TESIS USS

UNIVERSIDAD SEÑOR DE SIPÁN

FACULTAD DE INGENIERIA, ARQUITECTURA Y URBANISMO

ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA DE SISTEMAS

TESIS

DESARROLLO DE UN PLANIFICADOR DE RUTAS PARA RECOJO DE DESECHOS SÓLIDOS EN EL DISTRITO DE CHICLAYO UTILIZANDO ALGORITMO DE DIJKSTRA.

PARA OPTAR TÍTULO PROFESIONAL DE INGENIERO DE SISTEMAS

Autor (es):
ANTON BERNAL JUAN MANUEL

Asesor:
MG. BRAVO RUIZ JAIME ARTURO

Línea de Investigación:
Tecnologías de la Información y Comunicación

Pimentel – Perú
Año 2018

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Desarrollo de un planificador de rutas para recojo de desechos sólidos en el distrito de Chiclayo utilizando algoritmo de Dijkstra

Aprobación de la Tesis

Mg. Bravo Ruiz, Jaime Arturo
Presidente del jurado de tesis

Mg. Carrion Barco, Gilberto
Secretario del jurado de tesis

Mg. Ocampo Moreno, Rosa Liliana
Vocal del jurado de tesis

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

DEDICATORIA

Dedicada a mi familia entera, aquellas personas que me han apoyado directa e indirectamente durante toda mi carrera profesional, de manera incondicional, como lo son mis padres, mis hermanos, mi pareja, los docentes que me enseñaron con mucha entrega y a mi hermano mayor que Dios lo tiene en su gloria gracias a todos ustedes.

Juan Manuel Anton Bernal

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

AGRADECIMIENTO

Le agradezco a Dios porque él siempre está a mi lado para seguir adelante, gracias por darme la vida y permitir compartirla con los seres que amo, agradecerle en los momentos de adversidad porque me permite superarme día a día, a mis padres que siempre han estado conmigo dándome aliento, cariño y amor, a mis hermanos por ser aquellos con los que convivo a diario y me dan amor, a mi pareja por ser una de las personas que me acompañan siempre y me apoya incondicionalmente, a mis docentes por brindarme sus enseñanzas y consejos tanto profesionales como personales.

Juan Manuel Anton Bernal

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

INDICE

RESUMEN 11
ABSTRACT 12
INTRODUCCIÓN 13
CAPITULO I: PROBLEMA DE INVESTIGACIÓN 15
1.1. Situación problemática. 15
1.2. Formulación del problema 17
1.3. Delimitación de la investigación 17
1.4. Justificación e importancia de la investigación 18
1.5. Limitaciones de la investigación 19
1.6. Objetivos de la investigación 19
CAPITULO II: MARCO TEÓRICO 20
2.1. Antecedentes de estudios 20
2.2. Estado del arte 22
2.3. Base teórica científicas 27
2.3.1. Desechos Sólidos 27
2.3.2. Grafos 28
2.3.3. Algoritmos 33
2.3.4. Sistema 41
2.4. Definición de términos básicos. 42
2.4.1. Desechos sólidos. 42
2.4.2. Grafos. 42
2.4.3. Algoritmo. 43
2.4.4. Algoritmo Dijkstra 43
CAPITULO III: MARCO METODOLÓGICO 44
3.1. Tipo y diseño de investigación 44
3.1.1. Tipo de investigación 44
3.2. Población y muestra 44
3.2.1. Población 44
3.2.2. Muestra 44
3.3. Hipótesis 44
3.4. Variables 44

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

3.4.1. Variable Independiente 44
3.4.2. Variable Dependiente. 44
3.5. Operacionalización. 45
3.6. Abordaje metodológico, técnicas e instrumentos de recolección de datos. ... 46
3.6.1. Abordaje metodológico 46
3.6.2. Técnicas de recolección 46
3.6.3. Instrumentos de recolección de datos 46
3.7. Procedimiento para la recolección de datos 46
3.8. Principios éticos 47
3.9. Criterios de rigor científico 47
CAPÍTULO IV: ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS 48
4.1. Resultados en tablas y gráficos. 48
4.2. Discusión de resultados 51
CAPITULO V: PROPUESTA DE INVESTIGACIÓN 53
5.1. Generalidades de la propuesta. 53
5.2. Preparar el conjunto de datos. 54
5.2.1. Obtener puntos georreferenciados. 54
5.2.2. Convertir puntos georreferenciados a Grafos. 55
5.2.2.1. Poner en marcha el problema. 55
5.2.2.2. Modelado del problema. 55
5.2.2.3. Solución del problema. 56
5.3. Aplicar el algoritmo de Dijkstra sobre el grafo. 60
CAPITULO VI: CONCLUSIONES Y RECOMENDACIONES 124
6.1. Conclusiones 124
6.2. Recomendaciones 125
REFERENCIAS 126
ANEXOS 129
Anexo 1 - Guía de Observación científica. 129
Anexo 2 - Sector del Mapa. 131
Anexo 3 - Mapa con focos infecciosos. 132
Anexo 4 - Arcos que conforman el Grafo. 133
Anexo 5 - Manual de Usuario. 138
Anexo 6 – Mapa del sistema con la planificación de la ruta más corta de 14 focos infecciosos. 143

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Anexo 7 144
Anexo 8 145
Anexo 9 146
Anexo 10 147
Anexo 11 148

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

INDICE DE FIGURAS

Figura 1: Representación de un grafo etiquetado no dirigido 28
Figura 2: Representación de un camino a través de un grafo 30
Figura 3: Representación de un Dígrafo 32
Figura 4: Representación pseudoformal de una matriz de adyacencia. 32
Figura 5: Representación de un Dígrafo Acíclico 33
Figura 6: Ejemplo de Iteración con pseudo-programación 34
Figura 7: Ejemplo de Inducción 35
Figura 8: Ejemplo de Recursión 35
Figura 9: Tiempo que demora un algoritmo de f(n) pasos, si realiza 1 paso en 0.1 segundos 36
Figura 10: Grafo divido en grupos 37
Figura 11: Grafo ponderado y dirigido 39
Figura 12: Grafo con circuitos 40
Figura 13: Valores de longitud y latitud de un punto sobre la esfera 41
Figura 14: Ejes X, Y, Z del sistema de coordenadas cartesianas 42
Figura 15: Flujo de la realización de la investigación 53
Figura 16: Rutas simuladas obtenidas del servidor de aplicaciones de mapas “Google Maps” 55
Figura 17: Grafo no dirigido de la red de recolección de residuos sólidos 56
Figura 18: Grafo dirigido de la red de recolección de residuos sólidos 57
Figura 19: Matriz de adyacencia de un Grafo Dirigido. 59
Figura 20: Restricción del Algoritmo de Dijkstra sobre un Grafo. 60
Figura 21: Obtener la ruta optima v0-v9 (temporal) del Grafo dirigido de la red de recolección de residuos sólidos. 61
Figura 22: Obtener la ruta optima v0-v9 (permanente) del Grafo dirigido de la red de recolección de residuos sólidos 62
Figura 23: Obtener la ruta optima v9-v10 (temporal) del Grafo dirigido de la red de recolección de residuos sólidos 64
Figura 24: Obtener la ruta optima v9-v10 (permanente) del Grafo dirigido de la red de recolección de residuos solidos 65
Figura 25: Obtener la ruta optima v10 hacia sus adyacentes, del Grafo dirigido de la red de recolección de residuos sólidos 67
Figura 26: Obtener la ruta optima a partir del vértice v12, del Grafo dirigido de la red de recolección de residuos solidos 69
Figura 27: Obtener la ruta optima a partir del vértice v11, del Grafo dirigido de la red de recolección de residuos solidos 71
Figura 28: Obtener la ruta optima a partir del vértice v14, del Grafo dirigido de la red de recolección de residuos solidos 73
Figura 29: Obtener la ruta optima a partir del vértice v13, del Grafo dirigido de la red de recolección de residuos solidos 75
Figura 30: Obtener la ruta optima a partir del vértice v15, del Grafo dirigido de la red de recolección de residuos sólidos 77
Figura 31: Obtener la ruta optima a partir del vértice v7, del Grafo dirigido de la red de recolección de residuos sólidos. 79

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Figura 32: Trayectoria de la ruta optima de v0-v5, del Grafo dirigido de la red de recolección de residuos solidos 81
Figura 33: Obtener la ruta optima a partir del vértice v5, del Grafo dirigido de la red de recolección de residuos solidos 83
Figura 34: Obtener la ruta optima a partir del vértice v7, del Grafo dirigido de la red de recolección de residuos solidos 85
Figura 35: Obtener la ruta optima a partir del vértice v3, del Grafo dirigido de la red de recolección de residuos sólidos 87
Figura 36: Obtener la ruta optima a partir del vértice v15, del Grafo dirigido de la red de recolección de residuos sólidos 89
Figura 37: Obtener la ruta optima a partir del vértice v13, del Grafo dirigido de la red de recolección de residuos solidos 91
Figura 38: Obtener la ruta optima a partir del vértice v1, del Grafo dirigido de la red de recolección de residuos sólidos 93
Figura 39: Obtener la ruta optima a partir del vértice v6, del Grafo dirigido de la red de recolección de residuos sólidos 95
Figura 40: Obtener la ruta optima a partir del vértice v14, del Grafo dirigido de la red de recolección de residuos sólidos 97
Figura 41: Obtener la ruta optima a partir del vértice v11, del Grafo dirigido de la red de recolección de residuos sólidos 99
Figura 42: Obtener la ruta óptima a partir del vértice v8, del Grafo dirigido de la red de recolección de residuos sólidos. 101
Figura 43: Obtener la ruta optima a partir del vértice v0, del Grafo dirigido de la red de recolección de residuos sólidos. 103
Figura 44: Trayectoria de la ruta optima de v4, del Grafo dirigido de la red de recolección de residuos sólidos. 105
Figura 45: Trayectoria de la ruta optima de v5-v2, del Grafo dirigido de la red de recolección de residuos solidos 107
Figura 46: Trayectoria de la ruta optima de v0-v5-v2, del Grafo dirigido de la red de recolección de residuos sólidos 110
Figura 47: GUI del sistema informático. 123
Figura 48: GUI del sistema informático. 138
Figura 49: Selector del punto de partida**.** 138
Figura 50: Botón despliega focos infecciosos. 139
Figura 51: Ventana emergente con los focos infecciosos. 139
*Figura 52***:** Visor de focos infecciosos agregados 140
Figura 53: Agregar marcadores sobre el mapa del sistema. 140
Figura 54: Botón que genera la ruta más corta sobre el mapa del sistema. 141
Figura 55: Botón que borrar los marcadores del mapa del sistema 141
*Figura 56***:** Mapa del sistema planificador de rutas. 141
Figura 57: Resultados del algoritmo de Dijkstra sobre el planificador. 142

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

INDICE DE TABLAS

Tabla 1: 16
Tabla 2 54
Tabla 3 58
Tabla 4 59
Tabla 5 63
Tabla 6 66
Tabla 7 68
Tabla 8 70
Tabla 9 72
Tabla 10 74
Tabla 11: 76
Tabla 12: 78
Tabla 13: 80
Tabla 14: 82
Tabla 15 84
Tabla 16: 86
Tabla 17: 88
Tabla 18: 90
Tabla 19: 92
Tabla 20: 94
Tabla 21: 96
Tabla 22: 98
Tabla 23: 100
Tabla 24: 102
Tabla 25: 104
Tabla 26: 106
Tabla 27: 108
Tabla 28: 109

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

RESUMEN

En la presente investigación; se desarrolló un planificador de rutas para el recojo de desechos sólidos, el cual consiste brindar al operario del vehículo la ruta más corta que debe seguir para realizar la recolección de los desechos sólidos. Para realizar el sistema planificador de rutas para el recojo de desechos sólidos, se localizaron y ubicaron los puntos de acción; para luego construir la ruta óptima para poder llegar a cada uno de los puntos de acción; para ello se modeló el mapa del distrito de Chiclayo como un Grafo dirigido debido que será recorrido por un vehículo, el Grafo será almacenado mediante una Matriz de Adyacencia. Para obtener los datos que permitirán modelar el Grafo se usó el proveedor de aplicaciones de mapas Google Maps; para calcular la ruta más corta se implementó el Algoritmo de Dijkstra, tomando en cuenta el sentido de las calles, el punto de partida y los puntos de acción que vienen a ser los focos infecciosos. Los resultados del algoritmo de Dijkstra sobre un planificador de rutas, se observa una tendencia de mejora en el tiempo de ejecución del algoritmo a partir de 20 focos infecciosos en adelante, en un computador con procesador Corei7 en relación a un computador con procesador Corei5. Debido que, a más focos infecciosos el Algoritmo de Dijkstra realiza más iteraciones en busca del vértice más cercano por lo cual requiere más recursos computacionales. Asimismo se observó que el costo total de la trayectoria en cada caso no varía, siendo así independiente del procesador en el que se ejecute, por lo que el algoritmo de Dijkstra es confiable para la planificación de rutas. Se aplicó El algoritmo de Dijkstra para resolver el problema de la planificación de rutas de recojo de desechos sólidos, utilizando como fuente de datos un Grafo dirigido ponderado mediante una matriz de adyacencia, tomando en cuenta los focos infecciosos y el punto de partida, combinado con el lenguaje de programación PHP para darle funcionalidad.

Palabras claves: Algoritmo de Dijkstra, Grafo, Optimización de Rutas, Localización.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

ABSTRACT

In the present investigation; a route planner was developed for the collection of solid waste, which consists of providing the operator of the vehicle with the shortest route to follow in order to collect solid waste. To carry out the route planning system for the collection of solid waste, the action points were located and located; to then build the optimal route to reach each of the action points; For this purpose, the map of the district of Chiclayo was modeled as a directed graph that will be traversed by a vehicle, the Graph will be stored by means of an Adjacency Matrix. To obtain the data that will allow us to model the Graph, the Google Maps application provider was used; To calculate the shortest route, the Dijkstra Algorithm was implemented, taking into account the direction of the streets, the starting point and the action points that become the infectious centers. The results of the algorithm of Dijkstra on a route planner, we observe a tendency of improvement in the execution time of the algorithm from 20 infectious foci onwards, in a computer with Corei7 processor in relation to a computer with Corei5 processor. Because, to more infectious foci the Algorithm of Dijkstra realizes more iterations in search of the nearest vertex for which it requires more computational resources. Likewise, it was observed that the total cost of the trajectory in each case does not vary, thus being independent of the processor in which it is executed, for which the Dijkstra algorithm is reliable for route planning. The Dijkstra algorithm was applied to solve the problem of the planning of solid waste collection routes, using as a data source a Weighted Directed Graph using an adjacency matrix, taking into account the infectious foci and the starting point, combined with the PHP programming language to give it functionality.

Keywords: Dijkstra Algorithm, Graph, Route Optimization, Location.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

INTRODUCCIÓN

En la actualidad existen diversos problemas durante la recolección de los Residuos Sólidos Municipales (RSM), son presentados a continuación: Las rutas existentes de recolección no han pasado por un proceso de optimización, los operarios de los camiones no están capacitados, los camiones son sobrecargados dañando los mismo y el pavimento sobre el cual circulan, generando contaminación del aire debido al CO2 que producen, causando impacto sobre la salud y el bienestar de la población, por lo tanto para llegar a cumplir el servicio asignado de recolección de desechos sólidos, los camiones deben recorrer grandes distancias; debido que no son optimizadas las rutas de recolección, los operarios de los camiones trabajan más tiempo al pasar varias veces por el mismo punto o pasar por zonas innecesarias Cusco & Picón (2015).

En el distrito de Chiclayo se presentan estos mismos problemas; los vehículos que realizan el servicio de recojo de desechos sólidos cumplen con turnos y recorridos diferentes, sin embargo el servicio de recolección que brindan diariamente se realiza de manera no planificada, por lo que se atiende a lugares que acumulan residuos esporádicamente (Organizacion para el desarrollo sostenible & Ambiente y desarrollo sostenible S.A.C, 2013).

En relación a los problemas antes observados respecto a la planificación de rutas, diferentes autores implementaron mecanismos y/o algoritmos que permitan optimizar la planificación de rutas; como mencionan Zhang, Chen, & Li, (2015) que para empresas grandes que recorren grandes caminos, la optimización de rutas es importante en cuestión de logística, es por ello que implementaron el Algoritmo de dijkstra para resolver el problema de la ruta más corta asimismo se utilizó el método del mapa etiquetado. Por su parte Rosyidi, Pram Pradityo, & Gunawan (2014) mencionaron que para la planificación de rutas en una red de transporte con alta congestión vehicular, se implementó el algoritmo de Dijkstra mejorado incorporando un peso dinámico en base al tiempo.

Lo expresado anteriormente para poner bajo contexto se muestran algunos términos como el Algoritmo de Dijkstra que como explica Ruohonen (2013) el algoritmo de Dijkstra también llamado el algoritmo de los caminos mínimos, está conformado por vértices y aristas, donde las aristas contienen pesos no negativos y el peso de una trayectoria o camino es la suma de

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

los pesos de la trayectoria recorrida. Asimismo un grafo según Coto (2003) “es un conjunto de nodos unidos por un conjunto de líneas o flechas” (p. 4).

En la presente investigación se desarrolló un planificador de rutas para el recojo de desechos sólidos ubicados en diferentes puntos de acción (focos infecciosos), el cual establece una ruta en la menor distancia posible entre los mencionados puntos de acción y un determinado punto de partida. En el desarrollo de la investigación se aplicó el Algoritmo de Dijkstra para optimizar las rutas de recolección de desechos sólidos en un sistema informático, en la cual se ingresaron los focos infecciosos a recorrer y el punto de partida; de igual manera se empleó el API V3 de Google Maps para la obtención y visualización de mapas geográficos; las tecnologías mencionadas se aplicaron en el desarrollo del planificador de rutas con el fin de lograr el objetivo general.

