

**ANÁLISIS Y PROTOTIPADO DE UN ALGORITMO GENÉTICO MODIFICADO  
PARA SOLUCIONAR EL PROBLEMA DE RUTEO DE VEHÍCULOS CON  
VENTANAS DE TIEMPO (VRPTW), PRIORIDAD DE METAS ECONÓMICAS Y  
COMPONENTE MEDIO AMBIENTAL**

**CATALINA SEGURA LONDOÑO**

**CLAUDIA PATRICIA ARIAS HERNÁNDEZ**

**UNIVERSIDAD TECNOLÓGICA DE PEREIRA**

**FACULTAD DE INGENIERÍAS**

**PROGRAMA DE MAESTRÍA EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN**

**PEREIRA-2018**

**ANÁLISIS Y PROTOTIPADO DE UN ALGORITMO GENÉTICO MODIFICADO  
PARA SOLUCIONAR EL PROBLEMA DE RUTEO DE VEHÍCULOS CON  
VENTANAS DE TIEMPO (VPTWR), PRIORIDAD DE METAS ECONÓMICAS Y  
COMPONENTE MEDIO AMBIENTAL**

**CATALINA SEGURA LONDOÑO**

**CLAUDIA PATRICIA ARIAS HERNÁNDEZ**

**TRABAJO DE GRADO PARA OPTAR POR EL TÍTULO DE MAGISTER EN  
SISTEMAS Y COMPUTACIÓN**

**DIRECTOR**

**Ph.D (c) JORGE IVÁN RÍOS PATIÑO**

**UNIVERSIDAD TECNOLÓGICA DE PEREIRA**

**FACULTAD DE INGENIERÍAS**

**PROGRAMA DE MAESTRÍA EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN**

**PEREIRA-2018**

Nota de Aceptación

---

---

---

---

Presidente del Jurado

---

Jurado

---

Jurado

Pereira, Risaralda. Mayo 2018

*A Dios y a la Virgen y a nuestras  
familias por el apoyo y el amor  
incondicional que nos han  
brindado.*

## **AGRADECIMIENTOS**

Primero a Dios y a la Virgen, quienes guían y protegen a diario nuestro camino. A nuestras familias que nos brindaron su amor, comprensión y su compañía durante todo este proceso; así mismo agradecemos al PhD(c) Jorge Iván Ríos Patiño por su apoyo incondicional, confianza, orientación y colaboración en el desarrollo de este trabajo, a todos nuestros profesores, quienes nos orientaron con profesionalismo ético en la adquisición de conocimientos, afianzando nuestra formación. A la Universidad Tecnológica de Pereira y al programa de Maestría en Ingeniería de Sistemas y Computación por brindarnos la oportunidad de realizar nuestros estudios. A nuestros compañeros, quienes compartieron con nosotras durante este tiempo y a todas aquellas personas que estuvieron presentes en nuestro desarrollo personal y profesional.

Muchas Gracias

# TABLA DE CONTENIDO

INTRODUCCIÓN.....	19
CAPITULO 1. PLANTEAMIENTO DEL PROBLEMA .....	21
1.1 Definición Del Problema .....	21
1.2 Justificación.....	23
CAPITULO 2. MARCO TEÓRICO .....	25
2.1 Complejidad Computacional .....	25
2.2 Algoritmo Genético .....	26
2.3 Operadores Genéticos .....	27
2.4 Computación Evolutiva.....	31
2.5 Estrategias Evolutivas.....	31
2.6 Programación Evolutiva .....	32
2.7 Métodos Exactos .....	33
2.8 Heurísticas.....	33
2.9 Metaheurísticas.....	36
2.10 Problema Del Agente Viajero (TSP).....	38
2.11 El Problema De Enrutamiento De Vehículos (VRP).....	39
2.12 Problema con restricciones de Capacidad (CVRP).....	40
2.13 Problema con múltiples depósitos (MDVRP).....	41
2.14 Problema periódico (PVRP).....	42
2.15 Problema de entregas parciales (SDVRP) .....	44
2.16 Problema con valores al azar (SVRP) .....	45
2.17 Problema con entregas y devoluciones (VRPPD) .....	46
2.18 Problemas con viajes de regreso (VRPB).....	48
2.19 Problema con múltiples depósitos (MFVRP) .....	49
2.20 Problema de ruteo con ventanas de tiempo (VRPTW) .....	50
2.21 Generalizaciones de la VRPTW .....	51
2.22 Base De Datos Orientada A Grafos NEO4J .....	53
CAPITULO 3. ESTADO DEL ARTE .....	56
3.1 Vehicle Routing Problem with Simultaneous Pickups and Deliveries and Time Windows Considering Fuel Consumption and Carbon Emissions. ....	56
3.2 Drivers-Inspired Ants for Solving the Vehicle Routing Problem with Time Windows.....	57

<b>3.3 A Hybrid Genetic Algorithm for Vehicle Routing Problem With Time Windows.</b>	58
<b>3.4 Vehicle Routing with Time Windows Based on Two-stage Optimization Algorithm.</b>	59
<b>3.5 Models and Algorithms for the Vehicle Routing Problem with Time Windows and Other Conditions.</b>	60
<b>3.6 Improvement of Genetic Algorithm for Vehicle Routing Problems with Time Windows.</b>	61
<b>3.7 Solving Vehicle Routing and Scheduling Problems using Hybrid Genetic Algorithm.</b>	62
<b>3.8 Solving Vehicle Routing Problem with Time Windows with Hybrid Evolutionary Algorithm.</b>	63
<b>3.9 Vehicle Routing Problem with Time Windows: A Deterministic Annealing approach.</b>	64
<b>3.10 Selección óptima de operadores para el tratamiento de problemas VRP con Algoritmos Genéticos.</b>	65
<b>3.11 An algorithm for the routing problem with split deliveries and time windows (SDVRPTW) applied on retail SME distribution activities.</b>	66
<b>3.12 El problema de ruteo de vehículos [VRP] y su aplicación en medianas empresas colombianas.</b>	68
<b>3.13 Alternativa Heurística MCM para Problemas de Ruteo de Vehículos.</b>	69
<b>3.14 Algoritmos Meméticos Aplicados A La Resolución De Un Problema De Ruteo De Vehículos Periodico.</b>	71
<b>3.15 Diseño de Rutas de Transporte Escolar con Ventanas Temporales Móviles.</b>	72
<b>3.16 Método heurístico para el problema de ruteo de vehículos aplicado a la empresa distribuidora Representaciones Continental.</b>	74
<b>3.17 A set-covering based heuristic algorithm for the periodic vehicle routing problem</b>	75
<b>3.18 Metaheurísticas aplicadas al ruteo de vehículos. Un caso de estudio. Parte 1: formulación del problema y estudio.</b>	77
<b>3.19 Diseño de un Modelo de Ruteo de Vehículos: Caso de Estudio UMNG Sede Campus.</b>	78
<b>3.20 Modelo matemático para la planificación de servicios y programación de rutas en empresas prestadoras de servicios de control de plagas.</b>	79
<b>CAPITULO 4. METODOLOGÍA DE LA INVESTIGACIÓN</b>	81
<b>4.1 Enfoque De La Investigación</b>	81
<b>4.2 Población Objetivo De Estudio</b>	81
<b>4.3 Procedimiento De Investigación</b>	81

<b>CAPITULO 5. IMPLEMENTACIÓN DE LA SOLUCIÓN .....</b>	<b>82</b>
<b>5.1 Formulación Matemática Del VRPTW .....</b>	<b>82</b>
<b>5.2 Descripción Del Algoritmo Genético .....</b>	<b>85</b>
<b>5.3 Pseudocódigo.....</b>	<b>90</b>
<b>5.4 Cálculo De La Emisión De Gases.....</b>	<b>92</b>
<b>CAPÍTULO 6. ANÁLISIS DE RESULTADOS .....</b>	<b>95</b>
<b>6.1 Presentación de Resultados.....</b>	<b>95</b>
<b>6.1.1 Resultados Archivo RC206.....</b>	<b>95</b>
<b>6.1.2 Resultados Archivo R210.....</b>	<b>101</b>
<b>6.2 Ánàlisis De Resultados .....</b>	<b>106</b>
<b>CAPÍTULO 7. CONCLUSIONES.....</b>	<b>108</b>
<b>7.1 Conclusiones.....</b>	<b>108</b>
<b>BIBLIOGRAFÍA.....</b>	<b>109</b>
<b>ANEXOS.....</b>	<b>112</b>
<b>1. Vehicle Routing Problem with Simultaneous Pickups and Deliveries and Time Windows Considering Fuel Consumption and Carbon Emissions. ....</b>	<b>112</b>
<b>2 Drivers-Inspired Ants for Solving the Vehicle Routing Problem with Time Windows.....</b>	<b>112</b>
<b>3 A Hybrid Genetic Algorithm for Vehicle Routing Problem With Time Windows.....</b>	<b>113</b>
<b>4. Vehicle Routing with Time Windows Based on Two-stage Optimization Algorithm.....</b>	<b>114</b>
<b>5. Models and Algorithms for the Vehicle Routing Problem with Time Windows and Other Conditions.....</b>	<b>114</b>
<b>6. Improvement of Genetic Algorithm for Vehicle Routing Problems with Time Windows.....</b>	<b>115</b>
<b>7. Solving Vehicle Routing and Scheduling Problems using Hybrid Genetic Algorithm.....</b>	<b>115</b>
<b>8. Solving Vehicle Routing Problem with Time Windows with Hybrid Evolutionary Algorithm.....</b>	<b>116</b>
<b>9. Vehicle Routing Problem with Time Windows: A Deterministic Annealing approach.....</b>	<b>116</b>
<b>10. Selección óptima de operadores para el tratamiento de problemas VRP con Algoritmos Genéticos.....</b>	<b>117</b>
<b>11. An algorithm for the routing problem with split deliveries and time windows (SDVRPTW) applied on retail SME distribution activities. ....</b>	<b>118</b>



<b>12. El problema de ruteo de vehículos [VRP] y su aplicación en medianas empresas colombianas.....</b>	<b>118</b>
<b>13. Alternativa Heurística MCM para Problemas de Ruteo de Vehículos. ....</b>	<b>119</b>
<b>14. Algoritmos Meméticos Aplicados A La Resolución De Un Problema De Ruteo De Vehículos Periodico. ....</b>	<b>120</b>
<b>15. Diseño de Rutas de Transporte Escolar con Ventanas Temporales Móviles. ....</b>	<b>121</b>
<b>16. Método heurístico para el problema de ruteo de vehículos aplicado a la empresa distribuidora Representaciones Continental. ....</b>	<b>121</b>
<b>17. A set-covering based heuristic algorithm for the periodic vehicle routing problem.....</b>	<b>122</b>
<b>18. Metaheurísticas aplicadas al ruteo de vehículos. Un caso de estudio. Parte 1: formulación del problema y estudio. ....</b>	<b>123</b>
<b>19. Diseño de un Modelo de Ruteo de Vehículos: Caso de Estudio UMNG Sede Campus.....</b>	<b>123</b>
<b>20. Modelo matemático para la planificación de servicios y programación de rutas en empresas prestadoras de servicios de control de plagas. ....</b>	<b>124</b>
<b>Resultados RC206, AKT 125 NKD.....</b>	<b>125</b>
<b>Resultados RC206, SUZUKI GN125.....</b>	<b>131</b>
<b>RESULTADOS HONDA CB1 PRO.....</b>	<b>137</b>
<b>Resultados RC206, YAMAHA LIBERO 125.....</b>	<b>143</b>
<b>Resultados R210, AKT 125 NKD.....</b>	<b>149</b>
<b>Resultados R210, SUZUKI GN125.....</b>	<b>155</b>
<b>Resultados R210, HONDA CB1 PRO.....</b>	<b>161</b>
<b>Resultados R210, YAMAHA LIBERO 125.....</b>	<b>167</b>

## TABLA DE ILUSTRACIONES

Ilustración 1: Algoritmo heurístico basado en búsqueda local.....	35
<i>Ilustración 2. Variantes del VRDSPTW.</i> .....	48
<i>Ilustración 3 Base de Datos NEO4J</i> .....	54
Ilustración 4 Clientes, Coordenadas en (x,y).....	86
Ilustración 5 Asignación de agentes por moto en BD.....	87
Ilustración 6 Agente almacenado en base de datos.....	89
Ilustración 12 PROMEDIO CONSUMO EMISIÓN AKT 125 NKD .....	95
Ilustración 18 PROMEDIO CONSUMO EMISIÓN SUZUKI GN125 .....	96
Ilustración 24 PROMEDIO CONSUMO EMISIÓN, HONDA CB1 PRO .....	97
Ilustración 31 PROMEDIO CONSUMO EMISIÓN YAMAHA LIBERO 125.....	98
Ilustración 32 CONSUMO Y EMISIÓN DE LAS MOTOS .....	99
Ilustración 33 GRÁFICO RESUMEN, TIEMPOS ITERACIONES RC206 .....	100
Ilustración 39 PROMEDIOS CONSUMO EMISIÓN AKT 125 NKD .....	101
Ilustración 46 PROMEDIOS CONSUMO EMISIÓN SUZUKI GN125 .....	102
Ilustración 52 PROMEDIOS CONSUMO EMISIÓN HONDA CB1 PRO.....	103
Ilustración 58 PROMEDIO CONSUMO EMISIÓN YAMAHA LIBERO 125.....	104
Ilustración 59 CONSUMO Y EMISIÓN DE LAS MOTOS .....	105
Ilustración 60 GRÁFICO RESUMEN, TIEMPOS ITERACIONES R210 .....	106
Ilustración 7 Iteración1, AKT 125 NKD.....	126
Ilustración 8 Iteración 2, AKT 125 NKD.....	127
Ilustración 9 ITERACIÓN 3, AKT 125 NKD .....	128
Ilustración 10 ITERACIÓN 4, AKT 125 NKD .....	129
Ilustración 11 ITERACIÓN 5, AKT 125 NKD .....	130
Ilustración 13 ITERACIÓN 1, RC206, SUZUKI GN125 .....	132
Ilustración 14 ITERACIÓN 2 RC206, SUZUKI GN125.....	133
Ilustración 15 ITERACIÓN 3 RC206, SUZUKI GN125 .....	134
Ilustración 16 ITERACIÓN 4 RC206, SUZUKI GN125.....	135
Ilustración 17 ITERACIÓN 5 RC206, SUZUKI GN125 .....	136
Ilustración 19 ITERACIÓN 1, RC206, HONDA CB1 PRO.....	138
Ilustración 20 ITERACIÓN 2, RC206, HONDA CB1 PRO .....	139
Ilustración 21 ITERACIÓN 3, RC206, HONDA CB1 PRO.....	140
Ilustración 22 ITERACIÓN 4, RC206, HONDA CB1 PRO .....	141
Ilustración 23 ITERACIÓN 5, RC206, HONDA CB1 PRO.....	142
Ilustración 25 TIEMPOS ITERACIONES, RC206, YAMAHA LIBERO 125 .....	143
Ilustración 26 ITERACIÓN 1, RC206, YAMAHA LIBERO 125 .....	144
Ilustración 27 ITERACIÓN 2, RC206, YAMAHA LIBERO 125 .....	145
Ilustración 28 ITERACIÓN 3, RC206, YAMAHA LIBERO 125 .....	146
Ilustración 29 ITERACIÓN 4, RC206, YAMAHA LIBERO 125 .....	147

Ilustración 30 ITERACIÓN 5, RC206, YAMAHA LIBERO 125 .....	148
Ilustración 34 ITERACIÓN1, R210, AKT 125 NKD.....	150
Ilustración 35 ITERACIÓN2, R210, AKT 125 NKD.....	151
Ilustración 36 ITERACIÓN3, R210, AKT 125 NKD.....	152
Ilustración 37 ITERACIÓN4, R210, AKT 125 NKD.....	153
Ilustración 38 ITERACIÓN5, R210, AKT 125 NKD.....	154
Ilustración 40 ITERACIÓN 1, R210, SUZUKI GN125.....	156
Ilustración 41 ITERACIÓN 2, R210, SUZUKI GN125.....	157
Ilustración 42 ITERACIÓN 3, R210, SUZUKI GN125.....	158
Ilustración 43 ITERACIÓN 4, R210, SUZUKI GN125.....	159
Ilustración 44 ITERACIÓN 5, R210, SUZUKI GN125.....	160
Ilustración 45 ITERACIÓN 5, R210, SUZUKI GN125.....	160
Ilustración 47 ITERACIÓN 1, R210, HONDA CB1 PRO .....	162
Ilustración 48 ITERACIÓN 2, R210, HONDA CB1 PRO .....	163
Ilustración 49 ITERACIÓN 3, R210, HONDA CB1 PRO .....	164
Ilustración 50 ITERACIÓN 4, R210, HONDA CB1 PRO .....	165
Ilustración 51 ITERACIÓN 5, R210, HONDA CB1 PRO .....	166
Ilustración 53 ITERACIÓN1, R210, YAMAHA LIBERO 125.....	168
Ilustración 54 ITERACIÓN2, R210, YAMAHA LIBERO 125.....	169
Ilustración 55 ITERACIÓN3, R210, YAMAHA LIBERO 125.....	170
Ilustración 56 ITERACIÓN4, R210, YAMAHA LIBERO 125.....	171
Ilustración 57 ITERACIÓN5, R210, YAMAHA LIBERO 125.....	172

## INDICE DE TABLAS

Tabla 1 Caracterización Fisicoquímica de Combustible Líquido .....	92
Tabla 8 PROMEDIO CONSUMO EMISIÓN AKT 125 NKD.....	95
Tabla 15 PROMEDIO CONSUMO-EMISIÓN, SUZUKI GN125.....	96
Tabla 22 PROMEDIO CONSUMO EMISIÓN HONDA CB1 PRO.....	97
Tabla 28 PROMEDIO CONSUMO EMISIÓN YAMAHA LIBERO 125 .....	98
Tabla 29 CONSUMO Y EMISIÓN DE LAS MOTOS .....	99
Tabla 30 TABLA RESUMEN, TIEMPOS ITERACIONES RC206.....	99
Tabla 37 PROMEDIOS CONSUMO EMISIÓN AKT 125 NKD .....	101
Tabla 43 PROMEDIOS CONSUMO EMISIÓN SUZUKI GN125 .....	102
Tabla 50 PROMEDIOS CONSUMO EMISIÓN HONDA CB1 PRO .....	103
Tabla 57 PROMEDIO CONSUMO EMISIÓN YAMAHA LIBERO 125 .....	104
Tabla 58 CONSUMO Y EMISIÓN DE LAS MOTOS .....	105
Tabla 59 TABLA RESUMEN, TIEMPOS ITERACIONES R210 .....	105
Tabla 2 Tiempos en cada iteración, AKT 125 NKD .....	125
Tabla 3. iteración 1, RC206, AKT 125 NKD .....	125
Tabla 4 Iteración 2, RC206 , AKT 125 NKD .....	126
Tabla 5 Iteración 3, RC206, AKT 125 NKD .....	127
Tabla 6 ITERACIÓN 4, RC206.....	128
Tabla 7 ITERACIÓN 5, RC206, AKT 125 NKD .....	129
Tabla 9 TIEMPOS ITERACIONES, SUZUKI GN125.....	131
Tabla 10 Iteración 1 RC206, SUZUKI GN125 .....	131
Tabla 11 ITERACIÓN 2 RC206, SUZUKI GN125 .....	132
Tabla 12 ITERACIÓN 3 RC206, SUZUKI GN125 .....	133
Tabla 13 ITERACIÓN 4 RC206, SUZUKI GN125 .....	134
Tabla 14 ITERACIÓN 5 RC206, SUZUKI GN125 .....	135
Tabla 16 TIEMPOS ITERACIONES, HONDA CB1 PRO .....	137
Tabla 17 ITERACIÓN 1, RC206, HONDA CB1 PRO .....	137
Tabla 18 ITERACIÓN 2, RC206, HONDA CB1 PRO .....	138
Tabla 19 ITERACIÓN 3, RC206, HONDA CB1 PRO .....	139
Tabla 20 ITERACIÓN 4, RC206, HONDA CB1 PRO .....	140
Tabla 21 ITERACIÓN 5, RC206, HONDA CB1 PRO .....	141
Tabla 23 ITERACIÓN 1, RC206, YAMAHA LIBERO 125.....	143
Tabla 24 ITERACIÓN 2, RC206, YAMAHA LIBERO 125.....	144
Tabla 25 ITERACIÓN 3, RC206, YAMAHA LIBERO 125.....	145
Tabla 26 ITERACIÓN 4, RC206, YAMAHA LIBERO 125.....	146
Tabla 27 ITERACIÓN 5, RC206, YAMAHA LIBERO 125.....	147
Tabla 31 TIEMPOS ITERACIONES AKT 125 NKD.....	149
Tabla 32 ITERACIÓN1, R210, AKT 125 NKD.....	149
Tabla 33 ITERACIÓN2, R210, AKT 125 NKD .....	150

Tabla 34 ITERACIÓN3, R210, AKT 125 NKD .....	151
Tabla 35 ITERACIÓN4, R210, AKT 125 NKD .....	152
Tabla 36 ITERACIÓN5, R210, AKT 125 NKD .....	153
Tabla 38 TIEMPOS ITERACIONES, R210, SUZUKI GN125 .....	155
Tabla 39 ITERACIÓN 1, R210, SUZUKI GN125 .....	155
Tabla 40 ITERACIÓN 2, R210, SUZUKI GN125 .....	156
Tabla 41 ITERACIÓN 3, R210, SUZUKI GN125 .....	157
Tabla 42 ITERACIÓN 4, R210, SUZUKI GN125 .....	158
Tabla 44 TIEMPOS ITERACIONES, R210, HONDA CB1 PRO .....	161
Tabla 45 ITERACIÓN 1, R210, HONDA CB1 PRO .....	161
Tabla 46 ITERACIÓN 2, R210, HONDA CB1 PRO .....	162
Tabla 47 ITERACIÓN 3, R210, HONDA CB1 PRO .....	163
Tabla 48 ITERACIÓN 4, R210, HONDA CB1 PRO .....	164
Tabla 49 ITERACIÓN 5, R210, HONDA CB1 PRO .....	165
Tabla 51 TIEMPOS ITERACIONES, R210, YAMAHA LIBERO 125 .....	167
Tabla 52 ITERACIÓN1, R210, YAMAHA LIBERO 125 .....	167
Tabla 53 ITERACIÓN2, R210, YAMAHA LIBERO 125 .....	168
Tabla 54 ITERACIÓN3, R210, YAMAHA LIBERO 125 .....	169
Tabla 55 ITERACIÓN4, R210, YAMAHA LIBERO 125 .....	170
Tabla 56 ITERACIÓN5, R210, YAMAHA LIBERO 125 .....	171

## RESUMEN

El origen de los problemas de ruteo se da en el siglo XVIII cuando habitantes de Königsberg, un pueblo de Rusia, debate sobre cuál es la ruta que pasa una sola vez por los siete puentes que atravesaban el río Pregel regresando al punto de origen; este problema lo propuso el matemático suizo Leonhard Euler, quien en el año 1736 demostró que no existía ninguna, además de hacer solo referencia a la existencia de un camino y no a la búsqueda del óptimo, lo que conduce al mismo planteamiento de los problemas de rutas; uno de los más conocidos a lo largo de la historia es el problema de rutas de vehículo (VRP) que ha sido de gran importancia e influencia en investigaciones y estudios enfocados en implementar algoritmos que permitan encontrar una solución óptima. Este tipo de problemas son considerados difíciles de resolver y dentro de la optimización combinatoria son conocidos como problemas NP-Hard, dado que no se obtiene una solución de manera eficiente; así mismo dentro de la teoría de la complejidad computacional pertenecen a la clase NP-Completos, al no poderse garantizar hallar la mejor solución en un tiempo de cómputo razonable, ya que este aumenta de manera exponencial, generando así una búsqueda de soluciones aproximadas, siendo conveniente emplear métodos heurísticos y metaheurísticos que aplican el conocimiento del problema para acercarse a la solución en un tiempo de computo razonable.

El desarrollo de la presente tesis pretende llevar a cabo la realización de un prototipo de un algoritmo genético modificado para solucionar el problema del VRP con ventanas de tiempo y prioridad de metas, dado que algunas empresas que tienen empleados encargados del recaudo, requieren de mecanismos que consideren la optimización de los recursos disponibles para el desarrollo de las actividades asociadas al cumplimiento de rutas; para este caso concreto, se atenderán las necesidades basadas en los recursos económicos directamente relacionados con el gasto de combustible a lo largo de los recorridos, teniendo en cuenta el impacto ambiental generado por la emisión de gases (CO<sub>2</sub>) de los vehículos empleados.

## **ABSTRACT**

The origin of the problems of routing occurs in the 18th century when inhabitants of Königsberg, a town in Russia, debate about the route that passes only once through the seven bridges that crossed the river Pregel returning to the point of origin; this problem was proposed by the Swiss mathematician Leonhard Euler, who in 1736 showed that none existed, besides making only reference to the existence of a path and not to the search for the optimum, which leads to the same approach to the problems of routes; one of the best known throughout history is the problem of vehicle routes (VRP) that has been of great importance and influence in research and studies focused on implementing algorithms to find an optimal solution. These types of problems are considered difficult to solve and within combinatorial optimization are known as NP-Hard problems, since a solution is not obtained efficiently; likewise within the theory of computational complexity belong to the class NP-Completes, since it can not be guaranteed to find the best solution in a reasonable computation time, since this increases exponentially, generating a search for approximate solutions, being It is convenient to use heuristic and metaheuristic methods that apply the knowledge of the problem to approach the solution in a reasonable time.

The development of the present thesis intends to carry out the realization of a prototype of a modified genetic algorithm to solve the problem of the VRP with time windows and priority of goals, since some companies that have employees in charge of the collection, require mechanisms that consider the optimization of the available resources for the development of the activities associated with the fulfillment of routes; for this specific case, the needs based on economic resources directly related to fuel expenditure along the routes will be addressed, taking into account the environmental impact generated by the emission of gases (CO<sub>2</sub>) from the vehicles used.

## **GLOSARIO**

### **ALGORITMO**

Procedimiento computacional empleado para dar solución a un problema específico, el cual, mediante una entrada o estado inicial, realiza la ejecución de una secuencia de pasos o instrucciones con el fin de obtener una salida.

### **ALGORITMO GENÉTICO**

Tiene origen en los años 1970, de la mano de John Henry Holland y se inspiran en la teoría de la evolución biológica de Darwin, siendo un método de búsqueda que permite encontrar la solución a problemas, partiendo desde una población inicial en la cual se seleccionan los individuos más aptos o mejor capacitados para después reproducirlos y mutarlos con el fin de obtener una nueva población de individuos que estarán mejor adaptados y poseerán mejores condiciones y características que la generación anterior.

### **COMPLEJIDAD COMPUTACIONAL**

Medición de los recursos computacionales tales como, tiempo de ejecución y espacio de almacenamiento, empleados en la solución de un problema, con el fin de determinar la eficiencia y viabilidad de los algoritmos en cuanto a la optimización de costo y de tiempo.

### **COMPUTACIÓN EVOLUTIVA**

Es una rama de la inteligencia artificial que comprende las técnicas o herramientas computacionales basadas en la teoría de la evolución biológica, que tienen como enfoque la búsqueda de soluciones para problemas de optimización combinatoria.

### **CRUCE**

Operador genético que simula el proceso de intercambio de segmentos de material genético entre dos cromosomas, conocido como la reproducción sexual. Este se



realiza mediante la elección de una pareja de individuos para combinar sus características genéticas y producir una descendencia con mejores aptitudes.

## **ESTRATEGIAS EVOLUTIVAS**

Técnicas computacionales basadas en los procesos evolutivos biológicos tales como la selección, la mutación y la recombinación, empleados en la solución de problemas de optimización.

## **HEURÍSTICAS**

Los métodos heurísticos permiten resolver problemas de optimización, los cuales son procedimientos eficientes que garantizan encontrar soluciones de alta calidad con un costo computacional razonable, aunque estas soluciones no son exactas, establece un grado de optimalidad o factibilidad mayor.

## **METAHEURÍSTICAS**

Las metaheurísticas son empleadas en la búsqueda de la solución de aquellos problemas que generalmente no tienen una heurística o un algoritmo que les permita encontrar una solución adecuada, son diseñadas especialmente para aquellos problemas difíciles de resolver y que hacen parte de la optimización combinatoria. Las soluciones que se obtienen de estas son mejores que las que dan las heurísticas clásicas, aunque requieren de un tiempo de cómputo mayor. Las metaheurísticas son considerados algoritmos híbridos que permiten combinar diferentes métodos y mecanismos relacionados con la inteligencia artificial y la evolución biológica.

## **MUTACIÓN**

Operador genético que permite la preservación de la diversidad genética en una población, mediante la alteración o modificación aleatoria de los genes, evitando la convergencia prematura en la búsqueda de soluciones.

## **OPERADORES GENÉTICOS**

Procedimientos llevados a cabo dentro de los algoritmos genéticos que permiten mantener la diversidad dentro de una población, los cuales poseen similitud con el proceso evolutivo biológico. Dentro de los operadores genéticos que más se utilizan son la mutación, recombinación y cruce.

## **PROBLEMA DE RUTEO DE VEHÍCULO (VRP)**

El problema de rutas de vehículo es un problema de optimización en el cual se deben establecer un conjunto de rutas para una flota de vehículos que parten de uno o más depósitos con el fin de satisfacer la demanda de sus clientes ubicados geográficamente a diferentes distancias minimizando el costo que implica realizar esta operación.

## **SELECCIÓN**

Operador genético que basado en la teoría de Darwin, en donde se elige en una población aquellos individuos más aptos, con capacidad de adaptación y probabilidad de reproducción mayor.

## INTRODUCCIÓN

Dentro de los problemas de complejidad computacional más famosos y estudiados a lo largo de la historia se encuentra, el problema de rutas de vehículo (VRP), el cual ha sido un problema de gran importancia e influencia en la investigación en las diferentes áreas tales como la inteligencia artificial, las matemáticas, la física, la biología, la investigación de operaciones, el transporte, la genética, entre otras; es por esto que se han implementado diversos algoritmos que permitan encontrar una solución óptima a este tipo de problemas, buscando dar un aporte significativo en alguna de las áreas involucradas; debido a que estos problemas son considerados difíciles de resolver y dentro de la optimización combinatoria pertenecen a la clase NP-Hard, pues de estos no se obtiene una solución de manera eficiente; así mismo dentro de la teoría de la complejidad computacional pertenecen a la clase NP-Completos, lo que indica que no se puede garantizar encontrar la mejor solución en un tiempo de cómputo razonable, ya que este aumenta exponencialmente o en muchos casos factorialmente, generando así la búsqueda de soluciones aproximadas, para lo cual, es conveniente emplear métodos heurísticos o métodos metaheurísticos que aplican el conocimiento del problema para acercarse a la solución de este en un tiempo razonable.

Este problema se puede considerar desde dos puntos vista, el práctico y el teórico, en el primero estos no están resueltos y en el segundo las técnicas que se han ido empleando son solo aproximaciones suficientemente aceptables, pero no son una solución real de estos problemas. Buscando estas posibles soluciones se han empleado los algoritmos genéticos, que, simulando la evolución biológica, han basado su configuración en determinados aspectos e incluyendo distintas variaciones con el fin de encontrar una solución con un nivel de eficiencia y optimización mayor.

Los algoritmos genéticos a través de los años han mostrado que tienen la capacidad de encontrar soluciones óptimas para el problema de rutas de vehículo (VRP), muchas de las investigaciones que se conocen en la actualidad han propuesto

diversas formas de implementarlos, con el fin de encontrar la mejor solución o la óptima respuesta, por esta razón las implementaciones han ido más allá y muchos de estos estudios se han dirigido hacia el desarrollo de algoritmos híbridos que en conjunto con otras metodologías permitan que la búsqueda de estas soluciones sea más eficiente.

En el presente trabajo se desarrolla la implementación del algoritmo genético el cual presenta un proceso evolutivo manteniendo durante todo el proceso la diversidad de los individuos presentes en la población. La implementación del algoritmo genético se desarrollará de forma clásica llevando a cabo los procesos de selección, en el cual se determinan los padres aptos para pasar a la siguiente generación; de recombinación, en donde los cromosomas de los padres se comparten para generar nuevos individuos candidatos y así determinar cuál de los nuevos individuos sigue en el proceso, teniendo en cuenta que posea la mejor función objetivo; y finalmente la mutación en donde se controla la factibilidad nuevamente y se emplea una tasa que es ajustable dentro de los parámetros generales del algoritmo con el fin de alterar alguno de los alelos de forma aleatoria.

# **CAPITULO 1. PLANTEAMIENTO DEL PROBLEMA**

## **1.1 Definición Del Problema**

Los algoritmos genéticos a través de los años han mostrado que tienen la capacidad de encontrar soluciones óptimas para los problemas del agente viajero (TSP) y su variante el problema de ruteo (VRP) muchas de las investigaciones que se conocen hasta hoy han propuesto diversas formas de implementarlos, con el fin de encontrar la mejor solución o la óptima respuesta con un alto nivel de facilidad, por esta razón las implementaciones han ido más allá y muchos de estos estudios se han dirigido hacia el desarrollo de algoritmos híbridos que en conjunto con otras metodologías permitan que la búsqueda de estas soluciones sea más eficiente.

El problema de ruteo de vehículos es un problema de optimización combinatoria computacional que a pesar de tener un planteamiento simple ha sido de gran importancia, no solo por la complejidad de su resolución en comparación con otros problemas, sino por la gran variedad de aplicaciones que este posee; siendo un reto para investigadores desde hace varios años, debido a que no se ha encontrado ningún algoritmo que lo resuelva en un tiempo polinómico, pues el nivel de complejidad está ligado a la cantidad de nodos estimados y a las diferentes restricciones que se puedan apreciar, como la ubicación de los nodos, las ventanas de tiempo, el tipo de servicio, entre otros factores, que debido al contexto del problema tratado pueden variar.

En este trabajo se lleva a cabo la realización de un prototipo de un algoritmo genético modificado para solucionar el problema de ruteo con ventanas de tiempo (VRPTW) y prioridad de metas, dado que algunas empresas que tienen empleados encargados del recaudo, requieren de mecanismos que consideren la optimización de los recursos disponibles para el desarrollo de las actividades asociadas al cumplimiento de rutas; en el caso concreto de esta tesis, se atenderán las necesidades basadas en los recursos económicos directamente relacionados con el gasto de combustible a lo largo de los recorridos, así como también se analizará

y calculará la emisión de gases generado por los vehículos utilizados en los recorridos.

Se espera solucionar este problema mediante la implementación de un algoritmo genético, siguiendo la línea de inteligencia artificial y desarrollándolo de forma clásica, llevando a cabo los procesos de selección, en el cual se determinan los padres aptos para pasar a la siguiente generación; de recombinación, en donde los genes de los padres se comparten para generar nuevos individuos candidatos y así determinar cuál de estos sigue en el proceso, teniendo en cuenta que posea la mejor función objetivo; y finalmente la mutación en donde se controla la factibilidad nuevamente y se emplea una tasa que es ajustable dentro de los parámetros generales del algoritmo con el fin de alterar alguno de los alelos de forma aleatoria. Los procesos anteriormente descritos son aplicados en la generación de rutas óptimas que cumplan con las condiciones de tiempo y prioridad de metas económicas.

## 1.2 Justificación

El origen de los problemas de ruteo se da en el siglo XVIII cuando habitantes de Königsberg, un pueblo de Rusia, debate sobre cuál es la ruta que pasa una sola vez por los siete puentes que atravesaban el río Pregel regresando al punto de origen; este problema lo propuso el matemático suizo Leonhard Euler, quien en el año 1736 en un artículo demostró que no existía ninguna, además de hacer solo referencia a la existencia de un camino y no a la búsqueda del óptimo, lo que conduce al mismo planteamiento de los problemas de rutas; uno de estos, conocido en la Universidad de Princeton entre los años 1931 y 1932 bajo el nombre del “problema del agente viajero”, en donde su planteamiento será aplicado en la asignación de rutas a los empleados encargados del recaudo de dinero de algunas empresas, teniendo en cuenta factores como la localización de los clientes a visitar y los horarios en que se deben realizar dichas visitas; buscando que los recorridos de los empleados sean óptimos y el consumo de combustible sea menor (Dantzig, Fulkerson, & Johnson, Solution of a Large-Scale Traveling-Salesman Problem, 1954).

El problema del agente viajero (TSP) y el problema de ruteo de vehículos (VRP) han sido gran importancia e influencia en la investigación en las diferentes áreas, es por esto que se han implementado diversos algoritmos que permitan encontrar una solución óptima a estos problemas, buscando dar un aporte significativo; debido a que estos problemas son considerados difíciles de resolver y dentro de la optimización combinatoria son conocidos como problemas NP-Hard, pues de estos no se obtiene una solución de manera eficiente; así mismo dentro de la teoría de la complejidad computacional pertenecen a la clase NP-Complejos, lo que indica que no se puede garantizar encontrar la mejor solución en un tiempo de cómputo razonable, ya que este aumenta exponencialmente, generando así la búsqueda de soluciones aproximadas, para lo cual, es conveniente emplear algoritmos genéticos o métodos heurísticos y metaheurísticos que aplican el conocimiento del problema para acercarse a la solución de este en un tiempo razonable.

El estudio de este tipo de problemas permite que las empresas puedan optimizar no solo el tiempo que demoran en realizar sus empleados los diferentes recorridos,

sino además cumplir con las metas económicas establecidas, reflejadas de diversas formas, una de ellas representada en la disminución del consumo de combustible de los vehículos en el recorrido, lo que generaría no solo una disminución en los recursos económicos de la compañía sino también poder generar un impacto medioambiental positivo, mediante el cálculo de la emisión de gases de los vehículos empleador, además de permitir que se ofrezca un servicio oportuno y que se pueda tener control de cada una de las labores relacionadas con el recaudo.

Considerando lo anterior y conociendo la situación actual de estos problemas, en el presente trabajo se realizará el prototipado de un algoritmo genético, que permita un proceso evolutivo altamente eficiente, para contribuir en la búsqueda de soluciones más óptimas que puedan ser de gran utilidad en diversos ámbitos, como la industria y el comercio, en donde el transporte implica un alto porcentaje del valor agregado en los bienes, generando así ganancias significativas para las empresas. Con la implementación de este algoritmo se pretende ampliar las posibilidades en la búsqueda de la solución óptima al problema al problema de ruteo de vehículos (VRP).



## **CAPITULO 2. MARCO TEÓRICO**

En este capítulo se presentan las teorías que se relacionan con el problema de investigación, teniendo como objetivo ofrecer sustento al estudio que se está desarrollando en el presente trabajo.

### **2.1 Complejidad Computacional**

El principio de los estudios sistemáticos en la complejidad computacional se atribuye al trabajo realizado en 1965 "Sobre la complejidad computacional de algoritmos" por Juris Hartmanis y Richard E. Stearns, que estableció las definiciones de la complejidad del tiempo y la complejidad del espacio, y demostró los teoremas de la jerarquía. Además, en 1965 Edmonds sugirió considerar un algoritmo "bueno" para ser uno con tiempo de ejecución limitado por un polinomio del tamaño de entrada (Karp, 1986).

La complejidad computacional consiste en la medición de los recursos computacionales (tales como, tiempo de ejecución y espacio de almacenamiento), que son empleados en la solución de un problema, con el fin de determinar la eficiencia de los algoritmos considerando la viabilidad de este en cuanto a la optimización de costo y de tiempo.

La teoría de la complejidad propone un criterio formal para lo que significa que un problema matemático sea factiblemente decidible, es decir, que puede ser resuelto por una máquina de Turing convencional en una serie de pasos que es proporcional a una función polinómica del tamaño de su entrada. La clase de problemas con esta propiedad se conoce como P - o tiempo polinomial - e incluye el primero de los tres problemas descritos anteriormente. P se puede demostrar formalmente que es distinta de ciertas otras clases tales como EXP - o tiempo exponencial - que incluye el tercer problema de arriba. El segundo problema pertenece a una clase de complejidad conocida como NP- o tiempo polinomial no determinista - que consiste en aquellos problemas que pueden ser correctamente decididos por algún cálculo

de una máquina de Turing no determinista en una serie de pasos que es una función polinomial del tamaño de su entrada. Una conjetura famosa - a menudo considerada como la más fundamental en toda la informática teórica - afirma que P también está correctamente contenido en NP - es decir,  $P \subseteq NP$  (Dean, 2018).

## 2.2 Algoritmo Genético

Un algoritmo genético es un método de búsqueda que permite encontrar la solución a problemas, partiendo desde una población inicial en la cual se seleccionan los individuos más aptos o mejor capacitados para después reproducirlos y mutarlos con el fin de obtener una nueva población de individuos que estarán mejor adaptados y poseerán mejores condiciones y características que la generación anterior.

Los algoritmos genéticos (GAs) fueron inventados por John Holland en la década de 1960 y fueron desarrollados por Holanda y sus estudiantes y colegas en la Universidad de Michigan en los años 1960 y 1970. En contraste con las estrategias de evolución y la programación evolutiva, el objetivo original de Holanda no era diseñar algoritmos para resolver problemas específicos, sino más bien estudiar formalmente el fenómeno de adaptación tal como ocurre en la naturaleza y desarrollar formas de importar los mecanismos de adaptación natural en sistemas informáticos. En el año 1975 el libro de Holland, "Adaptation in Natural and Artificial Systems" presentó el algoritmo genético como una abstracción de la evolución biológica y dio un marco teórico.

GA es un método para pasar de una población de "cromosomas" (por ejemplo, cadenas de unos y ceros, o "bits") a una nueva población mediante el uso de una especie de "selección natural" junto con los operadores de genética de crossover, mutación e inversión. Cada cromosoma consta de "genes" (por ejemplo, bits), siendo cada gen una instancia de un "alelo" particular (por ejemplo, 0 ó 1). El operador de selección elige aquellos cromosomas en la población que se permitirá reproducir, y en promedio, los cromosomas más ajustados producen más

descendencia que los menos aptos. Crossover intercambia sub-partes de dos cromosomas, imitando aproximadamente la recombinación biológica entre dos organismos monocromosómicos ("haploides"); la mutación aleatoriamente cambia los valores del alelo de algunas localizaciones en el cromosoma. La introducción de Holland de un algoritmo basado en la población con crossover, inversión y mutación fue una gran innovación. (Las estrategias de evolución de Rechenberg comenzaron con una "población" de dos individuos, uno de los progenitores y uno de los descendientes, siendo la descendencia una versión mutada del padre, las poblaciones individuales y el crossover no se incorporó hasta más tarde) Además, Holland fue el primero en intentar poner la evolución computacional en una base teórica firme (Holland 1975). Hasta hace poco esta fundamentación teórica, basada en la noción de "esquemas", fue la base de casi todos los trabajos teóricos subsiguientes sobre algoritmos genéticos (Mitchell, 2002).

### 2.3 Operadores Genéticos

Los operadores genéticos son utilizados para guiar a los algoritmos genéticos a alcanzar una óptima solución, dichos operadores fueron creados con la idea de crear y conservar la diversidad genética, combinar soluciones existentes y a partir de estas nuevas soluciones, seleccionar la más apropiada.

Los operadores genéticos son (Estévez, 1997):

- **Selección:** El mecanismo de selección permite orientar la búsqueda a aquellos puntos más promisorios, con la mayor adaptación observada hasta el momento. El operador de selección genera a partir de la población actual una población intermedia del mismo tamaño, evaluando de acuerdo a la función objetivo (o de aptitud), reproduciendo con un mayor número de copias a los individuos más aptos y eliminando o asignando un menor número de copias a los individuos menos aptos. El operador de selección no produce puntos nuevos en el espacio de búsqueda, sino que determina qué individuos dejarán descendencia y en qué cantidad en la próxima generación.

