport scheduling problem, The Computer Journal, 10(1), 10-13, <https://doi.org/10.1093/comjnl/10.1.10>
- Qiu, X., Xu, G., & Huang, G, Q, (2014), A bilevel model for transportation service sharing in Supply Hub in Industrial Park (SHIP), IEEE International Conference on Industrial Engineering and Engineering Management, 986-990, <https://doi.org/10.1109/IEEM.2013.6962558>
- Riveros, D, P, B., & Silva, P, P, B, (2004), La logística competitiva y la administración de la cadena de suministros, Scientia et technica, 1(24), <https://doi.org/10.22517/23447214.7347>
- Sadjadi, S, J., Jafari, M., & Amini, T, (2009), A new mathematical modeling and a genetic algorithm search for Milk-run problem (an auto industry supply chain case study), International Journal of Advanced Manufacturing Technology, 44(1-2), 194-200, <https://doi.org/10.1007/s00170-008-1648-5>
- Servera-Francés, D, (2010), Concepto y evolución de la función logística, Innovar, 20(38), 217-234,
- Staab, T., Klenk, E., Galka, S., & Guenther, W, A, (2016), Efficiency in in-plant Milk-runsystems-The influence of routing strategies on system utilization and process stability, JOURNAL OF SIMULATION, 10(2, SI), 137-143, <https://doi.org/10.1057/jos.2015.6>

- Swanson, E, (1986), The Vehicle Routing Problem with Time Constraints (Tesis de lic, Department of Computer Science y Operations Research), North Dakota State University, Fargo,
- Vélez, M, C., & Montoya, J, A, (2007), Metaheurísticos: una alternativa para la solución de problemas combinatorios en administración de operaciones, *Revista EIA*, (8), 99-115,
- Wang, C., Mu, D., Zhao, F., & Sutherland, J, W, (2015), A parallel simulated annealing method for the vehicle routing problem with simultaneous pickup–delivery and time windows, *Computers & Industrial Engineering*, 83, 111-122, <https://doi.org/10.1016/j.cie.2015.02.005>
- Wang, L,-F., Liu, P., Huang, Y., Wang, Y,-C., & Kong, F,-S, (2013), Research on Milk-runrouting problem of automotive supply logistics based on genetic algorithm, *International Asia Conference on Industrial Engineering and Management Innovation: Core Areas of Industrial Engineering*, IEMI 2012 - Proceedings, 517-526, <https://doi.org/10.1007/978-3-642-38445-5-53>
- Xiong, T., Tang, Q., Huang, T., Shen, Z., Zhou, H., Hu, H, Y, K., & Li, Y, (2018), Delivery Vehicle Scheduling Modeling and Optimization for Automobile Mixed Milk-runMode Involved Indirect Suppliers, En S, Wang, M, Price, M, K, Lim, Y, Jin, Y, Luo, & R, Chen (Eds.), *Recent Advances in Intelligent Manufacturing* (pp, 169-178), Springer Singapore,
- Yepes Piqueras, V, (2008), Optimización heurística económica aplicada a las redes de transporte del tipo VRPTW, *Riunet*, <https://doi.org/10.4995/Thesis/10251/2664>