El informe se divide en seis capítulos, en el primer capítulo encontramos la situación problemática, la formulación del problema, delimitación de la investigación, justificación e importancia, limitaciones de la investigación y los objetivos. En el segundo capítulo se encuentra el marco teórico, donde se detalla los antecedentes de la investigación a nivel internacional, nacional y local. Asimismo se muestran conceptos referentes al estudio. En el tercer capítulo, se encuentra el marco metodológico donde se precisó la operacionalización de variables y la interpretación de instrumentos utilizados. En el cuarto capítulo: se muestra el análisis e interpretación de los resultados. En el quinto capítulo: se muestra la propuesta de investigación en donde se aplicó el algoritmo de Dijkstra (algoritmo de optimización de rutas), finalmente en el sexto capítulo: se presentan las conclusiones y recomendaciones de la investigación.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

CAPITULO I: PROBLEMA DE INVESTIGACIÓN

1.1. Situación problemática.

Los residuos sólidos municipales para las grandes ciudades se ha considerado un problema mundial debido a que los ciudadanos generan altos volúmenes de residuos sólidos, al no ser manejados adecuadamente, puede afectar la salud de los habitantes y al medio ambiente como indica Contreras (2008) al no tener una buena administración de los residuos sólidos la salud de la población corre riesgo por tres situaciones. La primera por transmitir enfermedades bacteriales y parasitarias tanto por agentes patógenos transferidos por los residuos; en segundo lugar el riesgo de lesiones e infecciones ocasionados por los objetos punzo penetrantes que se encuentran en los residuos, esto pone en alto riesgo a las personas que se desempeñan como recicladores; y en tercer lugar la contaminación ocasionada por la combustión de residuos, la cual afecta el sistema respiratorio de los ciudadanos.

Ante esta problemática mundial la Organización de las Naciones Unidas (ONU) realizó en 1992 la Cumbre de la Tierra en Río de Janeiro, donde se formularon cuatro áreas de programas, aún vigentes, relacionadas con los residuos: disminuir al mínimo los residuos, crecimiento máximo de la reutilización y reciclado ecológico, promover la eliminación y el tratamiento ecológicamente racional de los residuos y ampliar los alcances de los servicios que se ocupan de los desechos; con la finalidad de fomentar el desarrollo sostenible y ecológico racional para el siglo XXI en todos los países (Organización de las Naciones Unidas, 1992).

Debido a ello para mejorar el manejo de los residuos sólidos la OEFA (2014) menciona que el Perú cuenta con diez rellenos sanitarios acreditados y en funcionamiento para una población que supera los treinta millones de ciudadanos, provocando que los residuos sólidos terminen en espacios informales, llamados botaderos; demostrando con ello que existen graves problemas de disposición final de los residuos sólidos.

Asimismo el Organizacion para el desarrollo sostenible & Ambiente y desarrollo sostenible S.A.C (2013), expresan que la generación de residuos sólidos domiciliarios (RSD), guarda relación directa con diferentes aspectos como número de habitantes,

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

costumbres, etc. los cuales repercuten de manera directa en la generación per-cápita (GPC) de residuos sólidos.

Por otro lado en el distrito de Chiclayo en el año 2016, diariamente se generó 141.25 toneladas de residuos sólidos, como se muestra en el Plan integral de gestión ambiental de residuos sólidos de la provincia de Chiclayo.

De acuerdo a lo expresado en el distrito de Chiclayo se estima la cobertura del servicio de recogida de residuos sólidos, recolectados por los vehículos asignados.

Tabla 1:

Cobertura de recolección de residuos sólidos municipales del distrito de Chiclayo.

Distritos	Cobertura (%)	Generación Total de RSM T/día
Chiclayo	85.82	194.51(1)

Fuente: Plan integral de gestión ambiental de residuos sólidos de la provincia de Chiclayo, Departamento de Lambayeque.

Como se puede apreciar en el cuadro 1, diariamente los residuos sólidos que no son recogidos es el 14.18% el cual equivale a 28 toneladas.

En relación a los problemas antes observados respecto a la planificación de rutas, diferentes autores implementaron mecanismos y/o algoritmos que permitan optimizar la planificación de rutas; como mencionan Zhang, Chen, & Li, (2015) que para empresas grandes que recorren grandes caminos, la optimización de rutas es importante en cuestión de logística, es por ello que implementaron el Algoritmo de dijkstra para resolver el problema de la ruta más corta asimismo se utilizó el método del mapa etiquetado. Por su parte Rosyidi, Pram Pradityo, & Gunawan (2014) mencionaron que para la planificación de rutas en una red de transporte con alta congestión vehicular, se implementó el algoritmo de Dijkstra mejorado incorporando un peso dinámico en base al tiempo. , así como también, De la Cruz, y otros (2016), expresan que para proveer al cliente la ruta más corta para comprar sus artículos en un supermercado, es necesario implementar un mecánico de búsqueda de rutas más

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

cortas, por lo que implementaron tres algoritmos los cuales son el algoritmo de Dijkstra, Bellman-Ford y Floyd-Warshall, las pruebas para los tres algoritmos se llevaron a cabo en un supermercado, donde se determinó el mejor de ellos, asimismo Chen, Shen, Chen, & Rui, (2014) expresaron que para la evacuación optima ante eventos de emergencia, lo primero que se debe realizar es observar la red de carreteras para poder construir un modelo dinámico de la red de evacuación basada en el algoritmo de Dijkstra y así poder obtener la mejor ruta de evacuación.

Es por ello que para solucionar el problema de recojo de desechos sólidos se desarrolló un planificador de rutas utilizando el algoritmo de Dijkstra, el cual brindara la ruta más corta y conocida al operario del camión, que servirá como un mecanismo para la optimización de rutas, en relación al recojo de desechos sólidos en el distrito de Chiclayo, para que los habitantes puedan vivir en un ciudad menos contaminada.

1.2. Formulación del problema

¿De qué manera se podrá determinar la trayectoria óptima para la recolección de desechos sólidos en el distrito de Chiclayo?

1.3. Delimitación de la investigación

La investigación se realizó en el distrito de Chiclayo del departamento de Lambayeque, en donde se desarrolló un planificador de rutas para recojo de desechos sólidos, tomando como puntos de destino los focos infecciosos simulados en un planeador de rutas.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

1.4. Justificación e importancia de la investigación

Esta investigación:

- Es pertinente debido que en la actualidad el servicio de recojo de desechos sólidos que brindan diariamente en el distrito de Chiclayo lo realizan de manera no planificada es por ello que implementar un sistema planificador de rutas que provee al operario del camión recolector la ruta más corta y conocida, sería oportuno para mejorar problemas relacionados a la contaminación.

- Es viable debido que para la construcción del planificador de rutas no necesitara de muchos recursos tecnológicos y a su vez es útil y necesario su implementación en el distrito de Chiclayo por razones de contaminación ambiental.

- En el aspecto tecnológico, el sistema se justifica ya que permitirá optimizar el proceso de recojo de desechos sólidos que los operarios del camión recolector realizan, debido que el sistema admite la ubicación y posicionamiento de los puntos de acción o focos infecciosos por donde se realizará la recolección de los desechos sólidos; de igual manera se realizará la optimización de rutas debido que se le brindara al operario del vehículo la manera más rápida y conocida de llegar hacia los puntos de acción, además proporciona la visualización de los puntos de acción o focos infecciosos en un mapa geográfico.

- En el aspecto económico, se justifica debido que la trayectoria del recorrido de los camiones recolectores de basura serán más cortas, por lo que se considera menor consumo de combustible utilizados en los mencionados vehículos.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

- En el aspecto ecológico, se justifica ya que el distrito de Chiclayo reducirá sus focos infecciosos respecto a los residuos sólidos, por lo que la ciudad estará menos contaminada.

- En el aspecto social, se justificó ya que el planificador de rutas de recolección de desechos sólidos, mejorará el bienestar común de los ciudadanos del distrito de Chiclayo.

1.5. Limitaciones de la investigación

- No contempla el tamaño del vehículo.
- No contempla el ancho de las calles y/o avenidas.
- No contempla el grado de congestión vehicular que existe durante el día.
- No contempla el tiempo que demore el vehículo en recoger los desechos sólidos de los focos infecciosos.
- No contempla el funcionamiento del planeador de rutas cuando no haya acceso a internet.
- En la presente investigación se contempla que la realización de la optimización de rutas se realizara un día antes que los vehículos salgan a recorrer los puntos de acción.

1.6. Objetivos de la investigación

Objetivo general

Desarrollar un planificador de rutas para recojo de desechos sólidos en el distrito de Chiclayo utilizando el algoritmo de Dijkstra.

Objetivos específicos

a) Preparar el conjunto de datos.
b) Aplicar el algoritmo de Dijkstra sobre el grafo.
c) Realizar pruebas sobre el algoritmo de Dijkstra.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

CAPITULO II: MARCO TEÓRICO

2.1. Antecedentes de estudios

Cusco & Picón (2015) Ecuador. Nos presenta la deficiencia que existe en la administración de los residuos sólidos municipales (RMS), entre los principales problemas reconocidos durante la recolección de RMS en la ciudad de Cuenca se obtuvo que las actuales rutas de recojo no pasaron por un proceso de optimización, por lo que se diseñó de manera experimental, los operarios de los vehículos no acatan las zonas asignadas, los camiones recolectores son recargados, por lo que implica que se viajen considerables trayectos; el camión recolector pasa repetidas veces por el mismo punto, causando pasar por lugares reiterados, aumentando el tiempo y gastos de combustible. Por lo indicado los autores plantearon optimizar las rutas a través de los Sistemas de Información Geográfica (SIG), distribuyéndolo en 6 fases, fase 1 recopilar información sobre las rutas de recolección de residuos sólidos en la ciudad de Cuenca, fase 2 analizar los datos recopilados, fase 3 escoger las herramientas más adaptables al estudio, fase 4 modelar diferentes esquemas de recolección en software de información geográfica, fase 5 analizar e interpretar los resultados obtenidos, fase 6 presentar los resultados obtenidos de las nuevas rutas.

Lucero & Viñamagua (2016) Ecuador. La investigación tuvo como objetivo principal optimizar y rediseñar las rutas de recolección de residuos sólidos de la parroquia Cayambe, a través del uso de Sistema de Información Geográfica (SIG), con el fin de minimizar la distancia recorrida y tiempo total durante la recolección de los residuos sólidos. Se utilizó para la elaboración del rediseño de las rutas, el software ArcGis 10.3 aplicando la herramienta Network Analyst – Vehicle Routing Problem (VRP), que permite modelar las redes de transporte, simulando un sistema vial de una área determinada con sus respectivos bloqueos y restricciones de circulación. Así mismo se diseñó las rutas de recolección en base a parámetros como: velocidad del vehículo recolector, tipo de vías, longitud de vías, capacidad del carro recolector y horario de trabajo. Se concluye que las macro rutas y micro rutas se optimizaron en tiempo y distancia total de recolección, demostrando también que la aplicación de este software

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

para el diseño presenta una gran ventaja, ya que permite realizar simulaciones de acuerdo al crecimiento poblacional.

Fernández-Peteiro (2016) España. Está investigación permitió localizar la ruta más corta entre dos ubicaciones dentro de un mismo ambiente, obviando las dificultades entre el punto inicial y el punto final. Por ello se analizó una serie de algoritmos, que tendrán que leer un mapa o entorno de trabajo, en el que se mostrarán todas las dificultades a evadir entre el punto inicial y el punto final. Los algoritmos empleados fueron tres: Dijkstra, A estrella (A*) y RRT (Rapidly Exploring Random Trees). De tal manera que un celular, hombre mecánico o cualquiera que sea nuestro sujeto es apto de fijar un camino que esquive las dificultades, construyendo así una ruta óptima en diferentes ambientes, ya sean reales o virtuales.

Taquia (2013) Perú. La investigación alude a la implementación de un método para optimizar rutas para una empresa de transporte de residuos sólidos, debido a que los problemas relacionados con el transporte corresponden entre el 10% y 20 % del costo final de los bienes de la empresa, más aun si la empresa cuyo servicio principal es el mismo transporte, en esta tesis se propuso desarrollar la optimización de rutas dividido en 2 etapas: La etapa 1 es la sectorización de rutas, donde se define el número de rutas necesarias en el distrito, la etapa 2 el modelo de optimización se propone una solución adecuada al modelo.

Barrantes (2015) Perú. El objetivo general de esta investigación fue implementar un algoritmo meta-heurístico recocido simulado que genere un diseño de rutas de un área de Lima Metropolitana para disminuir el tiempo de viaje total y la demanda insatisfecha. Planteando una solución técnica al problema del transporte público, realizando un algoritmo de inteligencia artificial para optimizar el diseño de rutas para los buses del transporte público. Así mismo las herramientas utilizadas para llevar a cabo la investigación fueron PgAdmin III, Postguis, Code: Blocks, OpenStreetMap (OSM), Osmosis y API de Google Maps.

Sato (2017) Perú. En su estudio buscó dar solución al problema de definición de las frecuencias y definición de horarios implementando algoritmos meta heurísticos que ayuden a definir frecuencias y horarios para cada una de las rutas que componen la

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

red de transporte del Metropolitano y el Metro. Para poder llevar a cabo la implementación del programa se implementó las funciones objetivos, primeramente para realizar la generación de soluciones aleatorias y luego cada una de los algoritmos que contiene la solución. Utilizando la programación multi-hilos se buscó mejorar la eficiencia de los algoritmos, se siguió los mismos para la definición de frecuencias y la definición de horarios.

Marchena (2015) Pimentel. El autor indica que ante la mala toma de decisiones de los operarios de campo de la entidad prestadora de servicios EPSEL S.A respecto a las instalaciones de medidores de los respectivos predios, conlleva a que parte del trabajo realizado por estos operarios quede inconcluso, por lo que al culminar sus funciones diarias aún les quede mucho por realizar. Por lo que el autor se planteó utilizar el Algoritmo de Dijkstra para la optimización de rutas de trabajo, tomando como parámetros los predios en donde se realizan las instalaciones, además se realizó la sectorización geográfica utilizando diagramas de Voronoi, para obtener regiones definidas en donde se tomaran los puntos a optimizar, asimismo se utilizó el API V3 de Google Maps para la visualización de los mapas geográficos, con el fin de visitar los puntos en el menor tiempo posible, menor distancia y menor coste de recorrido.

Ruiz & Vidal (2016) Pimentel. La investigación tiene como objetivo general diseñar un modelo de optimización en el sistema de recojo de residuos sólidos en el distrito de Reque, debido a que en la Carretera Panamericana, pueblos jóvenes, y en algunas calles del distrito, se observa acumulación de basura. Por ello analizaron métodos de optimización de rutas mediante teoremas y aplicaciones prácticas, mencionando los fundamentos, restricciones y algoritmos adecuados para el caso de la empresa en estudio. Teniéndose como propuesta de estudio el desarrollo de la sectorización de rutas y el modelo de optimización.

2.2. Estado del arte

Timebase Dynamic Weight for Dijkstra Algorithm Implementation in Route Planning Software (Rosyidi, Pram Pradityo, & Gunawan, 2014)**,** los autores mencionaron en su artículo que en la actualidad los algoritmos que encuentran el

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

camino más corto no toman en cuenta el tráfico vehicular en base al tiempo de la red de carreteras, es decir tratar de calcular la distancia más corta en función del perfil de tráfico de la red de carreteras, tomando en cuenta los días laborales y fines de semana. Es por ello que los autores plantearon la implementación del algoritmo de Dijkstra mejorado, debido que para una red dependiente del tráfico, el algoritmo de Dijkstra clásico que obtiene la ruta más corta, no puede determinarse por la distancia debida que dependerá mucho del tráfico vehicular. La implementación del algoritmo de Dijkstra mejorado se adapta a la condición del tráfico, en base al tiempo de la red de carreteras, lo primero que realizaron fue implementar el factor del tráfico dinámico en función del tiempo, donde se le asigna un perfil de tráfico para cada carretera entre un par de nodos, entonces el perfil del tráfico se define como una función de tiempo, lo segundo que se realizó, es realizar el cálculo del algoritmo de Dijkstra incluyendo un peso de datos de tráfico durante un tiempo definido. Como resultado de la implementación del algoritmo de Dijkstra incluyendo un peso dinámico en base al tiempo, ayudo en la planificación de rutas, para obtener el máximo rendimiento es cuestiones de tiempo, consumo de combustible y costo.

Research On The Optimal Route Choice Based On Improved Dijkstra (Mingjun & Meng, 2014)**,** indicaron que en la actualidad; debido al incremento de vehículos sobre todo en las grandes ciudades empeora la congestión de tráfico, es por ello que el algoritmo de Dijkstra clásico si bien determina la ruta más corta, ya no es necesariamente la óptima, es por ello que los autores para resolver el problema de la ruta óptima presentaron el Algoritmo de Dijkstra mejorado, que introduce la función de la congestión del tráfico, donde esta función define el nuevo peso de acuerdo con la congestión del tráfico, entonces lo primero que realizan es los siguiente: (1) Determinan la longitud de la ruta como factor importante por ser la principal opción para los conductores, (2) luego determinan la calidad de la carretera tomando como parámetros el número del carril, el ancho de la carretera y la rigurosidad de la superficie de la carretera, (3) posteriormente determinan la congestión del tráfico que varía dependiendo de las horas, fechas festivos y hábiles, (4) se determina función de peso y se combina con el algoritmo de Dijkstra, entonces la ruta optima se calculara utilizando la distancia de congestión y esta puede variar dependiendo la congestión

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

del tráfico en una hora determinada en una ruta determinada, en términos generales se seleccionó la ruta óptima de acuerdo al algoritmo de Dijkstra mejorado en función a una ponderación basada en el análisis del Algoritmo de Dijkstra clásico, mostrando un mejor rendimiento, debido que da una ruta más razonable para los conductores.