Entre los operadores de selección más utilizados se encuentran:

- **Selección por torneo:** En este método se realizan comparaciones directas entre individuos elegidos al azar de la población, dichas comparaciones se denominan torneos, el ganador de cada torneo es el individuo que tenga un mayor valor de aptitud y es el que pasa a la recombinación. Se debe elegir el tamaño de los torneos a realizar, entre más grande el valor configurado, los individuos débiles tienen una menor posibilidad de ser seleccionados. El anterior método es denominado selección por torneo determinístico. Si, por el contrario, el ganador del torneo es seleccionado aleatoriamente al generar un número en el intervalo de  $[0,1]$  y se evalúa si es mayor que el parámetro  $p$  seleccionado, entonces se escoge el individuo con mayor aptitud y en el caso contrario, se selecciona el menos apto, este método es llamado selección por torneo probabilístico.
- **Selección por ruleta:** Es un método utilizado para seleccionar las soluciones potencialmente útiles para la recombinación. Cada posible solución es evaluada de acuerdo con la función objetivo, en la cual se le asigna un valor de aptitud y de acuerdo a este valor se otorga un tamaño proporcional en la ruleta, para esto, se debe normalizar la aptitud haciendo la división entre el valor de un individuo y la suma de la aptitud de todas las selecciones. Para seleccionar un individuo se genera un valor aleatorio y se devuelve el individuo situado en esa posición.
- **Selección Universal Estocástica:** Algoritmo de muestreo que se implementa en una sola fase, conocido como SUS, por sus siglas en inglés (Stochastic Universal Sampling). Dado un conjunto de  $n$  individuos y sus valores objetivos asociados, SUS los acomoda en una ruleta donde el tamaño de los cortes asignados a cada individuo es proporcional al valor objetivo (como en el algoritmo de ruleta). Después, una segunda ruleta, es marcada con *y marcadores*

igualmente espaciados entre sí, donde  $y$  es el número de selecciones que deseamos efectuar. Por último, se gira la ruleta y se selecciona un individuo por cada marcador. Las posiciones de los marcadores indican los individuos seleccionados. Si  $n$  marcadores caen sobre el mismo individuo, éste es seleccionado  $n$  veces. Esto garantiza que ningún individuo sea seleccionado ni más ni menos veces que las esperadas (Rodríguez Muro, 2005).

- **Recombinación:** El operador de recombinación (crossover) es el operador de búsqueda más importante en los algoritmos genéticos. Este es un operador sexuado que intercambia el material genético de un par de padres produciendo descendientes que normalmente difieren de sus padres. La idea central es que segmentos distintos de padres diferentes con alta adaptación deberían combinarse en nuevos individuos que tomen ventaja de esta combinación.

El algoritmo genético explota las regiones con mayor adaptación, ya que generaciones sucesivas de selección y recombinación producen un número creciente de puntos en estas regiones.

Los operadores de recombinación más utilizados son:

- Cruce de 1 punto (Single Point Crossover): Se escoge un punto de corte aleatoriamente en cada una de las cadenas de cromosomas padres para formar dos subcadenas en cada una, una a la izquierda del punto de corte y otra a la derecha. Se une la subcadena izquierda de un padre con la subcadena derecha del otro para formar la cadena hija. De manera similar se genera el segundo hijo, pero con las subcadenas restantes.
- Cruce de 2 puntos (Double Point Crossover): En vez de elegir un único corte como el método anterior, se eligen dos puntos de corte, evitando siempre que uno de los puntos sea el final del cromosoma. Para

obtener los hijos resultantes, se toman los segmentos laterales de un padre y se mezclan con el segmento central del otro padre.

- **Mutación:** Existen diferentes métodos de mutación desde una simple mutación de bits hasta el reemplazo de un gen con valores aleatorios. Las tasas de mutación pequeñas garantizan que un individuo no difiera mucho de sus padres en el genotipo. La mutación sirve para evitar la pérdida de diversidad genética entre las soluciones e intenta evitar que el algoritmo converja a un mínimo local al detener las soluciones que se acercan demasiado entre sí.

Los tipos de mutación más usados son (Moratilla, Fernández, Sánchez, & Vicario, 2014):

- Mutación basada en el desplazamiento (DM): Se selecciona una subcadena al azar del cromosoma y se inserta en un lugar aleatorio.
- Mutación basada en cambios (EM): Se seleccionan dos genes del cromosoma al azar y se cambian.
- Mutación basada en la inserción (ISM): Se elige aleatoriamente un gen, a continuación, se saca del cromosoma y se inserta nuevamente en un lugar al azar.
- Mutación basada en la inversión simple (SIM): Se selecciona aleatoriamente dos puntos de corte en el cromosoma, siguiente a esta elección, se revierte los genes situados entre dichos cortes.

## 2.4 Computación Evolutiva

La computación evolutiva es una familia de algoritmos de optimización inspirados en la evolución biológica y en la inteligencia artificial. En estos algoritmos se generan un conjunto inicial de soluciones candidatas y se actualizan iterativamente. Cada nueva generación se produce mediante la eliminación estocástica de las soluciones menos deseadas y la introducción de pequeños cambios aleatorios. En términos biológicos, una población de soluciones se somete a una selección natural (o artificial) y seguido de esto, se realiza una mutación de las soluciones elegidas, obteniendo como resultado que dicha población evolucionará gradualmente para aumentar la aptitud (Bäck, Fogel, & Michalewics, 1997).

Entre las técnicas más utilizadas de la computación evolutiva se encuentran (en su mayoría) algoritmos de optimización metaheurísticos como:

- Optimización de Colonias de Hormigas.
- Algoritmo de Colonias de Abejas Artificiales.
- Evolución de dos fases.
- Optimización de enjambre de partículas.

## 2.5 Estrategias Evolutivas

Es una técnica de optimización basada en las ideas de evolución, perteneciente a la rama de la computación evolutiva. Las estrategias de evolución utilizan representaciones naturales dependientes del problema, y principalmente utilizan mutación y selección, como operadores de búsqueda. En común con los algoritmos evolutivos, los operadores se aplican en un ciclo. Una iteración del ciclo se llama generación. La secuencia de generaciones se continúa hasta que se cumple un criterio de finalización.

En cuanto a los espacios de búsqueda de valores reales, la mutación se realiza normalmente añadiendo un valor aleatorio normalmente distribuido. La selección usada en las estrategias de evolución es determinista y sólo se basa en los rankings de fitness, no en los valores reales de aptitud física. El algoritmo resultante es, por tanto, invariante con respecto a las transformaciones monótonas de la función objetivo. La estrategia de evolución más simple opera sobre una población de tamaño dos: el punto actual (padre) y el resultado de su mutación. Sólo si la condición física del mutante es al menos tan buena como la del padre, se convierte en la madre de la siguiente generación (Bartz-Beielstein, 2003).

## **2.6 Programación Evolutiva**

La programación evolutiva es uno de los cuatro principales paradigmas de algoritmos evolutivos. Es similar a la programación genética, pero la estructura del programa a optimizar es fija, mientras que sus parámetros numéricos se permiten evolucionar.

La motivación original de Lawrence J. Fogel para la programación evolutiva se centró en generar un enfoque alternativo a la inteligencia artificial. En lugar de emular o simular a los seres humanos, ya sea en su estructura neurofisiológica o sus comportamientos particulares, la evolución natural fue modelada como un proceso que genera organismos de intelecto creciente a través del tiempo. La inteligencia se definió como la capacidad de un organismo para alcanzar metas en un rango de ambientes y se consideraba que la conducta inteligente requería la capacidad de predecir sucesos ambientales futuros junto con una traducción de esas predicciones en respuestas adecuadas (Fogel & Fogel, 1995).



## 2.7 Métodos Exactos

Son aquellos que parten de una formulación como modelos de programación lineal (enteros) o similares, y llegan a una solución factible (entera) gracias a algoritmos de acotamiento del conjunto de soluciones factibles.

## 2.8 Heurísticas

Un heurístico es un “procedimiento simple, a menudo basado en el sentido común, que se supone que ofrecerá una buena solución (aunque no necesariamente la óptima) a problemas difíciles, de un modo fácil y rápido” (Zanakis & Evans, 1981).

Los métodos heurísticos pueden ser utilizados cuando en un problema de optimización, determinístico o no, se encuentren algunas de las siguientes características (Martí, 2003):

- El problema es de una naturaleza tal que no se conoce ningún método exacto para su resolución.
- Aunque exista un método exacto para resolver el problema, su uso es computacionalmente muy costoso o inviable.
- El método heurístico es más flexible que un método exacto, permitiendo, por ejemplo, la incorporación de condiciones de difícil modelización.
- El modelo matemático es demasiado grande, demasiado NO lineal o demasiado complejo desde el punto de vista lógico.
- El asumir suposiciones o aproximaciones para simplificar el problema, tiende a destruir estructuras del modelo que son vitales en el contexto del mundo real, haciendo la solución no viable.

Existen métodos heurísticos (también llamados algoritmos aproximados, procedimientos inexactos, algoritmos basados en el conocimiento o simplemente heurísticas) de diversa naturaleza, por lo que su clasificación es bastante complicada. Se plantea la siguiente clasificación (Rodríguez Ortiz, 2010):

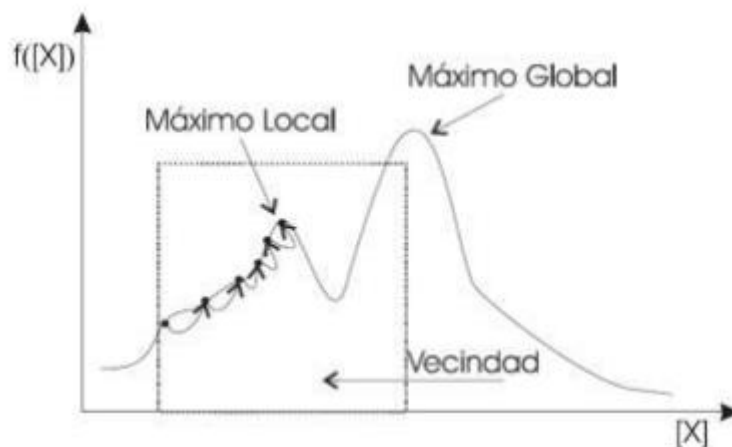
**Métodos constructivos:** Procedimientos que son capaces de construir una solución a un problema dado. La forma de construir la solución depende fuertemente de la estrategia seguida. Las estrategias más comunes son:

- **Estrategia voraz:** Partiendo de una semilla, se va construyendo paso a paso una solución factible. En cada paso se añade un elemento constituyente de dicha solución, que se caracteriza por ser el que produce una mejora más elevada en la solución parcial para ese paso concreto. Este tipo de algoritmos se dice que tienen una visión "miope" ya que eligen la mejor opción actual sin que les importe que ocurrirá en el futuro.
- **Estrategia de descomposición:** Se divide sistemáticamente el problema en subproblemas más pequeños. Este proceso se repite (generalmente de forma recursiva) hasta que se tenga un tamaño de problema en el que la solución a dicho subproblema es trivial. Después el algoritmo combina las soluciones obtenidas hasta que se tenga la solución al problema original. Los algoritmos más representativos de los métodos de descomposición son los algoritmos de divide y vencerás tanto en su versión exacta como aproximada.
- **Métodos de reducción:** Identifican características que contienen las soluciones buenas conocidas y se asume que la solución óptima también las tendrá. De esta forma se puede reducir drásticamente el espacio de búsqueda.
- **Métodos de manipulación del modelo:** Consisten en simplificar el modelo del problema original para obtener una solución al problema simplificado. A partir de esta solución aproximada, se extrapola la solución al problema original. Entre estos métodos se pueden destacar: la linealización, la agrupación de variables, introducción de nuevas restricciones, etc.

**Métodos de búsqueda:** Parten de una solución factible dada y a partir de ella intentan mejorarla. Algunos son:

- **Estrategia de búsqueda local 1:** Parte de una solución factible que la mejora progresivamente. Para ello examina su vecindad y selecciona el primer movimiento que produce una mejora en la solución actual (first improvement)
- **Estrategia de búsqueda local 2:** Parte de una solución factible que la mejora progresivamente. Para ello examina su vecindad y todos los posibles movimientos seleccionando el mejor movimiento de todos los posibles, es decir aquel que produzca un incremento (en el caso de maximización) más elevado en la función objetivo (best improvement).
- **Estrategia aleatorizada:** Para una solución factible dada y una vecindad asociada a esa solución, se seleccionan aleatoriamente soluciones vecinas de esa vecindad.

El principal problema que presentan los algoritmos heurísticos es su incapacidad para escapar de los óptimos locales. En la Figura 3.1 se muestra como para una vecindad dada el algoritmo heurístico basado en un método búsqueda local se quedaría atrapado en un máximo local.



**Ilustración 1: Algoritmo heurístico basado en búsqueda local.**

*Fuente: Rodríguez Ortiz, C. Algoritmos heurísticos y metaheurísticos para el problema de localización de regeneradores. Universidad Rey Juan Carlos.*

Cuando se resuelve un problema por métodos heurísticos, como la optimalidad no está garantizada se debe de medir la calidad de los resultados, evaluando la eficiencia del algoritmo para poder determinar su valía frente a otros. Para tal fin un buen algoritmo heurístico debe de tener las siguientes propiedades (Suarez, 2013):

- **Ser eficiente.** Que el esfuerzo computacional sea realista y adecuado para obtener la solución.
- **Ser bueno.** La solución debe de estar, en promedio, cerca del óptimo.
- **Ser robusto.** La probabilidad de obtener una mala solución (lejos del óptimo) debe ser baja.

## 2.9 Metaheurísticas

El término metaheurística o meta-heurística fue introducido por F. Glover en el año 1986. Con este término, pretendía definir un “procedimiento maestro de alto nivel que guía y modifica otras heurísticas para explorar soluciones más allá de la simple optimalidad local”.

La idea básica general de las metaheurísticas es enriquecer a los algoritmos heurísticos de forma que éstos puedan escapar de los óptimos locales.

Todas las técnicas metaheurísticas tienen las siguientes características (Sait & Youssef, 2000):

- Son ciegas, no saben si llegan a la solución óptima. Por lo tanto, se les debe indicar cuándo deben detenerse.
- Son algoritmos aproximativos y, por lo tanto, no garantizan la obtención de la solución óptima.
- Aceptan ocasionalmente malos movimientos (es decir, se trata de procesos de búsqueda en los que cada nueva solución no es necesariamente mejor – en términos de la función objetivo– que la inmediatamente anterior). Algunas

veces aceptan, incluso, soluciones no factibles como paso intermedio para acceder a nuevas regiones no exploradas.

- Son relativamente sencillos; todo lo que se necesita es una representación adecuada del espacio de soluciones, una solución inicial (o un conjunto de ellas) y un mecanismo para explorar el campo de soluciones.
- Son generales. Prácticamente se pueden aplicar en la resolución de cualquier problema de optimización de carácter combinatorio. Sin embargo, la definición de la técnica será más o menos eficiente en la medida en que las operaciones tengan relación con el problema considerado.

Dado que las metaheurísticas son estrategias para diseñar y/o mejorar los procedimientos heurísticos, el tipo de metaheurística estará en función de qué tipo de heurística se pretende mejorar; sin embargo, se pueden clasificar en (Melían, Pérez, & Vega, 2003):

- **Metaheurísticas Inspiradas en la física:** Recocido Simulado (Simulated Annealing). – Esta técnica se considera como un algoritmo de búsqueda. Está inspirado en el proceso de calentamiento y posterior enfriamiento de un metal, para obtener estados de baja energía en un sólido.
- **Metaheurísticas Inspiradas en la evolución:** Son métodos que van construyendo un conjunto de soluciones a diferencia de los otros métodos que sólo pasan de una solución a otra en cada iteración. El procedimiento consiste en generar, seleccionar, combinar y reemplazar un conjunto de soluciones en la búsqueda de la mejor solución (Díaz & González, 1996). Un ejemplo de metaheurística evolutiva son los Algoritmos Genéticos; el investigador de la Universidad de Michigan John Holland a fines de los 60s desarrolló esta técnica que permitió incorporarla a un programa, y su objetivo era lograr que las computadoras aprendieran por sí mismas. A la técnica que inventó Holland se le llamó originalmente “planes reproductivos”, pero se hizo popular bajo el nombre “algoritmo genético” tras la publicación de su libro en 1975. Esta técnica se basa en los mecanismos de selección que utiliza la

naturaleza, de acuerdo a los cuales los individuos más aptos de una población son los que sobreviven, al adaptarse más fácilmente a los cambios que se producen en su entorno. Hoy en día se sabe que estos cambios se efectúan en los genes de un individuo (unidad básica de codificación de cada uno de los atributos de un ser vivo), y que sus atributos más deseables (por ejemplo, los que le permiten adaptarse mejor a su entorno) se transmiten a sus descendientes cuando éste se reproduce sexualmente.

- **Metaheurísticas Inspiradas en la biología:** Una metaheurística bioinspirada relativamente reciente es la Optimización basada en Colonias de Hormigas (OCH) (“Ant Colony Optimization”, ACO en inglés), Metaheurística que se inspira en el comportamiento estructurado de las colonias de hormigas donde individuos muy simples de una colonia se comunican entre sí por medio de una sustancia química denominada feromona, la repetición de recorridos por los individuos establece el camino más adecuado entre su nido y su fuente de alimentos.

## 2.10 Problema Del Agente Viajero (TSP)

El problema del agente viajero plantea la siguiente pregunta: “Dada una lista de ciudades y las distancias entre cada par de ciudades, ¿Cuál es la ruta más corta posible en la cual el viajero visite cada ciudad exactamente una vez y regresa a la ciudad de origen?”. Este problema ha causado gran interés a matemáticos e informáticos puesto que el problema es fácil de describir, pero difícil de resolver (Hoffman, Padberg, & Rinaldi, 2001).

La importancia del TSP radica en que hace parte de un conjunto más grande de problemas conocidos como problemas de optimización combinatoria. El problema TSP pertenece a la clase de problemas conocidos como NP-Complejos.

El problema se formuló por primera vez en 1930 y es uno de los problemas más intensamente estudiados en la optimización. Se utiliza como punto de referencia para muchos métodos de optimización. Aunque el problema es computacionalmente difícil, se conoce un gran número de heurísticas y algoritmos

exactos, de modo que algunos casos con decenas de miles de ciudades pueden ser resueltos completamente e incluso los problemas con millones de ciudades pueden ser aproximados dentro de una pequeña fracción del 1%.

El primer paso para resolver grandes casos de TSP es encontrar una buena formulación matemática del problema. En el caso del problema del agente viajero, la estructura matemática es un grafo en el que cada ciudad se denomina un nodo y se dibujan líneas que conectan cada dos nodos (llamados arcos o bordes). Asociado con cada línea es una distancia (o costo). Cuando el viajero puede obtener una ruta de cada ciudad a otra ciudad directamente, entonces se dice que el grafo es completo. Un viaje de ida y vuelta de las ciudades corresponde a un subconjunto de las líneas, y se llama un tour o un ciclo hamiltoniano en la teoría de los grafos. La longitud de un recorrido es la suma de las longitudes de las líneas en la ida y vuelta.

## **2.11 El Problema De Enrutamiento De Vehículos (VRP)**

El problema de enrutamiento de vehículos puede describirse como el problema de diseñar rutas óptimas de entrega o recolección de uno o varios depósitos a un número de ciudades o clientes geográficamente dispersos, los cuales están sujetos a restricciones (Laporte, 1992). El VRP es un problema de programación de optimización combinatoria y entera que realiza la siguiente pregunta "¿Cuál es el conjunto óptimo de rutas que una flota de vehículos debe recorrer para entregar a un conjunto dado de clientes?". Este problema generaliza el conocido problema del agente viajero (TSP). La primera aparición de este concepto fue en un documento de George Dantzig y John Ramser en 1959 (Dantzig & Ramser, 1959), en el cual se modela el despacho de combustible a través de una flota de camiones a diferentes estaciones de servicio, desde una terminal. A menudo, el contexto es el de entregar mercancías ubicadas en un depósito central a los clientes que han hecho pedidos para tales bienes.

El objetivo del VRP es minimizar el costo total de la ruta. Este problema es uno de los más comunes en la optimización de operaciones logísticas y uno de los más

estudiados; puesto que plantea la búsqueda de la solución óptima con diferentes restricciones tales como: número de vehículos, su capacidad, lugares de destino (clientes) y demanda de los clientes, entre otras. Una formulación de éste tipo puede incluir un amplio número de variables y diversos parámetros (Rocha, González, & Orjuela, 2011).

Determinar la solución óptima de un VRP es NP-hard, por lo que el tamaño de los problemas que se pueden resolver, de manera óptima, utilizando la programación matemática o la optimización combinatoria puede ser limitado, además, no es posible resolverlos en un tiempo polinomial. El tiempo y esfuerzo computacional requerido para resolver este problema aumenta exponencialmente respecto al tamaño del problema, es decir, la cantidad de nodos a ser visitados por los vehículos.

Para este tipo de problemas es a menudo deseable obtener soluciones aproximadas, para que puedan ser encontradas suficientemente rápido y que sean suficientemente buenas para llegar a ser útiles en la toma de decisiones. Por esto se han ideado algoritmos que no garantizan optimalidad, pero que logran entregar buenas soluciones a estos problemas difíciles de resolver. Estos son los algoritmos heurísticos que conforman una clase de métodos muy extensa y taxonómicamente compleja, que en la última década han tenido un gran éxito resolviendo problemas pertenecientes a la clase NP-Hard.

## **2.12 Problema con restricciones de Capacidad (CVRP)**

El problema CVRP (*Capacited VRP*) es una variante del VRP que consiste en que dadas  $n$  ciudades, en donde se cuenta con un depósito único y una flota de vehículos que emplean habitualmente capacidad fija y que deben prestar un servicio al número de clientes desde el depósito central a un costo mínimo, se debe encontrar los recorridos de los vehículos de tal manera que se minimice la distancia y se satisfaga la demanda de los cliente cumpliendo con la restricción de no exceder la capacidad del vehículo.



Este tipo de problema es considerado como una generalización del TSP, teniendo en cuenta que el primer problema planteado tipo VRP, fue el agente viajero o TSP (Travelling Salesman Problem) por Flood en el año 1956. En el artículo “*Una revisión al estado del arte del problema de ruteo de vehículos: Evolución histórica y métodos de solución*” (Rocha, González, & Orjuela, 2011) se define este problema como el problema de los  $m$  agentes viajeros puede ser asumido como un VRP y aún más allá, como un CVRP (Capacited VRP), es decir como un problema de ruteo de vehículos donde la capacidad de la flota se convierte en restrictiva para la formulación. La función objetivo del CVRP minimiza el costo total para surtir a todos los consumidores. En este tipo de problema, a los consumidores les corresponde una demanda determinística, todos los vehículos son iguales y salen de un centro de distribución.

El CVRP encuentra una cantidad exacta de rutas con mínimo costo definido como la suma de los costos de los arcos pertenecientes a los recorridos, de tal manera que cada recorrido visita el centro de distribución, cada centro de consumo es visitado por exactamente un vehículo y la suma de las demandas de los centros de consumo visitados no excede la capacidad del vehículo. Cuando el costo de ir de un centro de consumo  $i$  a otro centro de consumo  $j$  es igual al costo de ir del centro de consumo  $j$  al centro de consumo  $i$  el problema es llamado CVRP Simétrico (Symmetric CVRP, SCVRP) en caso contrario se denomina CVRP Asimétrico (Asymmetric CVRP, ACVRP).

### **2.13 Problema con múltiples depósitos (MDVRP)**

El problema con múltiples depósitos conocido como MDVRP (por sus siglas en inglés, Multiple Depot VRP), tiene como objetivo brindar un servicio a todos los clientes minimizando el número de vehículos y la distancia recorrida por ellos. En este problema se tiene la variante de que existen varios depósitos desde donde se puede ofrecer el servicio a los clientes, cada uno de estos dispone de su propia flota de vehículos, los cuales deben salir del depósito realizar la ruta correspondiente y regresar a su depósito de origen.

El MDVRP se puede ver como un problema de agrupación en el sentido de que la salida es un conjunto de horarios de vehículos agrupados por depósito. Esta interpretación sugiere una clase de enfoque que agrupa a los clientes y luego programa los vehículos sobre cada grupo. El MDVRP se puede resolver en dos etapas: en primer lugar, los clientes deben ser asignados (asignados) a los depósitos; entonces se deben construir rutas que vinculen a los clientes asignados al mismo depósito. Idealmente, es más eficiente tratar los dos pasos simultáneamente. Sin embargo, cuando se enfrentan a problemas mayores, este enfoque ya no es manejable computacionalmente. Un enfoque razonable sería dividir el problema en tantos sub-problemas como hay depósitos y resolver cada sub-problema por separado (Tansini, Urguhart, & Viera, 2001).

## **2.14 Problema periódico (PVRP)**

El PVRP (Periodic VRP) tiene como objetivo minimizar la flota de vehículos y los tiempos de recorrido empleados para brindarles el servicio a los clientes. Su planteamiento consiste en establecer un horizonte de operación  $M$  días, en este periodo se debe visitar al menos una vez a cada cliente.

En los VRP clásicos, típicamente el periodo de planeación de las rutas es de un solo día, en el caso del PVRP el modelo es extendido a un período de planeación de  $M$  días.

El PVRP fue descrito por primera vez por Beltrami y Bodin (1974), donde consideraron dos aspectos del problema:

1. Desarrollar rutas que fueron asignadas a los días de entrega.
2. Asignar a los clientes los días de entrega y la ruta cada día por separado.

Tan & Beasley (1984) dieron una definición formal y propusieron tres enfoques:

- Asignación del cliente a los días de entrega mediante un algoritmo de agrupación en clúster, en el que los clústeres de cada día se formaron alrededor de los clientes con una única combinación de entrega permisible.

- Una adaptación de la ruta heurística de transporte de un solo día MTOUR de Russell, que se basa en la heurística de vendedor ambulante de Lin y Kernighan.
- Una adaptación de la heurística de enrutamiento de un solo día de Clarke y Wright.

Christofides y Beasley (1984) presentaron una formulación exacta de programación entera del PVRP. Para resolver esta formulación, propusieron un método de solución heurística que implicaba dos pasos: inicialización y mejora. Esta formulación es un programa cero-uno complejo con muchas variables, que difícilmente puede utilizarse para resolver grandes instancias.

Se basa en una elección inicial de los días de entrega del cliente para cumplir con los requisitos de nivel de servicio, seguido por un procedimiento de intercambio para mejorar la elección de los días de entrega. Sus algoritmos reemplazaron la VRP tradicional, con el fin de evaluar computacionalmente los intercambios en las combinaciones de entrega del cliente.

Russell y Gribbin (1991) presentaron un método de solución que consiste en un diseño inicial de la ruta, seguido de tres fases de mejora diferentes. Su método de solución muestra algunas similitudes con enfoques anteriores. Utilizaron una formulación modificada de la formulación LP de Tan y Beasley (1984) para encontrar una solución inicial.

Chao, Golden y Wasil (1995) desarrollaron un método metaheurístico para resolver el PVRP. Cordeau, Gendreau y Laporte (1997) presentaron un método de búsqueda tabú para resolver varios problemas de enrutamiento diferentes, incluyendo el PVRP. El método que utilizan es una heurística de dos fases: inicialización y búsqueda tabú. Cordeau, Laporte y Mercier (2001) ampliaron el método de Cordeau et al. (1997) con ventanas de tiempo.

Rusdiansyah y Tsao (2005) proporcionaron una formulación modelo con una serie de restricciones que son aplicables en esta tesis. Francis, Smilowitz y Tzur (2006)

desarrollaron un método de solución exacta para el PVRP con opción de servicio. El método de solución se basa en la relajación lagrangiana.

Mourgaya y Vanderbeck (2006) asignaron a los clientes vehículos con un método de solución que tiene un doble objetivo (regionalización y equilibrio). Groer, Golden y Wasil (2009) describieron un problema de enrutamiento de vehículos de ciclo de facturación equilibrado (BBCVRP); desarrollaron un algoritmo que utiliza una combinación de heurística y programación entera a través de un algoritmo de tres etapas. Él, Sun, Xu y Zu (2009) presentaron un algoritmo equilibrado de k-means para dividir áreas en problemas de enrutamiento de vehículos (Meerendonk, 2010).

## **2.15 Problema de entregas parciales (SDVRP)**

El problema SDVRP (Split Delivery VRP) o también conocido como VRP de entrega dividida, consiste en que un cliente podrá ser visitado por diferentes vehículos reduciendo los costos, siempre que la demanda del cliente sea mayor a la capacidad de los vehículos.

Es una relajación del VRP en donde se permite que el mismo cliente pueda ser atendido por diferentes vehículos siempre y cuando se reduzca el costo total. Esta condición es importante si los tamaños de las órdenes de los clientes copan la capacidad de un vehículo (Rocha, González, & Orjuela, 2011).

El objetivo es minimizar la flota de vehículos y la suma del tiempo de viaje necesario para abastecer a todos los clientes, donde una solución es factible si se cumplen todas las restricciones de VRP, excepto que un cliente puede ser suministrado por más de un vehículo.

Su formulación consiste en minimizar la suma del costo de todas las rutas. Una manera fácil de transformar un VRP en un SDVRP consiste en permitir repartos divididos dividiendo cada pedido del cliente en una cantidad de órdenes indivisibles más pequeñas ( Networking and Emerging Optimization, 2018)

## 2.16 Problema con valores al azar (SVRP)

El problema SVRP (Stochastic SVRP) tiene como objetivo minimizar la distancia recorrida por el vehículo de manera que considerando la aleatoriedad de las variables se pueda cumplir al cliente con sus demandas, servicios y tiempos de viaje. Este problema considera uno o varios componentes aleatorios, como por ejemplo el número de clientes, tiempo de recorrido, etc. Existen tres casos para este tipo de problema los cuales son:

- *Clientes estocásticos*. Cada cliente  $v_i$  está presente con una probabilidad  $p_i$  y está ausente con una probabilidad de  $1-p_i$
- *Demandas estocásticas*. La demanda  $d_i$  de cada cliente es una variable aleatoria.
- *Tiempos estocásticos*. Los tiempos de servicio  $ts_i$  y los tiempos de transporte  $tt_i$  son variables aleatorias.

El carácter estocástico del problema aparece tanto en los tiempos de desplazamiento por los arcos, a través de una matriz aleatoria, como en los tiempos de parada en los nodos, a través de un vector aleatorio. Estos parámetros pueden ser dependientes, en el caso de que todas las condiciones de la red cambien a la vez, por ejemplo, incrementos generalizados de la congestión, o pueden ser independientes, para representar la posibilidad de accidentes o retrasos localizados en determinados puntos de la red.

En un SVRP se llevan a cabo dos etapas para llegar a una solución. Una primera solución es determinada antes de saber el valor de las variables. En la segunda etapa, se lleva a cabo una acción correctiva, cuando los valores de las variables ya son conocidos (Rodríguez Perez, 2012).

Los autores proporcionan dos versiones distintas para la resolución de este problema, las cuales son (Cortés & Onieva, 2011):

1. Un modelo de probabilidad restringido, en el que el objetivo es minimizar los costes previstos para las rutas, a la vez que se garantiza que la probabilidad

de que la rutas sobrepasen la duración B no es superior a un determinado umbral.

2. Un modelo simple recursivo, en el que se busca minimiza la suma de los costes previstos para las rutas y los costes de penalización previstos.

## **2.17 Problema con entregas y devoluciones (VRPPD)**

El problema VRPPD (VRP with Pickup and delivery) fue tratado por primera vez por en 1989 y plantea que existe la posibilidad que los clientes puedan devolver algún tipo de mercancía, por lo que se debe tener en cuenta que estos artículos si quepan en el vehículo y no excedan su capacidad. El objetivo de este problema consiste en minimizar la flota de vehículos y la suma de los tiempos de transporte bajo la restricción de que cada uno de los vehículos debe contar con la capacidad suficiente para transportar los productos que deberán recogerles a los clientes con el fin de regresarlos al depósito.

Entre los años 1999 y 2009, a partir del Problema VRPPD se desarrollaron variaciones de dicho problema de tal manera que se pueden considerar dos grandes clasificaciones: la primera son problemas estáticos donde los datos del problema son conocidos antes de construir las rutas y la segunda son problemas dinámicos en los cuales algunos datos solo son conocidos durante el periodo de tiempo de operación, además que el horizonte de planeación puede ser no acotado (Berbeglia, Cordeau, & Laporte, 2010).

En el artículo *“Revisión del estado del arte del problema de ruteo de vehículos con recogida y entrega (VRPPD)”* se describe la taxonomía del VRPPD, donde según sus variantes puede tener diferentes enfoques. En un primer enfoque se tiene un esquema basado en tres campos: estructura, visitas y vehículos.

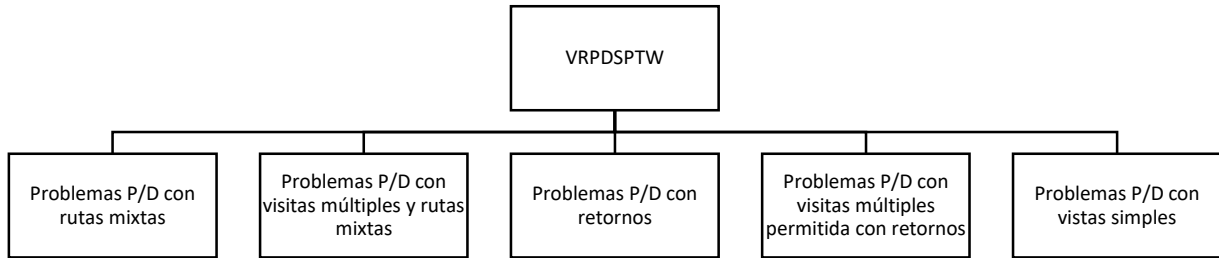
La estructura tiene tres subgrupos de problemas: muchos a muchos: "many to many", donde cualquier vértice puede utilizarse como fuente o como destino. Un ejemplo de este caso es el problema de intercambio: "Swapping problema"; uno-a

muchos-a uno: "one-to-many-to-one", por ejemplo, en la industria embotelladora, la distribución de bebidas y recolección de envases vacíos, y uno a uno: "one to one", donde cada producto se considera una solicitud que proviene de un origen y tiene un destino definido. Esta situación se observa en las operaciones de mensajería y el servicio de transporte puerta a puerta.

En las visitas se presenta información sobre la operación de recogida (P) y entrega (D): PD cuando las operaciones se combinan; P-D cuando las operaciones se pueden hacer conjuntamente o por separado y P/D cuando cada cliente tiene una entrega o un envío requerido, pero no ambos. La parte final del esquema está constituida por el número de vehículos utilizados en la solución del problema: puede ser uno o múltiples vehículos.

Entre 1999 y 2016 se desarrollaron muchas variantes del problema VRPPD, en las que se consideraron dos situaciones: *ambiente estático*, que se presenta cuando todos los datos de entrada del problema son conocidos antes de la construcción o diseño de las rutas y el horizonte de planeación es limitado, y *ambiente dinámico*, donde algunos datos de entrada se conocen o se actualizan durante el periodo en el que se efectúan las operaciones de entrega y recogida de los productos. El horizonte de planeación en este escenario es ilimitado. La mayoría de los problemas de entrega y recogida se han centrado en el escenario estático y pocos autores han trabajado la parte dinámica del problema.

Otro enfoque del problema de ruteo de vehículos con entregas y recogidas selectivas incluye ventanas de tiempo (VRPDSPTW) y su objetivo es minimizar la diferencia entre los costos de enrutamiento y los ingresos asociados con las recogidas. En este escenario se encuentran cinco variantes del problema según (Ballesteros Silva & Escobar Zuluaga, 2016), las cuales son:



**Ilustración 2. Variantes del VRDSPTW.**

**Fuente.** Ballesteros Silva, Pedro Pablo. Escobar Zuluaga, Antonio. *Revisión del estado del arte del problema de ruteo de vehículos con recogida y entrega (VRPPD)*. Revista Científica Ingeniería y Desarrollo, Vol 34, No 2 (2016).

**2.18 Problemas con viajes de regreso (VRPB)**

El problema VRPB (VRP with Backhauls) tiene un planteamiento similar al VRPPD que consiste en que los clientes pueden tanto recibir mercancías o productos como entregarlos, pero este tiene la variante de que existe una restricción en la que las entregas deben ser realizadas y completadas antes de que se realicen las devoluciones.

En el VRPB los consumidores pueden demandar o retornar algunas mercancías. Es necesario tener en cuenta cuales de los bienes que los consumidores devuelven al vehículo de reparto pueden caber en él. El supuesto más crítico en lo que respecta a todas las entregas, es que éstas deben ser realizadas en cada ruta antes de que alguna recogida pueda ser iniciada. Esto surge del hecho de que los vehículos son cargados en la parte trasera y las cargas acomodadas en los camiones y que los puntos de entrega son considerados económicamente factibles. Las cantidades que deben ser distribuidas y recogidas son fijas y conocidas con anticipación.

En el problema tipo VRPB el conjunto de clientes se divide en dos. El primero contiene n centros de consumo con línea de recorrido (linehauls), que requieren que una cantidad dada de productos sea entregada y el segundo contiene m centros de



consumo con recorrido de vuelta o retornos (backhauls), donde se requiere que una cantidad de productos dada deba ser recogida.

En esta variación de VRP hay una restricción importante entre linehaul y backhaul, cuando una ruta pueda servir a ambos tipos de consumidores todos los centros de consumo con linehauls deben ser atendidos antes que algún centro de consumo con backhauls, si los hay.

Este VRPB consiste en encontrar una cantidad de recorridos simples con mínimo costo de tal manera que cada ruta o recorrido visita el centro de distribución, cada centro de consumo es visitado solamente por una ruta y la suma de las demandas de los centros de consumo con linehaul y backhaul visitados por un recorrido no excede, individualmente, la capacidad del vehículo (Networkin and Emerging Optimization, 2018).

## **2.19 Problema con múltiples depósitos (MFVRP)**

El problema MFVRP (Mixed Fleet VRP) consiste en que los vehículos pueden tener distintas capacidades, por lo cual se deberá determinar que vehículo será el adecuado para realizar el recorrido de una ruta según la distancia y la demanda.

En este tipo de problemas no solo es posible que los vehículos cuenten con diferentes capacidades (como el problema de HFVRP) sino que es posible que cada uno cuente con distintos tiempos o costos de traslados o tengan alguna otra restricción adicional en función al tipo de vehículos.

Según Burchett y Campion, (2002), *“El Problema de Ruteo con Flota Mixta es una variación del Problema de Ruteo de Vehículos, pero incluyendo la decisión del tipo de vehículo que se utilizará. Cada uno de ellos es caracterizado por su capacidad, costos fijos y costos variables de transporte. Debido a la capacidad heterogénea de la flota de vehículos es necesario considerar esta en la ruta que seguirá cada bien, ya que un vehículo más grande podrá realizar una ruta más larga o que tenga mayor concentración de demanda”* (Arévalo, 2014).

## **2.20 Problema de ruteo con ventanas de tiempo (VRPTW)**

El problema VRPTW (VRP with time windows) tiene el mismo planteamiento del VRP, pero con la restricción de que existe un periodo de tiempo determinado para cumplir con el abastecimiento de los clientes.

El VRPTW que aparece en 1967, presenta las variaciones tales como el VRPTD (VRP with time deadlines, ventanas rígidas de tiempo) en 1986, el VRPMTW (VRP con ventanas de tiempo múltiples) el cual es formulado en 1988; y VRPSTW (VRP with soft time windows, ventanas blandas de tiempo) en 1992. (Toth & Vigo, 2002).

Es un problema de VRP con la restricción adicional de una ventana de tiempo asociada a cada consumidor, definiendo un intervalo dentro del cual el consumidor debe ser atendido, el intervalo en el depósito es llamado horizonte de programación.

En este tipo cada consumidor está asociado con un intervalo de tiempo llamado Time Window (Ventana de Tiempo). En el instante en el que los vehículos salen del centro de distribución, se da el tiempo de recorrido para cada arco y así mismo un tiempo de servicio adicional para cada consumidor. El servicio de cada consumidor debe empezar dentro de la ventana de tiempo asociada y el vehículo debe parar en el centro de consumo por instantes de tiempo (Restrepo, Medina, & Cruz, 2008).

En caso de que el vehículo llegue antes del tiempo establecido, este debe esperar hasta el instante de tiempo en el que el servicio deba empezar. Las ventanas de tiempo están definidas de tal manera que se asume que todos los vehículos salen del centro de distribución en el tiempo cero. En este problema se debe encontrar una cantidad de recorridos simples con el mínimo costo de tal manera que cada ruta o recorrido visita el centro de distribución, cada centro de consumo es visitado solamente por una ruta, la suma de las demandas de los centros de consumo visitados por un recorrido no excede la capacidad del vehículo, para cada centro de consumo el servicio comienza dentro de la ventana de tiempo y el vehículo se detiene por instantes de tiempo. Este tipo de problema tiene extensiones como VRPMTW, VRPTD y VRPSTW.

Este tipo de problemas están caracterizados por las siguientes restricciones adicionales (Rodriguez Perez, 2012):

- Una solución es inviable si un cliente es atendido después de su hora límite.
- Un vehículo que llega a un cliente antes de la hora programada causa un tiempo de espera adicional en la ruta.
- Cada ruta debe empezar y terminar dentro de la ventana de tiempo asociada a cada cliente.

## **2.21 Generalizaciones de la VRPTW**

Un número de restricciones o propiedades adicionales de problemas de enrutamiento más complejos se pueden modelar usando el marco que se acaba de desarrollar. En esta sección discutiremos brevemente cómo permitir vehículos no idénticos; trabajar con más de un depósito, vehículos de varios compartimentos, el uso de múltiples ventanas de tiempo o ventanas de tiempo suave, y recogida y entrega. (El-Sherbeny, 2010)

### **- *Vehículos no idénticos***

Los vehículos pueden ser no idénticos de varias maneras. La forma típica en que una flota heterogénea de vehículos se caracteriza por su capacidad, pero también podría ser diferente debido a los diferentes costes de arco para cada vehículo, los diferentes tipos de vehículos (que cubren y no cubren), los diferentes tiempos de viaje, las ventanas de tiempo u otras características (Tuyttens et al., 2004, El-Sherbeny, 2001, El-Sherbeny y Tuyttens, 2001).

### **- *Múltiples depósitos***

En problemas de la vida real puede haber más de un depósito. El VRPTW puede usarse para modelar situaciones donde existen depósitos múltiples. Los clientes son atendidos por varios depósitos, cada uno de los cuales tiene su propia flota de

vehículos (Lie y Simchi-Levi, 1990; Desaulniers et al., 1997). Normalmente se supone que los vehículos deben regresar al mismo depósito que comenzaron a formarse. De forma relajada sólo se exige que el número de vehículos que lleguen al depósito sea igual al número de vehículos que lo abandonan. En una relajación adicional que se utiliza poco, no hay restricciones sobre los depósitos a los que los vehículos deben regresar.

- ***Múltiples ventanas de tiempo***

En el VRPTW cada cliente tiene una ventana de tiempo donde el servicio debe tener lugar. Permitir que los clientes tengan ventanas de tiempo múltiples y disjuntas. Un vehículo que llega entre dos ventanas de tiempo debe esperar hasta el comienzo de la ventana de tiempo siguiente.

- ***Múltiples compartimentos***

Si los vehículos tienen dos o más compartimentos el problema de enrutamiento se conoce como VRPTW de Compartimiento Múltiple (MCVRPTW). El uso de múltiples compartimentos es relevante, cuando los vehículos transportan varios productos que deben permanecer separados durante el transporte. Un ejemplo es la distribución de productos petrolíferos a estaciones de servicio donde los camiones cisterna están divididos en varios compartimentos para transportar los diferentes tipos de gasolina.

De la misma manera se puede extender el modelo VRPTW para manejar las restricciones de capacidad multidimensional. En VRPTW la capacidad es unidimensional. Esta dimensión puede ser el peso, volumen, valor o piezas. Sin embargo, las restricciones de capacidad pueden ser multidimensionales, por ejemplo, peso y volumen para poder manejar casos en los que muchas cajas grandes no violen la restricción de peso, pero su volumen es grande para un vehículo, o al revés.