Path optimization study for vehicles evacuation based on dijkstra algorithm. (Chen, Shen, Chen, & Rui, 2014), mencionaron que ante eventos de emergencia tales como terremotos, incendios, etc. Que puedan poner en riesgo la integridad humana en especial para aquellas poblaciones con alta densidad poblacional, es necesario tener una ruta de evacuación óptima para salva guardar la integridad humana, es por ello que los autores plantearon realizar un modelo dinámico de la red de carreteras para vehículos basada en el algoritmo de Dijkstra tomando en cuenta diversos factores como el tránsito vehicular, intersecciones, impedancia, etc. En conclusión la investigación se compone en 5 fases: En la fase 1 se construye un modelo dinámico de la red de carreteras basado en el algoritmo de Dijkstra, donde se utilizaran autobuses u otro vehículo que evacuen multitudes de personas tomando en cuenta la impedancia de las carreteras e intersecciones y otros parámetros como los diferentes periodos establecidos tales como el pico de la mañana, común y noche, en la fase 2 se hizo un cálculo para el modelo introduciendo datos de la ruta urbana, se obtuvo el peso de las aristas y nodos utilizando el modelo de función de impedancia dinámica del camino, en la fase 3 se analizan los resultados en los diferentes periodos, en la fase 4 se selecciona el camino optimo según el periodo pico de la mañana, común y noche y en la fase 5 concluye el documento. Esta investigación nos muestra a través de un modelo matemático basado en el algoritmo de Dijkstra la selección óptima de rutas de evacuación en eventos de emergencia en 3 diferentes periodos.

Design of geographic information system for tracking and routing using dijkstra algorithm for public transportation. (Indrajaya, Affandi, & Pratomo, 2015). Indicaron en el artículo que problemas de seguimiento relacionados en especial al transporte público son muy importantes en la vida cotidiana, por lo cual se diseñó una aplicación de Sistemas de Información Geográfica (SIG) basado en el algoritmo de Dijkstra, para el transporte público el cual realizara un seguimiento en tiempo real de la posición de cada taxi y capaz de brindar la mejor ruta hacia cada pasajero, debido a que se

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

consume mucho tiempo en la espera por parte de los pasajeros al reservar un taxi o el consumo incensario de combustible. Por estas razones los autores plantearon realizar un sistema que se construye de la siguiente manera, primero un Smartphone android basado en receptores GPS colocados en cada vehículo (OBU) y una unidad de servidor que recibe información de la posición de cada receptor GPS, los cuales sirven para posicionar el taxi en los mapas de Google Maps, además los datos de longitud y latitud que fueron enviados por el receptor de GPS son almacenados en una base de datos que será procesado por el servidor para determinar la posición del taxi más cercano con el pasajero, a través del algoritmo de Dijkstra, tomando además como parámetros la posición del cliente, el tráfico de las calles urbanas, los resultados obtenidos son en base a la comparación con los servicios de transporte ya existentes respecto a la mejora en el tiempo de espera de cada pasajero, por lo tanto los beneficios reflejados son reducción en el tiempo de espera del cliente respecto a la reserva del taxi, reduce el consumo de combustible y tiempo.

En esta investigación se utilizó el algoritmo de Dijkstra para obtener una ruta óptima tomando como parámetros los receptores GPS de cada vehículo (latitud y longitud), posición del cliente y la congestión del tráfico, apoyándose de los Sistemas de Información Geográfica (SIG), tecnología GPS y teléfono inteligente Android.

Optimization of Logistics Route Based on Dijkstra. (Zhang , Chen, & Li, 2015) En su artículo demostraron que los principales problemas que se dan en empresas grandes o por lo menos medianas respecto a su logística es la distancia de cada ruta de distribución debido a que estas son grandes, además que la producción y el consumo de todo el material no puede hacerse sin transporte, es por ello que los autores plantearon utilizar en primer lugar el algoritmo de Dijkstra para resolver el problema de la ruta más corta y utilizaron el método del mapa etiquetado que representa la información del grafo dirigido ponderado, en segundo lugar utiliza el lenguaje de programación C, en la aplicación del algoritmo lo divide en 3 pasos: Paso 01 está conformado por: poner en marcha el problema y formulación del problema a tratar, paso 02 conformado por: el modelado del problema, donde se obtiene los componentes necesarios para poner en marcha el problema como lo son el camión y sus destinos a llegar, paso 03 solución del problema, se propone el nodo inicial el cual

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

posteriormente servirá para encontrar la ruta óptima, con el objetivo de reducir los costó logísticos y mejorar los ingresos de la empresa. En esta investigación se utilizó el algoritmo de Dijkstra para modelar la ruta de distribución a través de grafos combinando con el lenguaje C.

Items-mapping and Route Optimization in a Grocery Store using Dijkstra's, Bellman-Ford and Floyd-Warshall Algorithms. (De la Cruz, y otros, 2016). En su artículo, realizaron estudios donde nos presentan las experiencias que tienen los clientes al seleccionar los productos deseados en una tienda de comestibles y las dificultades que estos presentan al comprar dichos productos debido a que estos son agregados cada cierto tiempo como resultado hace que los clientes pasen más tiempo vagabundeando dentro de la tienda de comestibles, es por ello que los autores plantean realizar una evaluación entre los algoritmos de Dijkstra, Bellman-Ford y Floyd-Warshall, para así obtener la ruta de acceso más óptima, la aplicación se realizó bajo una arquitectura cliente-servidor, por el lado del cliente se encuentra una Tablet y por el lado del servidor un software desarrollado bajo el lenguaje C en un entorno Microsoft Visual Studio, se utilizó tecnologías como Wi-Fi.

La metodología propuesta se divide en 4 fases: Fase 1, base de datos, se realiza el modelado de la base de datos es el de un típico diseño de Tienda, compuesto por productos, categorías, dividido en 20 estantes que posteriormente servirán para realizar el mapeo del Grafo dirigido, en la fase 2, optimización de técnicas, se explica el funcionamiento de las técnicas a evaluar, posteriormente se representa el diagrama de flujo de la aplicación, fase 3, evaluación, se evaluaron los 3 algoritmos para obtener la ruta más óptima de acuerdo a los elementos seleccionados por los clientes, posteriormente se compararan los resultados, la evaluación se realizó en base a 2 criterios, el primero es el costo, que viene ser la distancia total calculada por el Algoritmo y el segundo es el tiempo computacional el cual consiste en el tiempo total que se toma la cadena de datos en encontrar la ruta más corta, finalmente en la fase 4, son las pruebas, se explican aspectos técnicos de la aplicación y lugar donde se desarrolló, la aplicación fue realizada con la finalidad de reducir el tiempo, distancia recorrida y el número de pasos en vano alrededor de la tienda de comestibles. En esta investigación se obtuvo la optimización de rutas de acuerdo a los productos

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

seleccionados por un cliente dentro de una tienda de comestibles lo cual genera un modelo dinámico para la ruta.

2.3. Base teórica científicas

2.3.1. Desechos Sólidos

Say (2005), explica que los desechos sólidos han ido incrementado en relación al desarrollo de la industria y la tecnología, estos se definen como desperdicios sólidos, semisólidos que carecen de valor actual por la persona que los desecha y que a su vez contaminan el medio el medio ambiente causando problemas en los seres vivos y su entorno. Además, el autor clasifica los desechos sólidos según su composición, tiempo que demoran en degradarse y su origen, según su composición se clasifica en basura orgánica e inorgánica. Basura orgánica, es todo aquello que estuvo vivo es decir de origen orgánico como las cascaras, hojas, huesos, etc. Basura Inorgánica, es todo aquello de origen industrial y materiales como plástico, telas sintéticas, etc. Según el tiempo que demoran los materiales en degradarse como los desechos biodegradables y no biodegradables.

Desechos biodegradables, son aquellos que se degradan en un tiempo corto como los alimentos. Desechos no biodegradables, son aquellos que demoran mucho tiempo en desaparecer como el plástico, televisor, etc.

Según su origen se componen de residuos sólidos urbanos (RSU) y residuos tóxicos y peligrosos (RTP). Los RSU son la basura generada a diario en los hogares, restaurantes, calles, mercados, entre otros. Los RTP son generados en procesos industriales y la manera de su recolección es de una manera especial ya que contienen sustancias que son poco comunes.

La explicación expresada por el autor define el concepto de Desechos Sólidos y su clasificación de acuerdo a ciertas características que lo componen.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

2.3.2. Grafos

2.3.2.1. Grafo No Dirigido.

Ruohonen (2013) explica que un grafo está formado por (V, E), donde V representa los vértices y E representa los bordes los cuales deben estar conectados a los vértices, los vértices y los bordes pueden ser etiquetados tanto por números como por letras. En el caso de los vértices cuando se etiquetan a través de letras se expresan de la siguiente manera "$a, b, c, \ldots$", o "$v1, v2, v3, \ldots$", cuando se etiqueta a través de números se expresan de la siguiente manera "$1,2,3, \ldots$", por otro lado los bordes cuando se etiquetan a través de letras se expresan de la siguiente manera "$a, b, c \ldots$", o "$e1, e2, e3 \ldots$", cuando se etiqueta a través de números se expresan de la siguiente manera "1, 2, 3,…", ver figura 1.

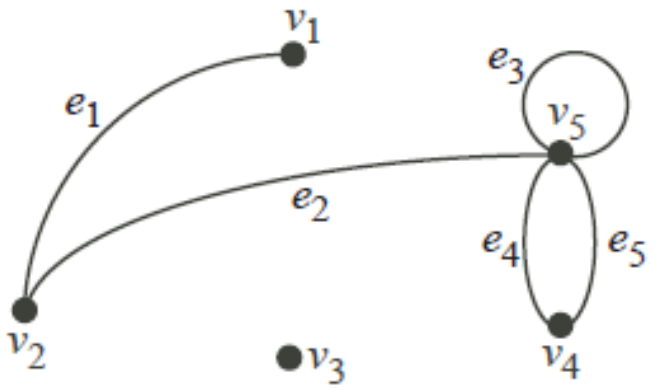


Figura 1: Representación de un grafo etiquetado no dirigido

Fuente: Teoría de Grafos.

Además el autor explica que un grafo para considerarse un grafo debe de tener un conjunto de características detalladas a continuación:

a. Los vértices u y v, son vértices finales del borde (u, v).
b. Los bordes que tienen los mismos vértices de destino son paralelos.
c. Un borde representado como (v, v), es un bucle.
d. Un grafo es simple si no tiene bordes o bucles paralelos.
e. Un grafo sin bordes, es decir que E esta vacío representa un grafo vacío

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

f. Si un grafo no tiene vértices, es decir que $V\ y\ E$ están vacíos representan un grafo nulo.
g. Si un grafo tiene un solo vértice es un grafo intranscendente o que no tiene importancia.
h. Los bordes son adyacentes si comparten un vértice final en común.
i. Los vértices (u, v), si están conectados por un borde son vértices adyacentes.
j. El grado del vértice V, representado como $d(v)$, es el número de bordes que entran en el vértice V.
k. Un vértice colgante es un vértice cuyo grado es 1.
l. El borde que tiene un vértice colgante como vértice final es un borde colgante.
m. Un vértice aislado es un vértice cuyo grado es 0.

Lo expresado por el autor, define el concepto y funcionamiento de un grafo así como también las características que debe tener para ser considerado un grafo, esto sirve como base para construir el Grafo y posteriormente aplicar el Algoritmo de Dijkstra sobre el grafo, asimismo saber representar los objetos o elementos abstraídos del sistema de aplicaciones de mapas Google Maps.

2.3.2.2. Caminos, Senderos, Rutas, Circuitos, Conectividad.

De acuerdo con Ruohonen (2013) un camino es un grafo representado por $G = (V, E)$, donde V son los vértices y E son los bordes, todo camino se inicia en un vértice, además si el vértice es un vértice inicial es representado como V_{i_0} y el vértice final representado por V_{i_k}, además k representa el tamaño del camino y si el vértice inicial es diferente al final se considera un camino abierto por el contrario es cerrado, ver figura 2.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

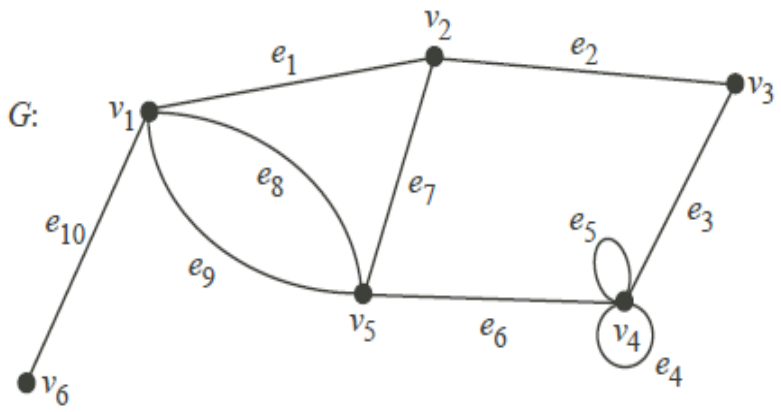


Figura 2: Representación de un camino a través de un grafo

Fuente: Teoría de Grafos.

Se tiene los siguientes vértices y aristas alternadas $v2, e7, v5, e8, v1, e8, v5, e6, v4, e5, v4, e5, v4$, el cual representa un camino abierto, por el contrario el camino $v4, e5, v4, e3, v3, e2, v2, e7, v5, e6, v4$ representa un camino cerrado.

Si cualquier vértice es visitado por más de una vez se le considera un Sendero, además si un camino es cerrado se le considera un circuito, por el contrario es una ruta.

Ejemplo: En el siguiente camino $v2, e7, v5, e6, v4, e3, v3$ representa una ruta, por el contrario $v2, e7, v5, e6, v4, e3, v3, e2, v2$ representa un circuito y si un camino u, v está conectado y v, w están conectados entonces existe un camino u, w, debido que existe una conexión a lo largo de todos los vértices.

Esta definición define el concepto y explica funcionamiento de un camino o ruta, por lo que en el presente proyecto los caminos o rutas se ven representados por las vías, calles o autopistas que vienen a ser las aristas, los focos infecciosos y/o lugares de concentración de desechos sólidos representan los vértices.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

2.3.2.3. Grafos dirigidos

Según Ruohonen (2013) un grafo dirigido o también llamado dígrafo se representa por un par (V, E) donde V es el conjunto de vértices y E es el conjunto de aristas dirigidas o arco (es decir tienen una orientación o dirección), que están conectados, además el autor describe algunas características que debe tener un grafo dirigido para ser considerado un "Grafo Dirigido", detallados a continuación:

a. El vértice inicial es el vértice u y el vértice final es el vértice v del arco (u, v).
b. El grado de salida del vértice V es el número de arcos fuera de él denotado $d^{+}(v))$.
c. El grado de entrada del vértice V es el número de arcos que ingresan denotado $d^{-}(v))$.
d. Un camino dirigido el cual puede ser un camino, ruta o circuito es representado de la siguiente manera $v_{i_0}, e_{j_1}, v_{i_1}, e_{j_2}, \dots, e_{j_k}, v_{i_k}$, es decir intercalando vértices y aristas.
e. Los vértices u y v están fuertemente conectados si existe un ruta $u - v$ dirigido asimismo una ruta $v - u$ dirigida en G.
f. Un grafo dirigido está fuertemente conectado si cada par de vértices están fuertemente conectados es decir de $u - v$ y $v - v$.
g. Por convención un grafo con un solo vértice es un grafo fuertemente conectado.

Ejemplo. En el siguiente grafo dirigido.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

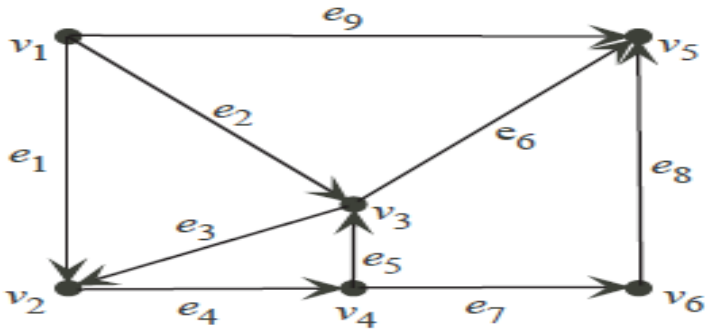


Figura 3: Representación de un Dígrafo

Fuente: Teoría de Grafos.

Esta explicación define como funciona un grafo dirigido (también llamado dígrafo), donde su arco u arista dirigida es un par ordenado de vértices es decir que tienen una orientación, el cual representara la dirección de las calles a ser mapeadas sobre el cual el algoritmo de Dijkstra construirá la ruta más corta.

2.3.2.4. Representación en el computador.

Coto (2003), explica que para representar un Grafo en un computador se puede realizar mendiante una matriz de adyacencia, ver figura 4.

```
CLASE Grafo
      Privado:
            Arreglo de <Tipo> Nodos de [1..N];
            Arreglo de logico MatrizDeAdyacencia de [1..N][1..N];
      Publico:
            # Todas las operaciones aquí
FCLASE
```

Figura 4*:* Representación pseudoformal de una matriz de adyacencia.

Fuente: Algoritmos Básicos de Grafos.

En la figura anterior se visualiza un arreglo de nodos y una matriz, esta ultima las filas y columnas representan los nodos del Grafo, se le asigna un valor verdadero (TRUE) si la posicion (i,j) de la matriz indica que hay una arista conectada del nodo i con el nodo j. Esta representacion matricial como indica Coto es muy util cuando se conoce el número exacto de nodos en el Grafo, debido que es una estructura estatica y por ende mas facil de acceder a sus datos.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

2.3.2.5. Grafos dirigidos Acíclicos

Ruohonen (2013) explica que un dígrafo con al menos un circuito dirigido es decir que el vértice de inicio es igual al vértice final y sus vértices tienen una orientación (por eje $v1, e1, v2, e3, v3, e3, v4, e4, v1$), representa un dígrafo cíclico por el contrario es Acíclico si $v_{i0}, e_{j1}, v_{i1}, e_{j2}, \dots, e_{jk}, v_{jk}$, por el cual la longitud máxima es k, dado que $v_{i0} \neq v_{ik}$, es decir que sus vértices de inicio y fin no son las mismas, ver figura 5.

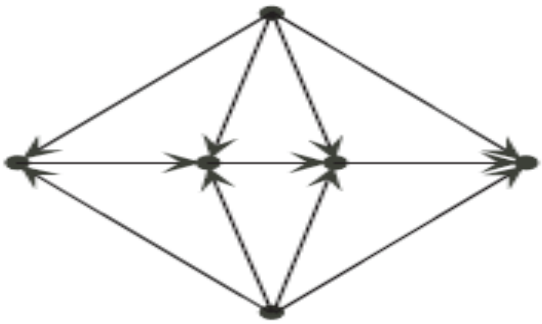


Figura 5: **Representación** de un Dígrafo Acíclico

Fuente: Teoría de Grafos.

La explicación expresada por el autor nos sirve para poder entender cuando diferenciar un dígrafo Acíclicos de un cíclico al momento de mapear el Grafo que nos servirá posteriormente para aplicar el Algoritmo de Dijkstra.

2.3.3. Algoritmos

2.3.3.1. Algoritmo

Joyanes, Rodriguez, & Matilde (2000), definen un algoritmo como un conjunto de pasos ordenados, no ambiguo (es decir cada acción a desarrollar dentro del algoritmo es única), que conducen a la solución del problema planteado y expresado en lenguaje natural (entendible por el ser humano), además indican que un algoritmo debe seguir 3 características:

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

a. Preciso, es decir que se especifica el orden en cómo se realizaran cada uno de los pasos.
b. Definido, que al utilizar el algoritmo ingresando los mismos datos, como resultado debe de entregar los mismos resultados.
c. Finito, es decir que el algoritmo en algún momento debe terminar (tener un número de pasos finitos).

Además los autores consideran 3 partes dentro del algoritmo:

a. La entrada, es la información que recibe el algoritmo, es decir los datos de entrada.
b. Proceso, son los cálculos que realiza el algoritmo para encontrar la solución del problema.
c. Salida, son los resultados que arroja el algoritmo luego de realizar el proceso de cálculo.

2.3.3.2. Tipos de algoritmo

Moreno & Ramírez (2011), los autores clasifican los algoritmos en tres tipos de ellos: iteración, inducción y recursión explicadas a continuación. La iteración es el proceso de repetir una rutina varias veces sin que esta sea repetida de manera manual es decir cada vez que se ejecute el proceso, un ejemplo en lenguaje de pseudo-programación es el siguiente, ver figura 6.

```
Entrada: n                /* número de escalones */
  para i = 1 hasta n
    Subir un escalón
  fin
```

Figura 6: Ejemplo de Iteración con pseudo-programación

Fuente: Grafos: fundamentos y Algoritmos.

La inducción es la prueba de la veracidad o falsedad de una proposición lógica $S(n)$, la cual depende de una variable n, los pasos a realizar son detallados a continuación.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

a. Se realiza una prueba de una base $S(n)$
b. Se prueba el paso inductivo, que se refiere a $S(n+1)$, el paso inductivo se puede identificar con la proposición lógica $S(n) => S(n+1)$, demostración de un prueba por inducción en la figura 7.

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}, \quad \forall n \in \mathbb{N},$$

Figura 7: Ejemplo de Inducción

Fuente: Grafos: fundamentos y Algoritmos.

Recursión es el proceso de llamar a procedimientos más pequeños y estos a su vez a otros, con el fin de llegar a un procedimiento $P(n)$ (es decir la implementación de un computador como una serie de pasos), ver figura 8.

Algoritmo 1.3 subir_escalera

```
Entrada: n                /* número de escalones */
  si n = 0 entonces
    Parar                 /* pues hemos llegado */
  sino
    Subir un escalón
    Ejecutar subir_escalera(n − 1)
  fin
```

Figura 8: Ejemplo de Recursión

Fuente: Grafos: fundamentos y Algoritmos.

2.3.3.3. Eficiencia de un algoritmo

Moreno & Ramírez (2011)**,** definen que la eficiencia de un algoritmo está ligada principalmente al tiempo razonable que toma este para resolver un problema determinado. Estos a su vez son determinados por la complejidad de un algoritmo explicados en el siguiente tema.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Los autores definen la eficiencia de un algoritmo, el cual para el presente proyecto es necesario saberlo, debido que los indicadores presentados guardan relación con el tiempo de demora.

2.3.3.4. Complejidad de un algoritmo

Moreno & Ramírez (2011) definen que la complejidad de un algoritmo se representa principalmente en dos aspectos. El tiempo que demora el algoritmo en resolver algún problema respecto al tamaño del problema y el tiempo de resolución en el peor de los casos como medida de eficiencia para un algoritmo, estos a su vez conformados por número de pasos u operaciones básicas que involucra la resolución como función del tamaño del problema, además explican que el cálculo de la complejidad de un algoritmo solo representa el tiempo de resolución en el peor caso, a continuación en la siguiente figura se visualiza la representación que denota el tiempo de resolución en diferentes casos, ver figura 9.

$n \setminus f$	$\log(n)$	n	$2n$	n^2	$2n^2$	2^n	$2 * 2^n$
1	0	0,1	0,2	0,1	0,2	0,2	0,4
2	0,1	0,2	0,4	0,4	0,8	0,4	0,8
3	0,158	0,3	0,6	0,9	1,8	0,8	1,6
4	0,2	0,4	0,8	1,6	3,2	1,6	3,2
5	0,232	0,5	1	2,5	5	3,2	6,4
6	0,258	0,6	1,2	3,6	7,2	6,4	12,8
7	0,28	0,7	1,4	4,9	9,8	12,8	25,6
8	0,3	0,8	1,6	6,4	12,8	25,6	51,2
9	0,317	0,9	1,8	8,1	16,2	51,2	102,4
10	0,332	1	2	10	20	102,4	204,8
100	0,664	10	20	1000	2000	$1{,}26765 \cdot 10^{29}$	$2{,}5353 \cdot 10^{29}$

***Figura 9**:* Tiempo que demora un algoritmo de f(n) pasos, si realiza 1 paso en 0.1 segundos

Fuente: Grafos: fundamentos y Algoritmos.

Los autores definen la complejidad de un algoritmo y principalmente por qué aspectos está compuesto, el cual es necesario para poder determinar la complejidad del algoritmo de Dijkstra, que a su vez determinan la eficiencia de este mismo.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

2.3.3.5. Algoritmo de Búsqueda en Amplitud para caminos de costo mínimo.

Meza & Ortega (2006), explican que el algoritmo de búsqueda en amplitud busca el grafo por etapas, es decir por cada etapa se consideran todos los caminos con la misma longitud (k), en la siguiente etapa se consideran todos los caminos con la longitud $(\mathrm{k}+1)$ y así sucesivamente, además consideran que un grafo $G=(\mathrm{V},\mathrm{E})$, se representa en grupos si V puede ser fraccionado en conjuntos $V_i, \mathrm{i}=1 \ldots \mathrm{n}$, de manera que si $\mathrm{v} \epsilon V_i$ entonces los sucesores de V están en V_{i+1} se puede ver en la figura 10.

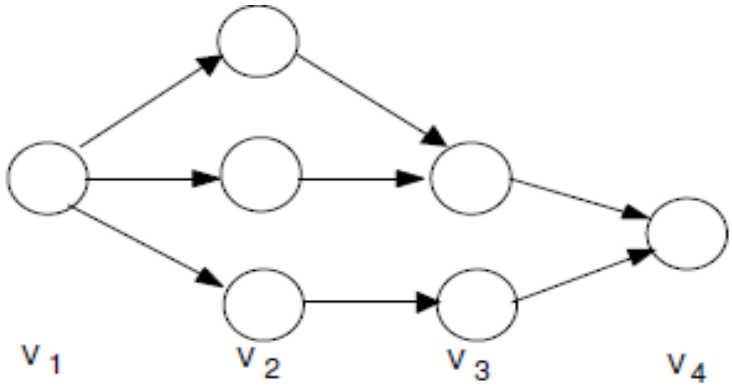


***Figura 10**:* Grafo divido en grupos

Fuente: Grafos y Algoritmos

Los autores exponen que el algoritmo de Búsqueda en Amplitud aplicado en grafos divididos en grupos es una técnica denominada "Programación Dinámica Progresiva", las características que debe seguir este algoritmo aplicado al grafo son las siguientes.

Se cierran todos los vértices de un grupo antes de comenzar a cerrar los del grupo siguiente, al terminar de cerrar los de un grupo, están abiertos todos los vértices del siguiente grupo. Los cerrados no son eliminados, estos garantizan que el número de iteraciones del algoritmo es igual al número de vértices alcanzables desde$S$, se podría implementar el algoritmo de manera que en cada iteración el número de operaciones realizables sea de orden $\Delta^{+}(\mathrm{G})$ $(representa\ el\ grado\ maximo\ exterior\ de\ G)$. Se

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis
Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

tendrá en memoria solo los elementos del grupo que se está expandiendo y el grupo resultante de la expansión.

La presente definición, explica el funcionamiento del algoritmo de búsqueda en amplitud; consiste en agrupar los vértices y utilizar poca memoria ya que solo se tiene en memoria el grupo expandido en ese momento sin embargo no toma en cuenta los costos y buscan indiscriminadamente todos los caminos o aristas que parten de "s".

2.3.3.6. Algoritmo de Dijkstra

Coto (2003), define al algoritmo de Dijkstra como como aquel conjunto de pasos que resuelve el problema de encontrar la ruta más corta a partir de un nodo de origen, en grafos ponderados y que no tengan pesos negativos $W(vi, vj) \geq 0$, el proceso en el que el Algoritmo de Dijkstra opera para encontrar la ruta más corta desde un nodo de inicio es el siguiente:

a. Debe existir un conjunto S de nodos cuya distancia más corta desde el nodo origen ya es conocida.
b. El conjunto S inicialmente contiene solo el nodo de origen.
c. Cada vez que va iterando el algoritmo, se va agregando el nodo iterado v al conjunto S, cuya distancia desde del nodo origen es la más corta posible (es posible que encontrar el camino más corto entre un par de nodos y estos deben pasar por la trayectoria de los nodos del conjunto S).
d. En cada iteración el algoritmo de Dijkstra utiliza un arreglo [D] el cual servirá para almacenar la longitud del camino más corto de cada nodo v, otra manera de expresar el arreglo [D], es que almacena la suma de los pesos de las aristas del camino más corto.
e. Cuando el conjunto S incluye todos los nodos iterados, el arreglo [D] almacenara la distancia más corta del origen a cada vértice de destino.
f. También se utiliza un arreglo [P] que almacenara la trayectoria más corta, es decir todos los nodos que se deben recorrer.
g. Repetir paso a.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Código 1.0 Pseudocodigo del Algoritmo de Dijkstra.

```
1. Data: G: a grafo dirigido ponderado, w: todos los pesos
   del borde,
2. s: fuente o estado inicial.
3. Result: El camino más corto de una sola fuente.
4. for each vertex v ∈ V [G]:
5.     d[v] ← ∞
6.     π[v] ← NIL
7.
8. d[s] ← 0
9. S ← Ø
10.      Q ← V [G]
11.
12.      while Q ≠ Ø:
13.          u ← EXTRACT-MIN(Q)
14.          S ← S ∪ [u]
15.
16.          for each vertex v ∈ Adj[u]:
17.              if d[u] > d[u] + w(u,v):
18.                 d[u] ← d[u] + w(u,v)
19.                 π[u] ← u
```

Una vez terminado el proceso de iteración del algoritmo de Dijkstra, cada nodo v del arreglo P tiene su nodo predecesor, y así sucesivamente hasta construir la ruta más corta hasta llegar al nodo de inicio, ver figura 11.

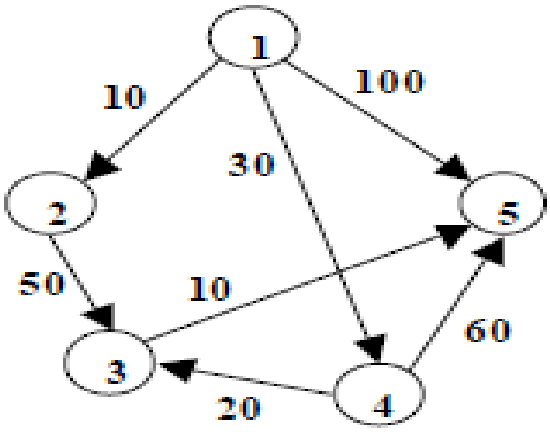


Figura 11: Grafo ponderado y dirigido

Fuente: Algoritmos Básicos de Grafos.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Por ejemplo si se desea llegar desde el nodo 1 (nodo inicial) hasta el nodo 5 (nodo final) el arreglo [P] debe tener el siguiente recorrido empezando desde los predecesores en orden inverso, es decir comenzar desde el 5; 1, 4, 3, 5.

Esta explicación define la manera en que el algoritmo de Dijkstra construye la ruta más corta a partir de un nodo de inicio, esto servirá para realizar el cálculo para hallar la ruta más corta respecto a los lugares de concentración de desechos sólidos establecidos por los usuarios, tomando en cuenta la posición de los camiones recolectores de desechos sólidos.

2.3.3.7. Algoritmo de Bellman-Ford.

Meza & Ortega (2006), definen el algoritmo de Bellman-Ford como aquel que permite calcular todos los caminos mínimos que parten de un vértice s, además se asume que el vértice s es el vértice inicial y que el grafo no tiene circuitos, esta técnica está basada en la formula recursiva representada de la siguiente manera. $C(v)$ Representa el costo de un camino mínimo desde s hasta v:

$$C(v) = min\left\{C(w) + \frac{c(w,v)}{w} es\ predecesor\ de\ v\right\}.$$

Nótese que si el grafo tiene circuitos entonces la formula recursiva anterior no funcionaria, ver figura 12.

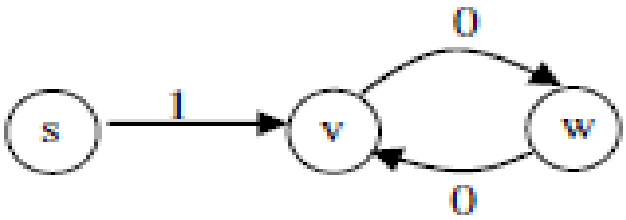


Figura 12: Grafo con circuitos

Fuente: Grafos y Algoritmos.

La presente explicación define el funcionamiento del Algoritmo de Bellman-Ford, definiendo las restricciones que tiene y la fórmula matemática para poder encontrar el costo mínimo desde un par de vértices (s, v).

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

2.3.4. Sistema

2.3.4.1.1. Sistemas de Coordenadas

Pérez Navarro (2011), define un sistema de coordenadas como el posicionamiento de un objeto o un fenómeno, los sistema de coordenadas considerados para representar la superficie terrestre son dos, sistema de coordenadas geográficas y sistema de coordenadas cartesiano, explicados a continuación.

Las coordenadas geográficas son aquellas que se utilizan para determinar una posición sobre la superficie terrestre utilizando una superficie esférica tridimensional determinada por 2 ángulos medidos tales como la latitud $(\propto)$ y longitud (β), en la figura 13 se representa cualquier posición sobre la superficie terrestre.

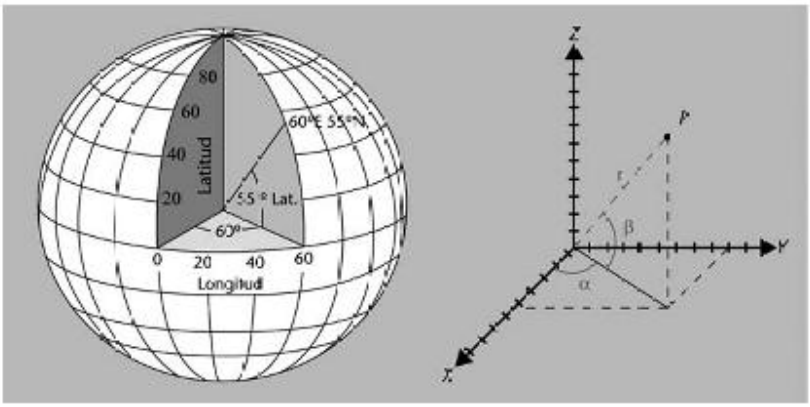


Figura 13: Valores de longitud y latitud de un punto sobre la esfera

Fuente: Introducción a los Sistemas de Información Geográfica y Geo-telemática.