## 2.22 Base De Datos Orientada A Grafos NEO4J

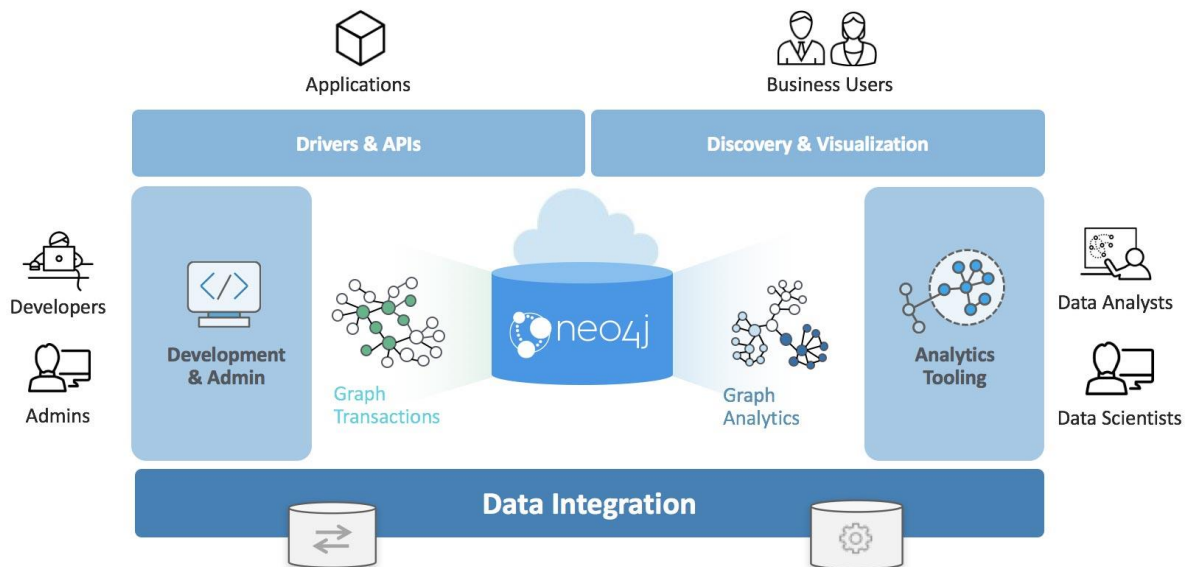
En el presente trabajo se empleó Neo4j debido a la naturalidad del problema y a las ventajas que brinda este tipo de bases de datos, en donde se tiene almacenamiento y procesamiento de gráficos nativos con una arquitectura escalable optimizada para la velocidad y el cumplimiento de **ACID** (*Atomicidad, Consistencia, Aislamiento y Durabilidad*) garantizando la previsibilidad de las consultas basadas en relaciones, además de permitir una ventaja en la velocidad de carga de grandes volúmenes de datos, con muy poco uso de memoria, lo cual es de gran importancia para tratar este tipo de problemas, en donde los datos van aumentando a medida que se están evaluando las diferentes posibilidades y los distintos escenarios que puedan generarse.

Otra de las razones por la cuales se eligió Neo4j es por su plataforma de gráficos, la cual mejora el análisis de los datos, ya que al mapear los puntos de datos y las relaciones entre ellos, se puede hacer un almacenamiento y un recorrido de las redes generadas de manera más precisa conociendo aquellas relaciones que se van creando; para problemas de transporte y logística es importante contar con esta plataforma gráfica, ya que permite reconocer fácilmente las rutas generadas y así llegar a conclusiones y resultados más precisos.

La plataforma de gráficos de Neo4j es compatible con aplicaciones transaccionales y análisis de gráficos, en donde dicho análisis permite obtener nuevas perspectivas sobre los datos, además de contar con la integración de datos, agilizando la condensación de estos, así como la visualización y el descubrimiento de gráficos que ayudan a comunicar los beneficios de dicha tecnología en aquellas empresas en donde se lleva a cabo su implementación.

La plataforma grafica de la neo4j está compuesta por diferentes herramientas que al integrarlas generan grandes ventajas como las que se mencionaron anteriormente; en el gráfico que se muestra a continuación se tiene su estructura en donde se encuentra la consola de desarrollo y administración, seguido de aplicaciones de transacción y recorrido en tiempo real y escalabilidad masiva,

finalizando con herramientas de análisis donde los algoritmos de gráficos admiten el PageRank, Centrality y Path Finding; transversal a toda esto se tiene la integración de los datos, en donde se revelan relaciones ocultas RDBMS al importar a un gráfico, Data Lake Integrator materializa gráficos de Apache Hadoop, Hive and Spark; también se tiene el descubrimiento y visualización que permiten la exploración de gráficos.



**Ilustración 3 Base de Datos NEO4J**

**Fuente:** <https://neo4j.com/product/>

Dentro de los conceptos básicos que se emplean en el modelo de gráfico se encuentran los siguientes:

### **Nodos**

Los nodos son los principales elementos de datos, están conectados a otros nodos a través de las relaciones los cuales pueden tener una o más propiedades (atributos almacenados como pares clave / valor) además tienen una o más etiquetas que describen su rol en el gráfico.

## **Relaciones**

Las relaciones conectan dos nodos y son direccionales. Los nodos pueden tener relaciones múltiples, incluso recursivas y las relaciones pueden tener una o más propiedades (es decir, atributos almacenados como pares clave / valor).

## **Propiedades**

Las propiedades se denominan valores donde el nombre (o clave) es una cadena y se pueden indexar y restringir. Los índices compuestos se pueden crear a partir de propiedades múltiples.

## **Etiquetas**

Las etiquetas se utilizan para agrupar nodos en conjuntos, en donde un nodo puede tener múltiples etiquetas y dichas etiquetas se indexan para acelerar la búsqueda de nodos en el gráfico. Los índices de etiquetas nativas están optimizados para la velocidad.

Algunas de las razones por las cuales se eligió neo4j y no una base de datos relacional son porque la estructura de almacenamiento de gráficos con adyacencia libre de índices da como resultado transacciones y procesos más rápidos para las relaciones de datos, comparadas con las RBDMS en donde el almacenamiento se genera en tablas fijas, predefinidas con filas y columnas con datos conectados a menudo desarticulados entre tablas, paralizando la eficiencia de la consulta; además el modelo de datos es flexible, sin discrepancias entre el modelo lógico y el físico. Los tipos de datos y las fuentes se pueden agregar o cambiar en cualquier momento, lo que lleva a tiempos de desarrollo mucho más cortos e iteraciones ágiles reales, mientras que en las RBDMS el modelo de base de datos debe desarrollarse con modeladores y traducirse de un modelo lógico a uno físico. Dado que los tipos de datos y las fuentes deben conocerse con anticipación y cualquier cambio requiere semanas de tiempo de inactividad para la implementación. En cuanto al rendimiento de consulta el procesamiento de gráficos garantiza una latencia cero y un rendimiento en tiempo real, independientemente del número o la profundidad de las relaciones, mientras que en las RBDMS el rendimiento del

procesamiento de datos se ve afectado con el número y la profundidad de JOIN (o relaciones consultadas).

## **CAPITULO 3. ESTADO DEL ARTE**

En este capítulo se presentan algunos de los estudios desarrollados a lo largo de los años, con el fin de analizar las soluciones encontradas y la forma como se han abordado este tipo de problemas en las diversas áreas, creando así una base importante en el desarrollo del presente trabajo, ya que es posible conocer las distintas soluciones de acuerdo con los algoritmos, métodos y herramientas empleadas.

### **3.1 Vehicle Routing Problem with Simultaneous Pickups and Deliveries and Time Windows Considering Fuel Consumption and Carbon Emissions.**

**Autores:** Hao Gang, Gou Zhijing, Yang Peng y Sun Junqing.

**Descripción:** En la búsqueda de mejorar los factores que promueven el calentamiento global, se plantea minimizar la cantidad de gasolina y emisiones de dióxido de carbono, puesto que el 22% de las emisiones producidas en el planeta provienen del transporte de mercancías. Por lo tanto, los autores buscan minimizar el costo total de las operaciones de transporte, incluyendo en el cálculo, el valor inicial de los vehículos, la depreciación de estos, el consumo de gasolina, las emisiones de dióxido de carbono y los costos de personal. Para plantear el modelo definen ciertas limitantes como un único centro de distribución el cual tendrá cierta cantidad de vehículos con las mismas características, la ubicación de los clientes y la cantidad de mercancía a recoger y/o entregar. Con el planteamiento, se determina la cantidad de vehículos con base a las necesidades de los clientes y la capacidad del vehículo, el orden de visita a cada cliente dependiendo de sus horarios de atención. Para dar solución a dicho problema, se utiliza un algoritmo genético.

**Aportes a la tesis:** El artículo sirve para entender la importancia de la optimización de las rutas y del combustible gastado durante los recorridos de entrega y/o recogida de productos. Es de gran utilidad, puesto que el problema a solucionar es



muy similar al planteado, pues se busca optimizar la cantidad de vehículos y gasolina gastada en cada recorrido, teniendo en cuenta los horarios de atención de los clientes y la ubicación de estos. Además, su modelo matemático sirve como base para la presente tesis, observando las posibles variables que se pueden utilizar durante la definición de este permitiendo que se defina de la mejor manera. Asimismo, proporciona las bases teóricas del algoritmo genético utilizado junto a los operadores manejados, brindando una idea del algoritmo a utilizar para dar solución al problema planteado en este proyecto.

### **3.2 Drivers-Inspired Ants for Solving the Vehicle Routing Problem with Time Windows.**

**Título:** Drivers-Inspired Ants for Solving the Vehicle Routing Problem with Time Windows.

**Autores:** Dalicia Bouallouche (dalia.bouallouche@u-bourgogne.fr), Jean-Baptiste Vioix (jean-baptiste.vioix@u-bourgogne.fr), Eric Busvelle (busvelle@univ-tln.fr) y Stéphane Millot (stephane.millot@tedies.fr).

**Descripción:** El objetivo de los autores es proporcionar una herramienta práctica que se base en las decisiones tomadas por el algoritmo planteado y las expectativas del transportista para resolver el VRPTW; dicha aplicación propone una interacción permanente y automática entre la experiencia profesional y el algoritmo de optimización, garantizando la fiabilidad de las soluciones óptimas. Para dar solución al problema planteado, se analiza y se encuentra que el algoritmo de TSP con colonias de hormigas encaja con el enfoque que se busca implementar, utilizando los datos de itinerarios de vehículos reales como datos de retroalimentación e inyectando dicha información en el rastro de feromonas de las hormigas, permitiendo influenciarlas por los planeadores humanos y las decisiones de los conductores. Con esta propuesta, no se alcanza instantáneamente la solución óptima, puesto que se propone un ciclo de optimización que incluye lo aprendido en las últimas rutas prácticas para utilizarlo en las futuras optimizaciones de enrutamiento de vehículos. Cabe aclarar, que el algoritmo se comporta de mejor manera cuando la empresa tiene clientes regulares.

**Aporte a la tesis:** Este artículo es de utilidad puesto que es aplicado a un problema de la vida real, mostrando desde el planteamiento del problema hasta la puesta en marcha de la solución utilizando el algoritmo de optimización de rutas de vehículos con ventanas de tiempo. Además, sirve como base para entender el funcionamiento del algoritmo de colonias de hormigas y determinar que, al retroalimentar la colonia, se puede obtener itinerarios de mejor calidad, no dejando de lado, la experiencia de los trabajadores de la empresa, pues no se deben aferrar completamente al resultado del algoritmo, sino que pueden mejorar la solución en cuanto a temas de costos más realistas y la complejidad de las vías. Asimismo, muestra las variables y las condiciones a tener en cuenta para definir correctamente el modelo y permitir que las hormigas y los humanos se complementen. Las conclusiones a las que se llega en el artículo son importantes pues muestra que el algoritmo se perfecciona cada vez más si se itera con el resultado anterior.

### **3.3 A Hybrid Genetic Algorithm for Vehicle Routing Problem With Time Windows.**

**Título:** A Hybrid Genetic Algorithm for Vehicle Routing Problem With Time Windows.

**Autores:** Sripriya. J, M.Tech (sripriya.sri10@gmail.com), Ramalingam.A (a.ramalingam1972@gmail.com), Rajeswari. K, M.Tech (rajeswari2009.it@gmail.com).

**Descripción:** Para poder controlar el problema de ruteo con múltiples objetivos, los autores plantean una búsqueda genética híbrida con control de la diversidad para dar solución a los problemas de ruteo con ventanas de tiempo. En dicho algoritmo hacen uso de operaciones de selección, crossover, mutación y espacio de búsqueda, también el enfoque de Pareto para encontrar así el conjunto de soluciones óptimas para grandes problemas de VRPTW. Además, dicha solución minimiza el número de vehículos y la distancia total simultánea en la que se logran múltiples objetivos. En este artículo muestran también diferentes trabajos

relacionados que han logrado soluciones efectivas para el problema de ruteo con ventanas de tiempo.

**Aportes a la tesis:** El artículo aborda la solución a los problemas de ruteo con ventanas de tiempo desde la perspectiva de una solución teórica basada en Inteligencia Artificial, la cual explica los diferentes métodos utilizados para alcanzar una solución óptima, lo que permite entender otras formas de abordar dicho problema cuando se tienen múltiples objetivos. Además, al mostrar los trabajos relacionados que han encontrado soluciones efectivas, permite ampliar el conocimiento adquirido y obtener criterios para decidir cual se adaptaría mejor al problema planteado en esta tesis.

### **3.4 Vehicle Routing with Time Windows Based on Two-stage Optimization Algorithm.**

**Autores:** Linling Liao, Xiushan Cai, Huadong Huang, Yanhong Liu.

**Descripción:** En este artículo, los autores proponen una optimización de dos etapas, mostrando inicialmente como definieron el problema matemáticamente y luego explicando en qué consiste el algoritmo. La primera fase del algoritmo consiste en utilizar una colonia de hormigas para pre-optimizar el problema VRPTW, intentando encontrar una ruta ideal pre-optimizada; Basado en los operadores de cortar y encajar de los algoritmos genéticos, se calcula una pre-ruta obteniendo así los puntos de segmentación, y utilizando el algoritmo heurístico de inserción basado en ventanas de tiempo, se calcula la longitud de la trayectoria y el tiempo de espera de los vehículos. Para validar la viabilidad del algoritmo, se prueba con dos algoritmos de la literatura, encontrando que el algoritmo propuesto puede resolver los problemas de VRPTW eficazmente.

**Aportes a la tesis:** Este artículo sirve como base puesto que aborda el problema de rutas con ventanas de tiempo de una manera diferente a otros métodos, mostrando que al mezclarlos se puede obtener soluciones óptimas. Aunque no realizan la implementación del algoritmo con valores del mundo real, muestra el

proceso descriptivo desde el modelo matemático hasta la puesta en marcha de la simulación. Además, muestra el método de prueba para determinar si el algoritmo planteado comparado con otros es viable, el cual puede ser utilizado en un futuro para determinar si el algoritmo propuesto en la presente tesis arroja soluciones efectivas.

### **3.5 Models and Algorithms for the Vehicle Routing Problem with Time Windows and Other Conditions.**

**Autores:** Guzairov M.B., Yusupova N.I., Smetanina O.N., Rassadnikova E.Yu.

**Descripción:** Los autores muestran el modelo matemático del clásico problema de ruteo (VRP), el cual es una generalización del problema del agente viajero (TSP) donde se tienen unas vías conectadas a través de un depósito en común; como características del modelo clásico se encuentran que, la demanda es determinista, la entrega a los clientes se hace por un vehículo, los vehículos tienen la misma capacidad y sólo hay un único depósito. Por lo tanto, en el artículo muestran los diferentes tipos derivados del VRP y se enfocan en proponer el uso del algoritmo OPT\_route como solución al problema de enrutamiento de vehículos con ventanas de tiempo cuando se tienen múltiples depósitos en empresas petroquímicas. Asimismo, incluye las precauciones de seguridad que se deben tener en cuenta para evitar accidentes, incluyendo dichos factores, como el estado de las vías en la función objetivo.

**Aportes a la tesis:** Este artículo es muy importante para el desarrollo de la presente tesis, porque ayuda a entender de mejor manera los problemas de ruteo (VRP) y las diferentes variaciones que se han presentado a lo largo del tiempo, permitiendo conocer las principales funciones objetivo y las limitaciones de estas en los modelos matemáticos. Asimismo, brinda una idea para solucionar el problema planteado, teniendo en cuenta que no se tenga únicamente un depósito. Además, tienen en cuenta un factor importante que debe incluirse en estos algoritmos que son las

precauciones de seguridad, pues toda empresa desea minimizar los posibles riesgos que se pueden presentar.

### **3.6 Improvement of Genetic Algorithm for Vehicle Routing Problems with Time Windows.**

**Autores:** Yanfang Deng, Jianling Xiang, Zhuoling Ou.

**Descripción:** En este artículo inicialmente definen el problema de ruteo con ventanas de tiempo y brindan diversas referencias ya que se ha propuesto gran variedad de algoritmos que incluyen métodos exactos y heurísticas eficaces para la solución de dichos problemas. Las búsquedas tabú, los algoritmos genéticos y las estrategias de evolución han demostrado el mejor rendimiento para los problemas de enrutamiento de vehículos. Los autores presentan una solución basada en una mejora realizada al algoritmo genético utilizando el operador crossover optimizado. El algoritmo propuesto ha sido probado frente a diferentes soluciones de la literatura, utilizando 56 problemas de Solomon con 100 clientes, obteniendo resultados altamente competitivos. Los pasos del modelo propuesto son los siguientes, se elige la forma de representación (en este caso es una cadena de enteros de tamaño N), se inicializa la población aleatoriamente y se utiliza el mecanismo de selección por torneo, después de esto, se aplica el operador crossover optimizado y la mutación; se itera hasta que se llegue a una condición de terminación.

**Aportes a la tesis:** Este artículo es de gran utilidad, puesto que muestra diferentes referencias en las cuales encuentran soluciones óptimas al problema de ruteo de vehículos con ventanas de tiempo, además, explican el algoritmo propuesto de manera que sirve de base al momento de definir la solución al problema definido en esta tesis. Es interesante observar cómo haciendo uso de un grafo bipartito no dirigido en el operador crossover se obtiene una mejora en el algoritmo genético haciéndolo competitivo con diferentes algoritmos propuestos en la literatura.

Además, brinda una idea para probar el algoritmo que dará solución al problema de este trabajo de grado, utilizando las seis clases del benchmark de Solomon.

### **3.7 Solving Vehicle Routing and Scheduling Problems using Hybrid Genetic Algorithm.**

**Autores:** Bhawna Minocha (bminocha@ascs.amity.edu), Saswati Tripathi (saswati@iift.ac.in) y C. Mohan.

**Descripción:** Aunque los algoritmos genéticos se desempeñan de buena manera en la búsqueda global, tardan mucho tiempo en converger a la solución óptima global. En cambio, en las búsquedas locales son rápidos para encontrar una solución eficaz. Por lo tanto, para mejorar la eficiencia de los algoritmos genéticos, los autores tratan de incorporar búsquedas locales. El algoritmo propuesto comienza construyendo la población inicial, todos los individuos son evaluados de acuerdo a los criterios de aptitud. La evolución continúa con la selección por torneos, donde se seleccionan buenos individuos para la reproducción. Se mantienen los dos mejores para la siguiente generación sin pasar por operadores genéticos. El operador crossover y la mutación se aplican para modificar los individuos seleccionados para generar una nueva generación factible. Para mejorar aún más a los individuos, se aplican heurísticas de búsqueda local. Se genera un número aleatorio  $r$ , entre 0 y 1, si  $r$  es menor que 0,8; uno de los algoritmos de búsqueda local se ejecuta, de lo contrario ningún algoritmo es ejecutado. El proceso continúa iterativamente hasta que la mejor solución no cambie durante un número específico de generaciones. Al comparar dicho algoritmo con los de la literatura, el rendimiento es competitivo y en algunos casos, incluso mejor, en términos de número de vehículos.

**Aportes a la tesis:** Este artículo muestra otra forma de mejorar el rendimiento de los algoritmos genéticos, explicando detalladamente el procedimiento a realizar y los posibles métodos a utilizar para lograr la mejoría. A través de este artículo, es posible entender claramente los conceptos relacionados a los algoritmos genéticos.

Es interesante comparar las diferentes maneras en las que los autores de los artículos incorporan métodos para lograr una solución óptima en el menor tiempo posible, puesto que, dicha teoría sirve como base para la definición de la solución al problema planteado en esta tesis, y se podría combinar los diferentes métodos intentando optimizar aún más los algoritmos genéticos.

### **3.8 Solving Vehicle Routing Problem with Time Windows with Hybrid Evolutionary Algorithm.**

**Autores:** Yong Mao(maodeyong@163.com), Yanfang Deng(dengyf@whut.edu.cn).

**Descripción:** Los autores proponen un nuevo algoritmo híbrido para dar solución a los problemas de enrutamiento de vehículos con ventanas de tiempo, dicho algoritmo es una combinación del algoritmo genético, el algoritmo GRASP, la estrategia ENS y la optimización por enjambre de partículas (PSO). La diferencia radica en que al momento de definir la población inicial se realiza de manera aleatoria sin conocer si dichos valores son buenos candidatos, para evitar que existan malas soluciones se utiliza el algoritmo GRASP para inicializar la población; se evalúa cada candidato y se mejora la aptitud de cada uno haciendo uso de la estrategia PSO; se seleccionan dos candidatos como padres a través del método de selección de la ruleta y se utiliza el operador crossover, se clonan para completar la descendencia de la nueva población usando el método de búsqueda local ENS, se aplica el operador de mutación y se inserta en la nueva población; se itera hasta encontrar la mejor solución. Los resultados computacionales muestran que a través del uso de una fase intermedia, se obtienen individuos más eficientes, y por lo tanto, se mejora la eficacia del algoritmo.

**Aportes a la tesis:** Dicho artículo muestra una forma de mejorar el algoritmo genético, agregando una nueva fase intermedia, en la cual se crea un algoritmo híbrido que hace uso de otras estrategias para lograr optimizar el algoritmo clásico de solución al problema de enrutamiento de vehículos con ventanas de tiempo (VRPTW) y brindar soluciones más eficientes. Además, sirve como base para

entender el algoritmo GRASP, la estrategia ENS como método de búsqueda local y la optimización de enjambre basada en la población (PSO), logrando obtener nuevas ideas que permitan dar solución al problema planteado en esta tesis.

### **3.9 Vehicle Routing Problem with Time Windows: A Deterministic Annealing approach.**

**Autores:** Mayank Baranwal (baranwa2@illinois.edu), Pratik M. Parekh (bpparekh2@illinois.edu), Lavanya Marla (lavanyam@illinois.edu), Srinivasa M. Salapaka (salapaka@illinois.edu), Carolyn L. Beck (beck3@illinois.edu).

**Descripción:** En este artículo los autores proponen una metodología de solución basada en el algoritmo de recocido determinístico (DA) aplicado al problema de enrutamiento de vehículos con ventanas de tiempo. DA se adapta con facilidad a problemas de asignación de recursos. El algoritmo propuesto tiene la habilidad de evitar obtener un óptimo local de baja calidad, además mantiene una velocidad de convergencia relativamente rápida. Aunque DA no ha sido utilizada para abordar las especificaciones y restricciones que surgen en el contexto de los problemas de enrutamiento y programación simultánea. El enfoque adapta las especificaciones y restricciones en términos de localización, permitiendo así el uso de DA; además, ofrece la flexibilidad de cambiar el enrutamiento y la programación en espacio y dimensiones de tiempo para identificar clústeres en el espacio combinado y permitir intercambiar decisiones de enrutamiento y programación. Los resultados de la simulación muestran que las restricciones son respetadas y demuestran resultados casi óptimos (cuando son verificables). Las soluciones proporcionadas por el DA son completamente independientes de las inicializaciones. De hecho, el algoritmo intenta para resolver un problema de optimización completamente determinista.

**Aportes a la tesis:** Este artículo es muy interesante puesto que muestra un método totalmente diferente a los encontrados con anterioridad, permitiendo buscar una solución sin necesidad de realizar la inicialización de las variables, sino que permite hacer uso de un algoritmo determinístico para encontrar resultados óptimos en



términos de horarios y duración del recorrido, también evita que el algoritmo se quede atrapado en mínimos locales. Además, el algoritmo presentado en el artículo permite realizar cambios en tiempo real a la programación y enrutamiento; lo que permitiría encontrar una buena solución al problema planteado en esta tesis. Asimismo, este artículo sirve como base teórica, puesto que especifica la descripción de cada uno de los problemas que se pueden encontrar en el VRPTW, así como el modelo matemático que representa dicha solución.

### **3.10 Selección óptima de operadores para el tratamiento de problemas VRP con Algoritmos Genéticos.**

**Autores:** Antonio Moratilla, Eugenio Fernández, Juan José Sánchez, Borja Vicario.

**Descripción:** En este artículo se da una definición general de que es el problema de enrutamiento de vehículos VRP y las diferentes tipologías que se pueden encontrar, además, muestra que para la solución de este tipo de problemas se han abordado diferentes técnicas que se pueden clasificar en tres grandes categorías: métodos exactos, heurísticas y metaheurísticas. Una de las técnicas metaheurísticas más utilizadas en la actualidad para dar solución a los problemas VRP son los algoritmos genéticos, los cuales utilizan conceptos como cruce y mutación, así como reglas de auto-reparación y adaptación. Uno de los problemas detectados por los autores es que la representación de cada individuo que compone la población, el tamaño de esta, las estrategias de cruce y mutación, y el resto de los diferentes parámetros del algoritmo, difieren según los autores y tipo de problemas a abordar, por lo que su uso no es tan fácil como debería ser, es por esto, que se definen los operadores más utilizados y una breve descripción de cada uno. Como resultado, los autores llegan a la conclusión que la combinación ideal para conseguir el mejor rendimiento del algoritmo está compuesta por el operador de selección por torneo (TS), el operador de cruce por correspondencia parcial (PMX) y el operador de mutación por intercambio (EM).

**Aportes a la tesis:** El artículo sirve como base para conocer la tipología del problema VRP y las diferentes técnicas que se han utilizado para encontrar una solución óptima, los autores se centran en la definición del algoritmo genético y los

parámetros necesarios para su utilización. Teniendo en cuenta que en la literatura se utilizan diversos operadores de mutación, cruce y selección, es interesante observar la manera en que son comparados en el artículo, encontrando así una combinación que permita obtener el mejor rendimiento del algoritmo genético, lo que haría que al momento de definir la solución al problema planteado se puedan utilizar los operadores recomendados por los autores, haciendo que se pueda obtener una solución eficaz.

### **3.11 An algorithm for the routing problem with split deliveries and time windows (SDVRPTW) applied on retail SME distribution activities.**

*Un algoritmo para el problema de ruteo de vehículos con entregas divididas y ventanas de tiempo (SDVRPTW) aplicado a las actividades de distribución de PYMEs del comercio al por menor.*

**Autores:** Juan Sepúlveda (jcsepulvedas@unal.edu.co), John Wilmer Escobar (jwescobar@javerianacali.edu.co) y Wilson Adarme-Jaimes (wadarme@unal.edu.co).

**Descripción:** Las pequeñas y medianas empresas (PYMEs), son compañías que generalmente están enfocadas a la distribución de productos en pequeñas cantidades y con frecuencias particulares para abastecer a sus clientes en diversas condiciones y medios de transporte. El estudio realizado permitió determinar que este tipo de empresas no tienen los conocimientos suficientes ni las capacidades económicas para llevar a cabo la implementación de tecnologías especializadas que les permita hacer una planeación óptima de las rutas al momento de hacer la distribución de sus productos a sus clientes; evidenciando además que los medios empleados para el transporte de las mercancías son en la mayoría de los casos bicicletas y vehículos motorizados. Con el fin de encontrar una solución a este tipo de problemas se emplea el enfoque de un SDVRPTW, en el cual se reduce el número de vehículos usados para satisfacer la demanda de los clientes y permite usar vehículos con menor capacidad que la requerida a través de su fraccionamiento. El método desarrollado permite dividir la demanda de los clientes

hasta en dos lotes, usando una heurística de inserción sencilla con un algoritmo de post-optimización primitivo, en donde el orden inicial del conjunto de clientes es aleatorio y usando una plataforma tecnológica de fácil acceso como Microsoft® Excel™. La implementación de esta herramienta resulta ser adecuada y pertinente para ser empleada por este tipo de empresas, debido a las ventajas económicas y de fácil manejo que permiten resolver problemas combinatoriales, respondiendo a problemas de la vida real en cortos periodos de tiempo con cierto grado de precisión.

**Aportes de la tesis:** El artículo brinda un aporte ya que presenta una investigación sobre el tipo de empresas que son objeto de estudio en el trabajo de grado, presentando información sobre su comportamiento y los medios de transporte más utilizados para llevar a cabo el proceso de visita de clientes. La investigación realizada muestra diversos caminos que son de utilidad, para afianzar el criterio que se debe tener de acuerdo al tema que se está abordando, ya que, al modelar un problema similar, pero con algunas variantes es fácil determinar cuál algoritmo podría ser más adecuado y pertinente implementar.

Por esta razón y teniendo en cuenta la naturaleza del problema que se aborda en el trabajo de grado, se genera una visión que permitirá determinar con más precisión las variables a considerar generando una solución óptima que brinde un aporte importante a las empresas en el desarrollo de sus actividades de distribución y servicio a sus clientes. La información suministrada por el artículo genera una base fundamental para determinar que aún estas empresas pequeñas desarrollan este tipo de actividades de manera empírica, lo que les dificulta la optimización de los tiempos y de gastos en combustible en los procesos de distribución, que en la mayoría de estas compañías es fundamental y es la naturaleza de su negocio.

### **3.12 El problema de ruteo de vehículos [VRP] y su aplicación en medianas empresas colombianas.**

*Vehicle Routing Problem [VRP] and its applications in Colombian medium-sized companies*

**Autores:** Jairo Arboleda Zúñiga, MSc (jarboledaz@usc.edu.co), Astrid Xiomara López (xioma1978@hotmail.com) y Yéssica Lorena Lozano (yessica.lozano.guzman@gmail.com).

**Descripción:** El artículo presenta una revisión a los diferentes tipos de VRP que se conocen a lo largo de la historia, además de los modelos matemáticos, heurísticos y metaheurísticos que han sido empleados en la búsqueda de soluciones óptimas para este tipo de problemas, considerando cada una de las variantes que hace compleja la búsqueda de una solución exacta. En la investigación realizada se presentan dos casos de estudio de empresas medianas colombianas, en donde se usan los diferentes tipos de VRP según la naturaleza de su empresa y la actividad a la que se dedica, con el fin de alcanzar resultados pertinentes en los procesos de logística de transporte y distribución de productos y servicios.

Las empresas que han sido objeto de estudio han permitido determinar, que cada una de las actividades que se realizan dentro de ellas pueden ser representadas por uno o más de los tipos de VRP que existen, aunque las dos empresas se dedican a sectores diferentes, considerando que una de ellas es una empresa de mensajería y la otra una empresa de aseo municipal, se podría pensar que por tratarse de distribución y recolección, se obtiene su solución de la misma manera, pero se evidencia que las diferentes restricciones y el número de nodos que se deben tener en cuenta hacen que se deba tratar de manera especial, debido a que esto genera un aumento en el tiempo de cálculo, por lo que los algoritmos exactos no son eficaces.

**Aportes a la tesis:** El artículo genera una base teórica y fundamental al trabajo de grado, debido a que presenta cada una de las variantes del VRP y la forma como se comporta de acuerdo a las situaciones que deben ser evaluadas en el momento

de planificar la distribución de las rutas dentro de las empresas, generando una visión más amplia en el momento de abordar este tipo de problemas, pues en sus casos de estudio se evidencia cómo un problema puede ser dividido y tratado con un tipo diferente de VRP, lo que permitirá llegar a una solución óptima y le brindará a este tipo de empresas prestar un servicio a sus clientes de manera oportuna, sin estar en desventaja de las grandes compañías que a diferencia de las medianas y pequeñas, la planeación y la logística de sus rutas se hace de manera más especializada y no empírica, como se ha demostrado en las diferentes investigaciones. El artículo además permite evidenciar que la investigación realizada tiene aplicabilidad en este tipo de empresas y que es necesario desarrollar software que permita fácil accesibilidad y uso extensivo de programas de ruteo, que contribuyan a mejorar su competitividad logística y que permitan el avance en el empleo de técnicas metaheurísticas en la búsqueda de soluciones óptimas.

### **3.13 Alternativa Heurística MCM para Problemas de Ruteo de Vehículos.**

*MCM Heuristic Alternative for Vehicle Routing Problem-Solving*

**Autores:** José Luis Flores Flores (lsigma@live.com.mx) y Manuel Alvarez-Madrigal (mmadrigal@itesm.mx).

**Descripción:** El problema de ruteo de vehículos (VRP) es un problema que a lo largo de la historia ha sido difícil de resolver debido a su complejidad matemática, por tal motivo han sido diversos los algoritmos y técnicas empleadas con el fin de buscar una solución factible, ya que las soluciones exactas son en general difíciles de encontrar, por ello de los métodos más utilizados han sido las heurísticas, las cuales han tenido una evolución constante, pues han sido objeto de estudio a lo largo de la historia, pero aunque han generado grandes aportes y énfasis en la velocidad de cálculo para obtener una solución factible, en la actualidad se busca desarrollar métodos más innovadores y que puedan ser utilizados por cualquier tipo de empresa, ya sea media, grande o pequeña, dado que con los métodos implementados hasta el momento son rápidos y eficientes desde el punto de vista

computacional, pero no son sencillos operacionalmente y requieren del apoyo de computadoras para efectuar todas las operaciones implicadas, por esta razón en la investigación se plantea un método con un enfoque para abordar el problema de lo general a lo particular, basado en un principio mecánico de “centro de masas” MCM el cual aunque no se ha implementado y las pruebas realizadas se han hecho manualmente, los resultados que ha generado han sido óptimos comparados con el programa comercial de optimización llamado “VRP solver” que ejecuta el algoritmo de Clarke-Wright.

**Aportes a la tesis:** El artículo genera un aporte debido a que muestra un nuevo punto de vista en la búsqueda de los métodos que se emplean en la solución de este tipo de problemas, teniendo en cuenta que su enfoque también se centra en la necesidad latente que tienen las medianas y pequeñas empresas a la hora de implementar un método que les permita generar una adecuada programación de rutas para la distribución de sus productos, pues a lo largo de la historia se han tenido métodos que requieren de una implementación computacional especializada, lo cual económicamente para muchas empresas no es posible cubrir y optan por realizar esta actividad empíricamente generando así que el servicio prestado a los clientes no sea de calidad y por ende no sean competitivos en el mercado. El aporte que genera este artículo permite que se pueda evaluar este método si se desea simplificar un problema con varios, vértices los cuales se pueden representar en centros de poblaciones pequeñas o puntos de consolidación de productos y así generar un híbrido con otro método que logre de manera ágil y rápida llegar a una solución óptima.

### **3.14 Algoritmos Meméticos Aplicados A La Resolución De Un Problema De Ruteo De Vehículos Periodico.**

**Autores:** A. Méndez, D. Palumbo, M. Carnero, J. Hernández (gop@ing.unrc.edu.ar)

**Descripción:** El trabajo propuesto en este artículo muestra una serie de estrategias para encontrar una solución óptima al problema de ruteo de vehículos, aplicado a un caso real; el cual consiste en la recolección de residuos infecciosos, en donde es necesario que se realice de manera oportuna, no solo por lo que implica económicamente, sino porque el manejo de dichos residuos está vinculado a la salud pública. La recolección se realiza diariamente, en donde se visita a una cantidad diferente de centros, variando día a día y atendiendo a la frecuencia con la que se requiera visitar cada centro.

Para modelar el problema anterior, se empleó un Problema de Ruteo de Vehículos Periódico (PVRP), en donde las técnicas y estrategias utilizadas para la solución, son una combinación de la computación evolutiva y las técnicas de búsqueda local. Dentro de las técnicas de la computación evolutiva empleadas están los algoritmos genéticos, ya que al realizar un híbrido con un mecanismo de búsqueda local se conjugan la capacidad exploratoria que poseen los algoritmos poblacionales con la capacidad de explotación de regiones atractivas del espacio de búsqueda, que son poco percibidas y exploradas difícilmente por los métodos basados en búsqueda local.

El problema fue dividido en dos niveles, el primer nivel consiste en el problema de la planificación semanal, en el cual se implementan cada una de las fases de los algoritmos genéticos, tales como la representación de los individuos, el método de selección de padres, los operadores genéticos, el diseño de la función de aptitud y los mecanismos de búsqueda local incorporados; y el segundo nivel corresponde el problema de determinación de rutas óptimas, en donde es implementada la búsqueda local.

**Aportes a la tesis:** El artículo permite identificar algunas características de las técnicas empleadas en la solución de problemas de ruteo de vehículos, con el fin de evidenciar que, al utilizar híbridos, se puede encontrar una solución óptima. Además, proporciona información sobre otro tipo de VRP, mostrando de esta manera, que cada problema a solucionar puede estar relacionado con un VRP diferente, por lo tanto, la forma de abordar su solución puede variar; aunque los algoritmos genéticos siguen siendo las técnicas que predominan; debido a las variaciones de este y a la complejidad de su solución, se pueden dividir y tratar con diferentes técnicas, lo que permitirá una mayor precisión en las respuestas encontradas.

La investigación genera una visión de la forma como se debe desglosar cada problema, con el fin de reducir la complejidad de su solución, ya que resulta ser más conveniente tratar con técnicas y mecanismos híbridos que con uno solo, ya que se puede potencializar sus capacidades y generar soluciones óptimas. Con la información que presenta el artículo se puede inferir, que un problema de ruteo de vehículos se puede convertir y tratar en diferentes tipos de VRP, lo importante es identificar las características del este buscando la forma más adecuada de fraccionarlo.

### **3.15 Diseño de Rutas de Transporte Escolar con Ventanas Temporales Móviles.**

**Autores:** José Luis Moura, Ángel Ibeas (mourajl@unican.es) y Luigi dell'Olio (ldell@ubu.es).

**Descripción:** El artículo plantea la solución a un problema presente en la Comunidad Autónoma de Cantabria, ya que el Transporte Escolar en España es un transporte público regular especial financiado por las Comunidades Autónomas, mediante un proceso de licitación pública, por ende, los Gobiernos Regionales ponen a disposición de los estudiantes, el correspondiente transporte desde sus viviendas hasta los centros de educación, lo cual genera un impacto económico alto



para las administraciones a tal punto de no poder asumir dichos costos. Las causas que provocan esta situación radican en la dificultad de aprovechar los vehículos en otros servicios a lo largo del día, dado que las horas de inicio de clase de los estudiantes es igual, existen rutas definidas desde hace varios años sin haber sido sometidas a un proceso de optimización.

Para la solución a este problema se plantea un modelo de asignación de costes desarrollado, en donde se busca reorganizar las rutas de transporte escolar de un área determinada, teniendo en cuenta la posibilidad de generar horarios diferentes de entrada en los colegios. Además, adoptaron un enfoque de programación binivel, compuesta por un nivel superior, en el cual se realiza la administración prefija de las horas de entrada de los distintos colegios y en un nivel inferior se resuelve un problema de optimización donde se obtiene la combinación de rutas factible que sean realizadas por un mismo vehículo. Este nivel inferior proporciona el número de autobuses necesario, la distancia y tiempos extras en los desplazamientos entre rutas.

**Aportes a la tesis:** La investigación realizada genera una base para identificar, que es necesario hacer una planificación adecuada de las rutas dentro de las empresas con el fin de optimizar los costos económicos que implica, debido a que estos tienden a aumentar de manera exponencial, considerando que allí se encuentran los gastos de gasolina, salarios de los conductores, mantenimiento de los vehículos, etc. El artículo proporciona un modelo de cálculo de costes que permite conocer la clasificación de los gastos que se deben considerar dentro de un problema de optimización, como lo son los fijos y los variables, dentro de los primeros se encuentran los impuestos del vehículo, amortización y financiación de este y dentro de los segundos se tienen los sueldos del personal (Conductores), combustible, lubricantes, mantenimiento, etc.

Adicional a lo anteriormente descrito, el artículo genera una solución a los problemas VRP empleando el algoritmo de HookeJeeves, el cual tiene como ventaja no exigir ningún atributo especial de la función objetivo, lo que proporciona una visión hacia la búsqueda e implementación de nuevos algoritmos con características

que permitan obtener una solución óptima; además de proponer el tratamiento de dichos problemas en fases, permitiendo así, convertir este tipo de problemas complejos en otros más simples, cumpliendo con las restricciones y supliendo las necesidades requeridas.

### **3.16 Método heurístico para el problema de ruteo de vehículos aplicado a la empresa distribuidora Representaciones Continental.**

**Autores:** Julio Fuentes Vidal, Marco Parra León y Hernando Alexánder Gutiérrez.

**Descripción:** El artículo presenta una implementación del problema de ruteo de vehículos con ventanas de tiempo y flota heterogénea aplicado a un caso de estudio real, en una empresa encargada de distribuir licores en la ciudad de Bogotá, cubriendo solo en el ejercicio de su actividad una de sus localidades. Esta empresa como muchas no tienen un algoritmo implementado para la planificación de las rutas, pues es el vendedor quien se encarga de indicar cómo se van a visitar los clientes según la secuencia que él realiza en la recopilación de los pedidos, por lo tanto, esta actividad operativa se está realizando sin ninguna técnica cuantitativa, sino basándose en la experticia del vendedor, exponiéndose la empresa no solo pérdidas económicas sino a prestar un mal servicio a sus clientes.

Con el fin de encontrar una solución a este tipo de problemas, se lleva a cabo la implementación de la heurística secuencial de Solomon, la cual inserta clientes en la última ruta creada, aunque esto presenta una desventaja, ya que los últimos clientes no visitados tienden a estar dispersos y dichas rutas generarían mayor costo y tiempo en su recorrido. Aun cuando la heurística presenta esta desventaja los resultados obtenidos en el estudio fueron satisfactorios respecto a la distancia, tiempo de recorrido y costos.

**Aportes a la tesis:** El artículo proporciona información sobre una heurística que puede ser empleada en la solución del problema objeto de estudio, lo cual permite determinar la viabilidad de emplear determinado algoritmo, pues al VRP ser un problema de tipo combinatorio, buscando de manera exhaustiva el mayor número

de soluciones no es posible resolverlos con programación lineal y entera, por ello se requiere abordarlos por medio de heurísticas y metaheurísticas; por lo tanto la implementación de heurísticas de inserción al ser métodos constructivos permiten crear una solución mediante sucesivas inserciones de clientes en las rutas, lo que hace que el algoritmo maneje una gran cantidad de datos y sea viable en la búsqueda de la solución a estos problemas.

A través del caso de estudio expuesto en el artículo se puede evidenciar que dependiendo del problema que se deba tratar la función objetivo dependerá de las características propias de este, que en la mayoría de los casos busca minimizar y maximizar determinados recursos, en ocasiones minimizando costos totales de operación y en otros minimizando los tiempos de transporte (bodega-clientes-bodega).

### **3.17 A set-covering based heuristic algorithm for the periodic vehicle routing problem**

**Autores:** V. Cacchiani, V.C. Hemmelmayr y F. Tricoireb.

**Descripción:** El artículo propone un algoritmo heurístico para la solución del problema de enrutamiento vehicular periódico (PVRP), el cual es una generalización del problema de ruteo de vehículos (VRP) y consiste en determinar un conjunto de rutas de coste mínimo para la planificación cada día de las rutas, con las restricciones de que los clientes deben ser visitados el número de veces necesarias y deben recibir siempre la cantidad requerida de producto y el número de rutas por día no exceda el número total de vehículos disponibles, respetando la capacidad del vehículo.

El algoritmo propuesto es una heurística de optimización híbrida, la cual puede ser aplicada a problemas de programación lineal de enteros mixtos, basándose en la solución de la relajación de programación lineal (LP) de una formulación de programación lineal entera del problema, dicha relajación puede ser resuelta por un solver de propósito general o por técnicas de generación de columnas, con

métodos exactos o heurísticos, además puede ser empleada para guiar el algoritmo heurístico a obtener una solución entera.

La implementación de este algoritmo se lleva a cabo con la finalidad de encontrar una solución factible al problema de enrutamiento periódico mediante la fijación y liberación de las variables, las cuales son liberadas después de transcurrido un tiempo, al convertirse en variables son fijas y ser consideradas contextualmente comprometedoras.

**Aportes a la tesis:** El artículo proporciona información útil para conocer algunos de los trabajos que se han llevado a cabo, relacionados con el VRP y sus generalizaciones, además de mostrar nuevas implementaciones de algoritmos que buscan mejores soluciones, ya que al hacer la comparación con algoritmos de última generación se evidencia que el algoritmo propuesto es eficaz y competitivo. Lo anterior permite que se amplíen las posibilidades para abordar este tipo de problemas y de esta manera dar una solución óptima que le garantice a las organizaciones brindar un mejor servicio, adquiriendo beneficios económicos y optimizando el tiempo empleado en las rutas.

La información suministrada en el artículo evidencia que es conveniente que este tipo de problemas se particione de tal manera que pueda tratarse de manera especializada y se puedan emplear diferentes métodos, que permitan la convergencia a una solución óptima, por ende, el algoritmo allí implementado se basa en la formulación de programación lineal de números enteros de partición del problema y tres relajaciones diferentes que son empleadas para derivar límites inferiores que puedan presentarse. Las dos fases en las que se fundamenta el algoritmo radican en, primero, asignar órdenes o pedidos a los días y segundo construir las rutas para cada uno de estos días; lo cual es necesario conocer para saber cómo abordar los problemas de este tipo.