Las coordenadas cartesianas son aquella que se utilizan para determinar la posición sobre la superficie terrestre utilizando un espacio tridimensional a través de las coordenadas (x, y, z), el eje (z), se desplaza por el centro de la tierra y los polos, el eje (x), se desplaza por el centro de la tierra y por el meridiano de Greenwich y por último el eje (y), forma un Angulo de 90% con el eje $(x \,\&\&\, z)$, estos son representados en la figura 14.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

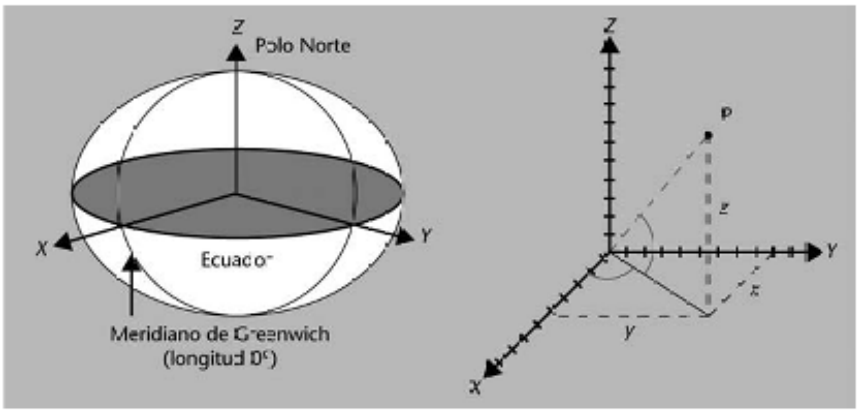


Figura 14: Ejes X, Y, Z del sistema de coordenadas cartesianas

Fuente: Introducción a los Sistemas de Información Geográfica y Geo-telemática.

La presente definición expuesta por los autores, explica el posicionamiento de un punto sobre la superficie terrestre, principalmente a través de dos sistemas de coordenadas como la geográfica y cartesiana.

2.4. Definición de términos básicos.

2.4.1. Desechos sólidos.

Para Gary & J.Glynn (1999) definen los desechos sólidos “como aquellos desperdicios que no son transportados por agua y que han sido rechazados porque no se van a utilizar” (p. 568).

Say (2005), explica que los desechos sólidos se definen como desperdicios sólidos y semisólidos que carecen de valor para una persona.

Por otro lado Organización de las Naciones Unidas para el Desarrollo Industrial (2007) describe a desechos sólido como aquella masa heterogénea, formada por algún producto generado de una actividad directa del hombre u organismos vivo.

2.4.2. Grafos.

Coto (2003) “un grafo es un conjunto de nodos unidos por un conjunto de líneas o flechas” (p. 4).

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Ruohonen (2013) explica que un grafo está formado por (V, E), donde V representa los vértices y E representa los bordes los cuales deben estar conectados a los vértices, los vértices y los bordes pueden ser etiquetados tanto por números como por letras.

Caicedo, Wagner &Méndez (2010) "Un grafo G consta de un conjunto de vértices o nodos V y un conjunto de arcos A, cada uno de los cuales une un vértice con otro. G = (V, A)=(N, A)" (p. 1).

2.4.3. Algoritmo.

Joyanes, Rodriguez, & Matilde (2000), definen un algoritmo como un conjunto de pasos ordenados, no ambiguo (es decir cada acción a desarrollar dentro del algoritmo es única), que conducen a la solución del problema planteado y expresado en lenguaje natural (entendible por el ser humano).

Así mismo Rodríguez, Santamaría, Rabasa & Martínez (2003) Mencionan que algoritmo es un método que da solución a un problema, siguiendo una secuencia ordenada de operaciones.

Para la Real academia española (2014) el algoritmo es "un conjunto ordenado y finito de operaciones que permite hallar la solución de un problema".

2.4.4. Algoritmo Dijkstra

Algoritmo Dijkstra para Coto (2003), resuelve problemas de encontrar caminos más cortos a partir de un origen, en grafos que no tengan pesos negativos.

Ruohonen (2013) menciona que el algoritmo de Dijkstra también llamado el algoritmo de los caminos mínimos, es conformado por vértices y aristas, donde las aristas son de pesos no negativos y el peso de una trayectoria o camino es la suma de los pesos de la trayectoria recorrida

Además Alonso (2008) señala que "algoritmo de dijkstra está diseñado para encontrar las rutas más cortas entre el nodo de origen y cada uno de los nodos de la red. Este algoritmo es de tipo "greedy" porque en cada iteración elige la mejor opción de las posibles con la esperanza de encontrar así la mejor solución global." (p. 59)

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

CAPITULO III: MARCO METODOLÓGICO

3.1. Tipo y diseño de investigación

3.1.1. Tipo de investigación:

La investigación se basó en la metodología cuantitativa; porque los indicadores están representados a través de cantidades numéricas, y está a su vez es continua debido a que se obtiene por medición o comparación una unidad o patrón de medida.

De acuerdo Hernández, Fernández & Baptista (2014) la investigación cuantitativa los datos son producto de mediciones y se representan mediante cantidades y se deben analizar con métodos estadísticos.

3.2. Población y muestra

3.2.1. Población

Ruohonen (2013) menciona que en la actualidad se consideran diez algoritmos de busqueda que trabajan sobre Grafos dirigidos, referenciado en el Anexo 7.

3.2.2. Muestra

La muestra de la investigación es elegida por conveniencia, donde el Algoritmo seleccionado es el Algoritmo de Dijkstra, referenciado en el Anexo 7.

3.3. Hipótesis

Utilizando el algoritmo de Dijkstra en un planificador de rutas se podrá determinar la trayectoria óptima para la recolección de desechos sólidos en el distrito de Chiclayo.

3.4. Variables

3.4.1. Variable Independiente.

El Algoritmo de Dijkstra.

3.4.2. Variable Dependiente.

Planificador de Rutas.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

3.5. Operacionalización.

Variable independiente	Dimensiones	Indicadores	Unidad de medida.	Formula	Descripción.
El Algoritmo de Dijkstra.	Tiempo de Ejecución.	Tiempo.	Segundos.	$TE = TF - TI$	TE = Tiempo de ejecución. TI = tiempo de inicio. TF = tiempo final.
Variable dependiente	**Dimensiones**	**Indicadores**	**Unidad de medida**	**Formula**	**Descripción**
Planificador de rutas.	Costo	Distancia	Metros	$W_i = L_i$	W = Arco de la ruta Li = Longitud de la ruta i.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

3.6. Abordaje metodológico, técnicas e instrumentos de recolección de datos.

3.6.1. Abordaje metodológico

El abordaje metodológico fue del subtipo cuasi – experimental, como lo indica Hernández, Fernández & Baptista (2014) que este tipo de diseño manipula la variable independiente para observar su efecto sobre una o más variables dependientes; teniendo en cuenta lo antes citado, la investigación estableció el efecto de el algoritmo de Dijkstra (variable independiente), sobre la ruta óptima para la recolección de desechos sólidos (variable dependiente).

3.6.2. Técnicas de recolección

Se eligió aplicar como técnica la observación científica debido que observaremos la realidad exterior con el propósito de obtener los datos que previamente, han sido definidos como de interés para la investigación.

Se eligió aplicar como técnica el análisis documental debido que este tiene fines de orientación científica e informativa el cual es un proceso transversal realizado durante la investigación como son las tesis, artículos y libros leídos.

3.6.3. Instrumentos de recolección de datos

Guía de Observación: Consiste en obtener un método de registro que no le dejen a su sola memoria el trabajo de retener lo importante, pero no es indispensable, referenciados en el anexo 01.

3.7. Procedimiento para la recolección de datos

El procedimiento para la recolección de datos se tomará en cuenta la simulación de los puntos establecidos por los usuarios, detallado a continuación.

a) En una primera fase se debe obtener principalmente las coordenadas de las intersecciones de las calles, mediante el proveedor de aplicaciones de mapas “Google Maps” para posteriormente poder realizar la conversión del mismo en un grafo a través de una matriz de adyacencia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

b) En una segunda fase se procede a aplicar el Algoritmo de Dijkstra sobre el grafo definido previamente para que este construya la ruta más corta.
c) En una tercera fase se procede a realizar la conversión del grafo con la ruta más corta en el mapa resultante.
d) En una cuarta fase los datos de interés obtenidos del planificador de rutas se almacenaran en la guía de observación.
e) En una quinta fase los datos guardados en la guía de observación, permitirán realizar la tabulación de los resultados.

3.8. Principios éticos

En la investigación se utilizó el principio ético de honestidad o integridad, ya que no se manipuló la información, mostrando los resultados tal cual se obtuvieron. Además se citó las referencias bibliográficas apropiadamente para no incurrir en plagio. Dik (2006), Fitzpatrick (2004) y Guterbock (2008) citado por Hernández, Fernández & Baptista (2014) señalan que este principio ético implica adherirse a los datos, resultados y hechos; no se debe manipular la información. A demás los colaboradores o deben incurrir en falsificación de recolección, codificación y análisis.

3.9. Criterios de rigor científico

Las características principales que deben tener los instrumentos para recolectar datos son:

Fiabilidad: La investigación cumplió con las expectativas expresadas en su contenido y su implementación con el estándar y políticas para su desarrollo.

Validez: Los datos obtenidos en el estudio fueron evaluados y analizados por los ingenieros especialistas en el tema para decretar su autenticidad.

Consistencia: La investigación representó material consistente y certificado por la comunidad científica.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

CAPÍTULO IV: ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS

4.1. Resultados en tablas y gráficos.

El algoritmo de Dijkstra se ejecutó en 7 casos diferentes, con diferentes puntos de acción bajo un mismo escenario; para construir la ruta más corta, se le proporcionó una matriz de adyacencia; además se utilizaron dos ordenadores diferentes cuyas especificaciones son las que se indican a continuación.

Intel(R) Core(TM) i7-4700MQ CPU @ 2.40GHz, 12.0 GB RAM, Sistema operativo Windows 8.1.

Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz, 6.00 GB RAM, Sistema operativo Windows 10.

Para la implementación del algoritmo de Dijkstra se utilizó el lenguaje de Programación PHP y para almacenar la matriz de adyacencia se utilizó el SGDB MySql.

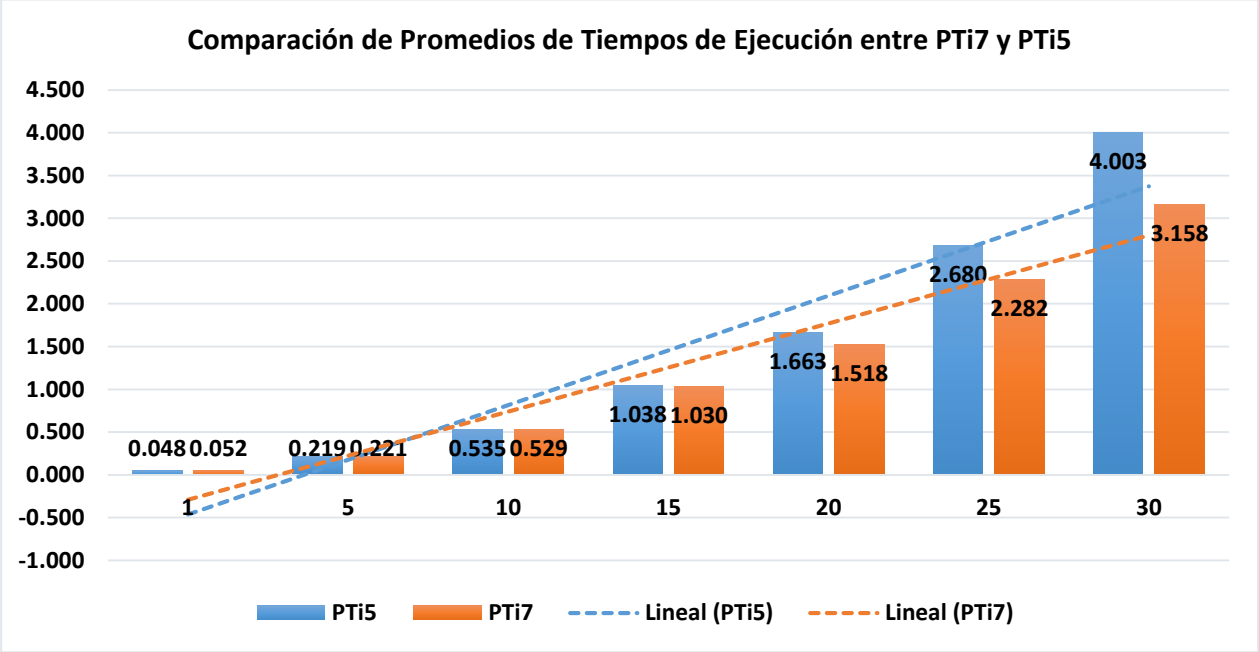
En el anexo 2, se puede observar que se extrajo una porción del mapa del distrito de Chiclayo equivalente a 1087 km2 de distancia, sobre el cual se aplicará el algoritmo de Dijkstra.

En resumen; lo analizado de los datos obtenidos del algoritmo de Dijkstra se muestra a continuación:

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Grafico 1: Comparación de Promedios de Tiempos de Ejecución entre PTi7 y PTi3 en cada caso.



Fuente: Elaboración propia**.**

Del gráfico anterior se concluye lo siguiente:

Para el computador con procesador Core i7 se aprecia que el algoritmo de Dijkstra para 1 foco infeccioso tiene un promedio de tiempo de ejecución de 0.052 s, para 5 focos infecciosos de 0.221 s, para 10 focos infecciosos de 0.529 s, para 15 focos infecciosos de 1.030 s, para 20 focos infecciosos de 1.518 s, para 25 focos infecciosos de 2.282 s y para 30 focos infecciosos se tiene un promedio de tiempo de ejecución de 3.158 s.

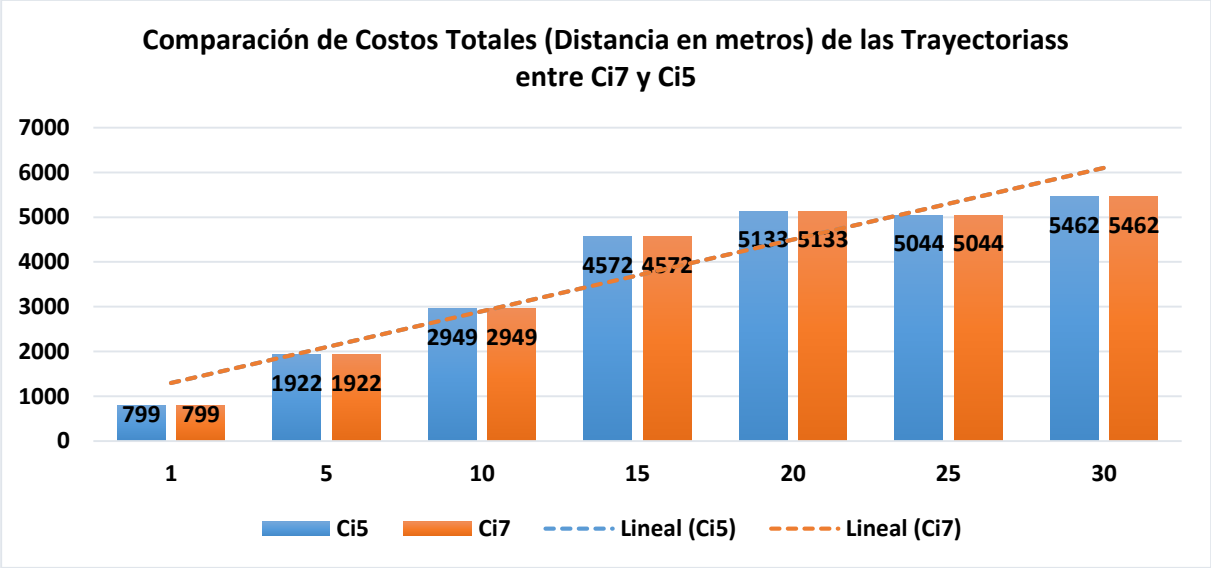
Para el computador con procesador Core i5 se aprecia que el algoritmo de Dijkstra para 1 foco infeccioso tiene un promedio de tiempo de ejecución de 0.048 s, para 5 focos infecciosos de 0.219 s, para 10 focos infecciosos de 0.535 s, para 15 focos infecciosos de 1.518 s, para 20 focos infecciosos de 1.663 s, para 25 focos infecciosos de 2.680 s y para 30 focos infecciosos se tiene un promedio de tiempo de ejecución de 4.003 s.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Se puede apreciar que de 1 a 15 focos infecciosos, los tiempos promedios de ejecución en un computador con procesador Corei7 son similares a los respectivos tiempos promedios de ejecución con un computador con procesador Corei5, observando que para 20, 25 y 30 focos infecciosos, en un computador con procesador Corei7 el Algoritmo de Dijkstra tiene un menor tiempo promedio de ejecución.

Grafico 2: Comparación de Costos Totales (Distancia en metros) de las Trayectorias entre Ci7 y Ci5



Fuente: Elaboración propia**.**

Del gráfico anterior se concluye lo siguiente:

Para el computador con procesador Core i7 se aprecia que el algoritmo de Dijkstra para 1 foco infeccioso tiene un costo total de 799 m; para 5 focos infecciosos de 1922 m; para 10 focos infecciosos de 2949 m; para 15 focos infecciosos de 4572 m; para 20 focos infecciosos de 5133 m; para 25 focos infecciosos de 5044 m y para 30 focos infecciosos se tiene un Costo Total de 5462 m.

Para el computador con procesador Core i5 se aprecia que el algoritmo de Dijkstra para 1 foco infeccioso tiene un costo total de 799 m; para 5 focos infecciosos de 1922 m; para 10 focos infecciosos de 2949 m; para 15 focos infecciosos de 4572 m; para 20 focos infecciosos de 5133 m; para 25 focos infecciosos de 5044 m y para 30 focos infecciosos se tiene un Costo Total de 5462 m.

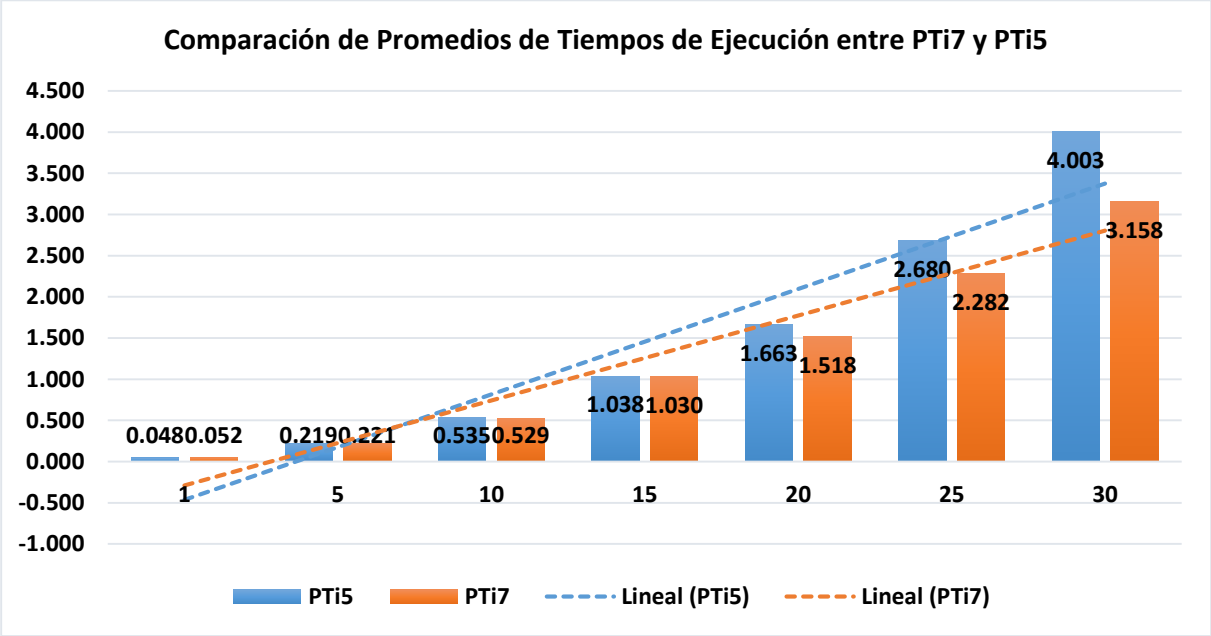
Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

4.2. Discusión de resultados

Comparación de los resultados del Algoritmo de Dijkstra sobre el planificador de rutas entre 2 computadores.

Grafico 3: Comparación de Promedios de Tiempos de Ejecución entre PTi7 y PTi3 en cada caso.



Fuente: Elaboración propia**.**

Del gráfico anterior se concluye lo siguiente:

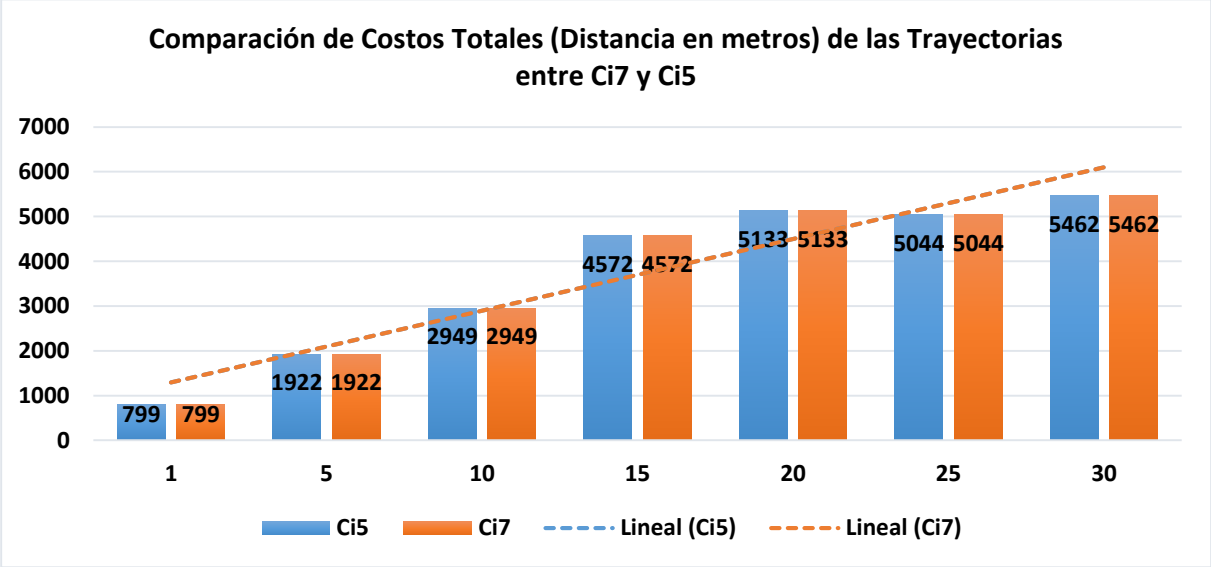
A través de la ejecución del algoritmo de Dijkstra sobre el planificador de rutas se puede observar que después de elegir los puntos de acción por donde se construirá la ruta más corta, es necesario buscar en el listado de vértices y actualizar el costo del vértice más cercano, es decir buscar los vértices en el grafo mediante la matriz de adyacencia. Luego, el siguiente vértice más cercano debería encontrarse buscando nuevamente en el listado de vértices y actualizar su costo. En el proceso de búsqueda del vértice adyacente más cercano, el algoritmo busca los puntos no adyacentes al mismo tiempo, lo que reduce el rendimiento del algoritmo es por ello que a más focos

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

infecciosos, realiza más iteraciones para buscar el vértice más cercano y por lo tanto consume más recursos computacionales he ahí donde el procesador del computador es necesario para obtener un mejor tiempo de ejecución del Algoritmo de Dijkstra, a partir de 20 focos infecciosos en adelante.

Grafico 4: Comparación de Costos Totales (Distancia en metros) de las Trayectorias entre Ci7 y Ci5.



Fuente: Elaboración propia**.**

Del gráfico anterior se concluye lo siguiente:

A través de la ejecución del algoritmo de Dijkstra sobre el planificador de rutas, se puede observar que después de elegir los puntos de acción por donde se construirá la ruta más corta, el costo de total de la trayectoria en cada caso es invariable por lo que el Costo de la trayectoria es indiferente del procesador en el que se ejecute retornando siempre el mismo costo, variando solo el tiempo de ejecución.

Esto da a entender que el algoritmo de Dijkstra para cada caso en diferentes ejecuciones siempre retorna el mismo resultado por lo que hace confiable al planeador de rutas para el recojo de desechos sólidos.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

CAPITULO V: PROPUESTA DE INVESTIGACIÓN

5.1. Generalidades de la propuesta.

La presente investigación se desarrolló un planificador de rutas para el recojo de desechos sólidos utilizando el algoritmo de Dijkstra. La investigación inició desde la preparación del conjunto de datos, estos consisten en convertir los puntos georreferenciados de recolección de desechos sólidos (obtenidos del servidor de aplicaciones de mapas Google Maps) en un grafo dirigido ponderado a través de una matriz de adyacencia. Posteriormente se aplicó el algoritmo de Dijkstra sobre el grafo (tomando en cuenta el costo de los arcos que conectan los vértices y la dirección de las calles mediante una matriz de Adyacencia), para una mejor elección se realizó una revisión de la literatura de distintas investigaciones similares realizadas con anterioridad. A continuación se realizó pruebas sobre el algoritmo de Dijkstra, y se verificaron los resultados obtenidos.

A continuación se muestra el flujograma de la presente investigación, ver figura 15.

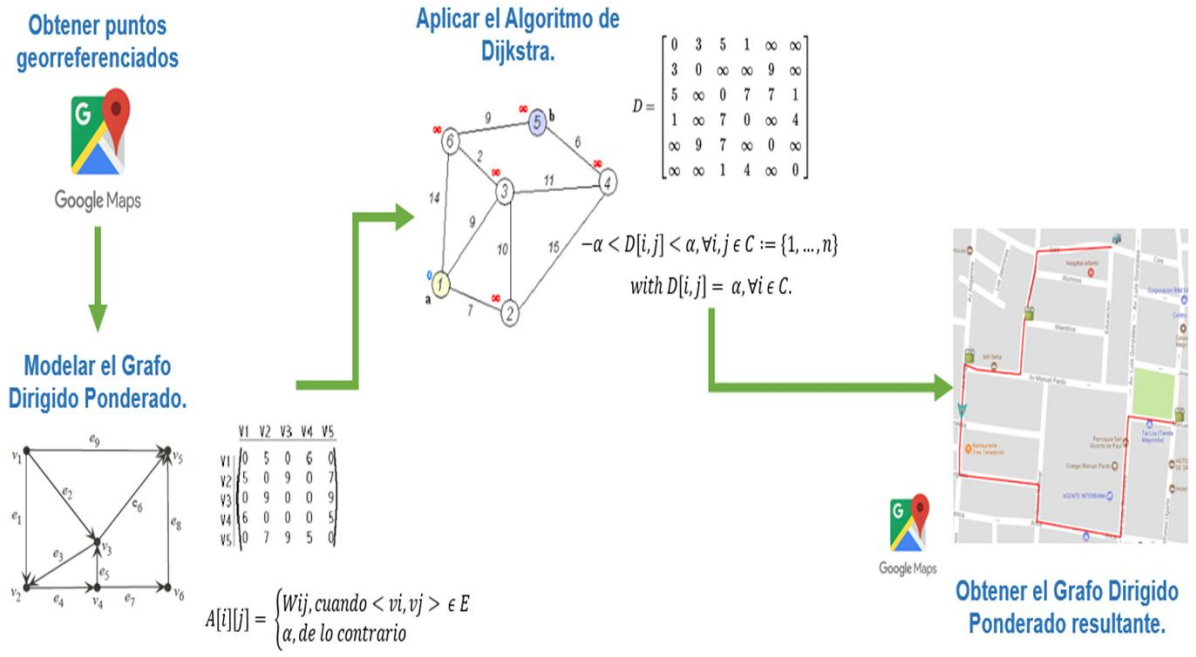


Figura 15: Flujo de la realización de la investigación

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

5.2. Preparar el conjunto de datos.

Para realizar el presente objetivo, se preparó el conjunto de datos, el conjunto de datos consiste en obtener los puntos georreferenciado mediante el servidor de aplicaciones de mapas "Google Maps" en el distrito de Chiclayo, para luego convertir dichos puntos en un Grafo ponderado. Como expresa Ruohonen (2013) un grafo está formado por (V, E), donde V representa el conjunto de vértices y E representa el conjunto de bordes los cuales deben estar conectados a los vértices.

5.2.1. Obtener puntos georreferenciados.

Los puntos georreferenciados a tomar en cuenta, se obtuvieron al azar, simulando un entorno donde los ciudadanos realicen denuncias, según la información obtenida del servidor de aplicaciones de mapas "Google Maps", se obtuvo un resumen de los datos de cada ruta simulada.

Tabla 2

Rutas simuladas en un entorno de denuncias de foco infeccioso.

Indicador del Mapa	Latitud && Longitud	Ubicación
A	-6.764400, -79.844146	Av. Manuel Pardo 682
B	-6.763466, -79.843228	Alumnos 133
C	-6.764718, -79.842076	Educación 109

Fuente: Elaboración propia

Los datos generados anteriormente en la tabla 2, fueron obtenidos de la representación del mapa del distrito de Chiclayo, haciendo uso del servidor de aplicaciones de mapas "Google Maps", como se indica en la figura 16.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

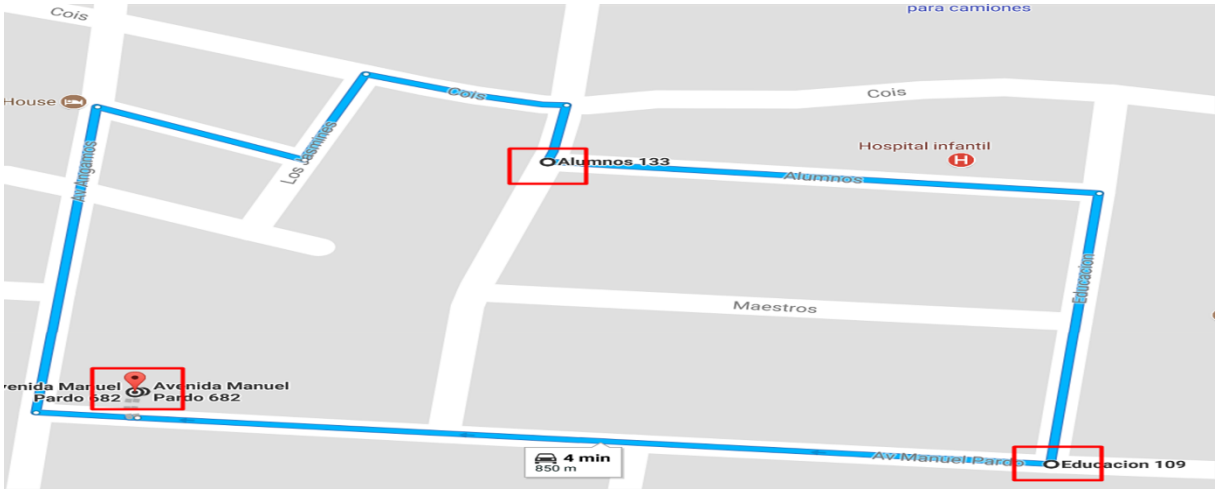


Figura 16: Rutas simuladas obtenidas del servidor de aplicaciones de mapas "Google Maps"

Fuente: Elaboración propia

En la figura 16 la trayectoria trazada no es la ruta óptima, sino es la trayectoria generada automáticamente por "Google Maps" al ubicar varios puntos sobre el mapa.

5.2.2. Convertir puntos georreferenciados a Grafos.

5.2.2.1. Poner en marcha el problema.

El vehículo recolector de desechos sólidos debe recorrer los puntos de focos infecciosos establecidos por el usuario, entonces ¿cómo elegir la ruta óptima para el camión recolector de desechos sólidos de modo que la longitud recorrida sea la más corta, tomando en cuenta la dirección de las calles y el punto de partida del camión?

5.2.2.2. Modelado del problema.

Desde el punto donde se encuentra el camión hacia los puntos de focos infecciosos se debe generar una trayectoria, ahora tanto el vehículo como los focos infecciosos de ahora en adelante se expresaran como vértices. Donde el vértice de partida el cual es el vehículo se le denominara "$v0$", y el resto de vértices son los focos infecciosos, los cuales serán "$v1, v2 \ldots$", y las rutas entre ellos se les denominara aristas.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

5.2.2.3. Solución del problema.

La red de recolección de residuos sólidos se abstrae de la siguiente manera, "$v0$" el vértice donde se encuentra posicionado el vehículo, y los otros vértices son los focos infecciosos, el peso de la arista entre los vértices es la longitud de la ruta representada en metros (m), ver figura 17.