### **3.18 Metaheurísticas aplicadas al ruteo de vehículos. Un caso de estudio. Parte 1: formulación del problema y estudio.**

*Metaheuristics applied to vehicle routing. A case study. Parte 1: formulating the problema.*

**Autores:** Guillermo González Vargas y Felipe González Aristizábal.

**Descripción:** El artículo muestra la implementación del problema de ruteo de vehículos en una empresa manufacturera que tiene la necesidad de definir la localización de una bodega desde la cual sea posible distribuir su producto a 53 centros de consumo, cada uno de los cuales tiene una demanda periódica constante, para ello plantean que este tipo de problemas es mejor particionarlo, donde la formulación matemática contenga familias de restricciones que imposibiliten la creación de ciclos o subtoures.

El artículo tiene como objetivo principal realizar una revisión bibliográfica en la que se pueda ilustrar el problema de ruteo de vehículo, sus generalizaciones y las técnicas utilizadas para resolverlo. Además de mostrar algunos estudios e implementaciones realizadas de dicho problema, con el fin de generar una formulación del problema y una formulación matemática general, que permita identificar el comportamiento de este y determinar para cada caso qué técnica o método se debe aplicar en la búsqueda de la solución.

**Aportes a la tesis:** El artículo proporciona una revisión bibliográfica que permite identificar las diferentes generalidades del problema de ruteo de vehículos, ya que alcanzar una solución óptima para este tipo de problemas depende de las características especiales de los clientes, las locaciones y los productos y los servicios que se ofrecen, por lo tanto, es necesario la elaboración de una metodología de solución específica que permita aproximarse a la mejor solución. Las diferentes variaciones y restricciones del problema generan una familia, de la cual el artículo expone ocho casos, en donde comparten características que pueden generar un universo de problemas VRP, estas generalidades son: CVRP (Capacited VRP), MDVRP (Multi-Depot VRP), PVRP (Period VRP), SDVRP (Split Delivery

VRP), SVRP (Stochastic VRP), VRPPD (VRP Pickup and Delivery), MFVRP (Mix Fleet VRP) y el VRPTW (VRP with Time Windows).

El artículo no solo da a conocer las generalidades del VRP, sino que también proporciona información de cómo reducir la complejidad cuando se aborda un problema con múltiples vehículos, acotando el universo de soluciones y disminuyendo el conjunto de clientes a ser visitados por cada vehículo o desde cada depósito, asignando a cada vehículo/deposito un conjunto de clientes para atender, lo que se conoce como set covering, o lo que algunos autores llaman clusterizar. Esta clusterización se puede llevar a cabo mediante la implementación de heurísticas, tales como Heurística de barrido o sweep, Heurística de asignación generalizada de Fisher y Jaikumur y Heurística de localización de Bramel y Simchi-Levi.

### **3.19 Diseño de un Modelo de Ruteo de Vehículos: Caso de Estudio UMNG Sede Campus.**

**Autores:** Yimy Alexander Hernández Ortiz (yimy.hernandez@unimilitar.edu.co ),

Wilfrido Javier Arteaga Sarmiento (wilfrido.arteaga@unimilitar.edu.co),

Nataly Lorena Guarín Cortés (nataly.guarin@unimilitar.edu.co).

**Aporte a la tesis:** El artículo genera una revisión bibliográfica del problema de ruteo de vehículos (VRP), con el fin de identificar aquellos estudios que sirvan para guiar las soluciones futuras que se implementen sobre este problema, así como también proporciona información sobre las técnicas utilizadas para la solución de este tipo de problemas, lo cual permite que en el momento de analizar cada problema conocer la manera más adecuada de abordarlo, identificando sus características y comportamiento, con el fin de determinar cuál o cuáles son las técnicas más adecuadas para encontrar la solución.

El artículo a través del estudio realizado permite determinar la importancia que tiene para las empresas la implementación de algoritmos que optimicen los recursos de

sus empresas, no solo porque así pueden brindar un mejor servicio, sino porque pueden optimizar sus recursos y tener un control de cada una de las actividades llevadas a cabo en el proceso de distribución, lo cual es una ventaja para ellas, haciéndolas competitivas en el mercado y atractivas para sus clientes.

### **3.20 Modelo matemático para la planificación de servicios y programación de rutas en empresas prestadoras de servicios de control de plagas.**

**Autor:** John Willmer Escobar (john.wilmer.escobar@correounivalle.edu.co)

**Descripción:** El artículo se basa en la propuesta de un modelo de programación lineal entera mixta, que considera la minimización de costos asociados a la distancia recorrida y el tiempo ocioso de los operarios que ejecutan las tareas, con el fin de mejorar la gestión logística en las empresas. Esta metodología fue implementada en una empresa colombiana que presta servicios de fumigación, presentando buenos resultados en donde se evidenció su importancia y eficiencia en la solución de este tipo de problemas. Para la solución de este problema se consideraron ciertas restricciones, tales como capacidad de servicios, eliminación de subtour y cumplimiento de demanda.

El modelo matemático propuesto se basó en los requerimientos y especificaciones que tienen las empresas del sector de servicios de control integral de plagas, por ello se consideraron aspectos de demanda, cantidad de operarios, etc., en donde las decisiones se relacionan con la asignación de operarios y la generación de rutas para cumplir con las demandas de los clientes; dicho modelo fue programado en AMPL y se ha ejecutado utilizando el solver CPLEX®.

El artículo presenta además dos tipos de problema de ruteo de vehículos con ventanas de tiempo (horario de atención): duras y suaves, donde las primeras, se especifican para cada cliente y se debe cumplir estrictamente un horario en el cual forzosamente debe ser visitado; mientras que las segundas el tiempo en el cual el cliente debería ser visitado es flexible.

**Aporte a la tesis:** El artículo propone una solución al problema de ruteo de vehículos empleando programación lineal entera mixta, lo que permite evidenciar que a través de este tipo de técnicas se puede llegar a soluciones óptimas, que brinden a las empresas mejorar su gestión logística con el fin de no afectar sus ingresos, el número de clientes y su reputación, ya que en el mercado actual se compite por precio, calidad y servicio al cliente.

La información suministrada en el artículo permite conocer e identificar que para solucionar problemas de pequeñas y medianas instancias es más viable utilizar métodos como la programación lineal mixta y para aquellos que tienen grandes instancias se emplean algoritmos heurísticos y metaheurísticos, los cuales permiten obtener soluciones aproximadas en tiempos computacionales reducidos.

El artículo también permite identificar que en la mayoría de las empresas el encargado de realizar la planificación de las rutas es un operario y no cuentan con metodologías y herramientas cuantitativas que les ayude en esta labor, por lo tanto, no es posible garantizar el buen funcionamiento en las actividades de secuenciación de rutas, así, cualquier decisión que se tome bajo este escenario será riesgoso, generando grandes pérdidas para las compañías. De esta manera las rutas que son programadas manualmente superan la capacidad productiva, impidiendo cumplir los objetivos trazados en la prestación de servicios.



## **CAPITULO 4. METODOLOGÍA DE LA INVESTIGACIÓN**

### **4.1 Enfoque De La Investigación**

El tipo de investigación que se realizó en el presente trabajo fue cuantitativa y analítica, ya que se basó en datos numéricos y estadísticos sobre aquellos algoritmos genéticos que han encontrado soluciones al problema de rutas de vehículo (VRP) y los tiempos computacionales empleados por estos, además de tomarse como referencia para el análisis, los datos suministrados por la página oficial del VRP con ventanas de tiempo.

### **4.2 Población Objetivo De Estudio**

Para esta investigación se tomó como población objetivo los datos suministrados por los problemas de referencia VRPTW de Solomon que se han expuesto desde 1987 para las instancias de 25, 50 y 100 clientes, en donde en la última versión se presentan con un objetivo jerárquico: 1) Minimizar el número de vehículos y 2) Minimizar la distancia total. La distancia y el tiempo deben calcularse con doble precisión, los resultados de la distancia total se redondean a dos decimales. Los métodos exactos generalmente usan un objetivo de distancia total monolítico y usan cálculos de distancia y tiempo integrales o de baja precisión. Por lo tanto, los resultados no son directamente comparables.

### **4.3 Procedimiento De Investigación**

El procedimiento llevado a cabo en esta investigación para la recolección de los datos consistió en primera instancia en la observación y análisis de los procedimientos realizados en empresas que desempeñan actividades de visitas a clientes con el fin de realizar cobros por los servicios prestados; seguido a esto se realiza la búsqueda en la página oficial del VRP con ventanas de tiempo, de los archivos de Solomon con los datos de referencia para ser empleados en los algoritmos implementados y así realizar la búsqueda de una mejor solución.

## **CAPITULO 5. IMPLEMENTACIÓN DE LA SOLUCIÓN**

En la búsqueda de una mejor solución para el problema objeto de estudio, se tuvo en cuenta el planteamiento del problema de ruteo de vehículos con ventanas de tiempo VRPTW, además de realizarse la implementación de un algoritmo genético en el lenguaje de programación Python y el uso de la base de datos Neo4j, adicional a esto se empleó como referencia el estudio realizado en la creación de FECOC 2016 (Consultoría técnica para el fortalecimiento y mejora de la base de datos de factores de emisión de los combustibles colombianos), ya que es una calculadora creada con el objetivo de facilitar el cálculo de emisiones de CO2 generados por el aprovechamiento energético de los combustibles que hacen parte importante en la actualidad de la canasta energética de Colombia.

### **5.1 Formulación Matemática Del VRPTW**

El VRPTW es una de las variantes más importantes del VRP clásico, posee su mismo planteamiento con una restricción adicional, que consiste en que el VRPTW asocia una ventana de tiempo con cada cliente definiendo un intervalo en donde el cliente tiene que ser atendido. Este problema busca minimizar la flota de vehículos y la suma de tiempo de viaje y tiempo de espera necesarios para abastecer y atender a todos los clientes en sus horas requeridas.

Dentro de las consideraciones que se deben tener en la formulación y búsqueda de la solución a dicho problema se encuentran:

- Una solución se vuelve inviable si un cliente es atendido después del límite superior de su ventana de tiempo.
- Un vehículo que llega antes del límite inferior de la ventana de tiempo causa un tiempo de espera adicional en la ruta.
- Cada ruta debe comenzar y finalizar dentro de la ventana de tiempo asociada al depósito.

- En el caso de rutas de horario reducido, un servicio posterior no afecta la viabilidad de la solución, pero se penaliza añadiendo un valor a la función objetivo.

En la formulación de dicho problema para el presente trabajo se tiene:

- El VRPTW se define como  $G = (V, A)$

Donde el depósito está representado por los dos nodos  $0$  y  $n + 1$ .

- Todas las rutas posibles del vehículo corresponden a rutas en  $G$  que comienzan en el nodo  $0$  y terminan en el nodo  $n + 1$ .
- Se asocia una ventana de tiempo con los nodos  $0$  y  $n + 1$ , es decir,  $[a_0, b_0] = [a_{n+1}, b_{n+1}] = [E, L]$ , donde  $E$  y  $L$  representan la salida más temprana posible del depósito y la última posible llegada al depósito, respectivamente.
- El vehículo puede llegar antes de  $a_i$  pero tendría que esperar a que el cliente esté listo para ser atendido, pero no podría llegar después del tiempo  $b_i$ .
- Un nodo se define de la siguiente manera:

$$\sum_{i=1}^n \text{caminos} \text{ dist}_i(a, b) \quad (1)$$

- Se definen cero demandas y tiempos de servicio para estos dos nodos, es decir,  $d_0 = d_{n+1} = s_0 = s_{n+1} = 0$ .
- Las soluciones viables existen solo si

$$\begin{aligned} a_0 = E &\leq \min_{i \in V \setminus \{0\}} b_i - t_{0i} \\ b_{n+1} = L &\geq \min_{i \in V \setminus \{0\}} a_i + s_i + t_{i0} \end{aligned}$$

- Un camino está definido como un conjunto de nodos que un agente debe visitar.

$$\sum_{i=2}^{nn} (d_{(n_i)} + (d_{(n_{i-1})})) * Consumo * Tipo de Vehículo \quad (2)$$

- **Consumo:** Galones de gasolina utilizados en el recorrido.
- **Tipo de Vehículos:** Para los recorridos se emplean motos como medio de transporte, donde se puede utilizar cualquier marca.

Para este trabajo se utilizaron como ejemplo las siguientes:

- 125 NKD (AKT MOTOS)<sup>1</sup>
- GN125 (Suzuki)<sup>2</sup>
- CB1 PRO (Honda)<sup>3</sup>
- Libero 125 (Yamaha)<sup>4</sup>

- **Factor de Emisión de gases:** El valor 8.8085  $KgCO_2$  se obtiene del estudio realizado en el numeral 5.3 Cálculo De La Emisión De Gases. Por lo tanto:

$$Fe = \sum_{i=2}^{nn} (d(n_i) + (d(n_{i-1}))) * Consumo * Tipo de Vehículo * 8.8085 KgCO_2 \quad (3)$$

De acuerdo con lo anterior se tiene:

$$\sum_{i=1}^{ng} g_i \sum_{j=i}^{nn} nodo_j \sum_{i=2}^{nn} ((d(n_i) + (d(n_{i-1}))) * Consumo * Tipo de Vehículo) \sum_{k=1}^{na} agente_k \quad (4)$$

Empleando los siguientes dos tipos de variables:

- Variables de flujo  $x_{ijk}$ ,  $(i, j) \in A$ ,  $k \in K$ , igual a 1 si el arco  $(i, j)$  es utilizado por el vehículo  $k$ , y 0 de otra manera
- Variables de tiempo  $w_{ik}$ ,  $i \in V$ ,  $k \in K$ , que especifican el inicio del servicio en el nodo  $i$  cuando son atendidas por el vehículo  $k$ .
- El vehículo  $k$  esta relacionado con el consumo de combustible, dado por  $C$ .

$$\min \sum_{k \in K} \sum_{(i,j) \in A} x_{ijk} \quad (5)$$

<sup>1</sup> AKT NKD 125. Recuperado de <https://www.aktmotos.com/motos/street/nkd-125>

<sup>2</sup> Suzuki GN 125. Recuperado de <https://www.suzuki.com.co/motocicletas/sport/gn-125-64356>

<sup>3</sup> Honda CB1 PRO. Recuperado de <http://motos.honda.com.co/bajo-cilindraje/cb1-pro-125>

<sup>4</sup> Yamaha Libero 125. Recuperado de <http://www.incolmotos-yamaha.com.co/site/productos/motos/urbanas-trabajo-y-deportivas/libero125>.

Sujeto a:

$$\sum_{k \in K} \sum_{j \in \Delta^+(i)} x_{ijk} = 1, \quad \forall i \in N \quad (6)$$

$$\sum_{j \in \Delta^+(0)} x_{0jk} = 1, \quad \forall k \in N \quad (7)$$

$$\sum_{i \in \Delta^-(j)} x_{ijk} - \sum_{i \in \Delta^+(j)} x_{jik} = 0, \quad \forall k \in K, \forall j \in N \quad (8)$$

$$\sum_{i \in \Delta^-(n+1)} x_{jik} = 1, \quad \forall k \in K \quad (9)$$

$$a_i(\sum_{j \in \Delta^+(i)} x_{ijk}) \leq w_{ik} \leq b_i(\sum_{j \in \Delta^+(i)} x_{ijk}), \quad \forall k \in K, \forall i \in N \quad (10)$$

$$E \leq W_{ij} \leq L, \quad \forall k \in K, \forall i \in \{0, n+1\} \quad (11)$$

Dado que:

$N = V \setminus \{0, n+1\}$  representa el conjunto de clientes.

(6) Restringe la asignación de cada cliente a una ruta de vehículo.

(7)-(9) Caracterizan el flujo del camino que seguirá el vehículo  $k$ .

(10) Para un  $k$  determinado esta restricción obliga que  $w_{ik}=0$  cuando el cliente  $i$  no es visitado por el vehículo  $k$ .

(11) Garantiza la viabilidad de la ruta con respecto a consideraciones de tiempo.

## 5.2 Descripción Del Algoritmo Genético

Para la búsqueda de la solución del problema planteado, se llevó a cabo la implementación de un algoritmo genético, siguiendo una serie de pasos los cuales se describen a continuación:

1. A partir de un archivo de prueba de Solomon, se crean  $n$  cantidad de clientes con sus respectivas coordenadas  $(x,y)$ ; además, se le asigna una hora aleatoria entre las 7 am y las 7 pm.

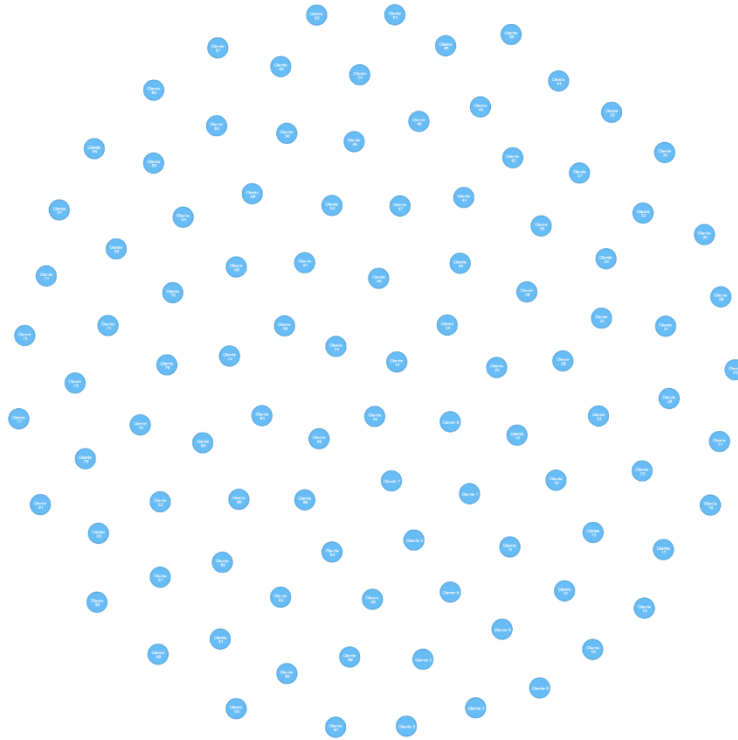


ILUSTRACIÓN 4 CLIENTES, COORDENADAS EN (X,Y)

2. Se determina un tiempo de servicio y la cantidad de horas laborables para cada uno de los agentes, de esta manera se calcula la cantidad de clientes que cada agente puede visitar en su jornada laboral. En este ejemplo, se define 20 minutos como tiempo de servicio y 6 horas laborables, se contempla una hora para contratiempos y se hace el cálculo para 5 horas laborables, se realiza la conversión de 5 horas a minutos (300) y luego se divide entre el tiempo de servicio (20), dando como resultado 15 clientes por agente.
3. Después, se dividen los clientes en dos franjas horarias: Mañana y Tarde. Teniendo en cuenta la cantidad de clientes en cada franja, se determina la cantidad de agentes necesarios para cumplir con las visitas durante la jornada laboral. Dicho cálculo se realiza haciendo un redondeo de la división de nodos por franja entre la cantidad de nodos por agente calculado en el punto anterior. Para este ejemplo, en la franja de la mañana se tienen 58 clientes, lo cual al dividir por 15 y redondear da como resultado 4 agentes

para cumplir satisfactoriamente con las visitas, a continuación, se realiza un balanceo de los 58 clientes entre los 4 agentes dando como resultado los siguientes nodos por agente [15, 15, 14, 14], de la misma manera se realiza el cálculo para la jornada de la tarde, la cual tiene 42 clientes y es necesario 3 agentes, balanceándolos de la siguiente manera [14, 14, 14].

4. Posteriormente de determinar la cantidad de agentes, se crean en base de datos y se asigna aleatoriamente una moto.



ILUSTRACIÓN 5 ASIGNACIÓN DE AGENTES POR MOTO EN BD

5. A continuación, se crea la matriz de adyacencia para cada una de las franjas horarias, para este cálculo se utiliza la distancia euclidiana.

$$d(n1, n2) = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$$

6. Se envían como parámetros al algoritmo genético la cantidad de nodos por franja, los agentes calculados y la matriz de adyacencia. Dicho algoritmo se corre n veces, donde n son la cantidad de agentes.
7. Se inicializan las variables del algoritmo, tales como, generaciones, individuos, diversidad y nodos por agente, las cuales en este ejemplo tienen los siguientes valores:
  - a. Generaciones: 5.
  - b. Individuos: 100.
  - c. Diversidad: Cantidad de nodos por franja. En este caso se mostrará el ejemplo con la franja de la mañana, que tiene 58 nodos.

- d. Nodos por agente: Cantidad de nodos que cada agente debe recorrer según el balanceo realizado con anterioridad. Para el primer agente de la jornada de la mañana, la cantidad de nodos son 15.
8. Se crea la población con una cantidad  $n$  de individuos. En este ejemplo, la cantidad es 100.
  9. Se calcula la función de adaptación para los padres y los hijos resultantes.
  10. Se ordenan los genomas mediante el criterio de menor camino.
  11. Se realiza el cruzamiento por emparejamiento parcial, en donde se elige un subsegmento de uno de los progenitores y se cruza manteniendo el orden y la posición de la mayor cantidad de genes posible del otro, conservando la coherencia, con el fin de obtener nuevos hijos mediante la función de adaptación.
  12. Se realiza un ordenamiento, el cual depende de las recombinaciones efectuadas.
  13. Se crea la próxima población y se obtiene una solución para un agente. En este ejemplo, para el primer agente de la franja de la mañana, se obtiene la siguiente ruta:  
[27, 44, 56, 15, 39, 6, 32, 51, 25, 9, 45, 17, 55, 20, 46]  
Recorriendo una distancia de 927.31 kms.
  14. Al encontrar una ruta óptima, se procede a crear un tabú en el cual se penalizan los nodos que ya han sido elegidos, haciendo que de esta manera el algoritmo en la siguiente iteración no vea dichos nodos en la matriz de adyacencia, buscando que la solución cada vez sea mejor.
  15. Al finalizar el cálculo de las rutas óptimas para los diferentes agentes, se realiza el cálculo del consumo de gasolina dependiendo de la moto que se le asignó al agente y además, se calcula la cantidad de emisiones almacenadas en base de datos de la siguiente manera:





**ILUSTRACIÓN 6 AGENTE ALMACENADO EN BASE DE DATOS**

El agente 0 tiene asignada la moto número 2 (Suzuki GN125) la cual tiene un consumo de 200km/galon, se realiza el cálculo de los galones que se consumiría el agente al realizar la ruta asignada (927,31kms) dando como resultado 4,64 galones. También, se realiza el cálculo de las emisiones de la siguiente manera: cantidad de galones \* 8,808 dando en total 40,87.

## 5.3 Pseudocódigo

### Función Principal

1. Leer archivo de clientes
2.  $g := \text{Grafo}()$
3. **Para cada cliente del archivo de clientes hacer**
  - i. Crear un Nodo de tipo Cliente  $a$  en  $g$  con su *nombre*, coordenada en  $x$ , coordenada en  $y$ , franja horaria  $t$  aleatoria
4. **Fin Para**
5. Crear los Nodos de tipo Moto en  $g$  con su *nombre*, *modelo*, *marca*, *cilindraje* y *consumo*
6. Invocar Función **DeterminarCantidadAgentes**
7. **Fin Función**

### Funcion DeterminarCantidadAgentes

1.  $\text{tiempoServicio} := 20$
2.  $\text{cantidadHorasLaborales} := 6$ .
3.  $\text{cantidadNodosAgente} := 18$ .
4.  $\text{listaFranjaDia} :=$  consultar nodos de la franja de la mañana
5.  $\text{listaFranjaNoche} :=$  consultar nodos de la franja de la noche
6.  $\text{lista} := [\text{listaFranjaDia}, \text{listaFranjaNoche}]$
7.  $\text{rutasOptimas} := []$
8. **Para cada franja dentro de la lista hacer**
  - i.  $\text{listaAgentes} :=$  Calcular la cantidad de agentes necesarios para cubrir la ruta de la franja
  - ii. **Para  $i < \text{tamaño de listaAgentes}$  hacer**
    1. Crear un Nodo  $a$  de tipo Agente con su *nombre*.
    2. Buscar un Nodo *moto* de tipo Moto de manera aleatoria.
    3. Crear una relación tipo “posee moto” entre Nodo  $a$  y *moto* y agregar arista en  $g$ .
  - iii. **Fin Para**
  - iv. Crear matriz  $m$  de adyacencia para la franja
  - v.  $m :=$  Distancia Euclidiana
  - vi.  $\text{rutaOptima} := \text{calcularMejorRuta} \leftarrow m, \text{cantidad nodos franja}, \text{listaAgentes}$
  - vii. Agregar *rutaOptima* a *rutasOptimas*
9. **Fin Para**
10. **FinFuncion**

**Funcion CalcularMejorRuta (m, cantidad nodos franja, listaAgentes)**

1. *cantidadAgentes* := 1
2. *generaciones* := 5
3. *numeroIndividuos* := 100
4. *diversidad* := cantidad nodos franja
5. *listaElegidos* := []
6. *listaOptimos* := []
7. **Mientras** *cantidadAgentes* <= tamaño de *listaAgentes*
  - i. *poblacionInicial* := Generar población inicial con los valores de *generaciones*, *numeroIndividuos*, *diversidad*
  - ii. *contadorGeneraciones* := 0
  - iii. *Individuo1*, *Individuo2* := Sacar de manera aleatoria dos individuos de la población
  - iv. **Mientras** *contadorGeneraciones* <= *generaciones*
    1. *listaFuncionAdaptacion* := []
    2. *punto1* := Obtener valor aleatorio de corte
    3. *punto2* := *punto1* + valor aleatorio
    4. *hijo1*, *hijo2* := Cruce (*Individuo1*, *Individuo2*, *punto1*, *punto2*)
    5. *individuo* := [*individuo1*, *individuo2*, *hijo1*, *hijo2*]
    6. *fit* := Calcular función de adaptación dado el *individuo* y la matriz de adyacencia *m*
    7. Agregar en *listaFuncionAdaptacion* [*individuo*, *fit*]
    8. Ordenar de manera ascendente la lista tomando como criterio de ordenamiento a *fit*.
    9. *rutaOptima* := primer elemento de la lista ordenada.
    10. Insertar los individuos que cumplan con la función de adaptación en la nueva generación.
    11. *contadorGeneraciones* := *contadorGeneraciones* + 1
  - v. **Fin Para**
  - vi. *cantidadAgentes* := *cantidadAgentes* + 1
  - vii. *m* := Penalizar (*rutaOptima*)
8. **Fin Mientras**
9. **FinFuncion**

## 5.4 Cálculo De La Emisión De Gases

Para realizar el cálculo de la emisión de gases en el problema que se aborda en el presente trabajo, se tomó como referencia los estudios realizados por el equipo de trabajo **INCOMBUSTIÓN**, en el proyecto “CONSULTORÍA TÉCNICA PARA EL FORTALECIMIENTO Y MEJORA DE LA BASE DE DATOS DE FACTORES DE EMISIÓN DE LOS COMBUSTIBLES COLOMBIANOS- FECOC”.

En el *Ejemplo de Cálculo de Factores de Emisión*, del informe de FECOC 2016, se muestran los cálculos para la emisión de sólidos y líquidos, el cual es tomado como referencia, dado que en este trabajo se requiere calcular la emisión generada por la gasolina.

A continuación, se muestra un breve resumen del ejemplo suministrado en el anexo 2:

- En primera instancia se tiene en cuenta que el modelo de combustión usado para la determinación de los FE para el CO<sub>2</sub> asume una combustión completa (100%). En este sentido todo el carbono contenido en el combustible será convertido a CO<sub>2</sub>. En el caso de CH<sub>4</sub> y N<sub>2</sub>O, se usaron valores por defecto IPCC.
- Se genera una caracterización fisicoquímica de combustible líquido, en donde se tienen datos de caracterización elemental (base seca), poder calorífico superior (PCS) y densidad obtenidos experimentalmente. Esta información se observa en la siguiente tabla:

%C	%H	%N	%S	%O	PCS (kJ/kg)	Densidad (kg/L)	%Humedad
85,76	13,69	0,33	0,0081	0,21	48317	07405	0,0

**TABLA 1 CARACTERIZACIÓN FISICOQUÍMICA DE COMBUSTIBLE LÍQUIDO**

*Fuente: CONSULTORÍA TÉCNICA PARA EL FORTALECIMIENTO Y MEJORA DE LA BASE DE DATOS DE FACTORES DE EMISIÓN DE LOS COMBUSTIBLES COLOMBIANOS- FECOC, pág. 46.*

- Para hallar el factor de emisión de CO<sub>2</sub> es necesario el contenido de carbono elemental en fracción molar o en peso, así para este combustible la fracción en peso sería 0,8576 kg C/ kg combustible, adicionalmente se requiere conocer el poder calorífico inferior, el cual puede ser determinado a partir del poder calorífico superior hallado experimentalmente de la siguiente manera:

$$PCI \left( \frac{kJ}{kg} \right) = PCS - 2441.8747 \times \frac{\%H}{100} \times \frac{18.01528}{2.01588} + \frac{\%humedad}{100} = 45329,53 \quad (1)$$

- El factor de emisión para CO<sub>2</sub> se halla teniendo en cuenta la Ecuación 1, reemplazando el contenido de carbono elemental en fracción masa, el peso molecular de CO<sub>2</sub>, el peso molecular de carbono y el poder calorífico inferior, obteniendo como resultado **69323,7 kg CO<sub>2</sub>/ TJ**.

**ECUACIÓN 1. Factor de emisión de CO<sub>2</sub> en unidades IPCC (kg CO<sub>2</sub>/TJ).  
Sólidos y líquidos**

$$FE \left( \frac{kgCO_2}{TJ} \right) = \frac{C \left( \frac{kgC}{kgcomb} \right) \times \frac{PM \left( \frac{kgCO_2}{kmolCO_2} \right) CO_2}{PM \left( \frac{kgC}{kmolC} \right) C}}{PCI \left( \frac{TJ}{kgcomb} \right)} \quad (2)$$

Reemplazando:

$$FE \left( \frac{kgCO_2}{TJ} \right) = \frac{0,8576 \left( \frac{kgC}{kgcomb} \right) \times \frac{44,0095 \left( \frac{kgCO_2}{kmolCO_2} \right) CO_2}{12,0107 \left( \frac{kgC}{kmolC} \right) C}}{45329,53 \left( \frac{TJ}{10^9 kJ} \right)} = 69323,7 \frac{kgCO_2}{TJ} \quad (3)$$

- Las anteriores unidades para el factor de emisión son las reportadas internacionalmente, de acuerdo con IPCC y con el objetivo de tener un factor de emisión en unidades de uso común y considerando que los combustibles líquidos se manejan en galones, la herramienta FECOC 2016 también reporta *FE (kg CO<sub>2</sub>/galon)*, reemplazando el FE de CO<sub>2</sub>, el PCI del combustible y la densidad del

combustible, aplicando el respectivo factor de conversión de litros a galones y obteniendo lo siguiente:

$$FE \left( \frac{kgCO_2}{galón} \right) = \left[ 69323,7 \frac{kgCO_2}{TJ} * 45329,53 \times 10^{-9} \left( \frac{TJ}{kg} \right) \right] * 0,7405 \left( \frac{kg}{L} \right) * \frac{3.7854118L}{1 Galón} =$$
$$8,808 \frac{kgCO_2}{galón} \quad (4)$$

## CAPÍTULO 6. ANÁLISIS DE RESULTADOS

### 6.1 Presentación de Resultados

#### 6.1.1 Resultados Archivo RC206

Los siguientes resultados están basados en el archivo de SOLOMON **RC206**, el cual contiene las coordenadas de 100 clientes y se generaron las siguientes soluciones en 5 iteraciones del algoritmo con los diferentes tipos de motos.

#### PROMEDIO CONSUMO Y EMISIÓN DE GASES PARA LA AKT 125 NKD

	PROMEDIO CONSUMO (Galón)	PROMEDIO EMISION (KgCO2)
ITERACION 1	5,65	49,75
ITERACION 2	5,51	48,53
ITERACION 3	5,56	48,99
ITERACION 4	5,47	48,17
ITERACION 5	5,43	47,84

TABLA 2 PROMEDIO CONSUMO EMISIÓN AKT 125 NKD

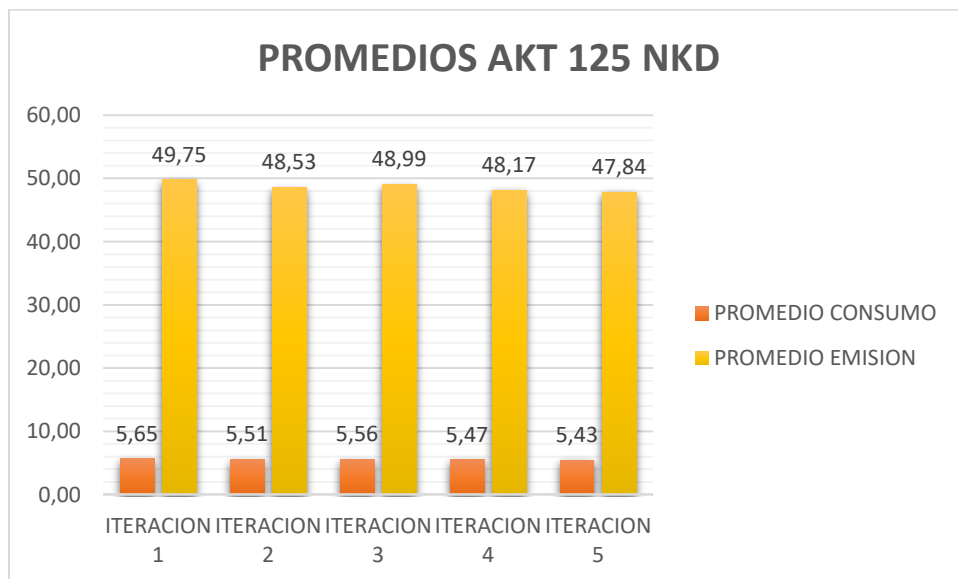


ILUSTRACIÓN 7 PROMEDIO CONSUMO EMISIÓN AKT 125 NKD

## PROMEDIO CONSUMO Y EMISIÓN DE GASES PARA LA SUZUKI GN125

	PROMEDIO CONSUMO (Galón)	PROMEDIO EMISIÓN (KgCO <sub>2</sub> )
ITERACIÓN 1	3,90	34,37
ITERACIÓN 2	3,91	34,47
ITERACIÓN 3	3,80	33,46
ITERACIÓN 4	3,87	34,09
ITERACIÓN 5	3,81	33,60

TABLA 3 PROMEDIO CONSUMO-EMISIÓN, SUZUKI GN125

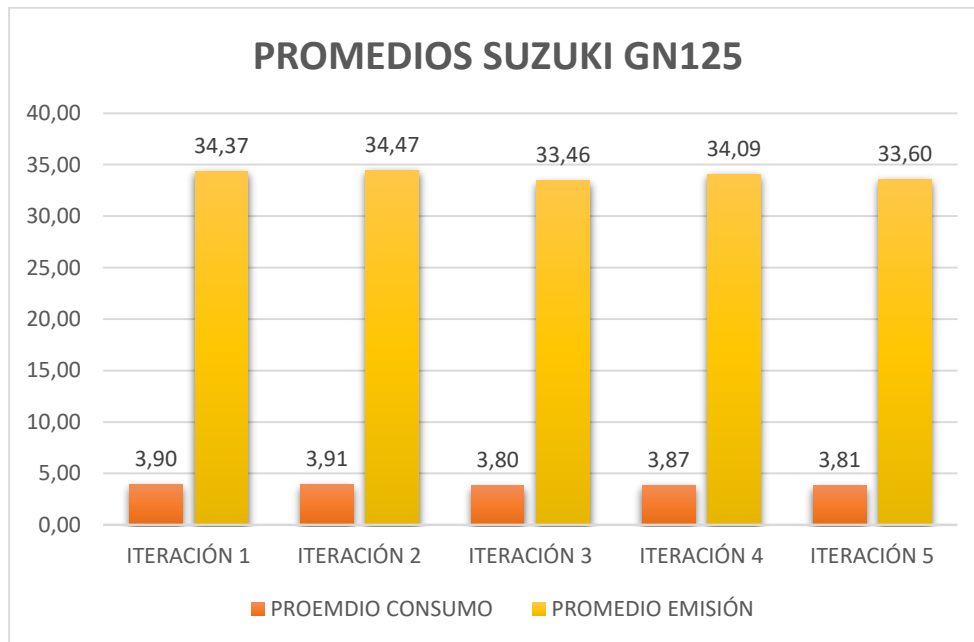


ILUSTRACIÓN 8 PROMEDIO CONSUMO EMISIÓN SUZUKI GN125



## Resultados RC206, HONDA CB1 PRO

### PROMEDIO CONSUMO Y EMISIÓN DE GASES PARA LA HONDA CB1 PRO

	PROMEDIO CONSUMO (Galón)	PROMEDIO EMISIÓN (KgCO2)
ITERACIÓN 1	2,93	25,83
ITERACIÓN 2	2,81	24,77
ITERACIÓN 3	2,90	25,58
ITERACIÓN 4	2,90	25,58
ITERACIÓN 5	2,86	25,18

TABLA 4 PROMEDIO CONSUMO EMISIÓN HONDA CB1 PRO

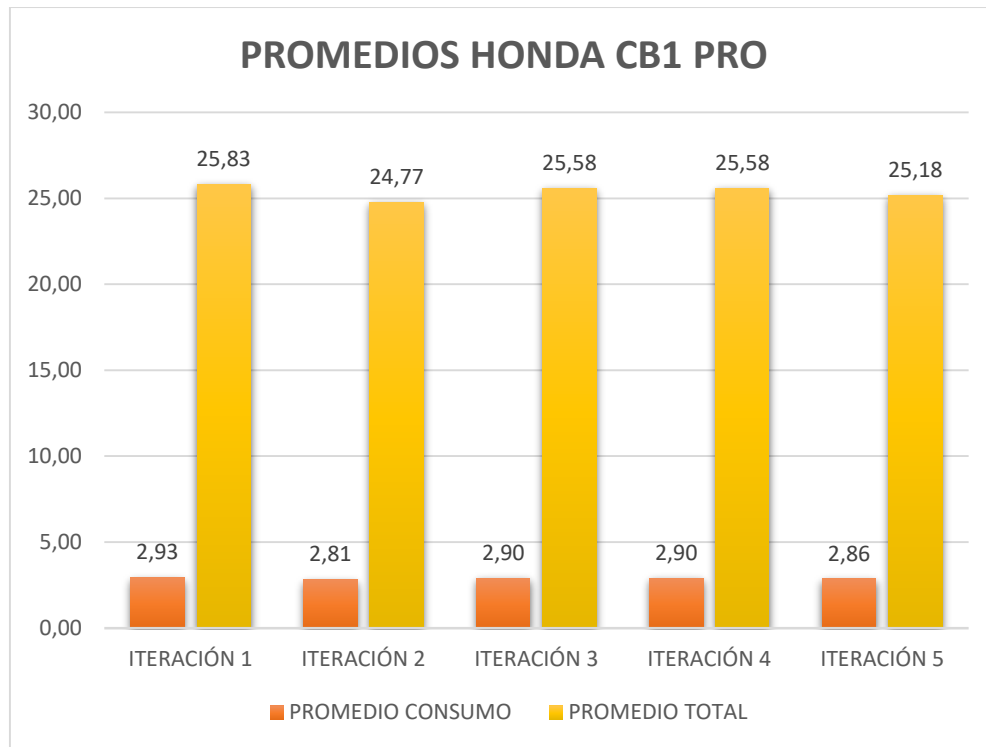


ILUSTRACIÓN 9 PROMEDIO CONSUMO EMISIÓN, HONDA CB1 PRO

## Resultados RC206, YAMAHA LIBERO 125

### PROMEDIO CONSUMO Y EMISIÓN DE GASES PARA LA YAMAHA LIBERO 125

	PROMEDIO CONSUMO (Galón)	PROMEDIO EMISIÓN (KgCO2)
ITERACIÓN 1	5,06	44,54
ITERACIÓN 2	5,09	44,82
ITERACIÓN 3	5,19	45,69
ITERACIÓN 4	5,15	45,39
ITERACIÓN 5	5,16	45,46

TABLA 5 PROMEDIO CONSUMO EMISIÓN YAMAHA LIBERO 125

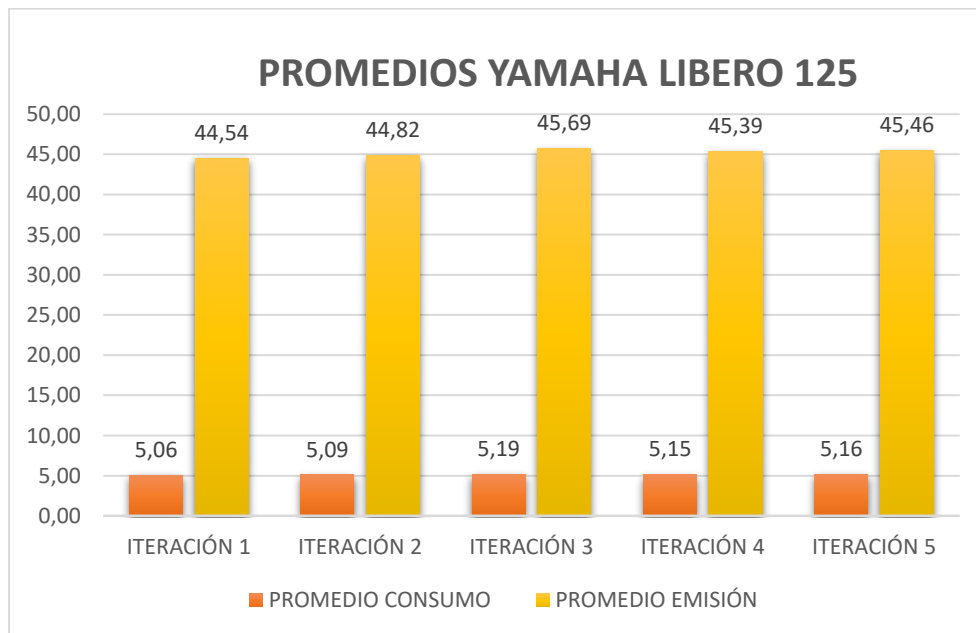
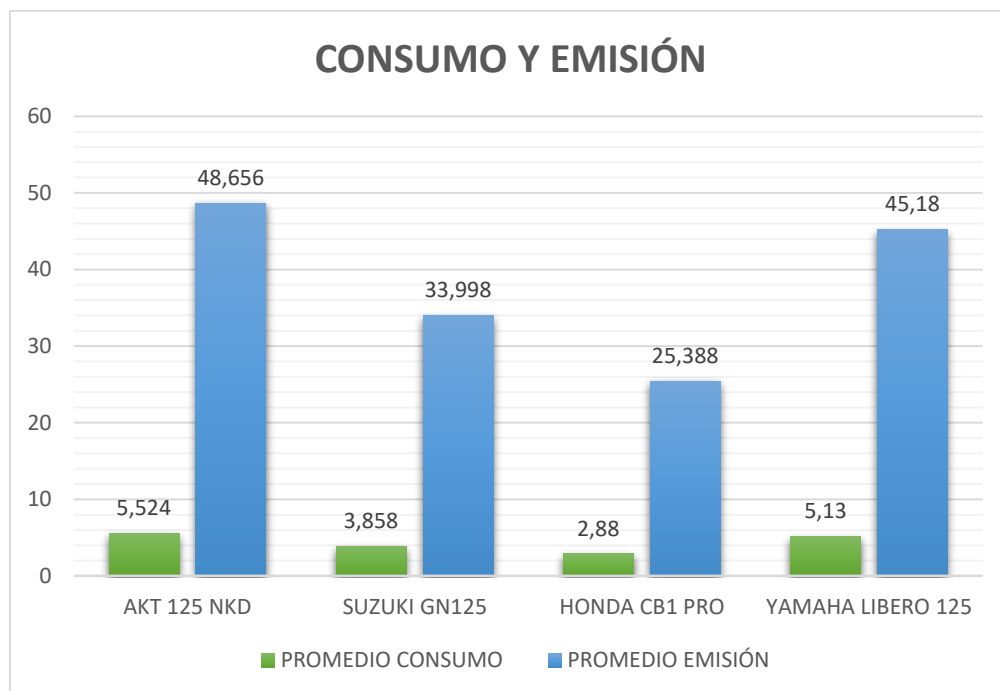


ILUSTRACIÓN 10 PROMEDIO CONSUMO EMISIÓN YAMAHA LIBERO 125

## CONSUMO DE GASOLINA Y EMISIÓN DE GASES EN LAS DIFERENTES MOTOS

	PROMEDIO CONSUMO (Galón)	PROMEDIO EMISIÓN (KgCO <sub>2</sub> )
AKT 125 NKD	5,524	48,656
SUZUKI GN125	3,858	33,998
HONDA CB1 PRO	2,88	25,388
YAMAHA LIBERO 125	5,13	45,18

**TABLA 6 CONSUMO Y EMISIÓN DE LAS MOTOS**



**ILUSTRACIÓN 11 CONSUMO Y EMISIÓN DE LAS MOTOS**

**TABLA 7 TABLA RESUMEN, TIEMPOS ITERACIONES RC206**

	MOTO AKT 125 NKD	Moto SUZUKI GN125	HONDA CB1 PRO	YAMAHA LIBERO 125
Iteración 1	0,187503576	0,183132648	0,230682611	0,189635038
Iteración 2	0,245173931	0,192136288	0,245674372	0,198158026
Iteración 3	0,226181507	0,260185242	0,306698799	0,200662851
Iteración 4	0,197158098	0,23718977	0,23617053	0,209649801
Iteración 5	0,206164837	0,25365901	0,19315505	0,191118002

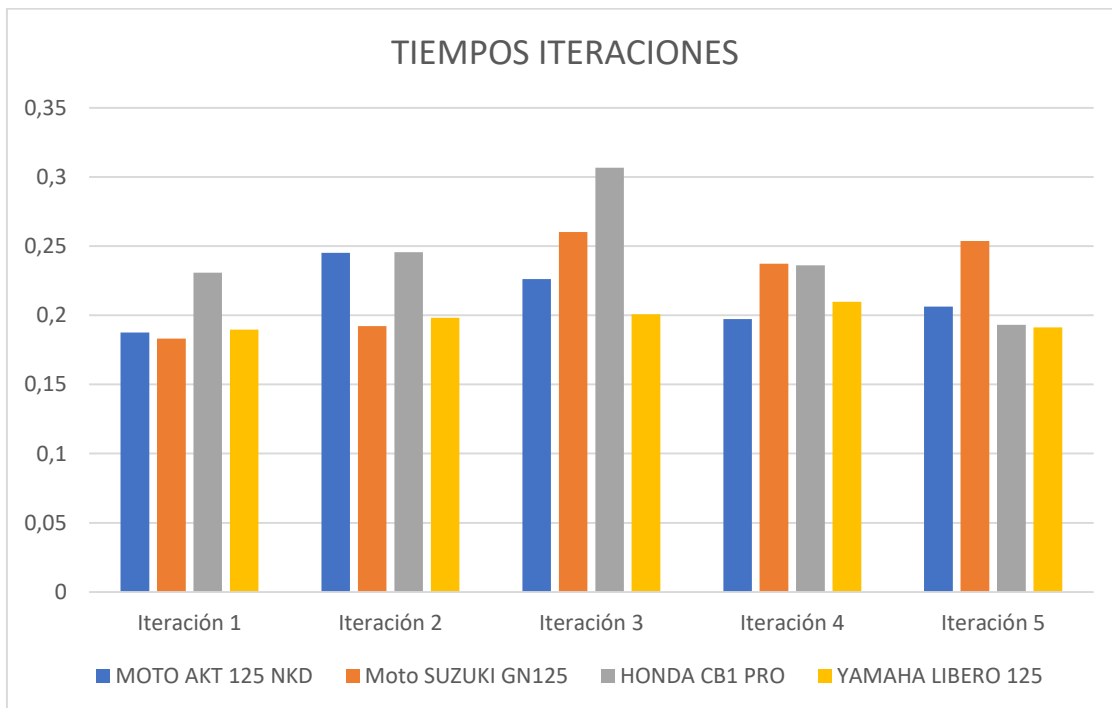


ILUSTRACIÓN 12 GRÁFICO RESUMEN, TIEMPOS ITERACIONES RC206

## 6.1.2 Resultados Archivo R210

Los siguientes resultados están basados en el archivo de SOLOMON **R210**, el cual contiene las coordenadas de 100 clientes y se generaron las siguientes soluciones en 5 iteraciones del algoritmo, con los diferentes tipos de motos.

### Resultados R210, AKT 125 NKD

#### PROMEDIO CONSUMO Y EMISIÓN DE GASES PARA LA AKT 125 NKD

	PROMEDIO CONSUMO (Galón)	PROMEDIO EMISIÓN (KgCO <sub>2</sub> )
ITERACIÓN 1	4,83	42,57
ITERACIÓN 2	4,84	42,59
ITERACIÓN 3	4,86	42,79
ITERACIÓN 4	4,92	43,31
ITERACIÓN 5	4,88	42,98

TABLA 8 PROMEDIOS CONSUMO EMISIÓN AKT 125 NKD

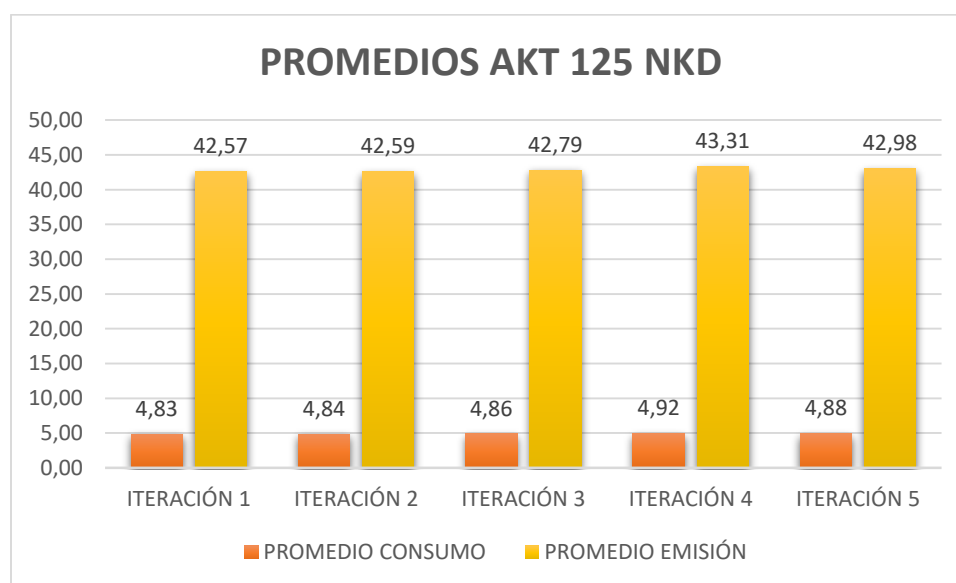


ILUSTRACIÓN 13 PROMEDIOS CONSUMO EMISIÓN AKT 125 NKD

## Resultados R210, SUZUKI GN125

### PROMEDIO CONSUMO Y EMISIÓN DE GASES PARA LA SUZUKI GN125

	PROMEDIO CONSUMO (Galón)	PROMEDIO EMISIÓN (KgCO <sub>2</sub> )
ITERACIÓN 1	3,40	29,92
ITERACIÓN 2	3,48	30,66
ITERACIÓN 3	3,47	30,53
ITERACIÓN 4	3,40	29,98
ITERACIÓN 5	3,47	30,58

TABLA 9 PROMEDIOS CONSUMO EMISIÓN SUZUKI GN125

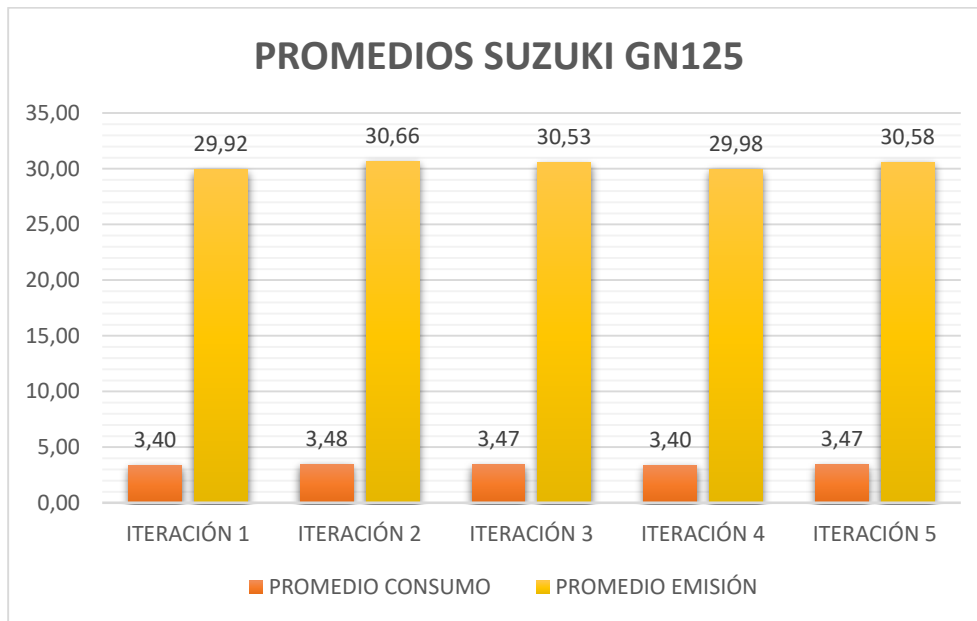


ILUSTRACIÓN 14 PROMEDIOS CONSUMO EMISIÓN SUZUKI GN125

## Resultados R210, HONDA CB1 PRO

### PROMEDIO CONSUMO Y EMISIÓN DE GASES PARA LA HONDA CB1 PRO

	PROMEDIO CONSUMO (Galón)	PROMEDIO EMISIÓN (KgCO2)
ITERACIÓN 1	2,55	22,49
ITERACIÓN 2	2,62	23,11
ITERACIÓN 3	2,63	23,12
ITERACIÓN 4	2,55	22,48
ITERACIÓN 5	2,54	22,48

TABLA 10 PROMEDIOS CONSUMO EMISIÓN HONDA CB1 PRO

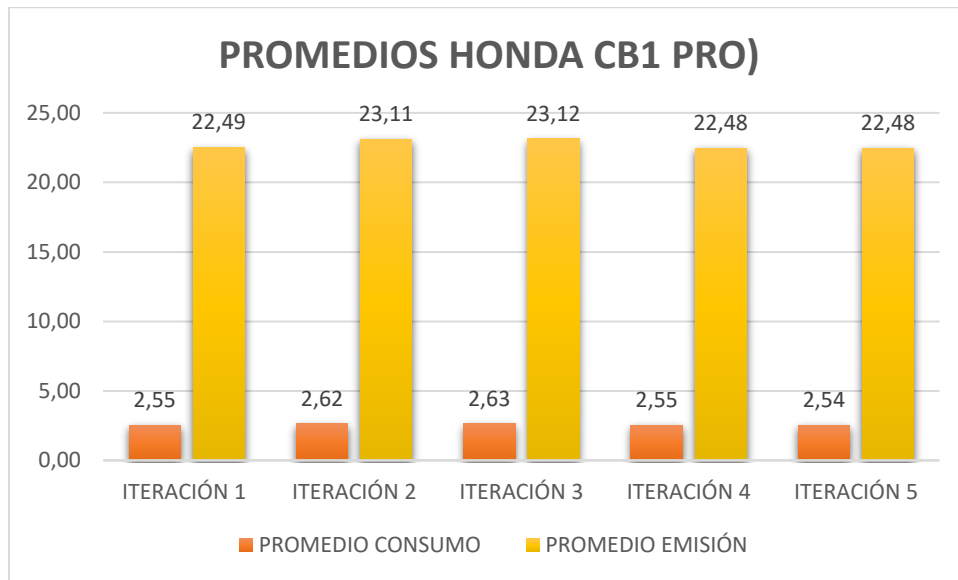


ILUSTRACIÓN 15 PROMEDIOS CONSUMO EMISIÓN HONDA CB1 PRO

## Resultados R210, YAMAHA LIBERO 125

### PROMEDIO CONSUMO Y EMISIÓN DE GASES PARA LA YAMAHA LIBERO 125

	PROMEDIO CONSUMO (Galón)	PROMEDIO EMISIÓN (KgCO2)
ITERACIÓN 1	4,61	40,59
ITERACIÓN 2	4,45	39,23
ITERACIÓN 3	4,64	40,84
ITERACIÓN 4	4,57	40,25
ITERACIÓN 5	4,57	40,22

TABLA 11 PROMEDIO CONSUMO EMISIÓN YAMAHA LIBERO 125

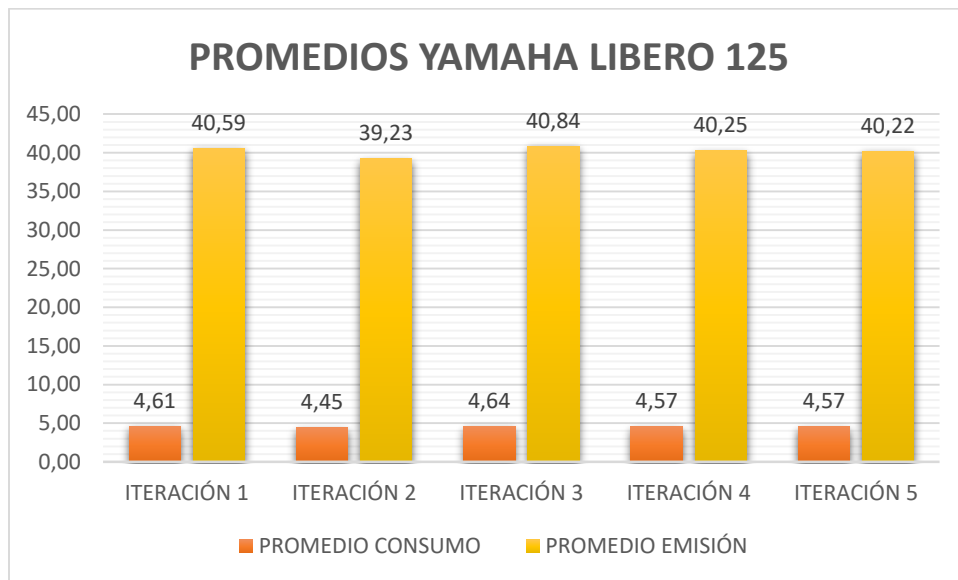


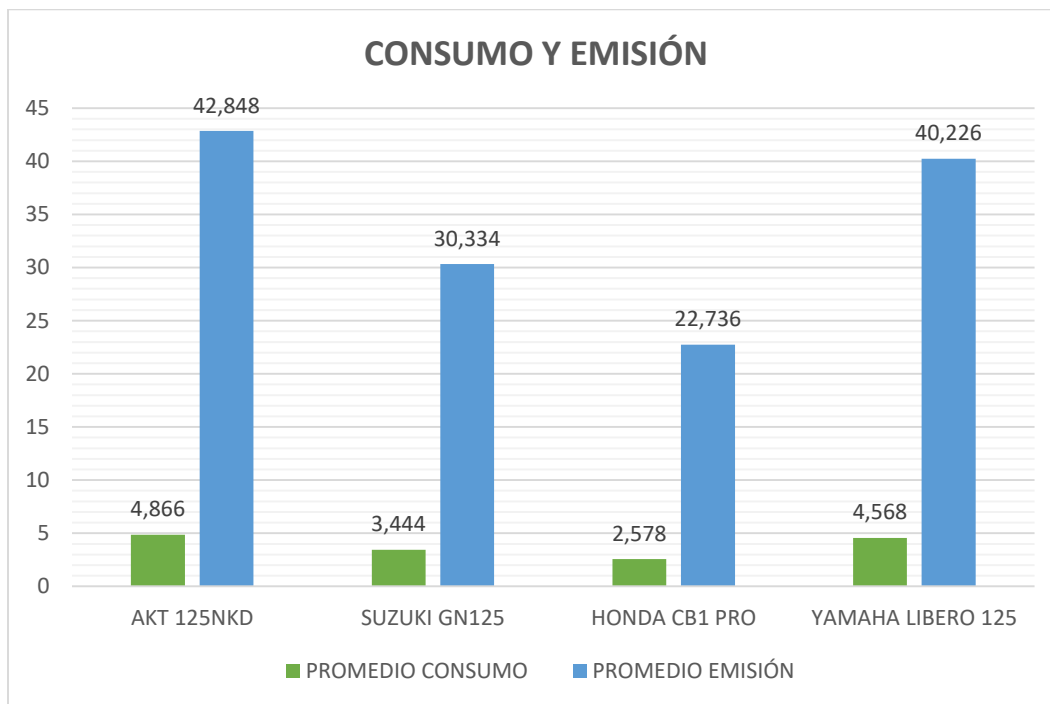
ILUSTRACIÓN 16 PROMEDIO CONSUMO EMISIÓN YAMAHA LIBERO 125



## CONSUMO DE GASOLINA Y EMISIÓN DE GASES EN LAS DIFERENTES MOTOS

	PROMEDIO CONSUMO (Galón)	PROMEDIO EMISIÓN (KgCO2)
AKT 125NKD	4,866	42,848
SUZUKI GN125	3,444	30,334
HONDA CB1 PRO	2,578	22,736
YAMAHA LIBERO 125	4,568	40,226

**TABLA 12 CONSUMO Y EMISIÓN DE LAS MOTOS**



**ILUSTRACIÓN 17 CONSUMO Y EMISIÓN DE LAS MOTOS**

**TABLA 13 TABLA RESUMEN, TIEMPOS ITERACIONES R210**

	AKT 125 NKD (ms)	SUZUKI GN125 (ms)	HONDA CB1 PRO (ms)	YAMAHA LIBERO 125 (ms)
ITERACIÓN 1	0,193118572	0,19515729	0,179616928	0,168011427
ITERACIÓN 2	0,187651157	0,216654539	0,173105717	0,18400836
ITERACIÓN 3	0,164615393	0,172121286	0,183229446	0,244016886
ITERACIÓN 4	0,171121597	0,171140671	0,176012754	0,176029921
ITERACIÓN 5	0,17013979	0,176620722	0,18803215	0,18403244

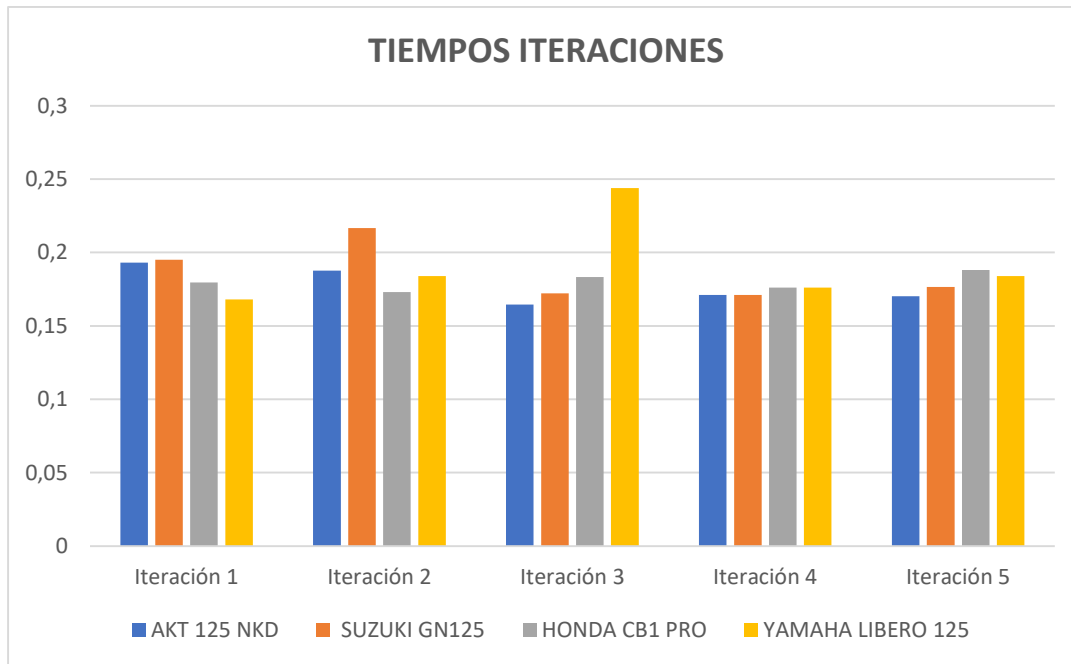


ILUSTRACIÓN 18 GRÁFICO RESUMEN, TIEMPOS ITERACIONES R210

## 6.2 Análisis De Resultados

- De acuerdo con los resultados obtenidos en la ejecución del algoritmo se evidencia claramente que este posee un 100% de efectividad en resolución de caminos, dado que siempre encuentra una ruta para los diferentes agentes, lo que garantiza que todos los problemas que sean abordados tendrán una solución.

- Teniendo en cuenta los resultados presentados anteriormente, en donde se toman como referencia los archivos de SOLOMON RC206 Y R210, para los cuales se realizan cinco iteraciones con cada tipo de moto, se evidencia claramente que el tiempo de respuesta para cada iteración no supera los 0,4 milisegundos, lo cuál demuestra la rapidez del algoritmo para encontrar las diferentes soluciones (**Ver tablas 30 y 59**).

- Considerando que el factor ambiental es un aspecto importante en esta investigación, el cual está reflejado en el cálculo de la emisión de gases de las motos empleadas en los diferentes recorridos, se evidencia en los datos obtenidos que el tipo de moto que tiene menor emisión es la **Honda CB1 PRO**, en donde en promedio para los resultados con el archivo de prueba R210 se tiene una emisión del **22,736** (**Ver tabla 58**) y para el archivo

RC206 es de **25,388 (Ver tabla 29)**; teniendo en cuenta que la presenta mayor emisión es la **AKT 125NKD** con **42,848** para el archivo R210 y para el archivo RC206 es de **48,656 (Ver tablas 29 y 58)**.

- Otro de los aspectos importantes que el algoritmo implementado permite calcular, es el consumo de gasolina, teniendo en cuenta la distancia del recorrido que cada agente debe realizar en la moto; este cálculo les permitirá a los usuarios tener un control mayor de los recursos destinados para la compra de este combustible. En los resultados obtenidos se evidencia claramente el consumo de cada una de las motos, pudiendo concluir en los dos ejemplos (RC206 Y R210) que la moto que menos combustible consume en promedio es la **HONDA CB1 PRO**, donde para las iteraciones del archivo R210 presenta un consumo promedio de **2,578** galones y para el archivo RC206 presenta un consumo de **2,88** galones (**Ver tablas 29 y 58**).

# CAPÍTULO 7. CONCLUSIONES

## 7.1 Conclusiones

- El algoritmo generó soluciones para las pruebas que se realizaron debido a sus características, en las cuales se tuvo control de la población y de cada uno de los individuos a través de los operadores genéticos.
- Los resultados obtenidos permitieron evidenciar que el uso de bases de datos no relacionales como la Neo4j, para este tipo de problemas, genera una eficiencia mayor en la búsqueda de las soluciones, además de brindar gráficamente una ventaja de visualización de los datos, que se obtienen en cada una de las ejecuciones del algoritmo.
- La implementación de este algoritmo para la solución de problemas de ruteo con ventanas de tiempo es de gran utilidad y genera resultados importantes en el área del transporte, debido a la sencillez de la codificación de los datos y a la eficiencia en la generación de las soluciones.
- La implementación del algoritmo para la solución del problema de ruteo de vehículos con ventanas de tiempo, no solo proporciona ventajas en la optimización de las rutas y el tiempo, sino que tiene un valor agregado importante, el cual está reflejado en la medición del consumo de combustible por parte de los vehículos utilizados en el servicio y la emisión de gases contaminantes, siendo estos dos aspectos importantes para las empresas, ya que así no solo podrán controlar sus recursos económicos sino también su factor de contribución a la contaminación del medio ambiente.

## BIBLIOGRAFÍA

- Bäck, T., Fogel, D., & Michalewics, Z. (1997). *Handbook of evolutionary computation*. Bristol: Institute of Physics Pub.
- Ballesteros Silva, P., & Escobar Zuluaga, A. (2016). Revisión del estado del arte del problema de ruteo de vehículos con recogida y entrega (VRPPD). *Ingeniería y Desarrollo*, 463-482.
- Bartz-Beielstein, T. (2003). *Experimental analysis of evolution strategies: Overview and comprehensive introduction*.
- Berbeglia, G., Cordeau, J., & Laporte, G. (2010). Dynamic pickup and delivery problems. *European Journal Of Operational Research*, 8-15.
- Cortés, P., & Onieva, L. (2011). *Ingeniería de organización: Modelos y aplicaciones*. Madrid: Ediciones Díaz de Santos.
- Dantzig, G., & Ramser, J. (1959). The Truck Dispatching Problem. *Management Science*, 80-91.
- Dantzig, G., Fulkerson, R., & Johnson, S. (1954). Solution of a Large-Scale Traveling-Salesman Problem. *Journal Of The Operations Research Society Of America*, 393-410.
- Dean, W. (2018). *Computational Complexity Theory*. Retrieved from <https://plato.stanford.edu/entries/computational-complexity>
- Díaz, V., & González, F. (1996). *Optimización heurística y redes neuronales*. Madrid: Paraninfo.
- El-Sherbeny, N. (2010). Vehicle routing with time windows: An overview of exact, heuristic and metaheuristic methods. *Journal of King Saud University-Science*, 123-131.
- Estévez, P. (1997). Optimización mediante algoritmos genéticos. *Anales del Instituto de Ingenieros de Chile*, 83-92.
- Fogel, L., & Fogel, D. (1995). An Introduction to Evolutionary Programming. *European Conference on Artificial Evolution*, (pp. 21-33).
- Hoffman, K., Padberg, M., & Rinaldi, G. (2001). *Traveling Salesman Problem*. Kluwer Academic Publishers.
- Karp, R. (1986). Combinatorics, complexity and randomness. *Communications of the ACM*, 98-109.

- Laporte, G. (1992). The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 345-358.
- Martí, R. (2003). Procedimientos Metaheurísticos en Optimización Combinatoria. *Matemáticas, Universidad de Valencia*, 3-62.
- Meerendonk, Y. (2010). Extended Period Vehicle Routing Problem Balancing the week of merchandisers in beverage industry. *Faculty of Economics and Business Administration Tilburg University*.
- Melían, B., Pérez, J., & Vega, J. (2003). Metaheurísticas: Una visión global. *Revista Iberoamericana de Inteligencia Artificial*.
- Mitchell, M. (2002). *An introduction to genetic algorithms*. New Delhi: Prentice-Hall of India.
- Moratilla, A., Fernández, E., Sánchez, J., & Vicario, B. (2014). Selección óptima de operadores para el tratamiento de problemas VRP con Algoritmos Genéticos. *Cuarta Conferencia Iberoamericana de Complejidad, Informática y Cibernética: CICIC*.
- Networking and Emerging Optimization. (2018). *Split Delivery VRP | Vehicle Routing Problem*. Retrieved from <http://neo.lcc.uma.es/vrp/vrp-flavors/split-delivery-vrp/>
- Networking and Emerging Optimization. (2018). *VRP WEB*. Retrieved from <http://neo.lcc.uma.es/radi-aeb/WebVRP/>
- Restrepo, J., Medina, P., & Cruz, E. (2008). Un problema logístico de programación de vehículos con ventanas de tiempo (VRPTW). *Scientia Et Technica*, 229-234.
- Rocha, L., González, C., & Orjuela, J. (2011). Una revisión al estado del arte del problema de ruteo de vehículos: Evolución histórica y métodos de solución. *Ingeniería*, 35-55.
- Rodríguez Muro, M. (2005). *Optimización de la genotipificación de ADN como un problema de Selección de Características*. Puebla.
- Rodríguez Ortiz, C. (2010). *Algoritmos heurísticos y metaheurísticos para el problema de localización de regeneradores*.
- Rodriguez Perez, J. (2012). Caracterización, Modelado y Determinación de las rutas de la flota en una empresa de rendering.
- Sait, S., & Youssef, H. (2000). *Iterative Computer Algorithms with Applications in Engineering: Solving Combinatorial Optimization Problems*. Wiley.

- Suarez, O. (2013). Una aproximación a la heurística y metaheurísticas. *INGE@UAN-Tendencias en la Ingeniería*.
- Tansini, L., Urguhart, M., & Viera, O. (2001). Comparing assignment algorithms for the Multi-Depot VRP. *Reportes técnicos*, 01-08.
- Toth, P., & Vigo, D. (2002). *The vehicle routing problem*. Philadelphia: Society for Industrial and Applied Mathematics.
- Zanakis, S., & Evans, J. (1981). Heuristic "Optimization": Why, When, and How to Use It. *Interfaces*, 84-91.

## ANEXOS

### **1. Vehicle Routing Problem with Simultaneous Pickups and Deliveries and Time Windows Considering Fuel Consumption and Carbon Emissions.**

**Autores:** Hao Gang, Gou Zhijing, Yang Peng y Sun Junqing.

**Resumen:** Taking the cost of fuel consumption and carbon dioxide emissions into account, this paper studies the vehicle routing problem with simultaneous pickups and deliveries and time windows (VRP-SPDTW). Firstly, the mathematical model is proposed to describe the problem VRP-SPDTW. The model takes minimizing the total cost as its objective function and decides how many vehicles are employed to service the clients, which clients each vehicle services and in what order each vehicle service its clients. In the objective function the total cost is composed of fuel consumption and carbon dioxide emissions, the startup cost of vehicles, depreciation charge of vehicles and personnel cost. Then an improved genetic algorithm, which adopts effective encoding and a specific crossover operator, is designed to solve the problem. The simulation experiment results show that the algorithm can get a global satisfactory solution, and the loading rate of each vehicle is very high. Hence the mathematical model and the algorithm proposed in this paper can provide a good idea to solve the similar combinational optimization problems.

### **2 Drivers-Inspired Ants for Solving the Vehicle Routing Problem with Time Windows.**

**Título:** Drivers-Inspired Ants for Solving the Vehicle Routing Problem with Time Windows.

**Autores:** Dalicia Bouallouche (dalia.bouallouche@u-bourgogne.fr), Jean-Baptiste Vioix (jean-baptiste.vioix@u-bourgogne.fr), Eric Busvelle (busvelle@univ-tln.fr) y Stéphane Millot (stephane.millot@tedies.fr).

**Resumen:** In our study, we develop a method that merges two information sources within ant's colony optimization heuristic. Namely artificial ants which occurs for short term optimization and transporter's vehicles that occurs in long term and continuous



optimization toward solving the real-world vehicle routing problem. This study is supported by a transporter (Upsilon) of the region of l'Yonne in France and a transport and logistics software development company (Tedies). Our method suits for transporters that use human planners to make decisions about their tours and intending to move to computer planners without drastically upsetting the driver's habits. Hence, the pledge of this study is to take advantage from transport operator's practices to achieve solutions which are as close as possible to the real-world vehicle routing planning, and keep a human control on the way optimal paths are computed and applied.

### **3 A Hybrid Genetic Algorithm for Vehicle Routing Problem With Time Windows.**

**Título:** A Hybrid Genetic Algorithm for Vehicle Routing Problem With Time Windows.

**Autores:** Sripriya. J, M.Tech (sripriya.sri10@gmail.com), Ramalingam.A (a.ramalingam1972@gmail.com), Rajeswari. K, M.Tech (rajeswari2009.it@gmail.com).

**Resumen:** The Vehicle Routing Problem with Time Windows (VRPTW) consists of a homogenous set of vehicles and a set of customer located in a city. In VRPTW all the vehicles starts from the depot visit the customer and end at the depot. Each customer is visited exactly by one vehicle within the specified time window. The objective is to minimize the number of vehicles and total distance travelled simultaneously. This represents the multiobjective Vehicle Routing Problem with Time Windows. The proposed work consists of Hybrid Genetic Search with Diversity Control using the Genetic Algorithm for solving the VRPTW. The Pareto approach is used for finding the set of optimal solutions for achieving the multiobjective. The crossover operator is used for exchanging the best routes, which have shortest distance. Two mutation operators such as relocation mutation operator and split mutation operator were used in this application. In this, it accounts penalty for an infeasible solutions with respect to time-window and duration constrains. The

computations are performed using the instances which are obtained from the VRPLIB.

#### **4. Vehicle Routing with Time Windows Based on Two-stage Optimization Algorithm.**

**Autores:** Linling Liao, Xiushan Cai, Huadong Huang, Yanhong Liu.

**Resumen:** For vehicle routing problem with time windows, a two-stage optimization algorithm with the help of ant colony algorithm and plug-in heuristic algorithm are proposed to solve vehicle routing problem. The first stage is the use of ant colony algorithm AS model to solve the pre-path, and the second stage is pre-path segmentation based on distribution of customer point and the constraints in the model and sub-path re-optimization which take insert heuristic based on the time window classification. Finally, simulation results are used to illustrate that the proposed algorithm can solve the vehicle path planning with time windows easily and effectively compared with other algorithms. Two-stage optimization algorithm proposed in this paper is a new approach to vehicle routing problem with time windows problem.

#### **5. Models and Algorithms for the Vehicle Routing Problem with Time Windows and Other Conditions.**

**Autores:** Guzairov M.B., Yusupova N.I., Smetanina O.N., Rassadnikova E.Yu.

**Resumen:** In the article the authors examine the mathematical modeling of vehicle routing problems. The authors present the mathematical model of the classic vehicle routing problem, the vehicle routing problem with time windows, and the classification of vehicle routing problems, taking into account different aspects of the transport route. The mathematical model and method of solution for Multi Depot Heterogeneous Vehicle Routing Problem with Time Windows as well as safety precautions for the transportation of petrochemical products have been proposed.

## **6. Improvement of Genetic Algorithm for Vehicle Routing Problems with Time Windows.**

**Autores:** Yanfang Deng, Jianling Xiang, Zhuoling Ou.

**Resumen:** Vehicle routing problem with time windows (VRPTW) is of crucial importance in today's industries, especially in logistics distribution. Improvement of genetic algorithm (GA) using an optimized crossover operator is proposed by a complete undirected bipartite graph in order to find an optimal set of delivery routes satisfying the requirements and giving minimal total cost. We tested our algorithm with benchmark instances and compared it with some other heuristics in the literature. The results showed that the proposed algorithm is competitive in terms of the quality of the solutions found. It can be concluded that the proposed algorithm is competitive when compared with other heuristics in the literature.

## **7. Solving Vehicle Routing and Scheduling Problems using Hybrid Genetic Algorithm.**

**Autores:** Bhawna Minocha (bminocha@ascs.amity.edu), Saswati Tripathi (saswati@iift.ac.in) y C. Mohan.

**Resumen:** Vehicle Routing Problem with Time windows (VRPTW) is an example of scheduling in constrained environment. It is a well known NP hard combinatorial scheduling optimization problem in which minimum number of routes have to be determined to serve all the customers within their specified time windows. So far different analytic and heuristic approaches have been tried to solve such problems. In this paper we proposed an algorithm which incorporates a new local search technique with genetic algorithm approach to solve VRPTW scheduling problems in various scenarios.

## **8. Solving Vehicle Routing Problem with Time Windows with Hybrid Evolutionary Algorithm.**

**Autores:** Yong Mao(maodeyong@163.com), Yanfang Deng(dengyf@whut.edu.cn).

**Resumen:** A new hybrid evolutionary algorithm, which based on genetic algorithm (GA), greedy randomized adaptive search procedure (GRASP), the expanding neighborhood search (ENS) strategy and particle swarm optimization (PSO), is introduced in order to solve vehicle routing problem with time windows (VRPTW). The work makes full use of the advantages of each algorithm. The computational experiments were carried out on typical Solomon benchmark problems. The results demonstrate that the proposed method is highly competitive, providing the best-known solutions to minimal distance.

## **9. Vehicle Routing Problem with Time Windows: A Deterministic Annealing approach.**

**Autores:** Mayank Baranwal (baranwa2@illinois.edu), Pratik M. Parekh (bpparekh2@illinois.edu), Lavanya Marla (lavanyam@illinois.edu), Srinivasa M. Salapaka (salapaka@illinois.edu), Carolyn L. Beck (beck3@illinois.edu).

**Resumen:** The Vehicle Routing Problem with Time-Windows (VRPTW) is an important problem in allocating resources on networks in time and space. We present in this paper a Deterministic Annealing (DA)-based approach to solving the VRPTW with its aspects of routing and scheduling, as well as to model additional constraints of heterogeneous vehicles and shipments. This is the first time, to our knowledge, that a DA approach has been used for problems in the class of the VRPTW. We describe how the DA approach can be adapted to generate an effective heuristic approach to the VRPTW. Our DA approach is also designed to not get trapped in local minima, and demonstrates less sensitivity to initial solutions. The algorithm trades off routing and scheduling in an n-dimensional space using a tunable parameter that allows us to generate qualitatively good solutions. These solutions differ in the degree of intersection of the routes, making the case for

transfer points where shipments can be exchanged. Simulation results on randomly generated instances show that the constraints are respected and demonstrate near optimal results (when verifiable) in terms of schedules and tour length of individual tours in each solution.

## **10. Selección óptima de operadores para el tratamiento de problemas VRP con Algoritmos Genéticos.**

**Autores:** Antonio Moratilla, Eugenio Fernández, Juan José Sánchez, Borja Vicario.

**Resumen:** Los problemas conocidos como VRP (Vehicle Routing Problem) consisten, como es sabido, en asignar a cada vehículo una ruta de clientes a los que entregar un conjunto de productos, de manera que se minimice el coste del transporte, partiendo, habitualmente, de un depósito central, una flota de vehículos fija y un conjunto de clientes fijos. Existen numerosas variantes de este problema, siendo la más conocida la que trabaja con capacidades (CVRP), que limita la cantidad de productos que puede transportar cada vehículo. Para la resolución de este tipo de problemas existen numerosas aproximaciones, siendo las técnicas metaheurísticas las más utilizadas y, entre ellas, los Algoritmos Genéticos (AG). No obstante, el número de diferentes aproximaciones existentes en la literatura a los diferentes parámetros de un AG (cruce, mutación, ...) es tal, que en la actualidad no es fácil abordar una resolución de un problema CVRP en la práctica de manera directa. En este trabajo se pretende simplificar esta tarea, analizando las aproximaciones más conocidas con conjuntos de datos CVRP estándares, y mostrando las configuraciones de parámetros que ofrecen mejores resultados.

## **11. An algorithm for the routing problem with split deliveries and time windows (SDVRPTW) applied on retail SME distribution activities.**

*Un algoritmo para el problema de ruteo de vehículos con entregas divididas y ventanas de tiempo (SDVRPTW) aplicado a las actividades de distribución de PYMEs del comercio al por menor.*

**Autores:** Juan Sepúlveda (jcsepulvedas@unal.edu.co), John Wilmer Escobar (jwescobar@javerianacali.edu.co) y Wilson Adarme-Jaimes (wadarme@unal.edu.co).

**Resumen:** In this paper, particular conditions of retail trade SMEs was analyzed, identifying not enough financial resources for using powerful tools for solve vehicle routing problem (VRP). On the other hand, in literatura revised could not be identified studies about application of current approaches for solving VRP in SMEs. Additionally because of high cost, commercial software do not fit investment budget of those companies. Through a simple insertion heuristic for VRP with split deliveries and time windows (SDVRPTW), developed on an accessible technology platform like Microsoft® Excel™, was validated that SDVRPTW is an appropriate approach for solving vehicle routing problem on retail trade SMEs. Computational results show that the heuristic proposed can reduce about 50% the fleet size.

## **12. El problema de ruteo de vehículos [VRP] y su aplicación en medianas empresas colombianas.**

*Vehicle Routing Problem [VRP] and its applications in Colombian medium-sized companies*

**Autores:** Jairo Arboleda Zúñiga, MSc (jarboledaz@usc.edu.co), Astrid Xiomara López (xioma1978@hotmail.com) y Yéssica Lorena Lozano (yessica.lozano.guzman@gmail.com).

**Resumen:** This article presents a review of the literature about some of the most important mathematical, heuristic and meta-heuristics models contributing to the solution of the Vehicle Routing Problem [VPR] –considered as an complex problem

by being stochastic and polynomial—, which is determined by a set of routes that begin and end at a warehouse or deposit and each route is carried out by a single vehicle, on a route, through which global transportation costs are minimized and the demand and other operating constraints are satisfied. The paper also includes the advance of results obtained in two medium-sized Colombian companies. As a part of research project, they are beginning its implementation, obtaining satisfactory results in their process of transportation logistics and products and services distribution.

### **13. Alternativa Heurística MCM para Problemas de Ruteo de Vehículos.**

*MCM Heuristic Alternative for Vehicle Routing Problem-Solving*

**Autores:** José Luis Flores Flores (lsigma@live.com.mx) y Manuel Alvarez-Madrigal (mmadrigal@itesm.mx).

**Resumen:** The Vehicle Routing Problem (VRP) involves a major mathematical complexity to solve it. This impedes its use in small and medium size organizations, because an investment in specialized software and trained personnel is required. The methods used to find an optimal solution to the VRP start with an improving workable solution. This initial solution can be generated randomly, calculated by some other method, or even a solution provided by the user can also be used. In this paper, we present an algorithm to obtain a feasible solution to the problem of VRP called Mass Center Method (MCM). The method is easy to perform and its performance differs little from the final solutions generated by commercial algorithms, therefore, it could be used as an approximation to the solution of the problem. This can help to extend the application of the VRP.

## **14. Algoritmos Meméticos Aplicados A La Resolución De Un Problema De Ruteo De Vehículos Periodico.**

**Autores:** A. Méndez, D. Palumbo, M. Carnero, J. Hernández (gop@ing.unrc.edu.ar)

**Resumen:** La provisión de servicios tales como la recolección de residuos infecciosos, residuos urbanos, recolección y transporte de leche en los tambos, transporte de personas, etc, tienen a menudo grandes problemas de prestación y altos costos de recolección y procesamiento, por lo que la optimización de sus recorridos tiene un fuerte incentivo económico. Los problemas de ruteo representan una de las más importantes clases de problemas dentro de la logística computacional. En este trabajo se presenta una metodología para la resolución de un problema que incluye la planificación de un esquema de visitas y construcción del conjunto de rutas óptimas, que se describe como un problema de ruteo de vehículo periódico (PVRP, Periodic Vehicle Routing Problem), aplicado a un caso real de recolección de residuos infecciosos en la ciudad de Río Cuarto. Es conocido que el PVRP está incluido dentro de los problemas catalogados como NP-Hard, razón por la cual las heurísticas aparecen como métodos de resolución atractivos por su capacidad de brindar soluciones de alta calidad en tiempos razonables, para instancias de tamaño considerable, tal como aparecen en las aplicaciones reales. En el procedimiento de resolución propuesto, en este trabajo, se construye un Algoritmo Memético, que está basado en técnicas de Computación Evolutiva equipadas con diferentes y variados mecanismos de búsqueda local que aseguran la explotación intensiva de regiones promisorias del espacio de búsqueda. Se presenta la metodología y su desempeño para la optimización de la prestación del servicio de recolección diferenciada y transporte de residuos patógenos.



## **15. Diseño de Rutas de Transporte Escolar con Ventanas Temporales Móviles.**

**Autores:** José Luis Moura, Ángel Ibeas (mourajl@unican.es) y Luigi dell'Olio (ldell@ubu.es).

**Resumen:** This article puts forward a methodology for designing a system for school transport which, apart from designing the routes, specifies the school opening times. Traditionally school time tables have always been identical in the same area meaning many buses have to be used at once. This work suggests staggering school opening times in order to minimize the number of buses and thereby reduce the enormous costs involved in running them. The methodology is based on: a first phase which addresses the classic routing problem using mixed integer lineal programming and a second phase which uses bi-level programming to find the vector for school opening times, which, when staying within the constraints of the problem, minimizes the direct costs of the system. At the upper level of the bi-level optimization problem an evaluation is made of the system costs based on a model developed by the authors assigning costs within school transport services. The lower level finds the best combination of optimum routes for the same bus.

## **16. Método heurístico para el problema de ruteo de vehículos aplicado a la empresa distribuidora Representaciones Continental.**

**Autores:** Julio Fuentes Vidal, Marco Parra León y Hernando Alexánder Gutiérrez.

**Resumen:** This paper presents the feasibility of applying a heuristic model of insertion for the vehicle routing problem with time windows and heterogeneous fleet of a liquor distribution company. By analyzing the initial data supplied by the company and the results obtained, the purpose is to determine whether the heuristic model improves in terms of time and cost of logistics operation, as well as to validate the application of the heuristic method. Based on the results obtained, the article focuses on claiming how the insertion heuristic applied to the routing problem behaves in an expected and correct manner. Finally, based on the results obtained from the investigation, a few suggestions are made, such as exporting the reading

of the customer information and the demands generated each day to a database, investigating with other heuristic models proposed by other authors, developing a new program based on a different heuristics and testing how the programming of these models would behave in other programs.