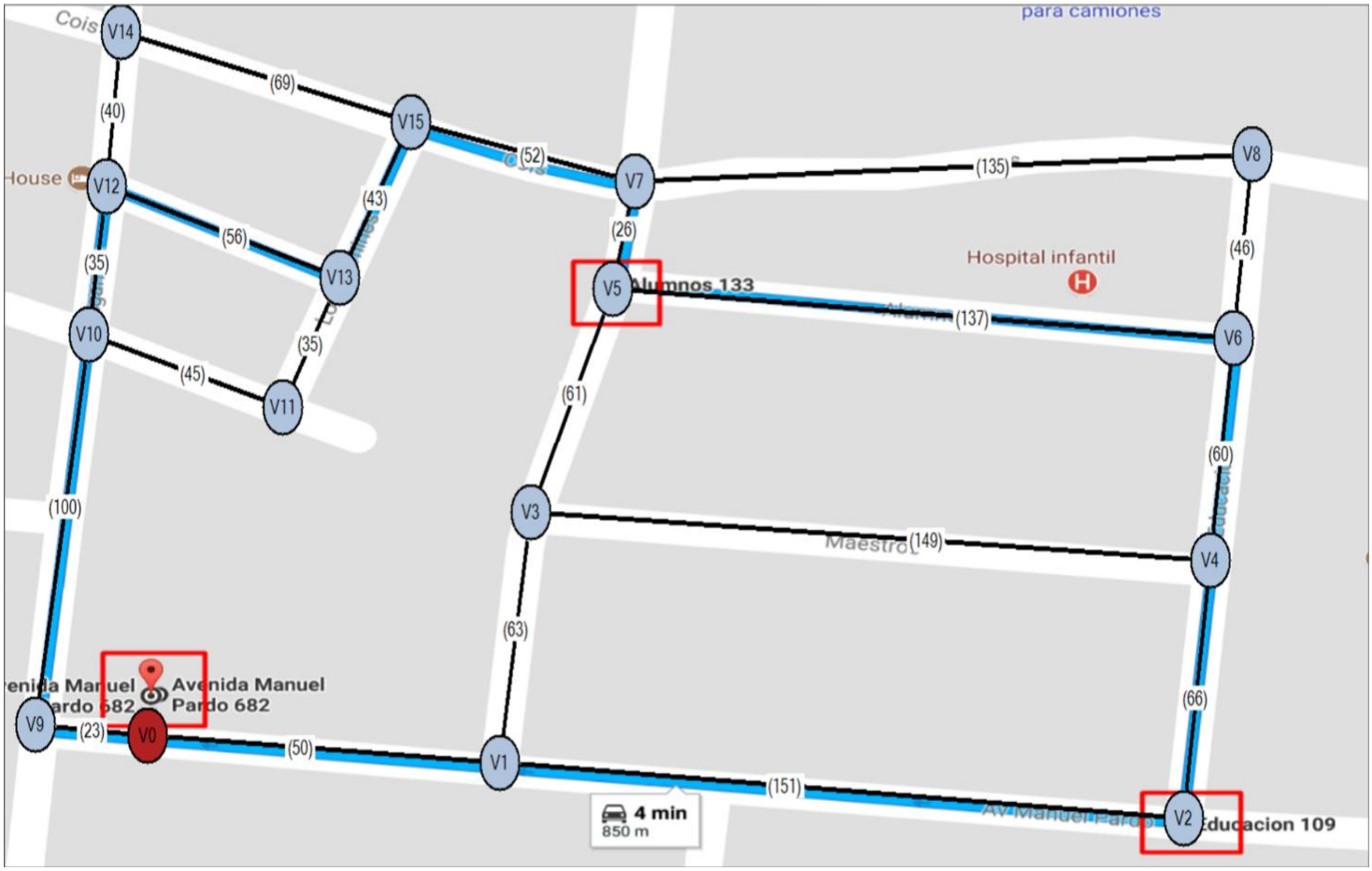


Figura 17: Grafo no dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

El grafo mostrado en la figura 17 es un grafo no dirigido, este se convirtió en un grafo dirigido debido que el recorrido es a través de un vehículo, es por ello que se tomó en cuenta la señalización de las calles representados en la aplicación de mapas de Google Maps, quedando de la siguiente manera en la figura 18.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

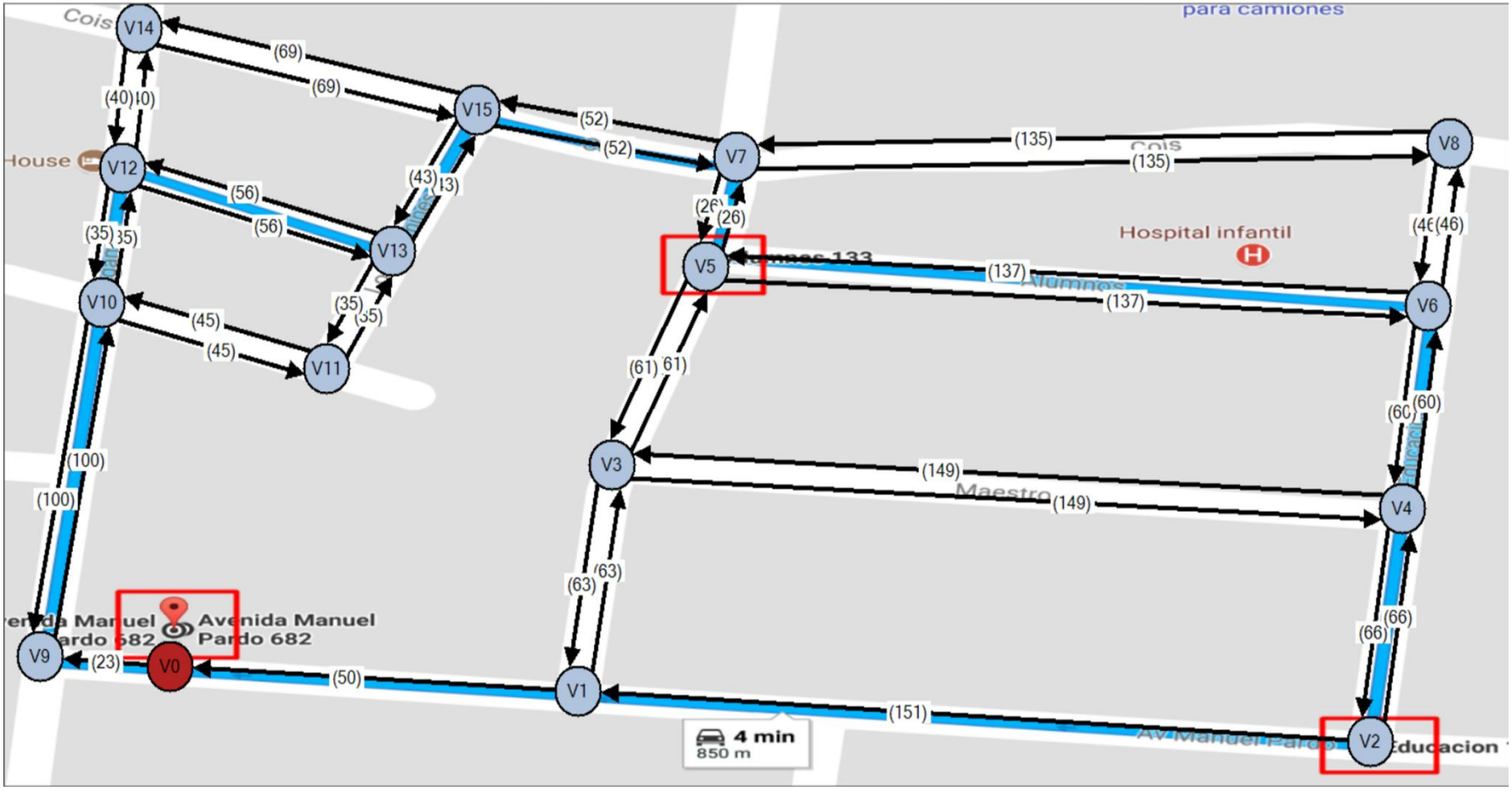


Figura 18: Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

El proceso para construir la matriz de adyacencia se puede simplificar abstrayendo la longitud entre un par de vértices que estén conectados y su vez sean accesibles, del grafo dirigido de la figura 18, ver tabla 3.

Tabla 3

Distancia entre intersecciones del grafo.

Número del arco (i, j)	Tamaño de la ruta (m) d_{ij}
$(v0, v9)$	23
$(v1, v0)$	50
$(v1, v3)$	63
$(v2, v1)$	151
$(v2, v4)$	66
$(v3, v1)$	63
$(v3, v4)$	149
$(v3, v5)$	61
$(v4, v2)$	66
$(v4, v3)$	149
$(v4, v6)$	60
$(v5, v3)$	61
$(v5, v6)$	137
$(v5, v7)$	26
$(v6, v4)$	60
$(v6, v5)$	137
$(v6, v8)$	46
$(v7, v5)$	26
$(v7, v8)$	135
$(v7, v15)$	52
$(v8, v6)$	46
$(v8, v7)$	135
$(v9, v10)$	100
$(v10, v9)$	100
$(v10, v11)$	45
$(v10, v12)$	35
$(v11, v10)$	45
$(v11, v13)$	35
$(v12, v10)$	35
$(v12, v13)$	56
$(v12, v14)$	40
$(v13, v11)$	35
$(v13, v12)$	56
$(v13, v15)$	43
$(v14, v12)$	40
$(v14, v15)$	69
$(v15, v13)$	43
$(v15, v14)$	69
$(v15, v7)$	52

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Ahora se procede a convertir dicho grafo en una matriz de adyacencia donde:

$$A[i][j] = \begin{cases} Wij, cuando < vi, vj > \ \epsilon\ E \\ \alpha, de\ lo\ contrario \end{cases}$$

Figura 19: Matriz de adyacencia de un Grafo Dirigido.

Fuente: Optimization of Logistics Route Base on Dijkstra.

La matriz de adyacencia deberá cumplir dos condiciones:

(1) Debe existir un arco W_{ij}, siempre y cuando $vi\ y\ vj$, estén conectados por una arista $< vi, vj > \epsilon\ E$, en donde E representa el conjunto de aristas de un grafo $G = (V, E)$. (2) De lo contrario, no se considera adyacente por lo tanto se interpretara como α.

Tabla 4

Matriz de adyacencia

	$v0$	$v1$	$v2$	$v3$	$v4$	$v5$	$v6$	$v7$	$v8$	$v9$	$v10$	$v11$	$v12$	$v13$	$v14$	$v15$
$v0$	α	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
$v1$	50	α	α	63	α	α	α	α	α	α	α	α	α	α	α	α
$v2$	α	151	α	α	66	α	α	α	α	α	α	α	α	α	α	α
$v3$	α	63	α	α	149	61	α	α	α	α	α	α	α	α	α	α
$v4$	α	α	66	149	α	α	60	α	α	α	α	α	α	α	α	α
$v5$	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
$v6$	α	α	α	α	60	137	α	α	46	α	α	α	α	α	α	α
$v7$	α	α	α	α	α	26	α	α	135	α	α	α	α	α	α	52
$v8$	α	α	α	α	α	α	46	135	α	α	α	α	α	α	α	α
$v9$	α	α	α	α	α	α	α	α	α	α	100	α	α	α	α	α
$v10$	α	α	α	α	α	α	α	α	α	100	α	45	35	α	α	α
$v11$	α	α	α	α	α	α	α	α	α	α	45	α	α	35	α	α
$v12$	α	α	α	α	α	α	α	α	α	α	35	α	α	56	40	α
$v13$	α	α	α	α	α	α	α	α	α	α	α	35	56	α	α	43
$v14$	α	α	α	α	α	α	α	α	α	α	α	α	40	56	α	69
$v15$	α	α	α	α	α	α	α	52	α	α	α	α	α	43	69	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

5.3. Aplicar el algoritmo de Dijkstra sobre el grafo.

La idea del algoritmo de Dijkstra como expresa Coto (2003), es resolver problemas relacionados a encontrar el camino más corto a partir de un vértice de origen, en grafos que no tengan pesos negativos. A partir de esta definición se empezó a demostrar la manera en que el algoritmo de Dijkstra realiza el cálculo para encontrar la ruta más corta entre un par de vértices.

Sin embargo existen algunas restricciones que el Algoritmo de Dijkstra debe verificar para poder aplicarse sobre un Grafo, ver figura 20.

$$-\alpha < D[i,j] < \alpha, \forall i,j \in C := \{1, \dots, n\}$$

$$with\ D[i,j] = \alpha, \forall i \in C.$$

***Figura 20**:* Restricción del Algoritmo de Dijkstra sobre un Grafo.

Fuente: Timbebase Dynamic Weight for Dijkstra Algorithm Implementation in Route Planning Software.

Se cumplen dos condiciones:

(1) La matriz de adyacencia debe ser mayor que menos infinito y menor que más infinito para todo i, j perteneciente al conjunto de vértices.

(2) Con la matriz de adyacencia igual que infinito, se interpreta como una indicación que no hay un enlace directo del nodo i al nodo j, para toda i perteneciente al conjunto C.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Lo primero que se realizo fue buscar los vértices adyacentes (vértices destino conectados directamente por un arista al vértice actual analizado) del vértice inicial $(v0)$, ver figura 21.

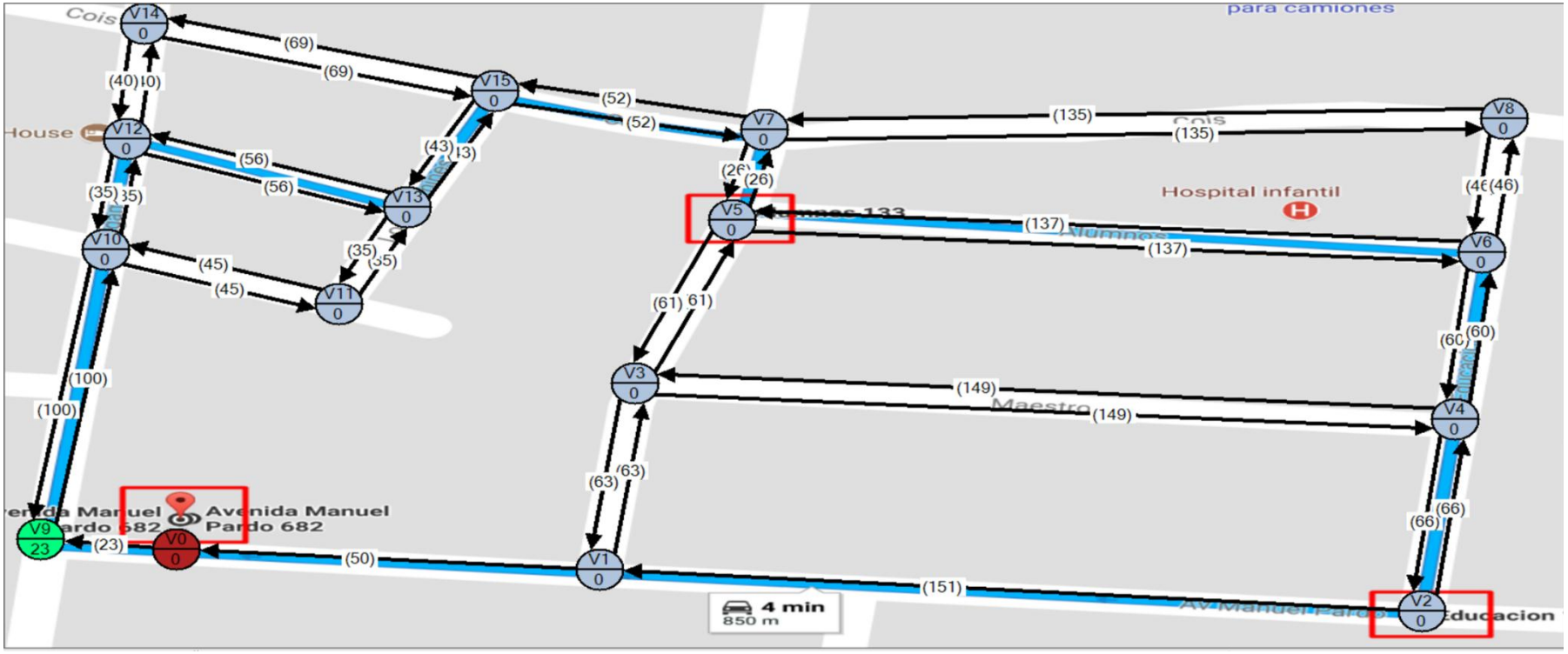


Figura 21: Obtener la ruta optima v0-v9 (temporal) del Grafo dirigido de la red de recolección de residuos sólidos.

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Al ser $(v9)$ el único vértice adyacente de $(v0)$ y por ende tener menor costo, se convierte de un vértice temporal (color verde claro) a un vértice permanente (color verde oscuro) y se actualiza el costo del vértice $(v9)$, ver figura 22.

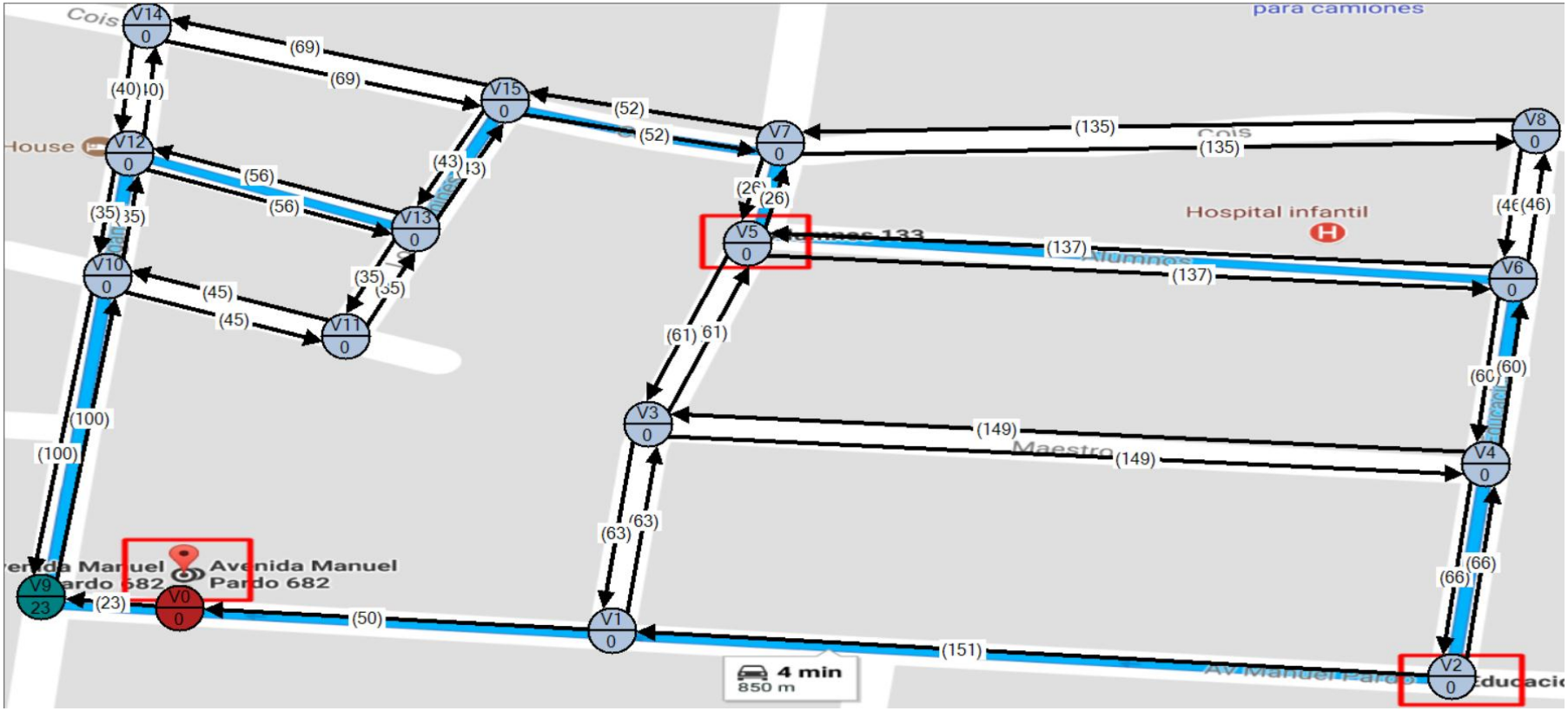


Figura 22: Obtener la ruta optima v0-v9 (permanente) del Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración de un par de vértices finaliza actualizando la matriz de adyacencia $(v0, v09)$.