### **17. A set-covering based heuristic algorithm for the periodic vehicle routing problem**

**Autores:** V. Cacchiani, V.C. Hemmelmayr y F. Tricoireb.

**Resumen:** We present a hybrid optimization algorithm for mixed-integer linear programming, embedding both heuristic and exact components. In order to validate it we use the periodic vehicle routing problem (PVRP) as a case study. This problem consists of determining a set of minimum cost routes for each day of a given planning horizon, with the constraints that each customer must be visited a required number of times (chosen among a set of valid day combinations), must receive every time the required quantity of product, and that the number of routes per day (each respecting the capacity of the vehicle) does not exceed the total number of available vehicles. This is a generalization of the well-known vehicle routing problem (VRP). Our algorithm is based on the linear programming (LP) relaxation of a set-covering-like integer linear programming formulation of the problem, with additional constraints. The LP-relaxation is solved by column generation, where columns are generated heuristically by an iterated local search algorithm. The whole solution method takes advantage of the LP-solution and applies techniques of fixing and releasing of the columns as a local search, making use of a tabu list to avoid cycling. We show the results of the proposed algorithm on benchmark instances from the literature and compare them to the state-of-the-art algorithms, showing the effectiveness of our approach in producing good quality solutions. In addition, we report the results on realistic instances of the PVRP introduced in Pacheco et al. (2011) [24] and on benchmark instances of the periodic traveling salesman problem (PTSP), showing the efficacy of the proposed algorithm on these as well. Finally, we report the new best known solutions found for all the tested problems.

**18. Metaheurísticas aplicadas al ruteo de vehículos. Un caso de estudio.  
Parte 1: formulación del problema y estudio.**

*Metaheuristics applied to vehicle routing. A case study. Parte 1: formulating the problema.*

**Autores:** Guillermo González Vargas y Felipe González Aristizábal.

**Resumen:** This paper deals with VRP (vehicle routing problem) mathematical formulation and presents some methodologies used by different authors to solve VRP variation. This paper is presented as the springboard for introducing future papers about a manufacturing company's location decisions based on the total distance traveled to distribute its product.

**19. Diseño de un Modelo de Ruteo de Vehículos: Caso de Estudio UMNG Sede Campus.**

**Autores:** Yimy Alexander Hernández Ortiz (yimy.hernandez@unimilitar.edu.co ),

Wilfrido Javier Arteaga Sarmiento (wilfrido.arteaga@unimilitar.edu.co),

Nataly Lorena Guarín Cortés (nataly.guarin@unimilitar.edu.co).

**Resumen:** El artículo muestra la investigación que se lleva a cabo para diseñar un sistema de ruteo de vehículos para la UMNG, la cual es una sede de un campus universitario, en donde se pueda optimizar los costos de transporte y el servicio a la comunidad académica, mediante el uso de los principios de programación matemática y el empleo de técnicas heurísticas y metaheurísticas, teniendo en cuenta que la naturaleza del problema a abordar requiere de estos métodos para alcanzar una solución óptima. El problema consiste en localizar una o varias rutas con origen desconocido y destino común, en donde se han establecido paraderos por los cuales los buses deben recoger a los funcionarios del campus que allí estén.

La importancia y la eficiencia de este tipo de problemas radica en que al ser aplicados en sistemas de distribución se requiere que generen impacto sobre los

costos asociados a las operaciones de la empresa, teniendo en cuenta además la complejidad de la distribución física, las exigencias del servicio, el cambio de legislación constante, etc.

## **20. Modelo matemático para la planificación de servicios y programación de rutas en empresas prestadoras de servicios de control de plagas.**

**Autor:** John Willmer Escobar (john.wilmer.escobar@correounivalle.edu.co)

**Resumen:** This paper addresses the problem of scheduling of services and planning of routes for companies which offer the service of pest control (CP) by considering the minimization of costs related to the distance traveled by the used vehicles and the cost of the cost of idle time of operators. The problem considers scheduled activities, dates not available and installed capacity, and data of demand previously provided by the customers. The problem consists of the scheduling and planning of the routes by considering time windows. In particular, it is proposed a mixed integer linear programming model to improve the logistic management process of companies belonging to this sector. The model has been tested with data obtained from a Colombian company that provides the CP services in the main Colombian cities. The results show the importance and efficiency of the proposed methodology as an alternative to the solution of the considered problem.

**Resultados RC206, AKT 125 NKD**  
**ITERACIÓN 1**

Esta iteración se realiza teniendo en cuenta que la moto utilizada para hacer el recorrido es **AKT 125 NKD**

**Cantidad nodos franja 1 = 53 Cantidad agentes = [14, 13, 13, 13]**

**Cantidad nodos franja 2 = 47 Cantidad agentes = [16, 16, 15]**

ITERACIÓN	TIEMPO (ms)
Iteración 1	0.18750357627868652
Iteración 2	0.24517393112182617
Iteración 3	0.2261815071105957
Iteración 4	0.1971580982208252
Iteración 5	0.20616483688354492

**TABLA 14 TIEMPOS EN CADA ITERACIÓN, AKT 125 NKD**

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	960.09	6.86	60.42
Agente 1	867.98	6.2	54.61
Agente 2	770.84	5.51	48.53
Agente 3	445.71	3.18	28.01
Agente 4	1048.47	7.49	65.97
Agente 5	806.45	5.76	50.73
Agente 6	635.39	4.54	39.99

**TABLA 15. ITERACIÓN 1, RC206, AKT 125 NKD**

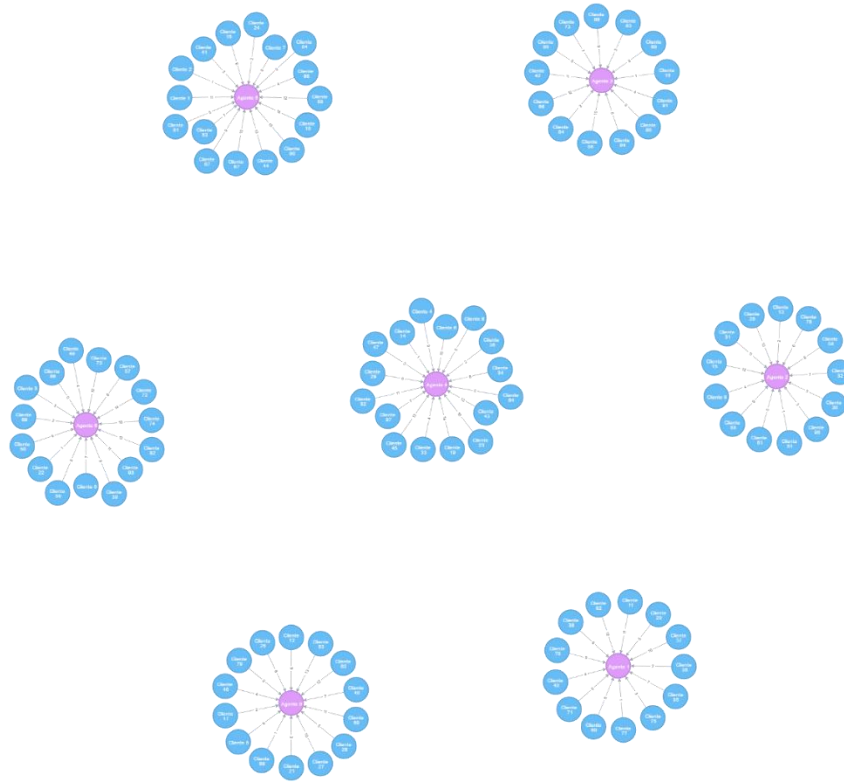


ILUSTRACIÓN 19 ITERACIÓN1, AKT 125 NKD

## ITERACIÓN 2

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	994.8	7.11	62.62
Agente 1	826.83	5.91	52.06
Agente 2	697.22	4.98	43.86
Agente 3	403.03	2.88	25.37
Agente 4	1018.45	7.27	64.03
Agente 5	871.29	6.22	54.79
Agente 6	588	4.2	36.99

TABLA 16 ITERACIÓN 2, RC206 , AKT 125 NKD

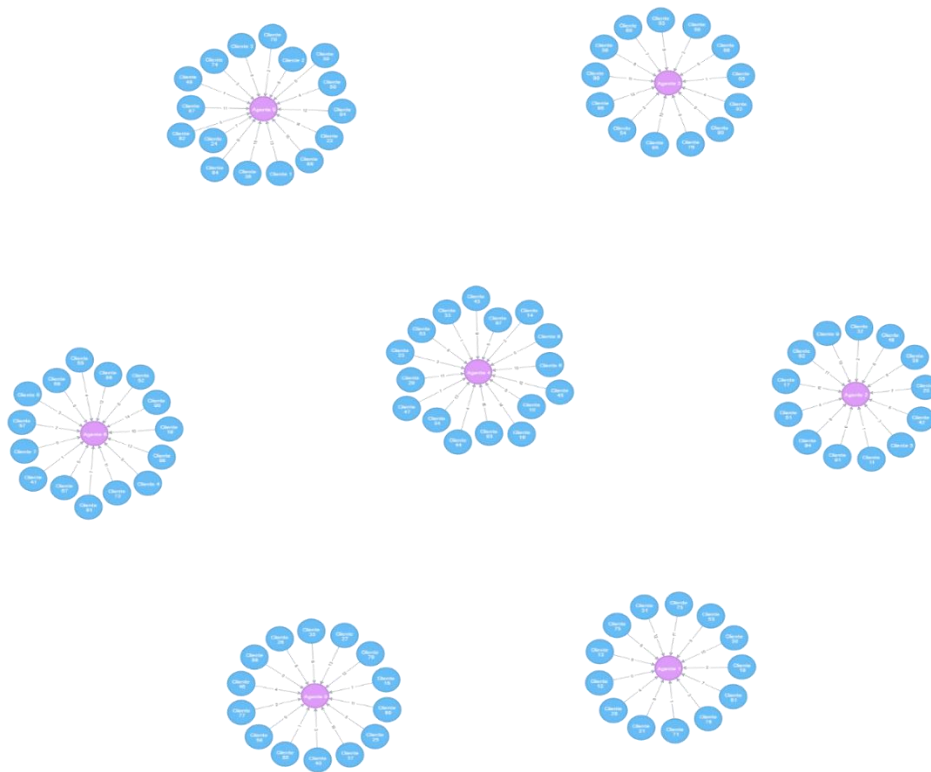


ILUSTRACIÓN 20 ITERACIÓN 2, AKT 125 NKD

### ITERACIÓN 3

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	971.05	6.94	61.13
Agente 1	839.91	6	52.85
Agente 2	696.74	4.98	43.86
Agente 3	459.81	3.28	28.89
Agente 4	970.43	6.93	61.04
Agente 5	875.36	6.25	55.05
Agente 6	636.99	4.55	40.08

TABLA 17 ITERACIÓN 3, RC206, AKT 125 NKD

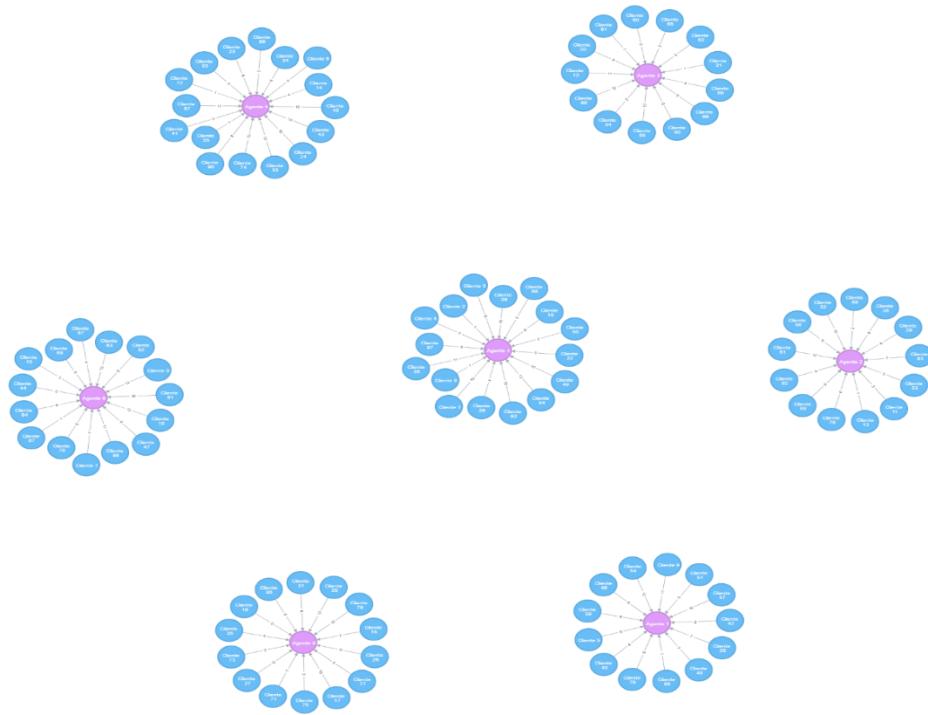


ILUSTRACIÓN 21 ITERACIÓN 3, AKT 125 NKD

#### ITERACIÓN 4

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	993.58	7.1	62.54
Agente 1	838.01	5.99	52.76
Agente 2	706.22	5.04	44.39
Agente 3	412.08	2.94	25.9
Agente 4	976.16	6.97	61.39
Agente 5	896.36	6.4	56.37
Agente 6	538.16	3.84	33.82

TABLA 18 ITERACIÓN 4, RC206



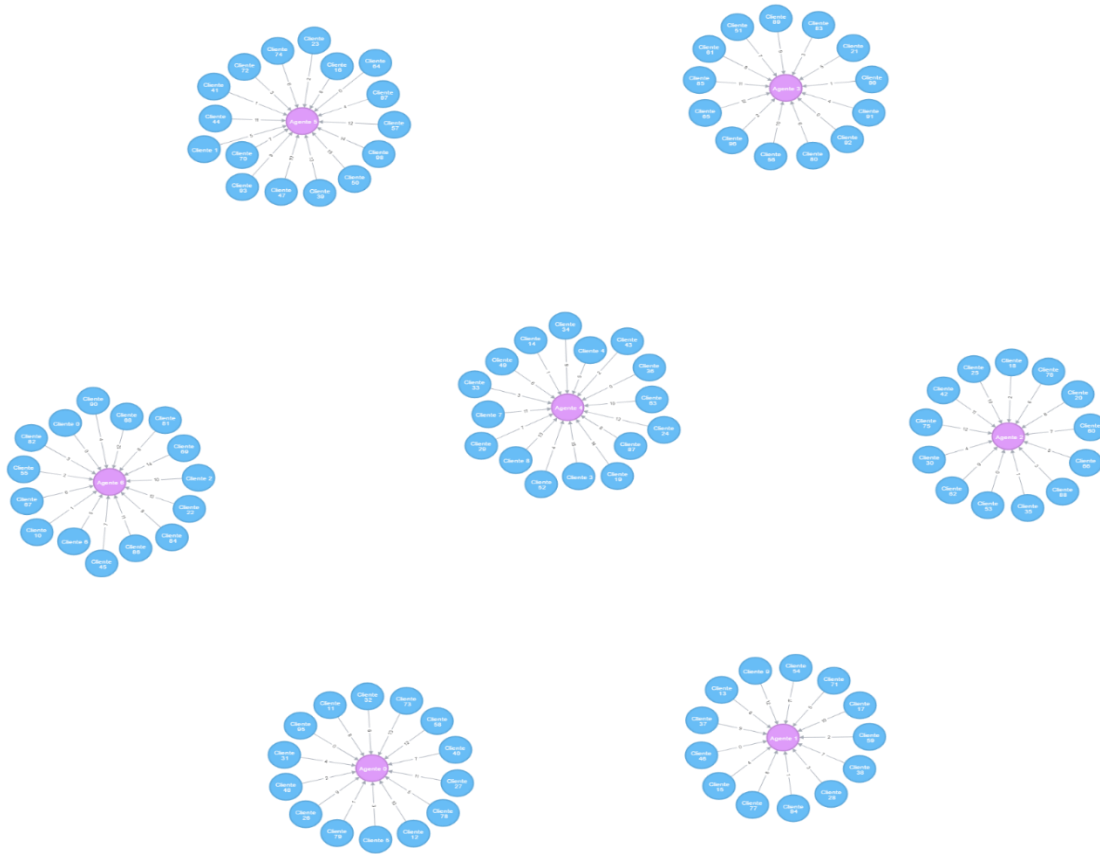


ILUSTRACIÓN 22 ITERACIÓN 4, AKT 125 NKD

### ITERACIÓN 5

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	945.48	6.75	59.45
Agente 1	822.09	5.87	51.7
Agente 2	701.09	5.01	44.13
Agente 3	451.13	3.22	28.36
Agente 4	949.73	6.78	59.72
Agente 5	889.39	6.35	55.93
Agente 6	565.86	4.04	35.58

TABLA 19 ITERACIÓN 5, RC206, AKT 125 NKD

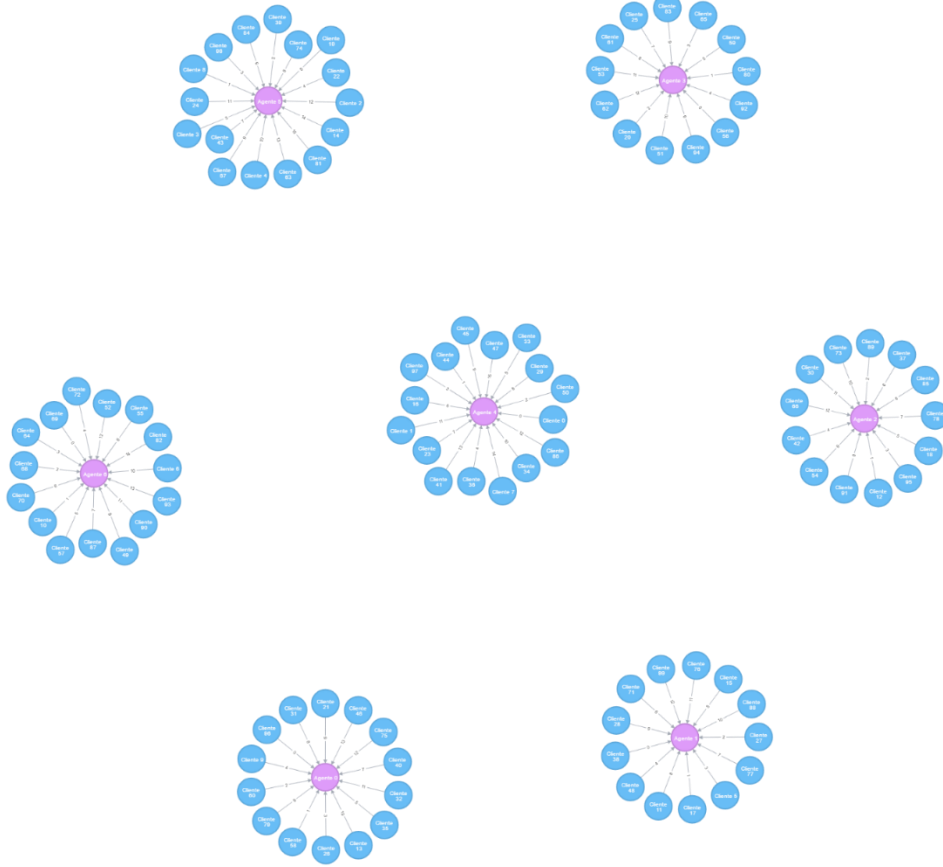


ILUSTRACIÓN 23 ITERACIÓN 5, AKT 125 NKD

**Resultados RC206, SUZUKI GN125**  
**ITERACIÓN 1**

Esta iteración se realiza teniendo en cuenta que la moto utilizada para hacer el recorrido es **SUZUKI GN125**

ITERACIÓN	TIEMPO (ms)
ITERACIÓN 1	0.18313264846801758
ITERACIÓN 2	0.19213628768920898
ITERACIÓN 3	0.26018524169921875
ITERACIÓN 4	0.23718976974487305
ITERACIÓN 5	0.2536590099334717

**TABLA 20 TIEMPOS ITERACIONES, SUZUKI GN125**

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	994.76	4.97	43.78
Agente 1	777.53	3.89	34.26
Agente 2	686.5	3.43	30.21
Agente 3	461.3	2.31	20.35
Agente 4	1085.66	5.43	47.83
Agente 5	848.07	4.24	37.35
Agente 6	607.37	3.04	26.78

**TABLA 21 ITERACIÓN 1 RC206, SUZUKI GN125**

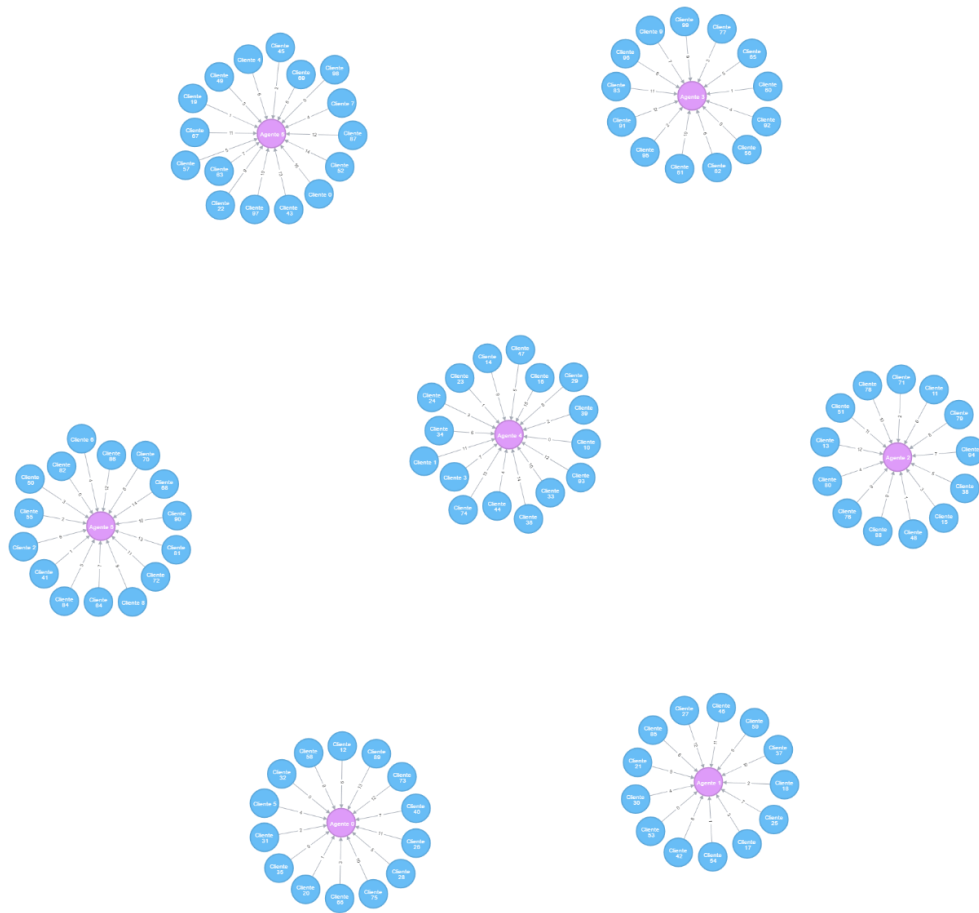


ILUSTRACIÓN 24 ITERACIÓN 1, RC206, SUZUKI GN125

## ITERACIÓN 2

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	914.85	4.57	40.25
Agente 1	816.28	4.08	35.94
Agente 2	729.46	3.65	32.15
Agente 3	533.55	2.67	23.52
Agente 4	1043.78	5.22	45.98
Agente 5	894.62	4.47	39.37
Agente 6	545	2.73	24.05

TABLA 22 ITERACIÓN 2 RC206, SUZUKI GN125

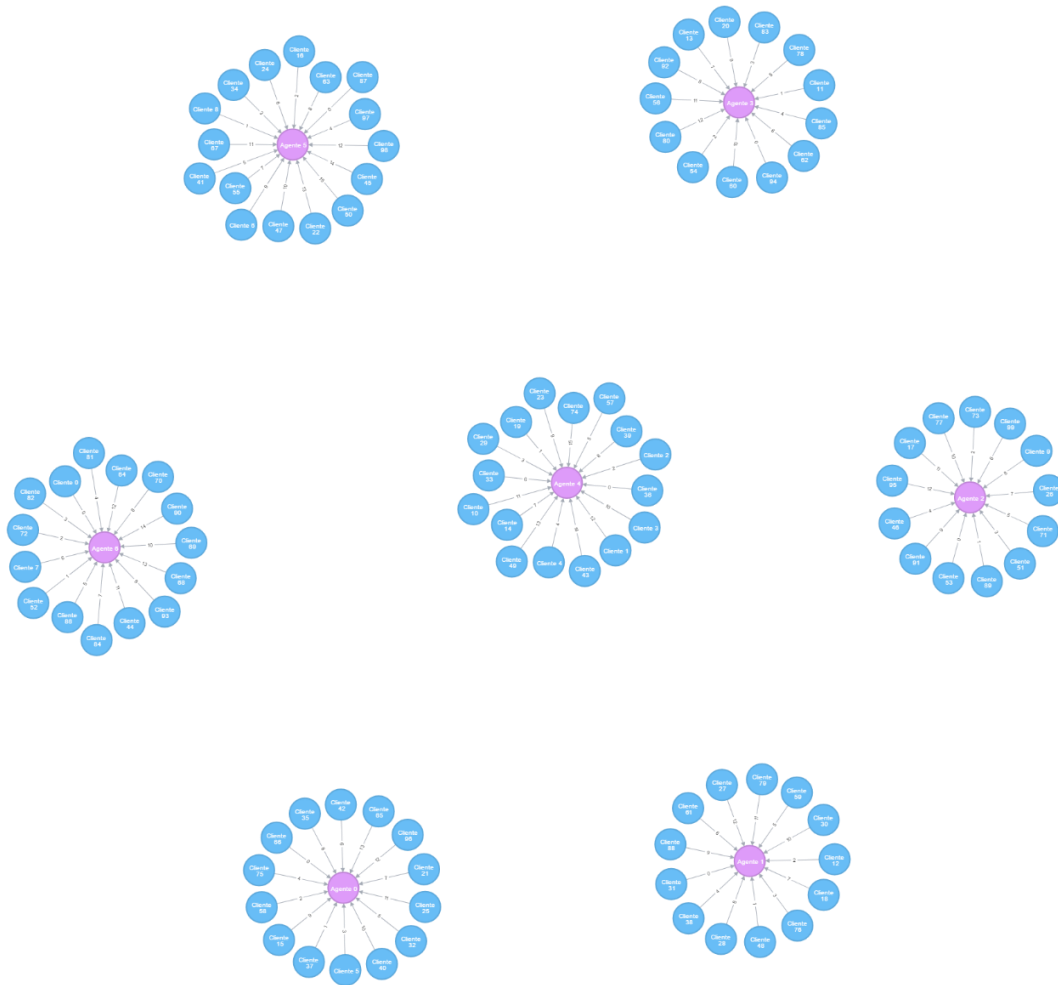


ILUSTRACIÓN 25 ITERACIÓN 2 RC206, SUZUKI GN125

### ITERACIÓN 3

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	920.22	4.6	40.52
Agente 1	827.28	4.14	36.47
Agente 2	722.17	3.61	31.8
Agente 3	425.75	2.13	18.76
Agente 4	931.13	4.66	41.05
Agente 5	872.23	4.36	38.4
Agente 6	618.38	3.09	27.22

TABLA 23 ITERACIÓN 3 RC206, SUZUKI GN125

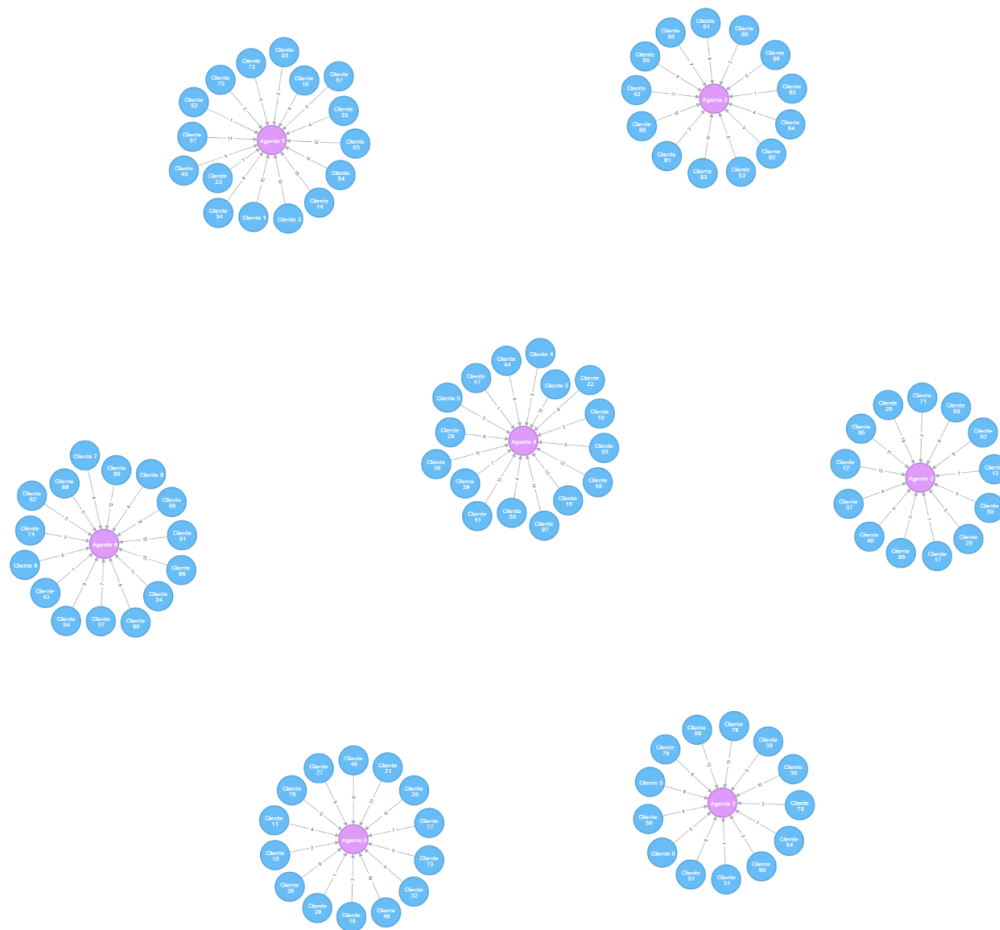


ILUSTRACIÓN 26 ITERACIÓN 3 RC206, SUZUKI GN125

**ITERACIÓN 4**

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	958.19	4.79	42.19
Agente 1	857.94	4.29	37.79
Agente 2	709.27	3.55	31.27
Agente 3	473.44	2.37	20.87
Agente 4	1004.95	5.02	44.22
Agente 5	866.41	4.33	38.14
Agente 6	547.55	2.74	24.13

TABLA 24 ITERACIÓN 4 RC206, SUZUKI GN125

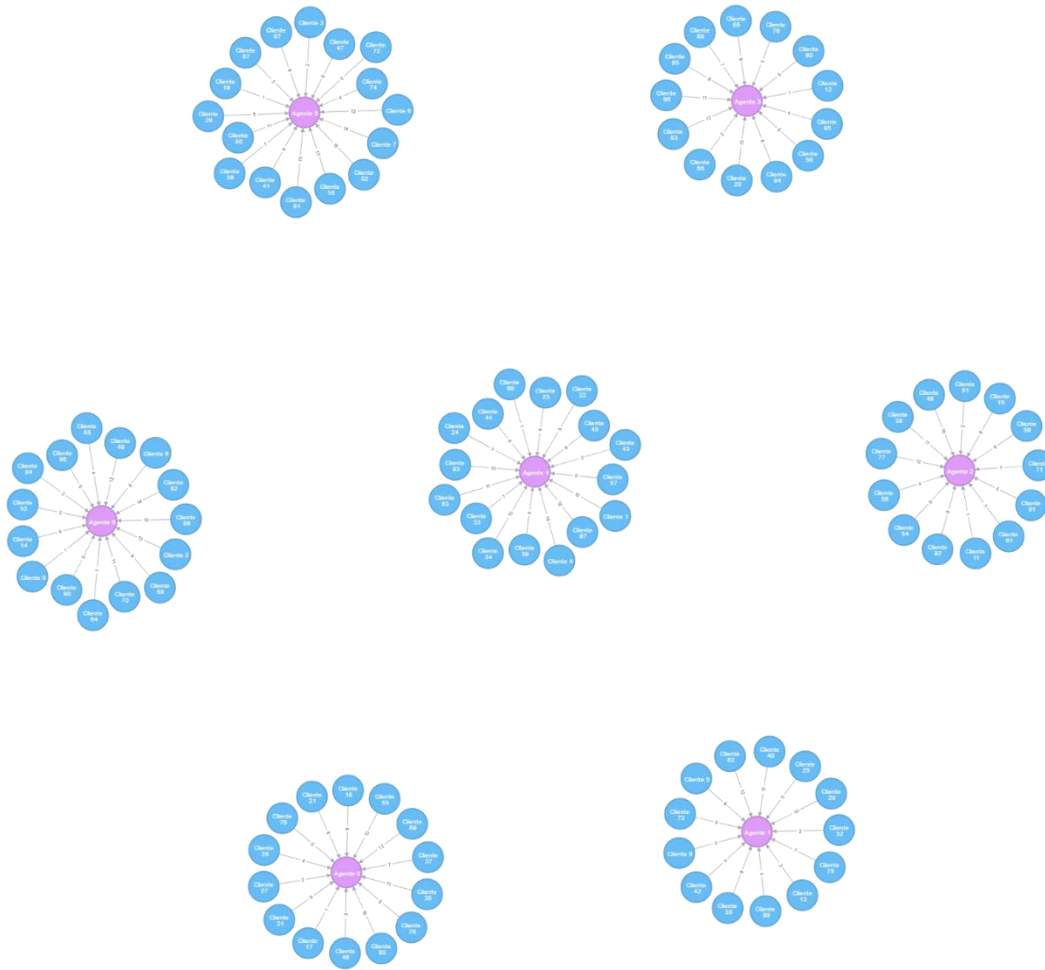


ILUSTRACIÓN 27 ITERACIÓN 4 RC206, SUZUKI GN125

### ITERACIÓN 5

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	940.58	4.7	41.4
Agente 1	926.47	4.63	40.78
Agente 2	631.36	3.16	27.83
Agente 3	389.3	1.95	17.18
Agente 4	957	4.79	42.19
Agente 5	957.97	4.79	42.19
Agente 6	536.33	2.68	23.61

TABLA 25 ITERACIÓN 5 RC206, SUZUKI GN125

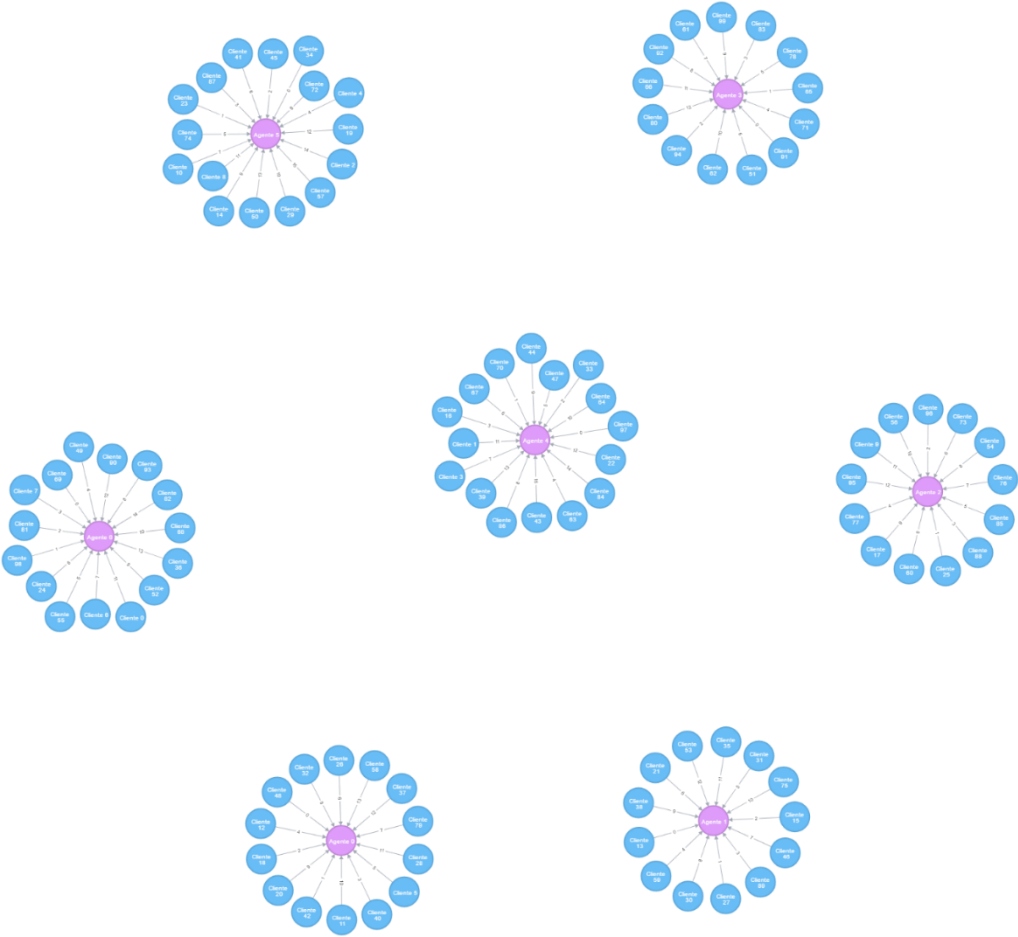


ILUSTRACIÓN 28 ITERACIÓN 5 RC206, SUZUKI GN125



## RESULTADOS HONDA CB1 PRO ITERACIÓN 1

Esta iteración se realiza teniendo en cuenta que la moto utilizada para hacer el recorrido es la **HONDA CB1 PRO**.

ITERACIÓN	TIEMPO (ms)
ITERACIÓN 1	0.2306826114654541
ITERACIÓN 2	0.24567437171936035
ITERACIÓN 3	0.3066987991333008
ITERACIÓN 4	0.23617053031921387
ITERACIÓN 5	0.19315505027770996

TABLA 26 TIEMPOS ITERACIONES, HONDA CB1 PRO

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	995.28	3.74	32.94
Agente 1	806.96	3.03	26.69
Agente 2	751.84	2.83	24.93
Agente 3	407.51	1.53	13.48
Agente 4	963.33	3.62	31.88
Agente 5	898.84	3.38	29.77
Agente6	639.35	2.4	21.14

TABLA 27 ITERACIÓN 1, RC206, HONDA CB1 PRO

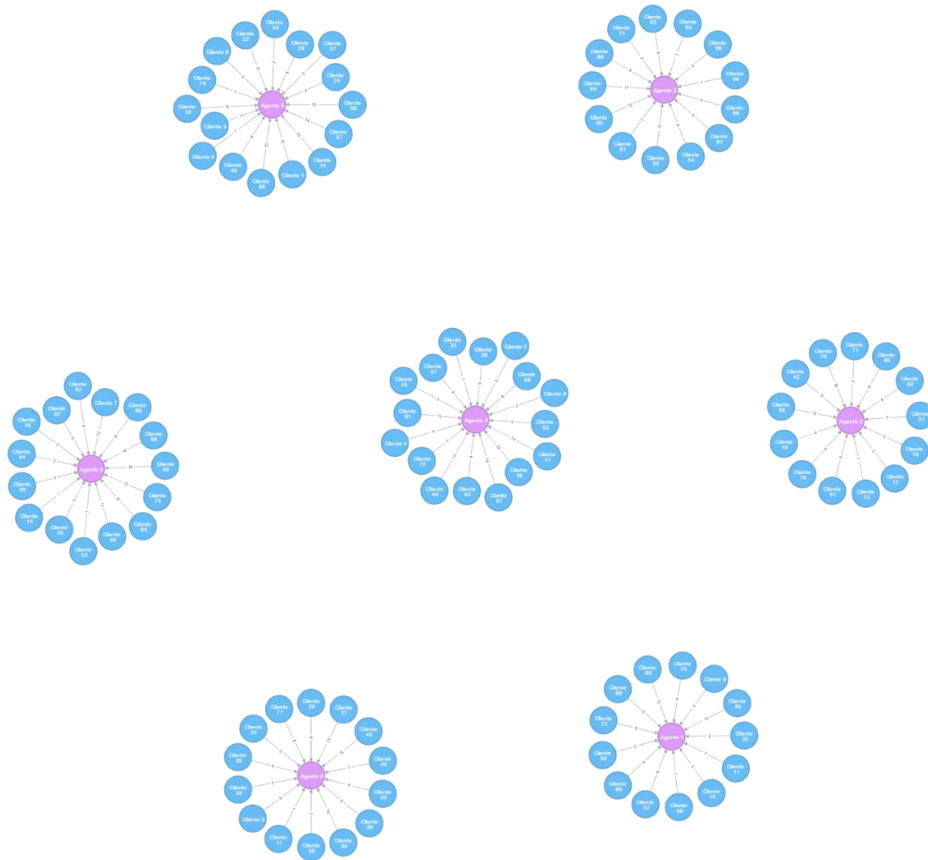


ILUSTRACIÓN 29 ITERACIÓN 1, RC206, HONDA CB1 PRO

## ITERACIÓN 2

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	942.35	3.54	31.18
Agente 1	840.35	3.16	27.83
Agente 2	681.66	2.56	22.55
Agente 3	395.79	1.49	13.12
Agente 4	915.65	3.44	30.3
Agente 5	852.9	3.21	28.27
Agente 6	609.64	2.29	20.17

TABLA 28 ITERACIÓN 2, RC206, HONDA CB1 PRO

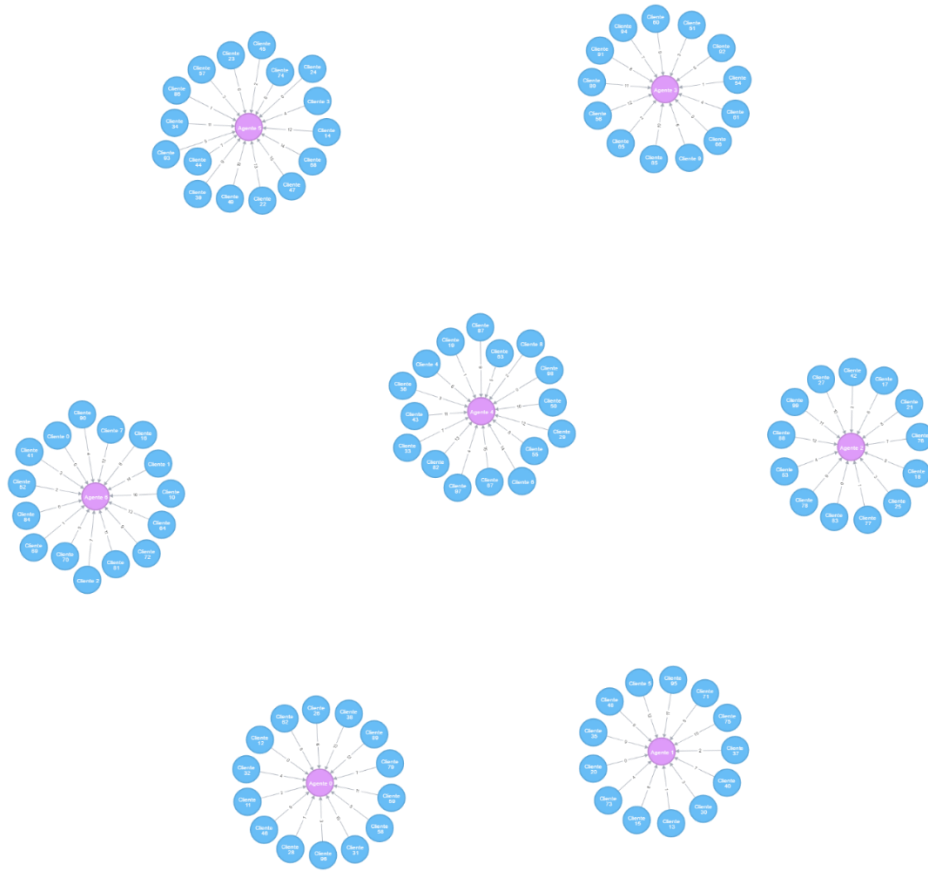


ILUSTRACIÓN 30 ITERACIÓN 2, RC206, HONDA CB1 PRO

### ITERACIÓN 3

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmissionTotal (KgCO2)
Agente 0	931.22	3.5	30.83
Agente 1	830.12	3.12	27.48
Agente 2	697.04	2.62	23.08
Agente 3	427.5	1.61	14.18
Agente 4	976.52	3.67	32.33
Agente 5	875.73	3.29	28.98
Agente 6	670.07	2.52	22.2

TABLA 29 ITERACIÓN 3, RC206, HONDA CB1 PRO

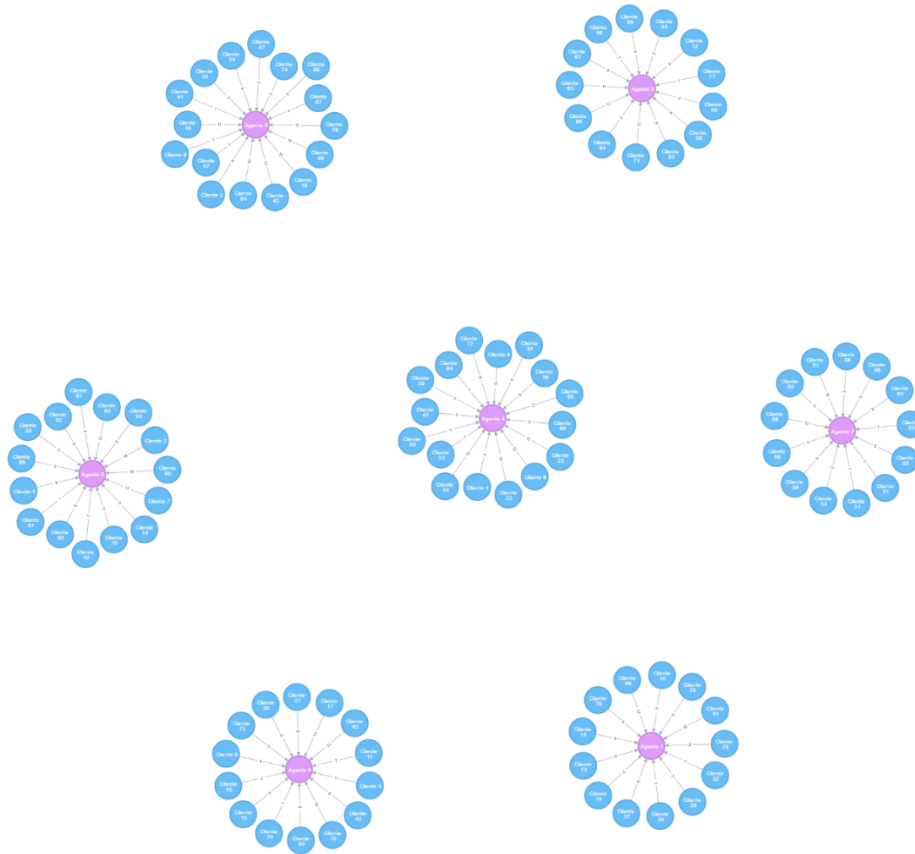


ILUSTRACIÓN 31 ITERACIÓN 3, RC206, HONDA CB1 PRO

### ITERACIÓN 4

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	933.28	3.51	30.92
Agente 1	840.69	3.16	27.83
Agente 2	726.24	2.73	24.05
Agente 3	465.43	1.75	15.41
Agente 4	1013.15	3.81	33.56
Agente 5	850.89	3.2	28.19
Agente 6	577.64	2.17	19.11

TABLA 30 ITERACIÓN 4, RC206, HONDA CB1 PRO

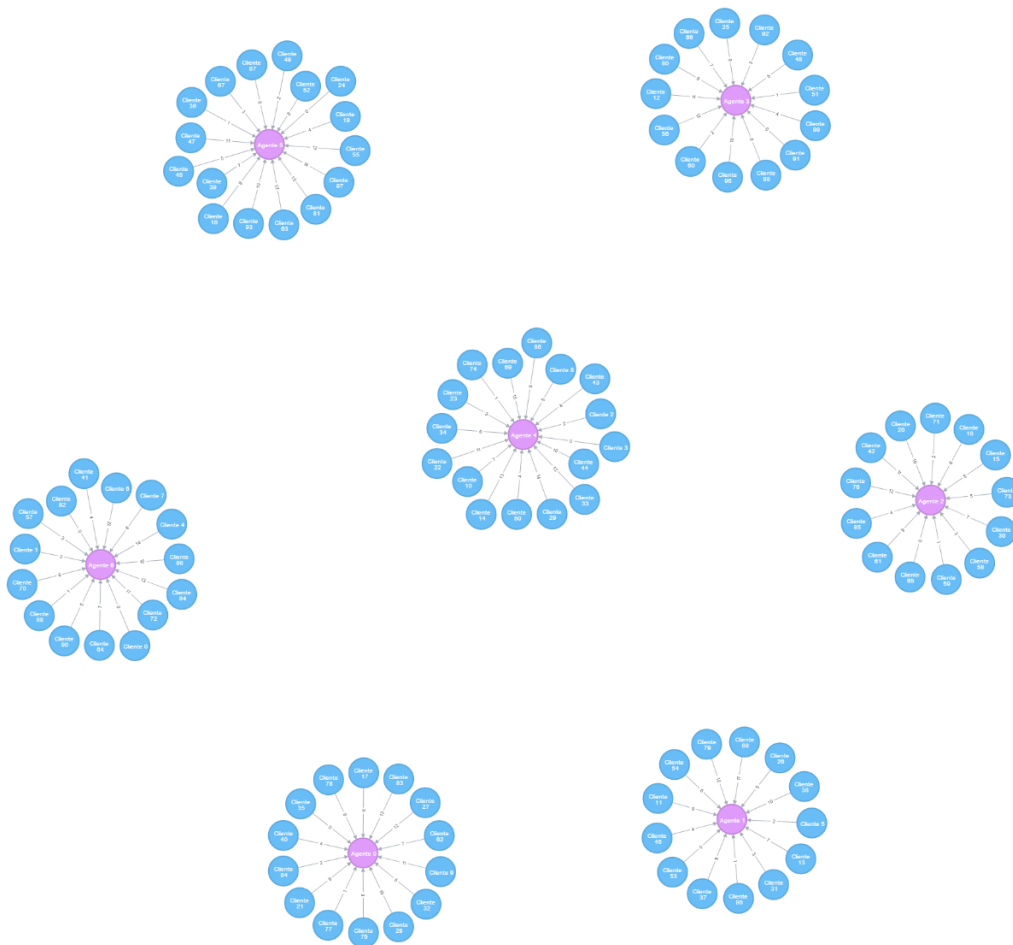


ILUSTRACIÓN 32 ITERACIÓN 4, RC206, HONDA CB1 PRO

### ITERACIÓN 5

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	912.48	3.43	30.21
Agente 1	840.06	3.16	27.83
Agente 2	762.86	2.87	25.28
Agente 3	417.92	1.57	13.83
Agente 4	1006.66	3.78	33.29
Agente 5	837.18	3.15	27.75
Agente 6	544.93	2.05	18.06

TABLA 31 ITERACIÓN 5, RC206, HONDA CB1 PRO



ILUSTRACIÓN 33 ITERACIÓN 5, RC206, HONDA CB1 PRO

**Resultados RC206, YAMAHA LIBERO 125  
ITERACIÓN 1**

Esta iteración se realiza teniendo en cuenta que la moto utilizada para hacer el recorrido es la **YAMAHA LIBERO 125**.

ITERACIÓN	TIEMPO (ms)
ITERACIÓN 1	0.1896350383758545
ITERACIÓN 2	0.19815802574157715
ITERACIÓN 3	0.20066285133361816
ITERACIÓN 4	0.20964980125427246
ITERACIÓN 5	0.1911180019378662

**ILUSTRACIÓN 34 TIEMPOS ITERACIONES, RC206, YAMAHA LIBERO 125**

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	937.82	6.25	55.05
Agente 1	882.96	5.89	51.88
Agente 2	699.57	4.66	41.05
Agente 3	399.09	2.66	23.43
Agente 4	963.28	6.42	56.55
Agente 5	821.9	5.48	48.27
Agente 6	606.4	4.04	35.58

**TABLA 32 ITERACIÓN 1, RC206, YAMAHA LIBERO 125**

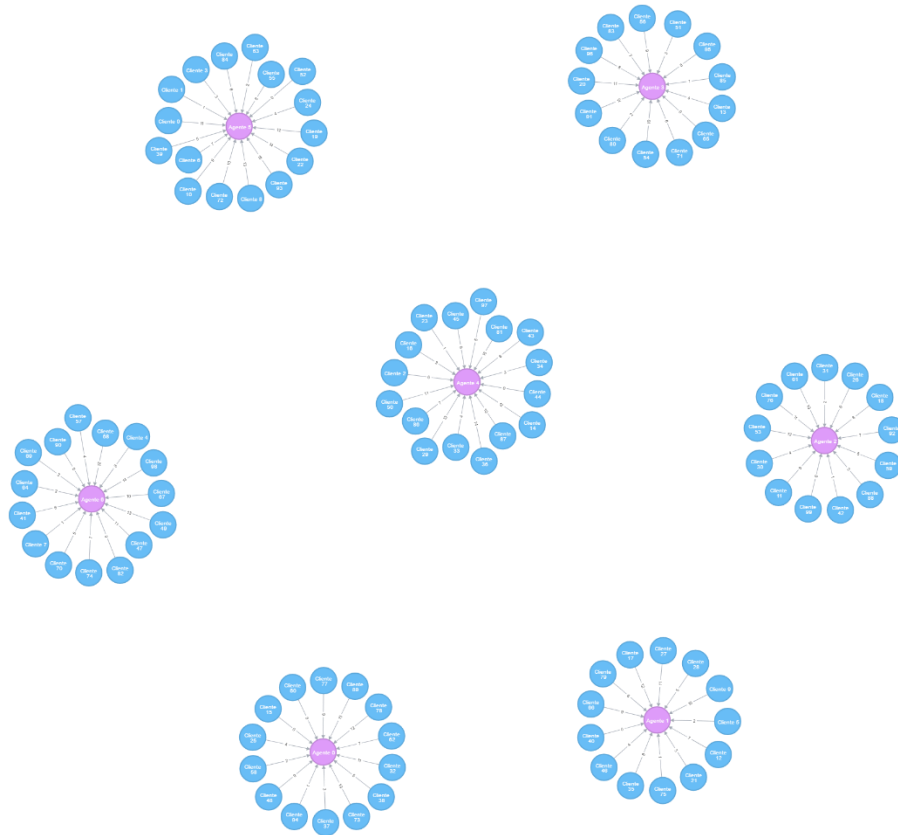


ILUSTRACIÓN 35 ITERACIÓN 1, RC206, YAMAHA LIBERO 125

## ITERACIÓN 2

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 1	873.4	5.82	51.26
Agente 2	647.1	4.31	37.96
Agente 3	402.48	2.68	23.61
Agente 4	960.94	6.41	56.46
Agente 5	889	5.93	52.23
Agente 6	604.71	4.03	35.5
Agente 0	966.56	6.44	56.72

TABLA 33 ITERACIÓN 2, RC206, YAMAHA LIBERO 125



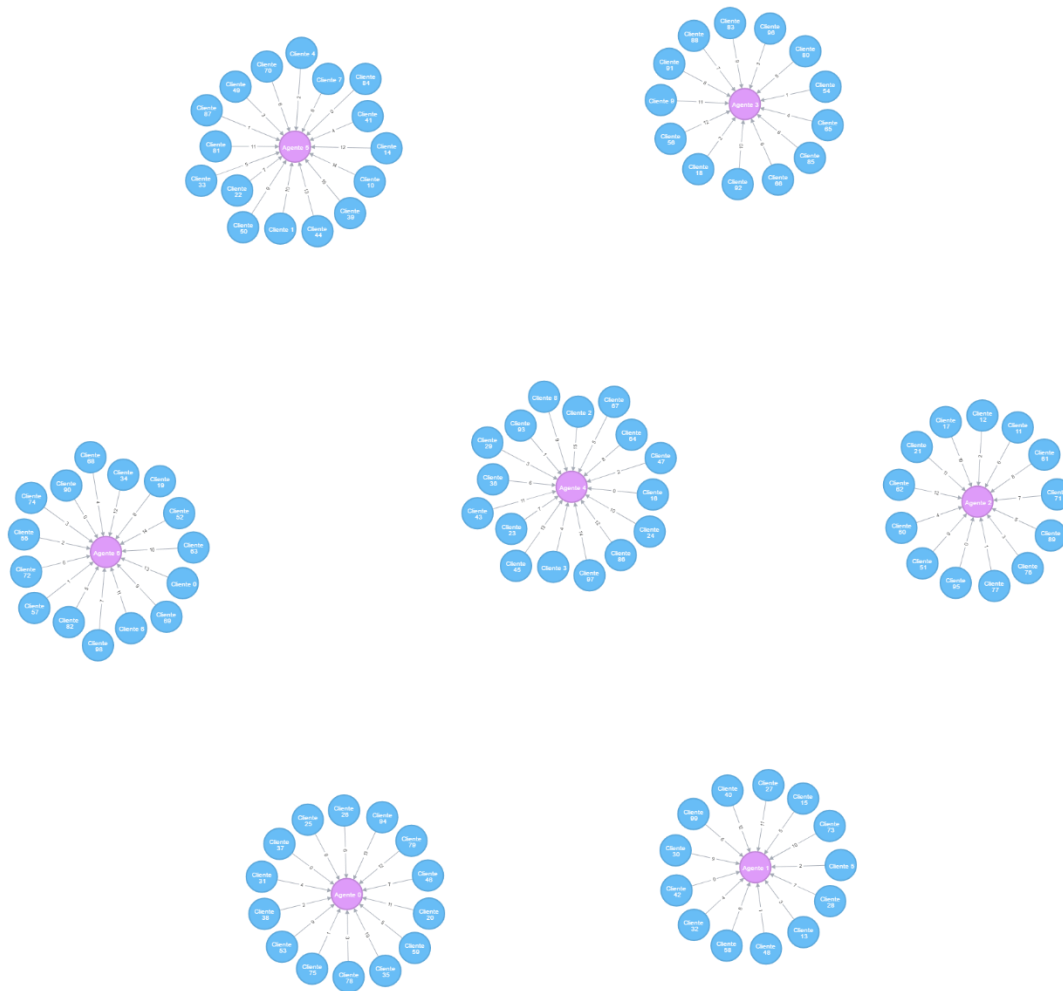


ILUSTRACIÓN 36 ITERACIÓN 2, RC206, YAMAHA LIBERO 125

### ITERACIÓN 3

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	991.26	6.61	58.22
Agente 1	820.03	5.47	48.18
Agente 2	679.98	4.53	39.9
Agente 3	471.47	3.14	27.66
Agente 4	988.67	6.59	58.04
Agente 5	846.06	5.64	49.68
Agente 6	649.83	4.33	38.14

TABLA 34 ITERACIÓN 3, RC206, YAMAHA LIBERO 125

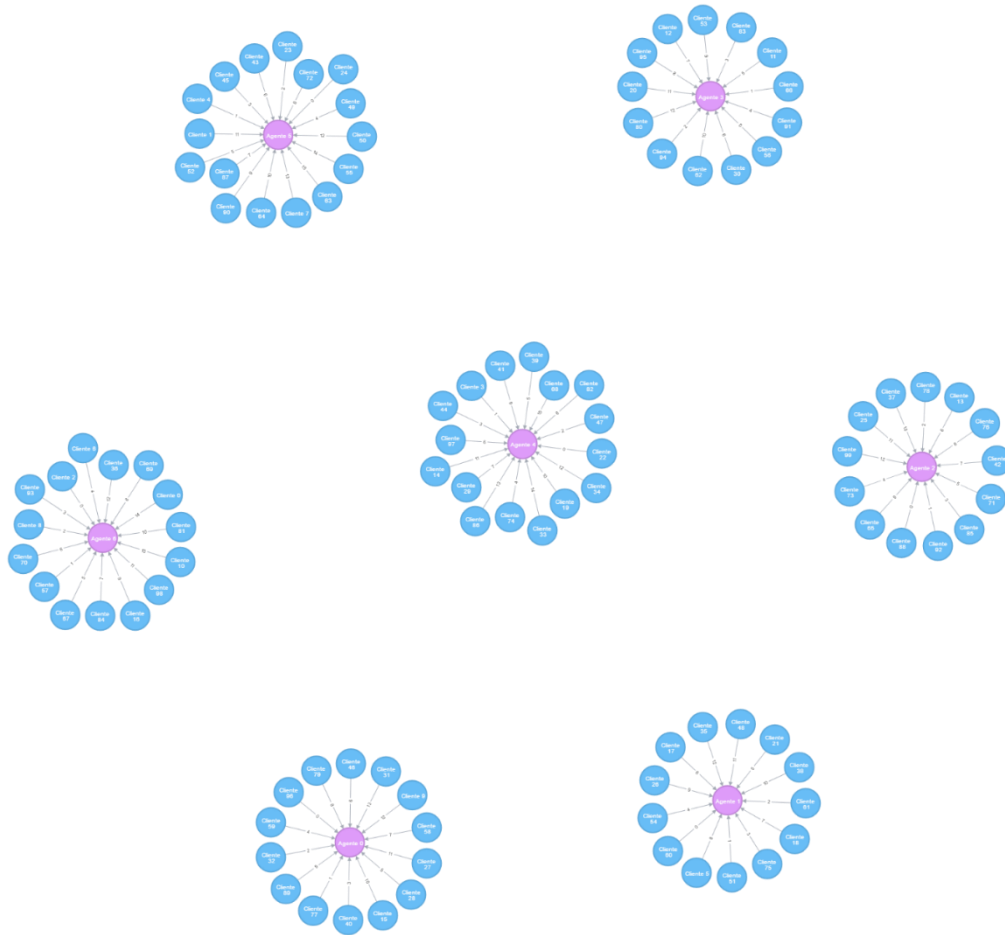


ILUSTRACIÓN 37 ITERACIÓN 3, RC206, YAMAHA LIBERO 125

### ITERACIÓN 4

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmissionTotal (KgCO2)
Agente 0	981.61	6.54	57.6
Agente 1	812.04	5.41	47.65
Agente 2	772.29	5.15	45.36
Agente 3	407.51	2.72	23.96
Agente 4	975.48	6.5	57.25
Agente 5	931.75	6.21	54.7
Agente 6	530.86	3.54	31.18

TABLA 35 ITERACIÓN 4, RC206, YAMAHA LIBERO 125

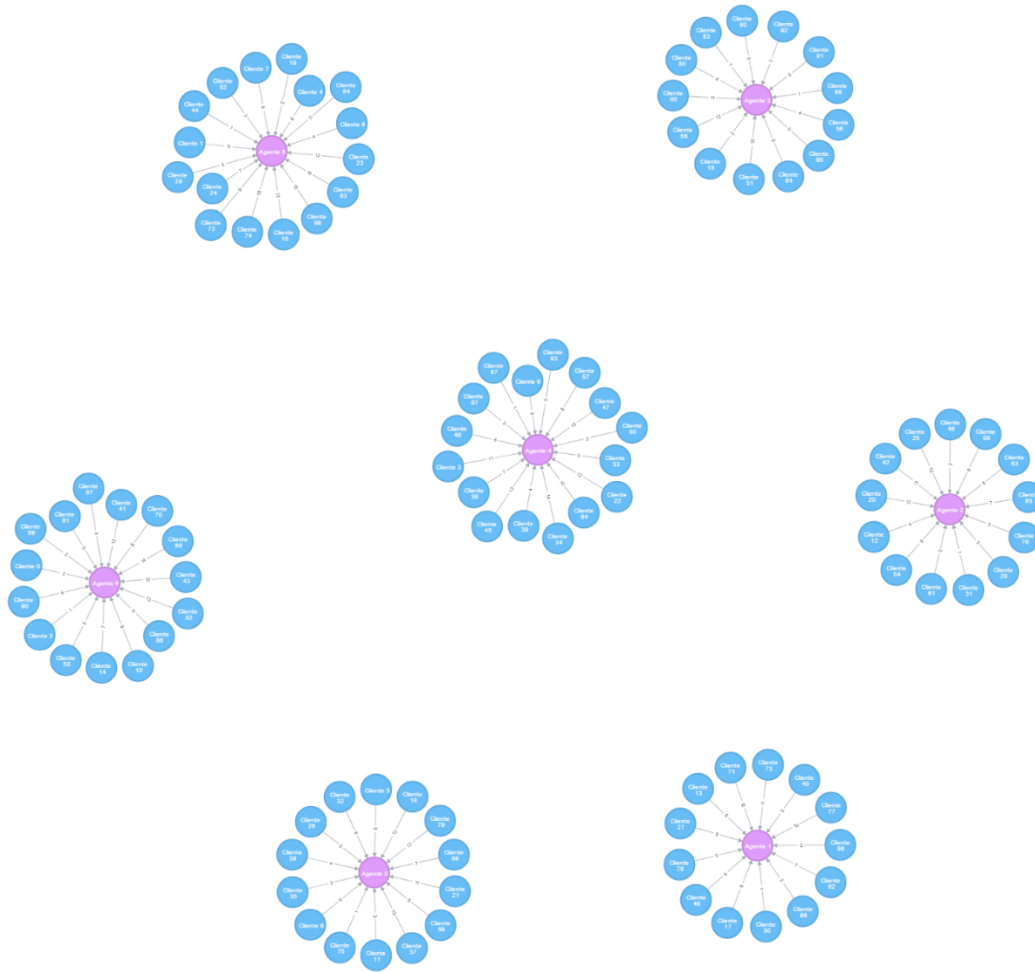


ILUSTRACIÓN 38 ITERACIÓN 4, RC206, YAMAHA LIBERO 125

### ITERACIÓN 5

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	979.08	6.53	57.52
Agente 1	784.46	5.23	46.07
Agente 2	683.27	4.56	40.16
Agente 3	480.92	3.21	28.27
Agente 4	1025.77	6.84	60.25
Agente 5	883.35	5.89	51.88
Agente 6	581.24	3.87	34.09

TABLA 36 ITERACIÓN 5, RC206, YAMAHA LIBERO 125

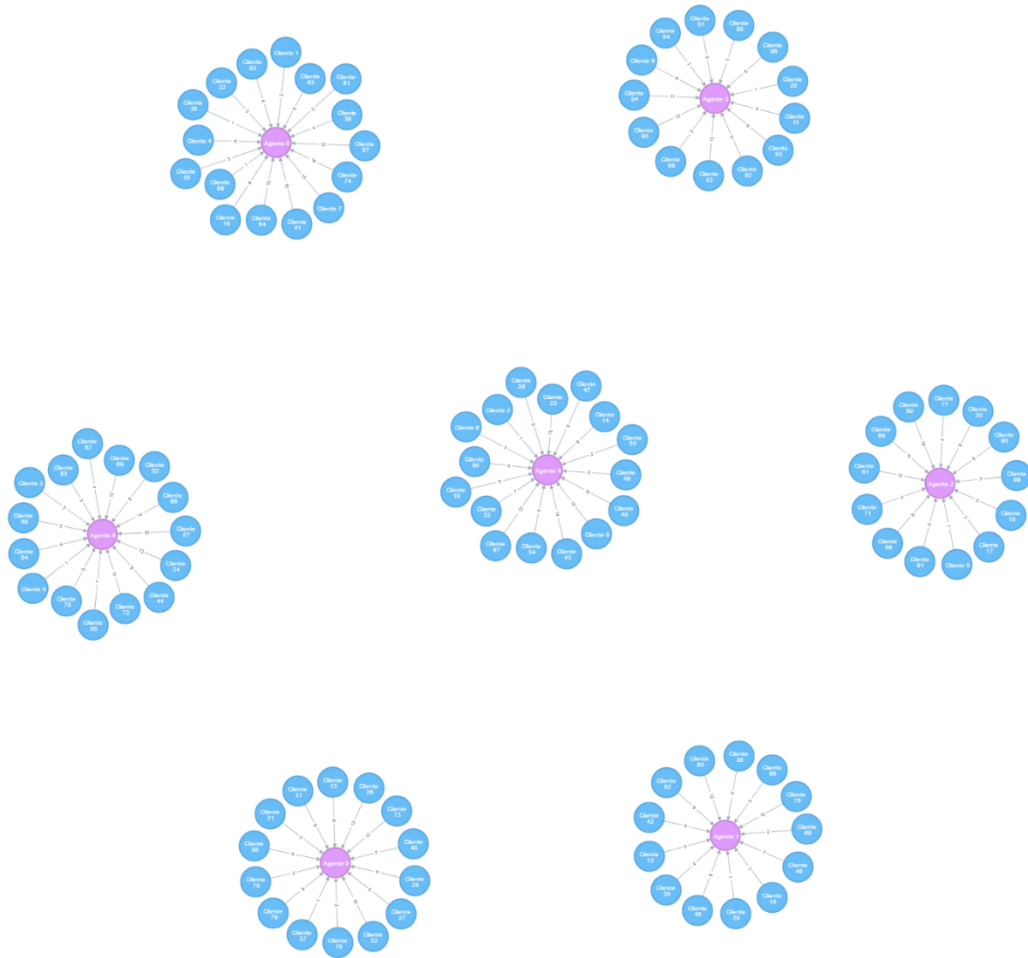


ILUSTRACIÓN 39 ITERACIÓN 5, RC206, YAMAHA LIBERO 125

**Resultados R210, AKT 125 NKD**  
**ITERACIÓN 1**

Esta iteración se realiza teniendo en cuenta que la moto utilizada para hacer el recorrido es **AKT 125 NKD**

**Cantidad nodos franja 1 = 51 Cantidad agentes = [17, 17, 17]**

**Cantidad nodos franja 2 = 49 Cantidad agentes = [17, 16, 16]**

ITERACIÓN	TIEMPO (ms)
ITERACIÓN 1	0.19311857223510742
ITERACIÓN 2	0.1876511573791504
ITERACIÓN 3	0.16461539268493652
ITERACIÓN 4	0.17112159729003906
ITERACIÓN 5	0.17013978958129883

**TABLA 37 TIEMPOS ITERACIONES AKT 125 NKD**

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	900.18	6.43	56.64
Agente 1	767.81	5.48	48.27
Agente 2	585.26	4.18	36.82
Agente 3	785.15	5.61	49.41
Agente 4	623.25	4.45	39.2
Agente 5	398.91	2.85	25.1

**TABLA 38 ITERACIÓN1, R210, AKT 125 NKD**

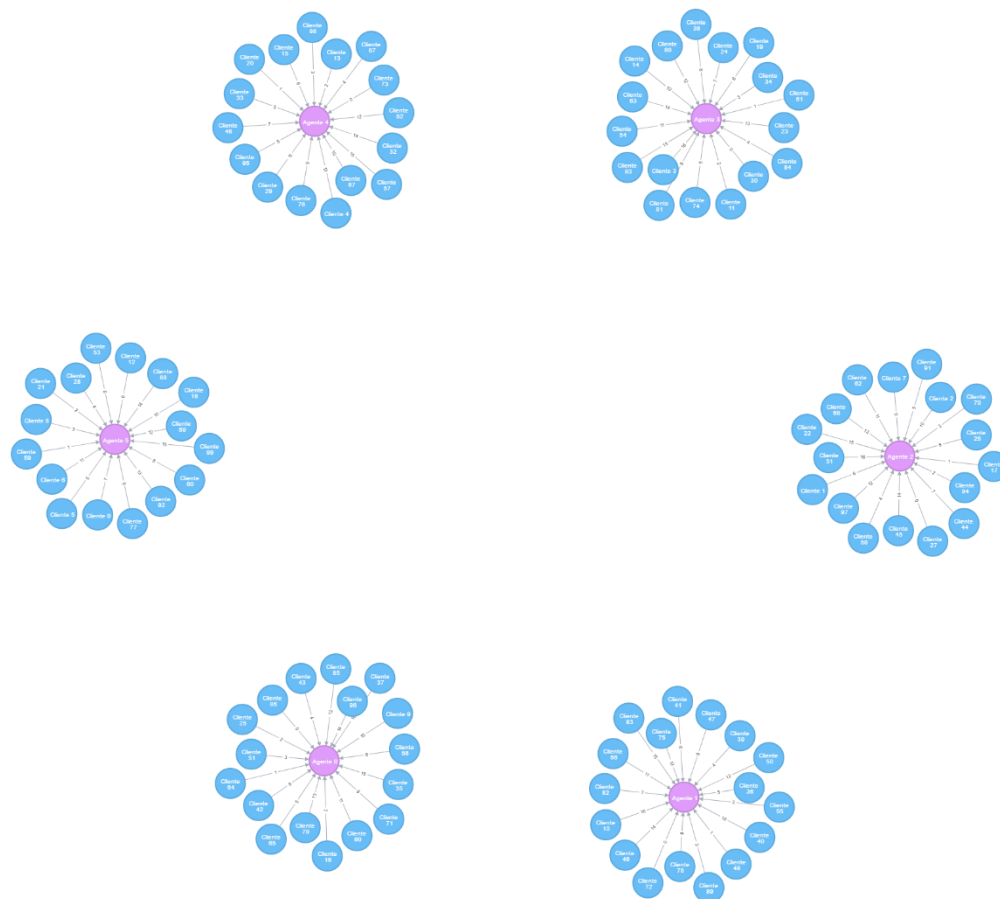


ILUSTRACIÓN 40 ITERACIÓN1, R210, AKT 125 NKD

## ITERACIÓN 2

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmissionTotal (KgCO2)
Agente 0	861.73	6.16	54.26
Agente 1	795.28	5.68	50.03
Agente 2	627.28	4.48	39.46
Agente 3	744.41	5.32	46.86
Agente 4	600.63	4.29	37.79
Agente 5	430.61	3.08	27.13

TABLA 39 ITERACIÓN2, R210, AKT 125 NKD

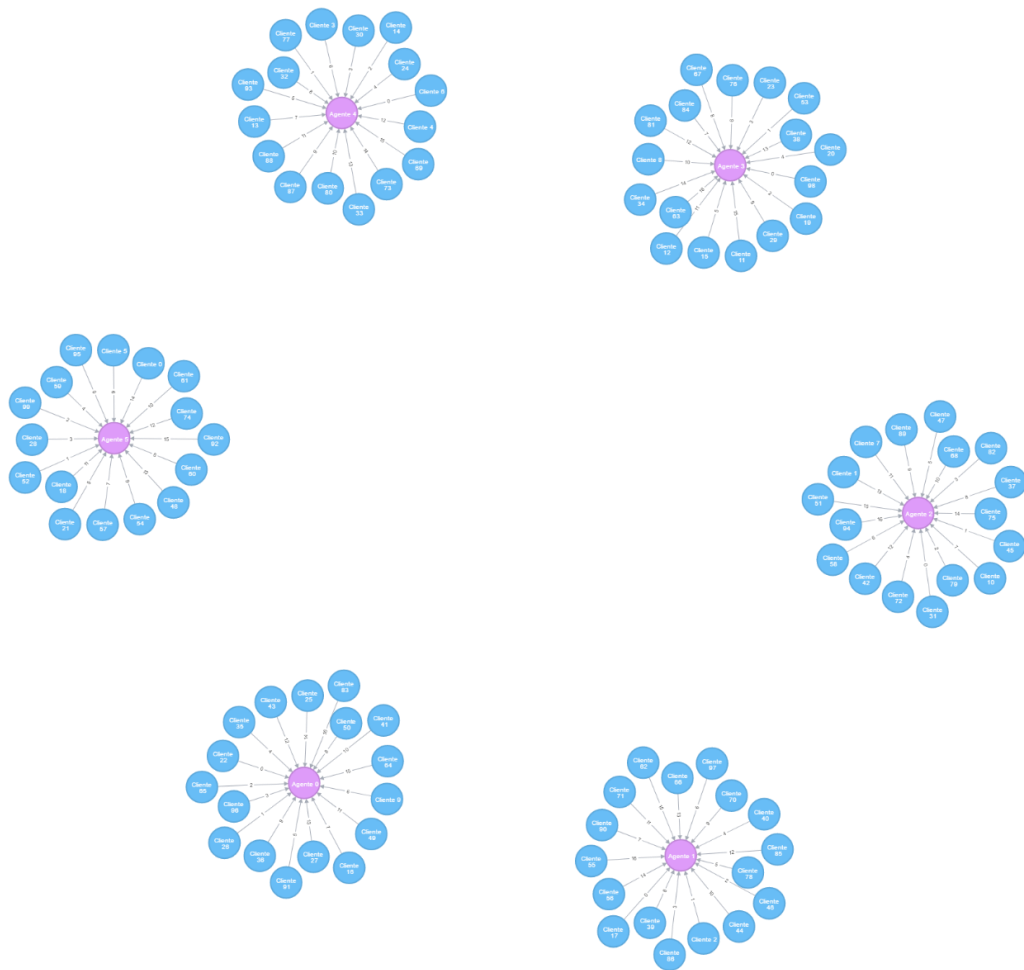


ILUSTRACIÓN 41 ITERACIÓN2, R210, AKT 125 NKD

### ITERACIÓN 3

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmissionTotal (KgCO2)
Agente 0	888.88	6.35	55.93
Agente 1	805.22	5.75	50.65
Agente 2	566.68	4.05	35.67
Agente 3	776.08	5.54	48.8
Agente 4	579.06	4.14	36.47
Agente 5	464.59	3.32	29.24

TABLA 40 ITERACIÓN3, R210, AKT 125 NKD

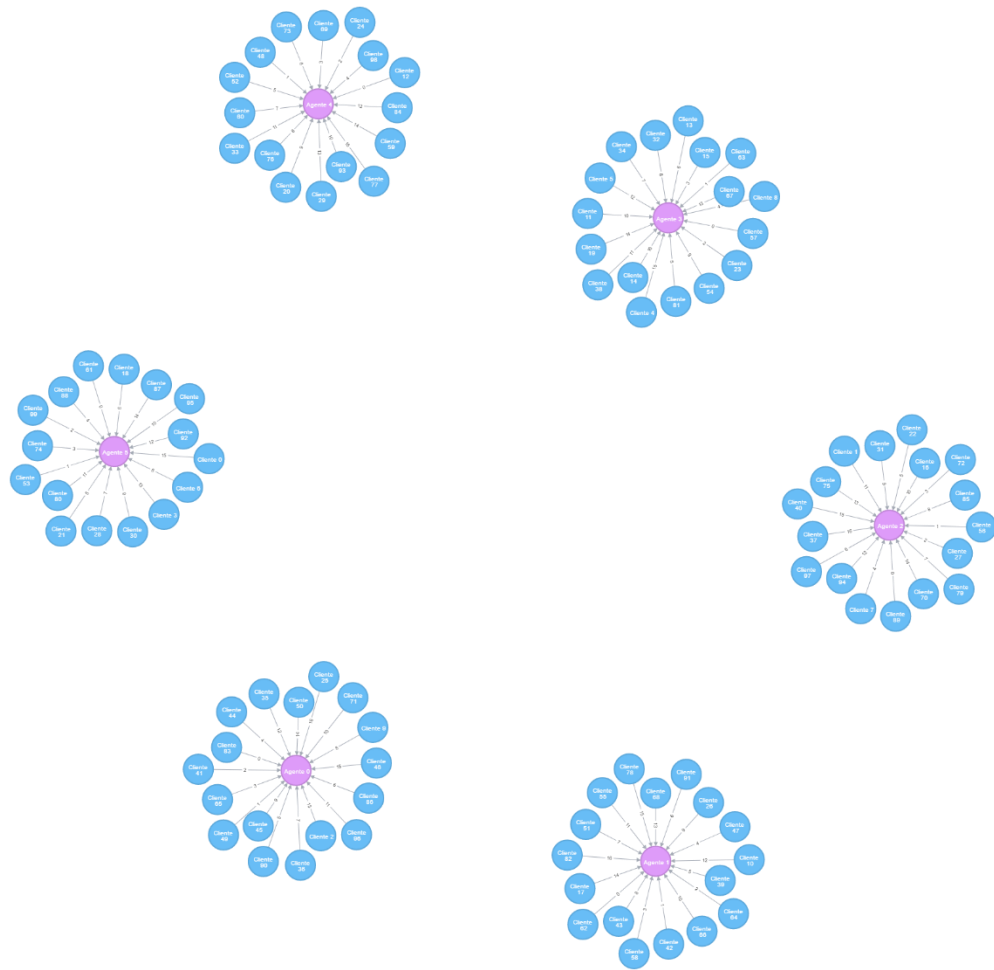


ILUSTRACIÓN 42 ITERACIÓN3, R210, AKT 125 NKD

### ITERACIÓN 4

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmissionTotal (KgCO2)
Agente 0	969.34	6.92	60.95
Agente 1	746.41	5.33	46.95
Agente 2	606.88	4.33	38.14
Agente 3	773.67	5.53	48.71
Agente 4	610.92	4.36	38.4
Agente 5	424.36	3.03	26.69

TABLA 41 ITERACIÓN4, R210, AKT 125 NKD



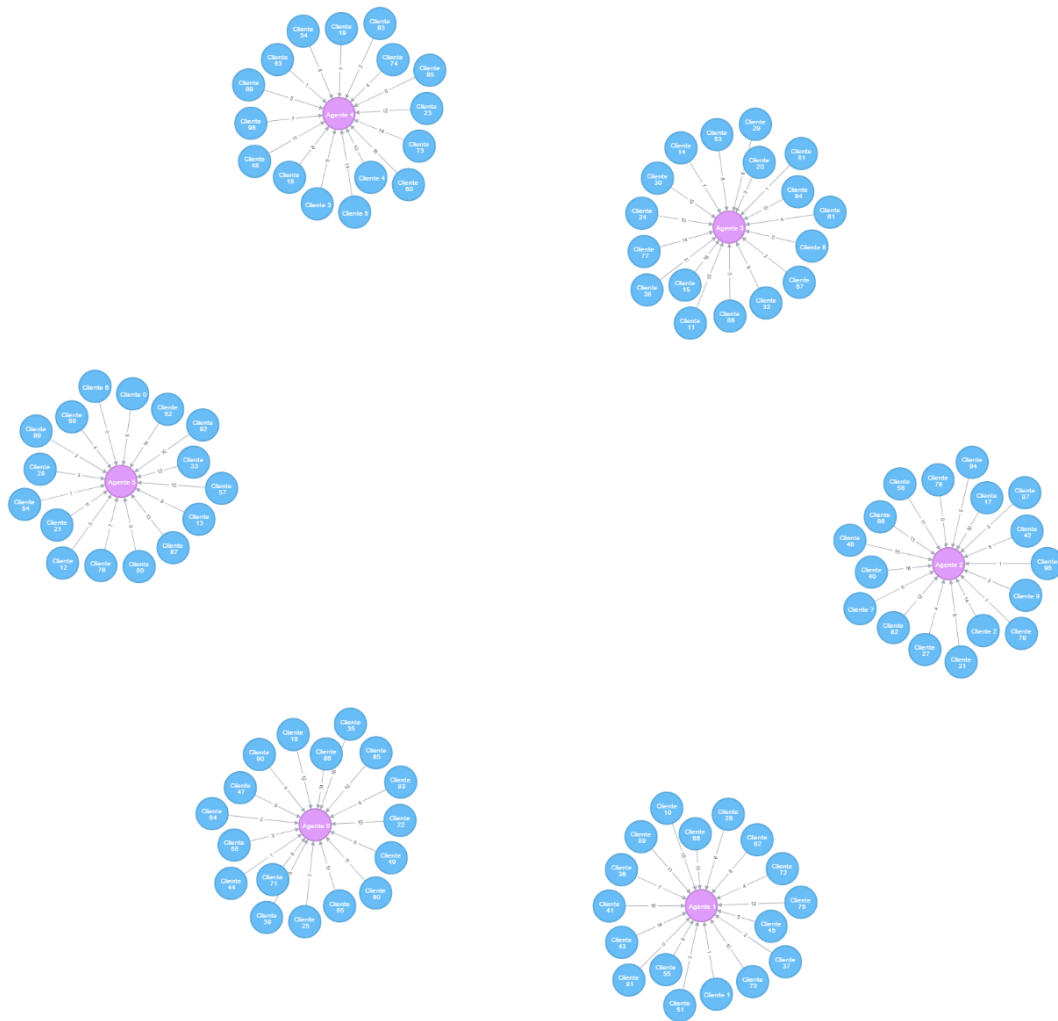


ILUSTRACIÓN 43 ITERACIÓN4, R210, AKT 125 NKD

### ITERACIÓN 5

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	882.18	6.3	55.49
Agente 1	775.74	5.54	48.8
Agente 2	553.59	3.95	34.79
Agente 3	747.35	5.34	47.03
Agente 4	665.76	4.76	41.93
Agente 5	474.73	3.39	29.86

TABLA 42 ITERACIÓN5, R210, AKT 125 NKD

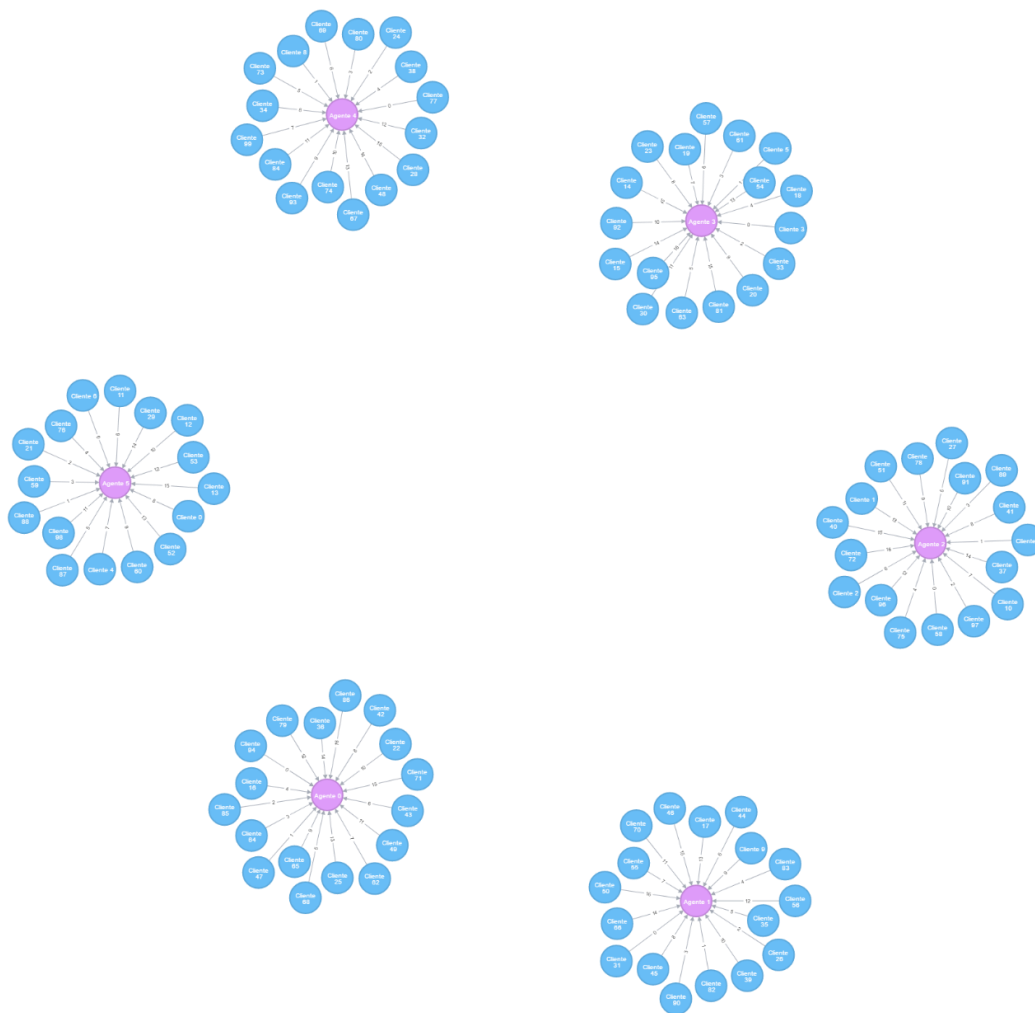


ILUSTRACIÓN 44 ITERACIÓN5, R210, AKT 125 NKD

## Resultados R210, SUZUKI GN125

### ITERACIÓN 1

Esta iteración se realiza teniendo en cuenta que la moto utilizada para hacer el recorrido es **SUZUKI GN125**.

ITERACIÓN	TIEMPO (ms)
ITERACIÓN 1	0.19515728950500488
ITERACIÓN 2	0.21665453910827637
ITERACIÓN 3	0.17212128639221191
ITERACIÓN 4	0.1711406707763672
ITERACIÓN 5	0.1766207218170166

TABLA 43 TIEMPOS ITERACIONES, R210, SUZUKI GN125

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	863.34	4.32	38.05
Agente 1	797.62	3.99	35.14
Agente 2	556.72	2.78	24.49
Agente 3	786.43	3.93	34.62
Agente 4	616.19	3.08	27.13
Agente 5	456.55	2.28	20.08

TABLA 44 ITERACIÓN 1, R210, SUZUKI GN125

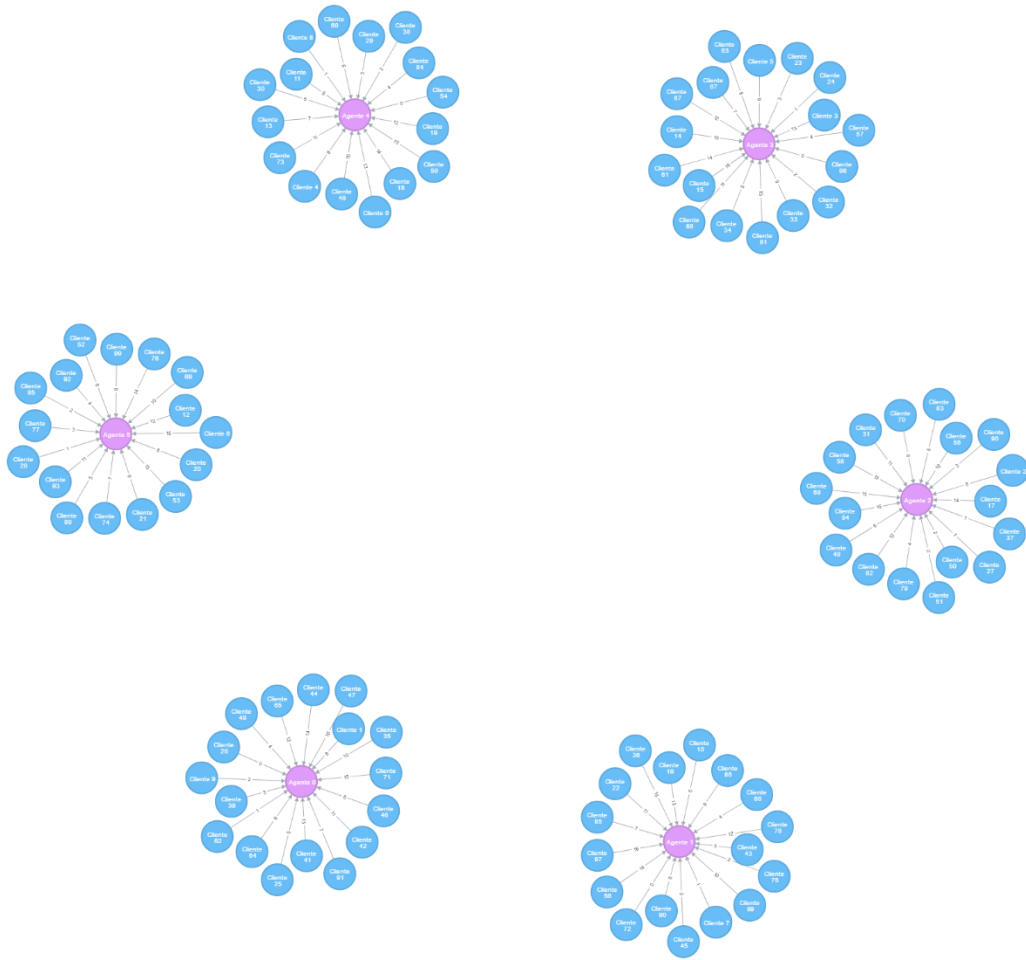


ILUSTRACIÓN 45 ITERACIÓN 1, R210, SUZUKI GN125

## ITERACIÓN 2

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	871.02	4.36	38.4
Agente 1	817.15	4.09	36.02
Agente 2	652.12	3.26	28.71
Agente 3	757.46	3.79	33.38
Agente 4	610.92	3.05	26.86
Agente 5	467.09	2.34	20.61

TABLA 45 ITERACIÓN 2, R210, SUZUKI GN125

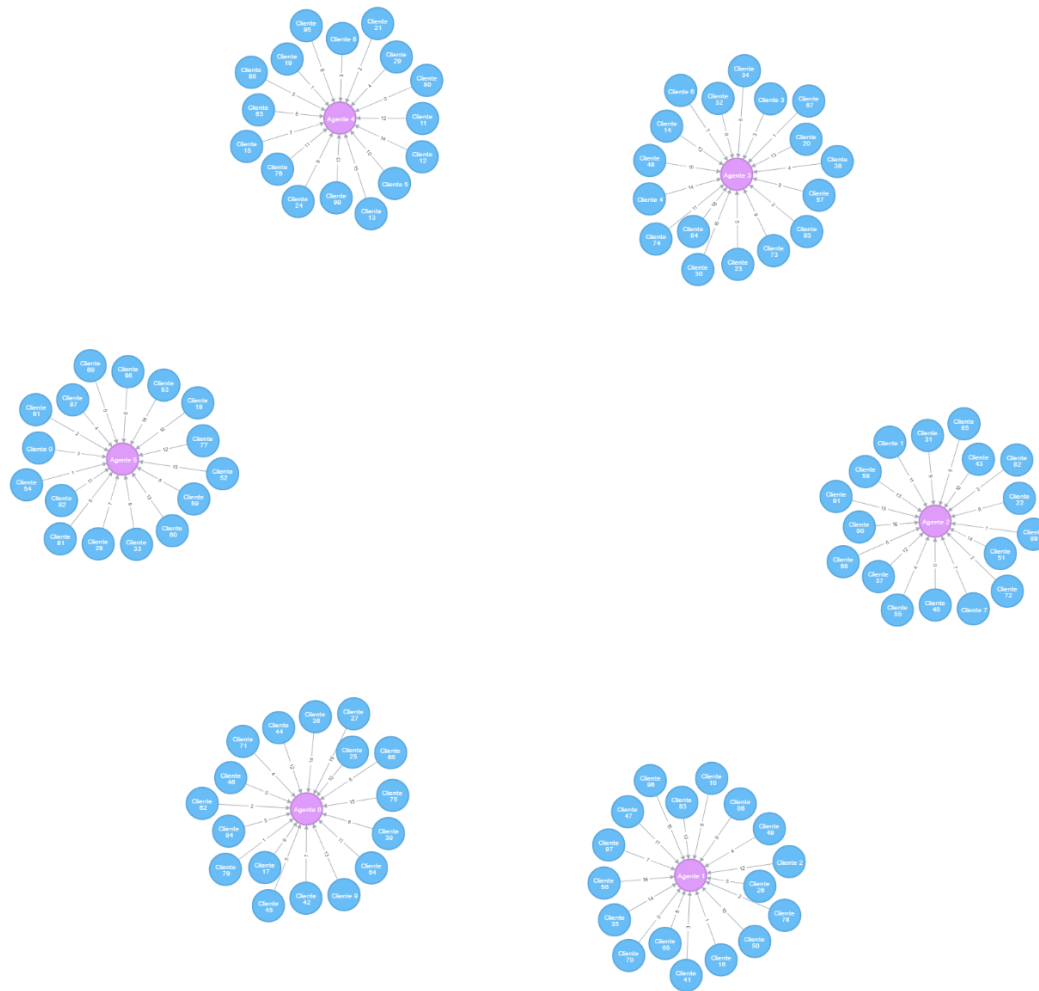


ILUSTRACIÓN 46 ITERACIÓN 2, R210, SUZUKI GN125

### ITERACIÓN 3

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmissionTotal (KgCO2)
Agente 0	892.13	4.46	39.28
Agente 1	787.08	3.94	34.7
Agente 2	577.32	2.89	25.46
Agente 3	789.01	3.95	34.79
Agente 4	636.32	3.18	28.01
Agente 5	476.25	2.38	20.96

TABLA 46 ITERACIÓN 3, R210, SUZUKI GN125

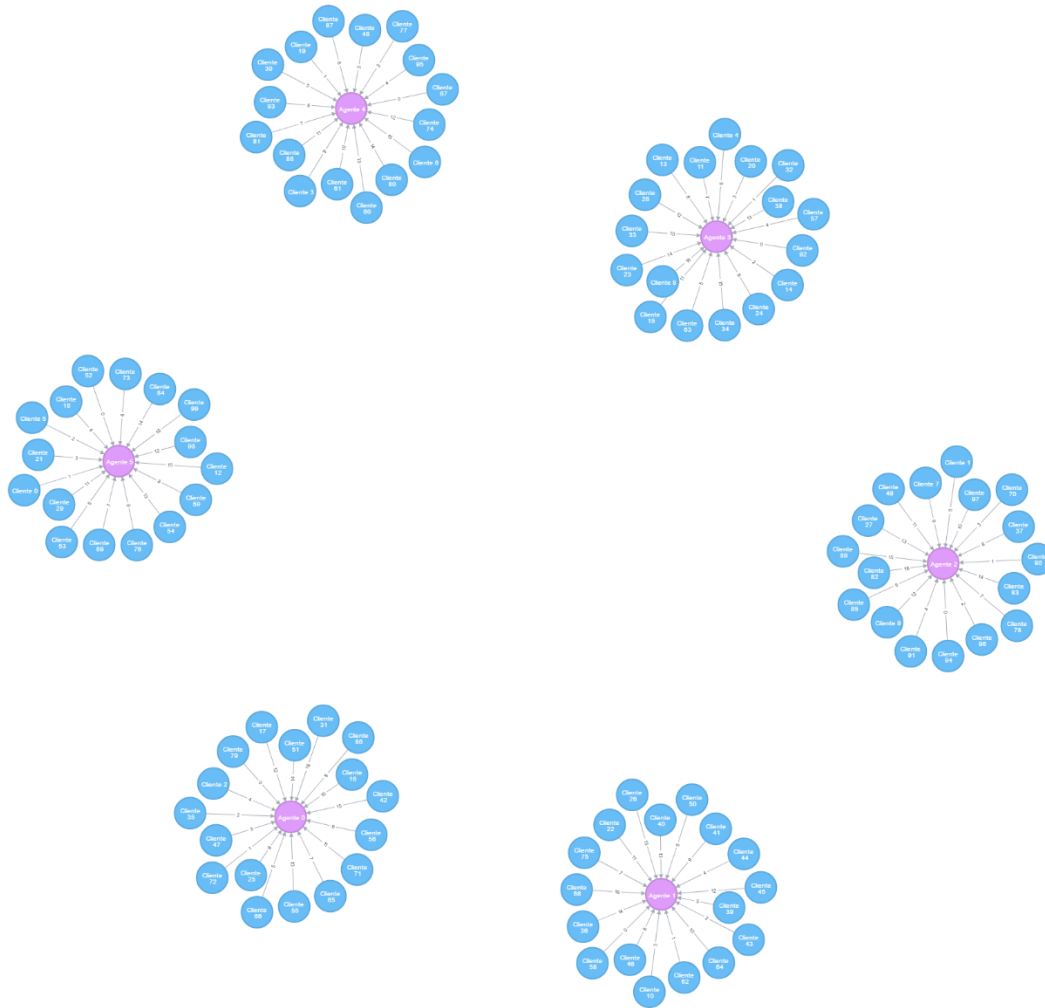


ILUSTRACIÓN 47 ITERACIÓN 3, R210, SUZUKI GN125

### ITERACIÓN 4

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal	EmisionTotal
Agente 0	887.5	4.44	39.11
Agente 1	823.96	4.12	36.29
Agente 2	588.66	2.94	25.9
Agente 3	741.73	3.71	32.68
Agente 4	644.06	3.22	28.36
Agente 5	397	1.99	17.53

TABLA 47 ITERACIÓN 4, R210, SUZUKI GN125

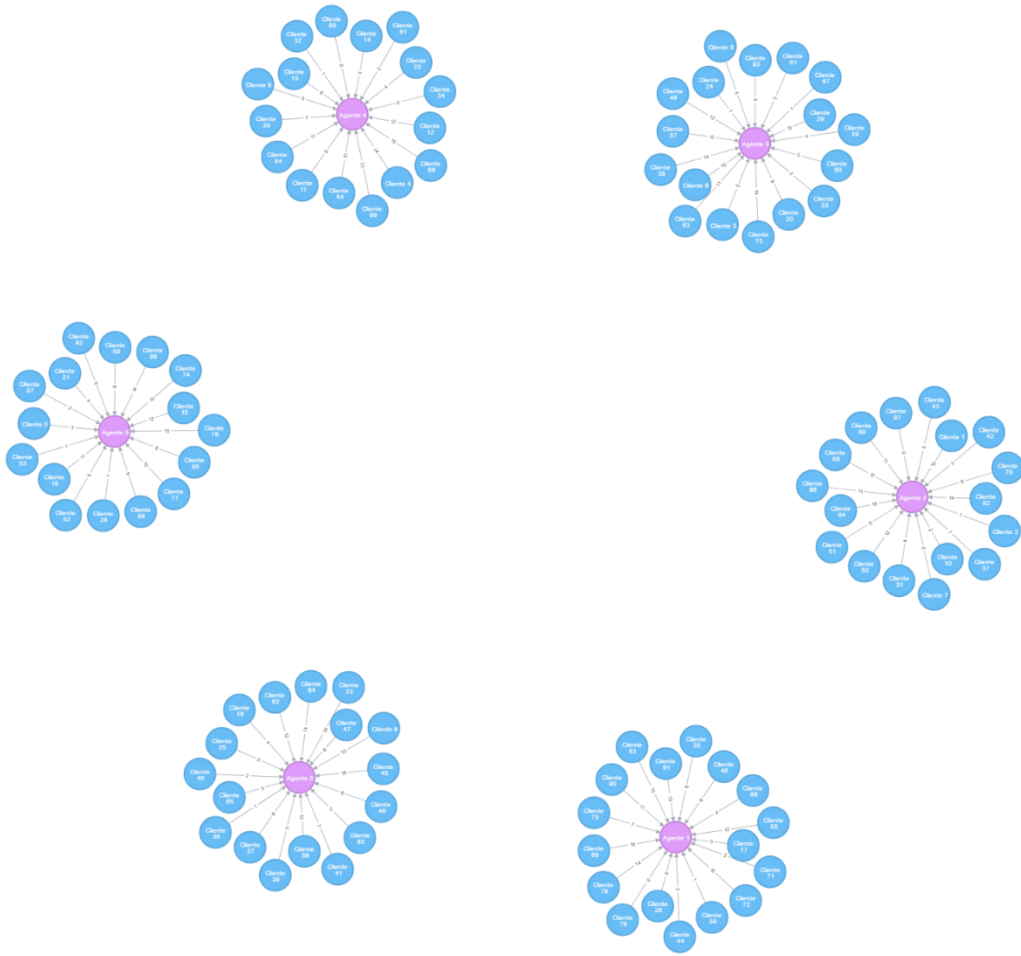


ILUSTRACIÓN 48 ITERACIÓN 4, R210, SUZUKI GN125

## ITERACIÓN 5

NombreAgente	DistanciaRecorrida	ConsumoTotal	EmisionTotal
Agente 0	961.89	4.81	42.37
Agente 1	804.53	4.02	35.41
Agente 2	565.17	2.83	24.93
Agente 3	785.6	3.93	34.62
Agente 4	616.2	3.08	27.13
Agente 5	431.03	2.16	19.03

ILUSTRACIÓN 49 ITERACIÓN 5, R210, SUZUKI GN125

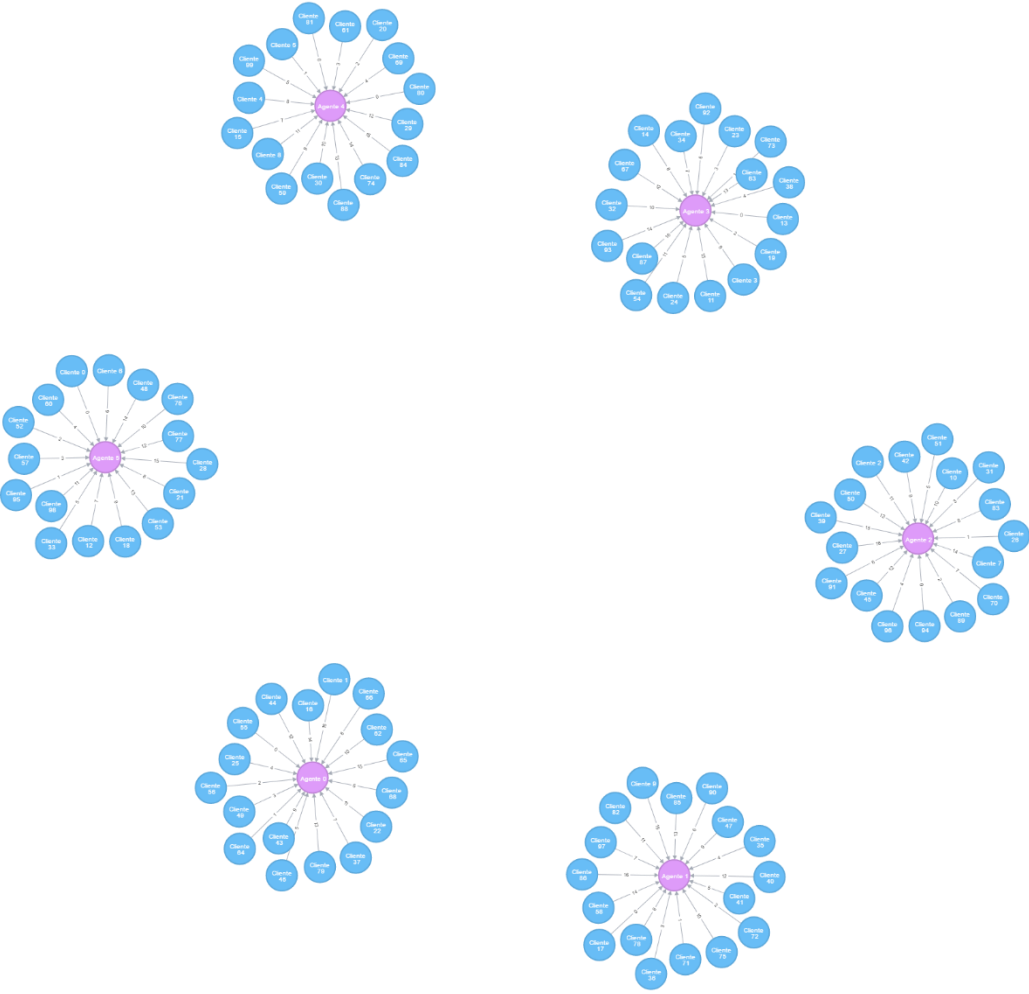


ILUSTRACIÓN 50 ITERACIÓN 5, R210, SUZUKI GN125



**Resultados R210, HONDA CB1 PRO  
ITERACIÓN 1**

Esta iteración se realiza teniendo en cuenta que la moto utilizada para hacer el recorrido es **HONDA CB1 PRO**.

ITERACIÓN	TIEMPO (ms)
Iteración 1	0.17961692810058594
Iteración 2	0.17310571670532227
Iteración 3	0.1832294464111328
Iteración 4	0.17601275444030762
Iteración 5	0.1880321502685547

**TABLA 48 TIEMPOS ITERACIONES, R210, HONDA CB1 PRO**

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón )	EmisionTotal (KgCO2)
Agente 0	896.45	3.37	29.68
Agente 1	792.47	2.98	26.25
Agente 2	579.89	2.18	19.2
Agente 3	778.56	2.93	25.81
Agente 4	602.36	2.26	19.91
Agente 5	426.23	1.6	14.09

**TABLA 49 ITERACIÓN 1, R210, HONDA CB1 PRO**

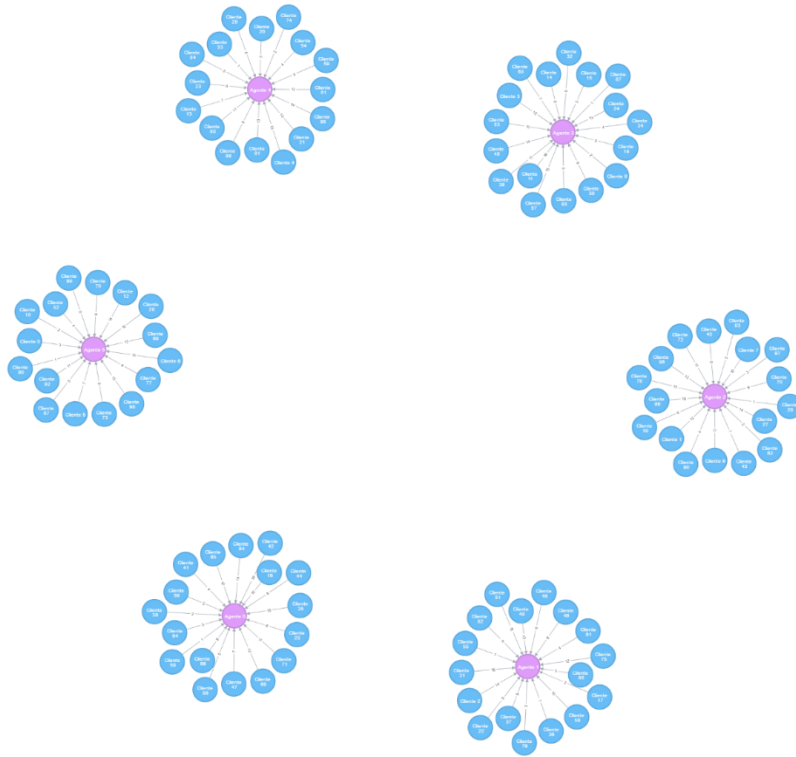


ILUSTRACIÓN 51 ITERACIÓN 1, R210, HONDA CB1 PRO

**ITERACIÓN 2**

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	924.06	3.47	30.56
Agente 1	803.9	3.02	26.6
Agente 2	538.97	2.03	17.88
Agente 3	783.82	2.95	25.98
Agente 4	670.89	2.52	22.2
Agente 5	464.37	1.75	15.41

TABLA 50 ITERACIÓN 2, R210, HONDA CB1 PRO



ILUSTRACIÓN 52 ITERACIÓN 2, R210, HONDA CB1 PRO

### ITERACIÓN 3

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	862.63	3.24	28.54
Agente 1	810.45	3.05	26.86
Agente 2	613.43	2.31	20.35
Agente 3	747.75	2.81	24.75
Agente 4	657.43	2.47	21.76
Agente 5	497.46	1.87	16.47

TABLA 51 ITERACIÓN 3, R210, HONDA CB1 PRO

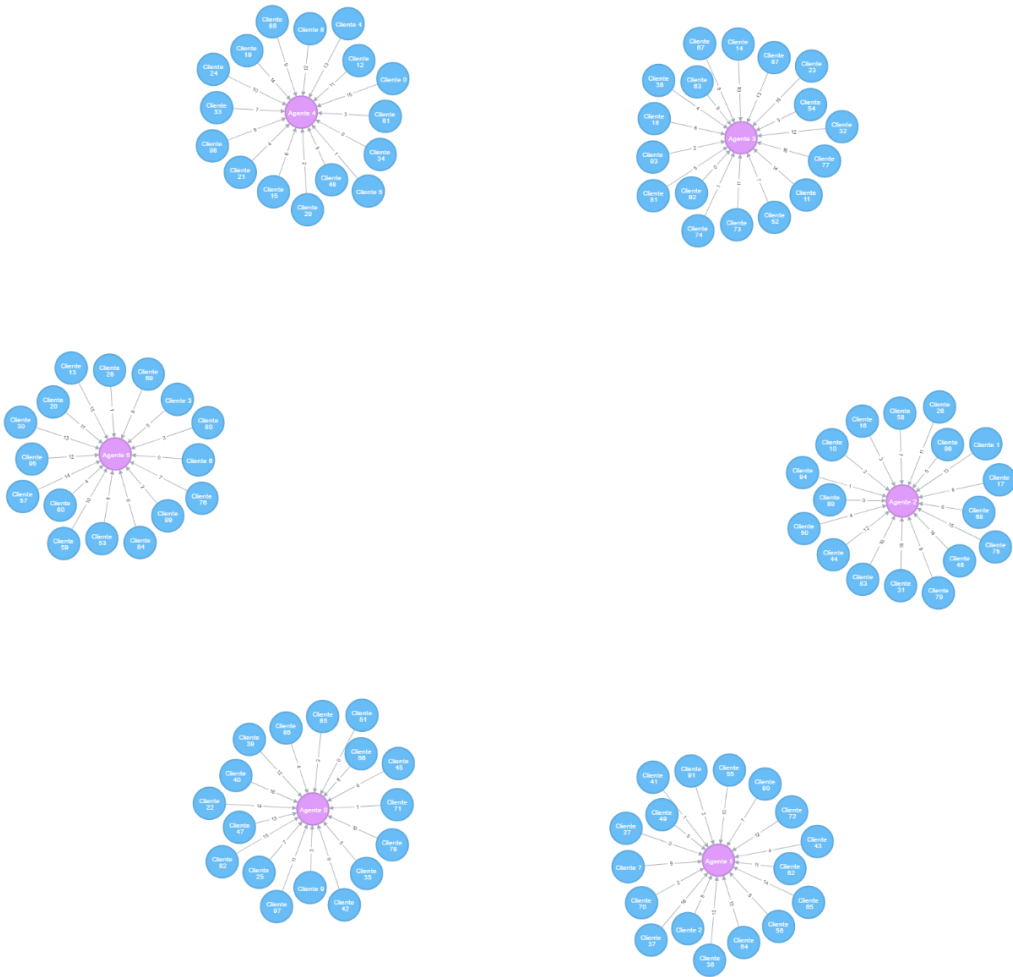


ILUSTRACIÓN 53 ITERACIÓN 3, R210, HONDA CB1 PRO

**ITERACIÓN 4**

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmissionTotal (KgCO2)
Agente 0	878.77	3.3	29.07
Agente 1	763.72	2.87	25.28
Agente 2	514.58	1.93	17
Agente 3	787.48	2.96	26.07
Agente 4	656.17	2.47	21.76
Agente 5	474.46	1.78	15.68

TABLA 52 ITERACIÓN 4, R210, HONDA CB1 PRO

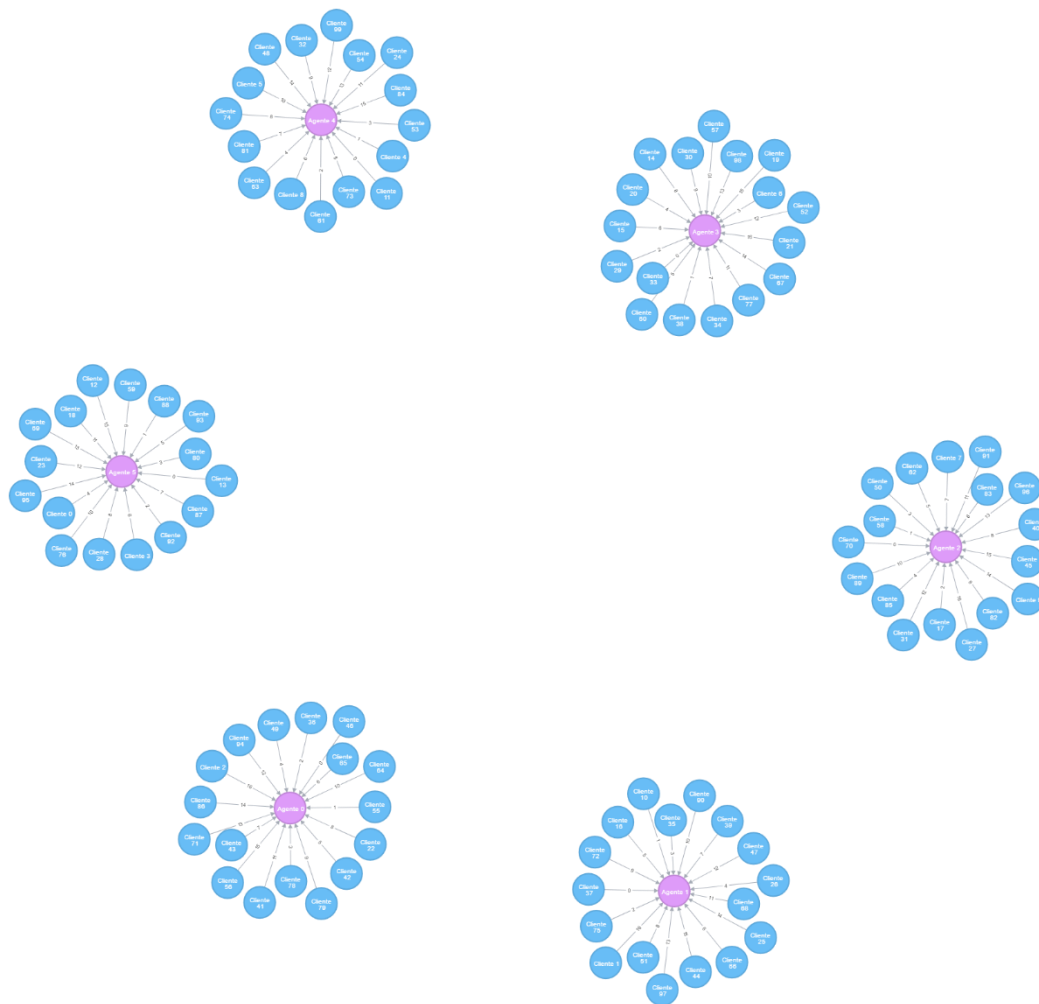


ILUSTRACIÓN 54 ITERACIÓN 4, R210, HONDA CB1 PRO

### ITERACIÓN 5

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	886.78	3.33	29.33
Agente 1	759.63	2.86	25.19
Agente 2	530.62	1.99	17.53
Agente 3	772.02	2.9	25.54
Agente 4	633.83	2.38	20.96
Agente 5	476.66	1.79	15.77

TABLA 53 ITERACIÓN 5, R210, HONDA CB1 PRO



ILUSTRACIÓN 55 ITERACIÓN 5, R210, HONDA CB1 PRO

## Resultados R210, YAMAHA LIBERO 125 ITERACIÓN 1

Esta iteración se realiza teniendo en cuenta que la moto utilizada para hacer el recorrido es **YAMAHA LIBERO 125**.

ITERACIÓN	TIEMPO (ms)
ITERACIÓN 1	0.16801142692565918
ITERACIÓN 2	0.18400835990905762
ITERACIÓN 3	0.2440168857574463
ITERACIÓN 4	0.17602992057800293
ITERACIÓN 5	0.18403244018554688

TABLA 54 TIEMPOS ITERACIONES, R210, YAMAHA LIBERO 125

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	892.94	5.95	52.41
Agente 1	808.78	5.39	47.48
Agente 2	556.87	3.71	32.68
Agente 3	791.01	5.27	46.42
Agente 4	616.24	4.11	36.2
Agente 5	482.69	3.22	28.36

TABLA 55 ITERACIÓN1, R210, YAMAHA LIBERO 125

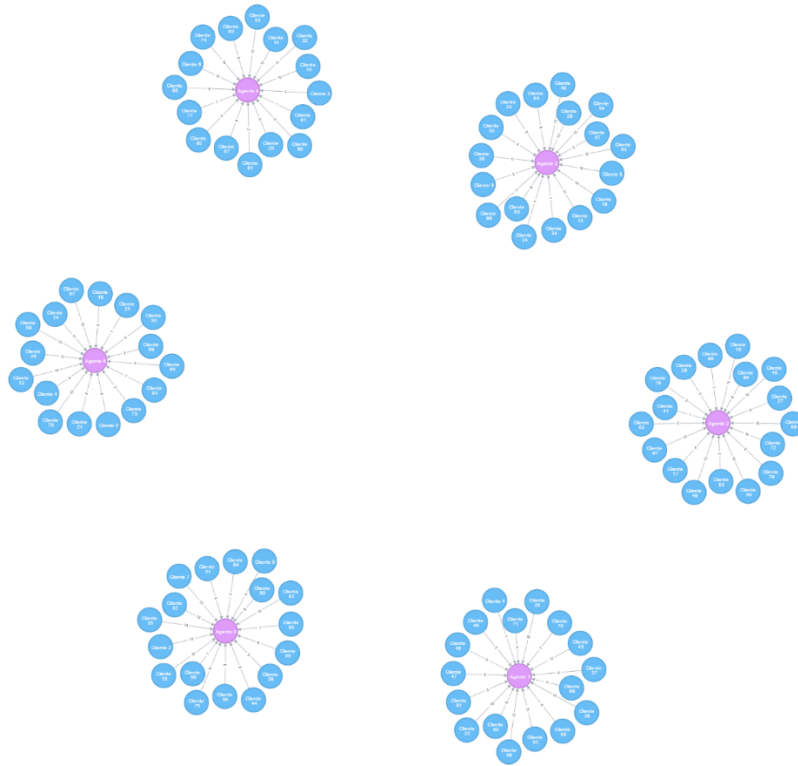


ILUSTRACIÓN 56 ITERACIÓN1, R210, YAMAHA LIBERO 125

**ITERACIÓN 2**

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	860.65	5.74	50.56
Agente 1	738.92	4.93	43.42
Agente 2	576.41	3.84	33.82
Agente 3	816.75	5.45	48
Agente 4	604.26	4.03	35.5
Agente 5	410.24	2.73	24.05

TABLA 56 ITERACIÓN2, R210, YAMAHA LIBERO 125



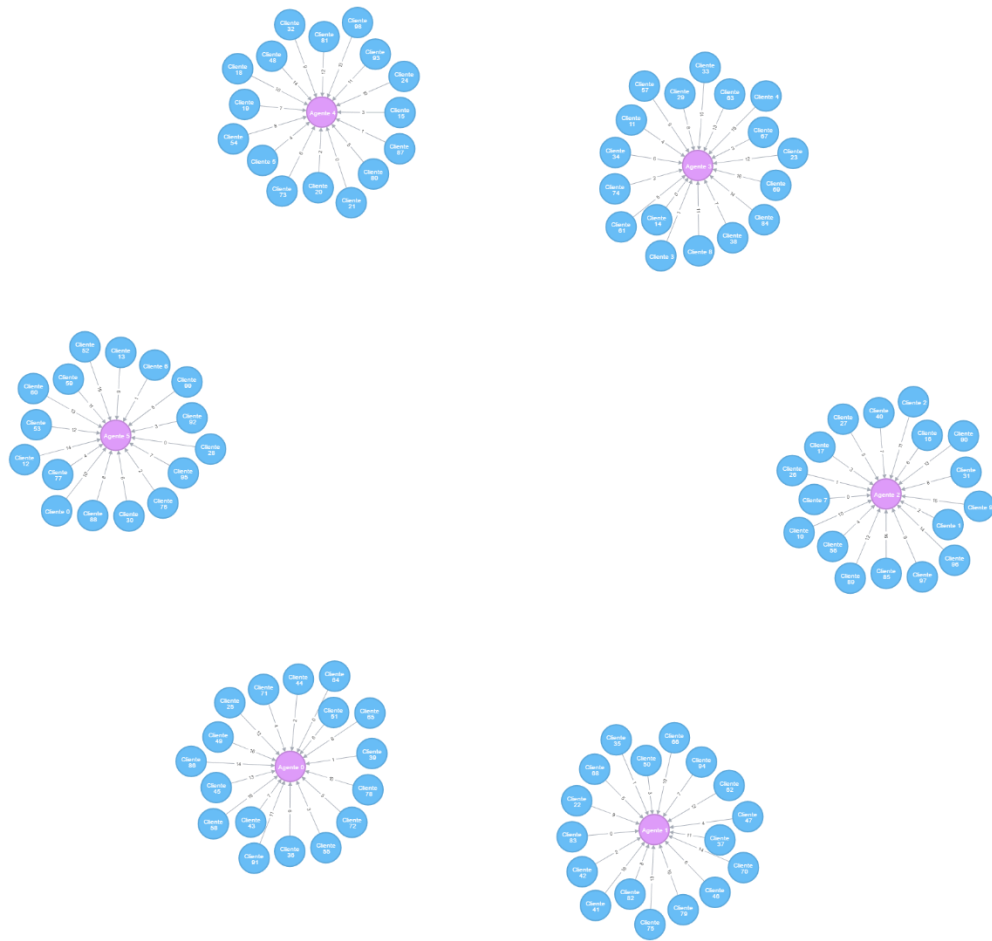


ILUSTRACIÓN 57 ITERACIÓN2, R210, YAMAHA LIBERO 125

### ITERACIÓN 3

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmissionTotal (KgCO2)
Agente 0	862.95	5.75	50.65
Agente 1	821.29	5.48	48.27
Agente 2	616.45	4.11	36.2
Agente 3	775.18	5.17	45.54
Agente 4	628.59	4.19	36.91
Agente 5	467.27	3.12	27.48

TABLA 57 ITERACIÓN3, R210, YAMAHA LIBERO 125

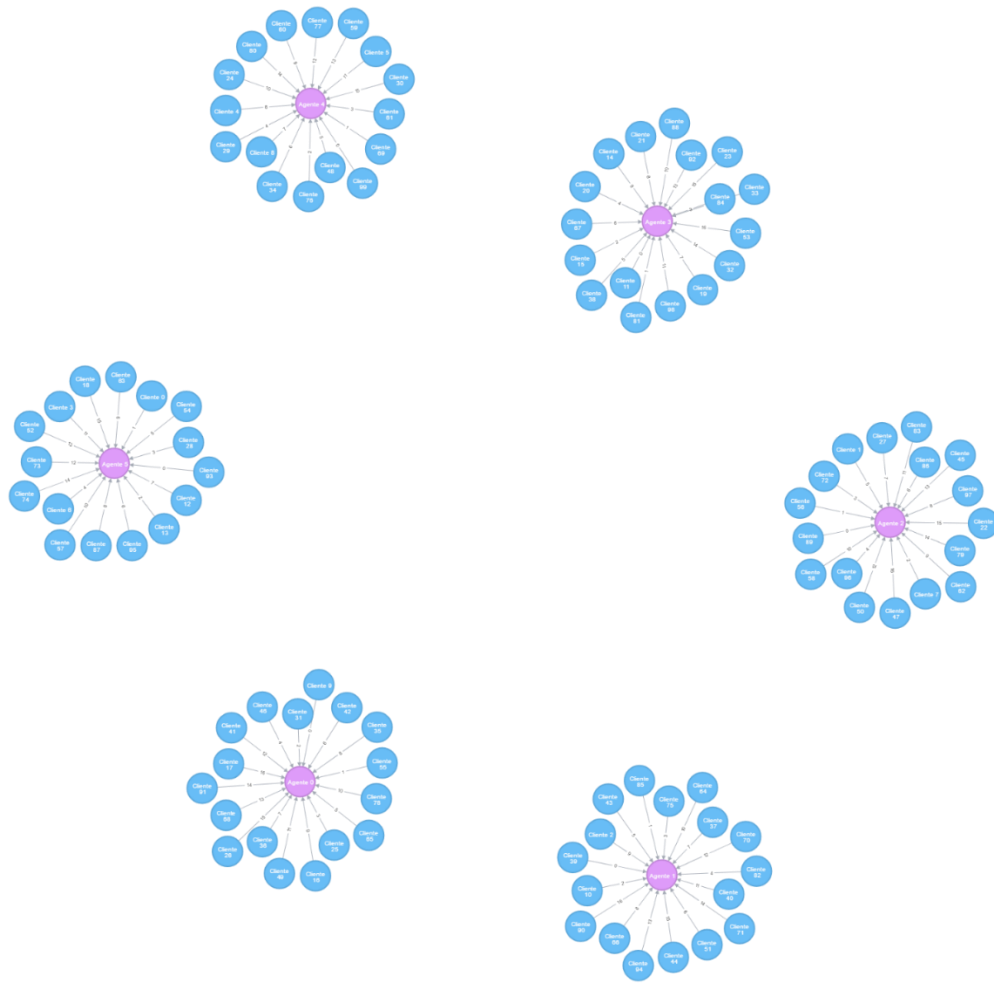


ILUSTRACIÓN 58 ITERACIÓN3, R210, YAMAHA LIBERO 125

#### ITERACIÓN 4

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	909.75	6.07	53.46
Agente 1	715.75	4.77	42.01
Agente 2	647.29	4.32	38.05
Agente 3	731.1	4.87	42.89
Agente 4	629.58	4.2	36.99
Agente 5	478.9	3.19	28.1

TABLA 58 ITERACIÓN4, R210, YAMAHA LIBERO 125

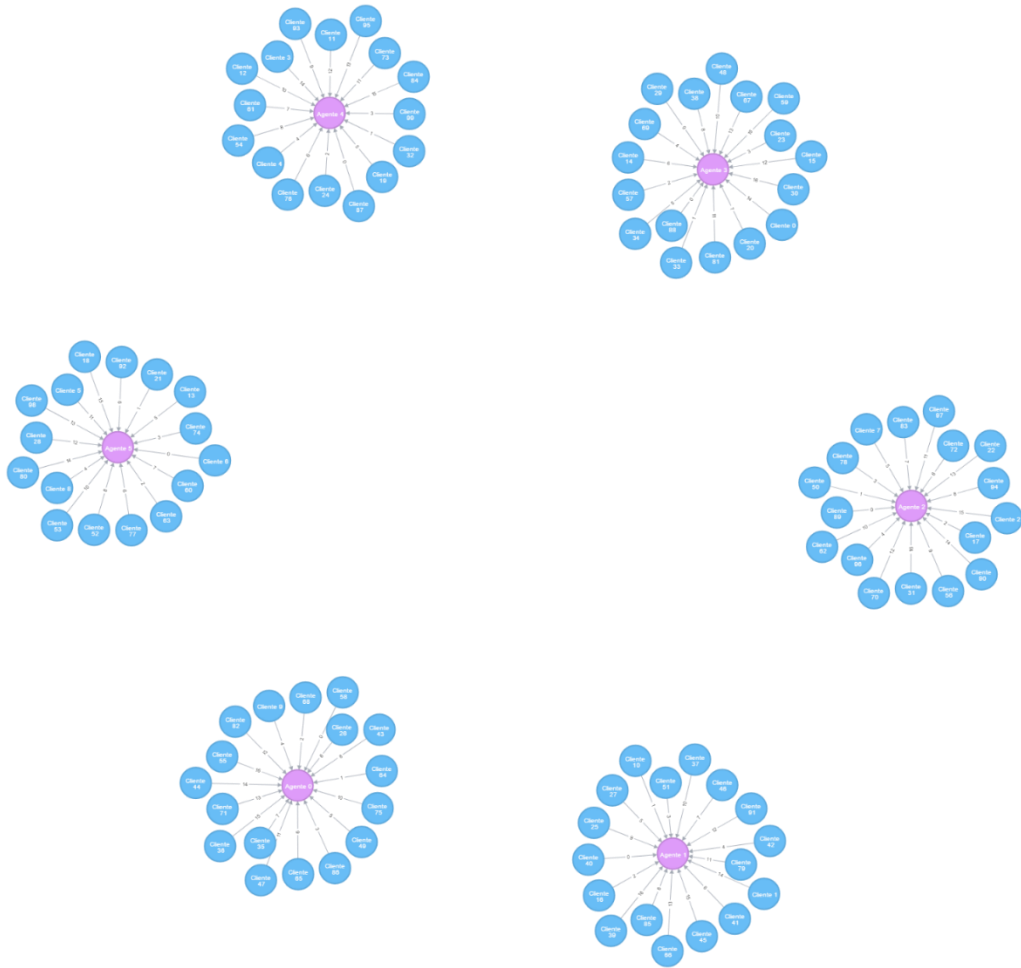


ILUSTRACIÓN 59 ITERACIÓN4, R210, YAMAHA LIBERO 125

### ITERACIÓN 5

NombreAgente	DistanciaRecorrida (Km)	ConsumoTotal (Galón)	EmisionTotal (KgCO2)
Agente 0	894.71	5.96	52.5
Agente 1	770.26	5.14	45.27
Agente 2	565.26	3.77	33.21
Agente 3	812.97	5.42	47.74
Agente 4	609.48	4.06	35.76
Agente 5	458.21	3.05	26.86

TABLA 59 ITERACIÓN5, R210, YAMAHA LIBERO 125



ILUSTRACIÓN 60 ITERACIÓN5, R210, YAMAHA LIBERO 125