Cabe recalcar que el color azul del vértice de la matriz de adyacencia representa el o los vértices óptimos de la trayectoria recorridos hasta ese momento y el color celeste representa una ruta recorrida u explorada.

Tabla 5

Matriz de adyacencia (v0, v09).

	$v0$	$v1$	$v2$	$v3$	$v4$	$v5$	$v6$	$v7$	$v8$	$v9$	$v10$	$v11$	$v12$	$v13$	$v14$	$v15$
$v0$	0	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
$v1$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v2$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v3$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v4$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v5$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v6$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v7$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v8$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v9$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v10$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v11$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v12$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v13$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v14$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v15$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Segundo paso, fue buscar los vértices adyacentes del vértice $(v09)$, debido que este es el nuevo vértice a analizar, ya que se encontró la ruta optima hasta ese tramo de la trayectoria, ver figura 23.

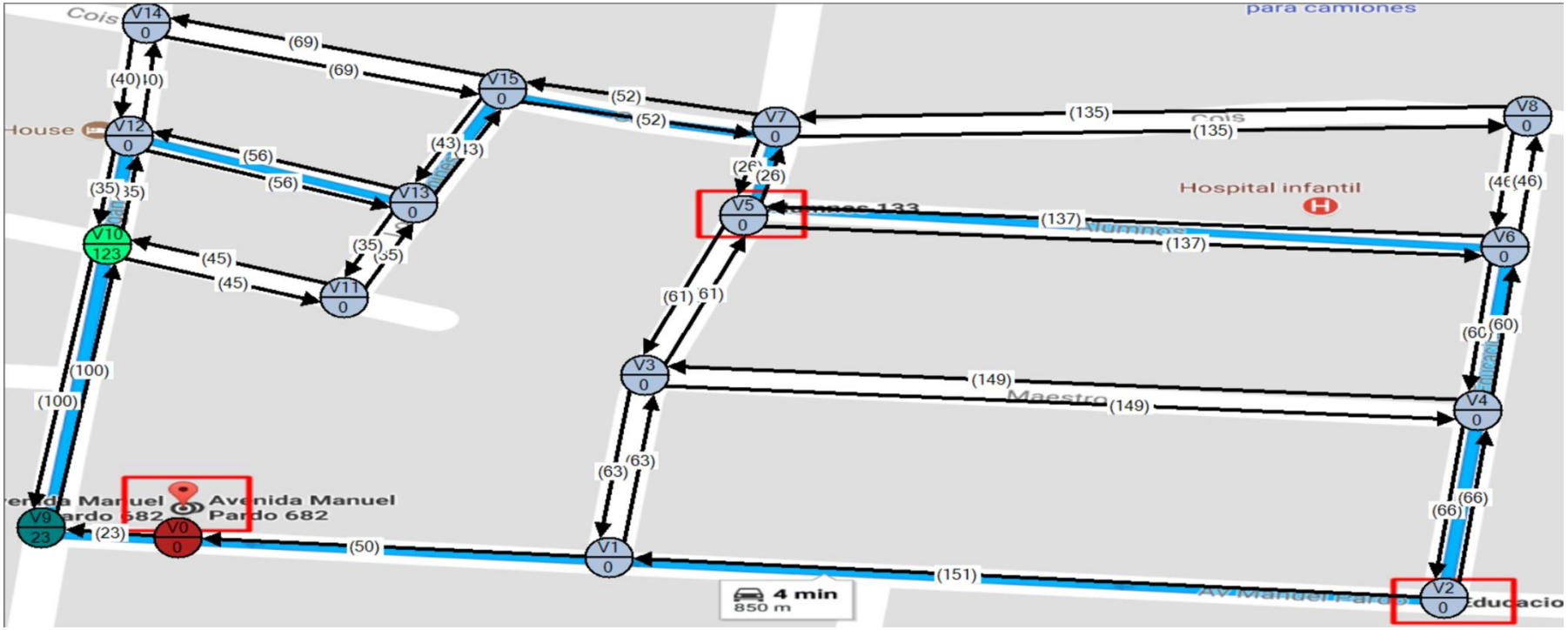


Figura 23: Obtener la ruta optima v9-v10 (temporal) del Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Al ser $(v10)$ el único vértice adyacente de $(v9)$ y por ende tener menor costo, se convierte de; un vértice temporal (color verde claro) a un vértice permanente (color verde oscuro) y se actualiza el costo del vértice $(v10)$, ver figura 24.

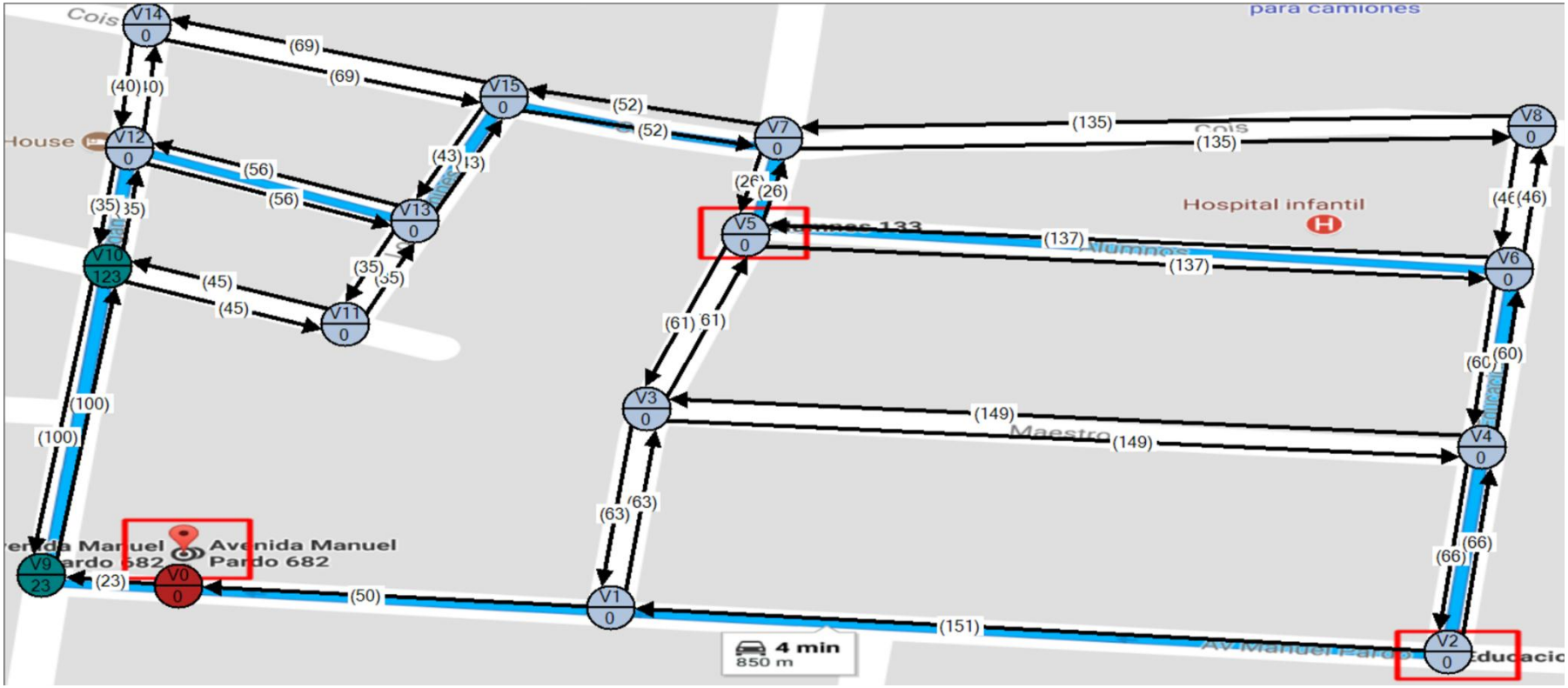


Figura 24: Obtener la ruta optima v9-v10 (permanente) del Grafo dirigido de la red de recolección de residuos solidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración de un par de vértices finaliza actualizando la matriz de adyacencia $(v09, v10)$.

Tabla 6

Matriz de adyacencia (v09, v10).

	$v0$	$v1$	$v2$	$v3$	$v4$	$v5$	$v6$	$v7$	$v8$	$v9$	$v10$	$v11$	$v12$	$v13$	$v14$	$v15$
$v0$	α	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
$v1$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v2$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v3$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v4$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v5$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v6$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v7$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v8$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v9$	α	α	α	α	α	α	α	α	α	α	123	α	α	α	α	α
$v10$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v11$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v12$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v13$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v14$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v15$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Tercer paso, el vértice $(v10)$, tiene dos posibles caminos $(v11, v12)$, se debe tomar el camino de menor costo (distancia), el costo actual del vértice $(v10)$, tiene una distancia acumulada de 123m, por lo tanto el camino que debe seguir es el vértice $(v12)$, debido que tiene menor distancia, ver figura 25.

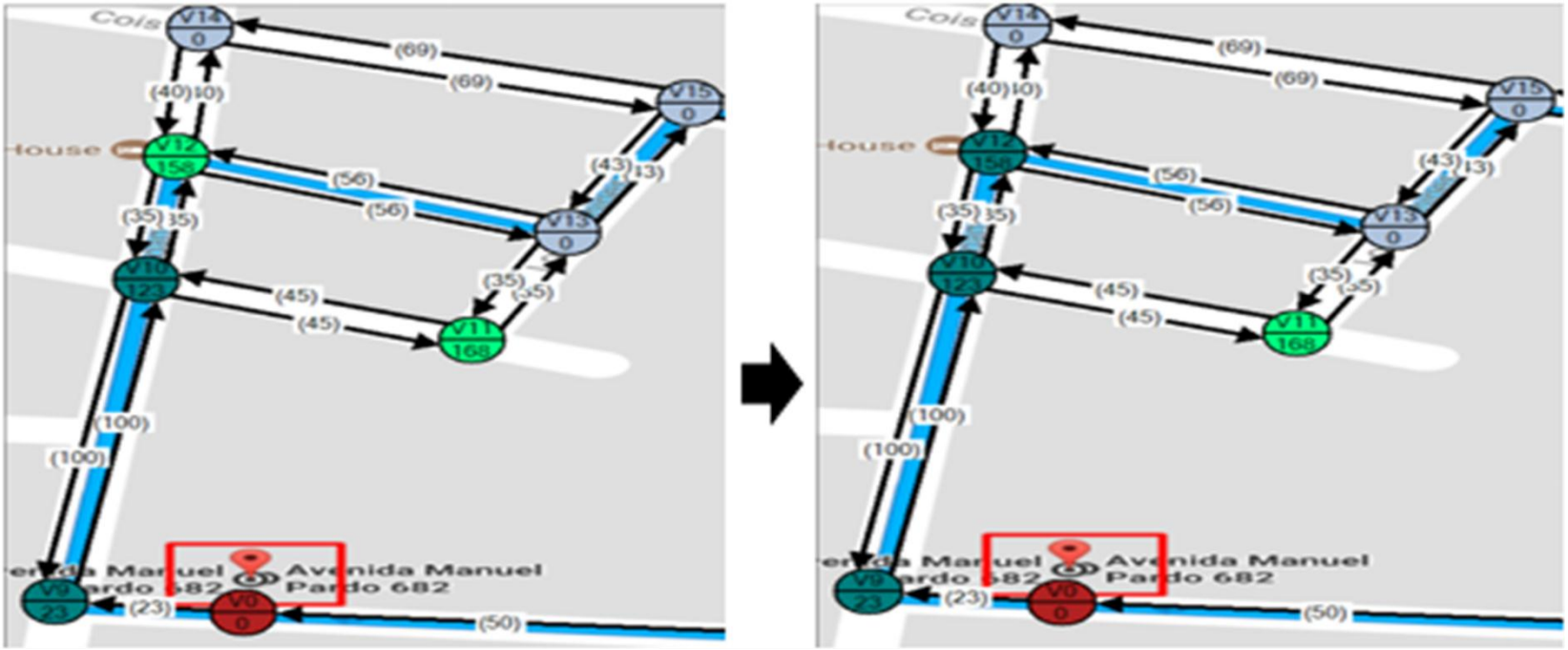


Figura 25: Obtener la ruta optima v10 hacia sus adyacentes, del Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración de un par de vértices finaliza actualizando la matriz de adyacencia $(v10, v11)$ y $(v10, v12)$.

Tabla 7

Matriz de adyacencia (v10, v11) y (v10, v12).

	$v0$	$v1$	$v2$	$v3$	$v4$	$v5$	$v6$	$v7$	$v8$	$v9$	$v10$	$v11$	$v12$	$v13$	$v14$	$v15$
$v0$	α	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
$v1$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v2$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v3$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v4$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v5$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v6$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v7$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v8$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v9$	α	α	α	α	α	α	α	α	α	α	123	α	α	α	α	α
$v10$	α	α	α	α	α	α	α	α	α	α	α	168	158	α	α	α
$v11$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v12$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v13$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v14$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v15$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Cuarto paso, el vértice $(v12)$, tiene dos posibles caminos $(v14, v13)$, se debe tomar el camino de menor costo (distancia), el costo actual del vértice $(v12)$, tiene una distancia acumulada de 158m, por lo tanto el camino que debería seguir es el vértice $(v14)$, debido que tiene la menor distancia, sin embargo al actualizar el costo de los vértices $(v14, v13)$, estos tienen un costo de $(198\ y\ 214)$ metros respectivamente, entonces ahora existen tres vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v11)$, con un costo de 168m, como se visualiza en la figura 26:

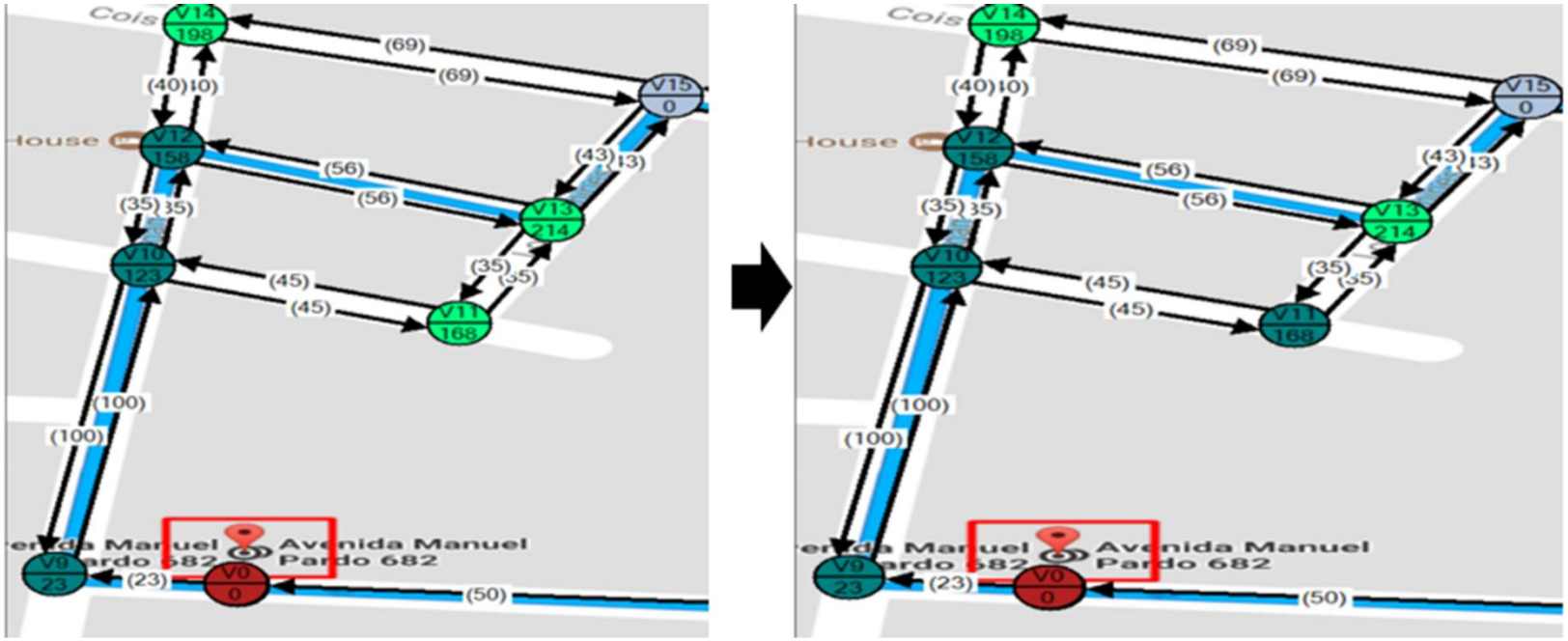


Figura 26: Obtener la ruta optima a partir del vértice v12, del Grafo dirigido de la red de recolección de residuos solidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $(v12, v14)$ y $(v12, v13)$ y $v11$ como permanente.

Tabla 8

Matriz de adyacencia (v12, v14), (v12, v13) y v11.

	$v0$	$v1$	$v2$	$v3$	$v4$	$v5$	$v6$	$v7$	$v8$	$v9$	$v10$	$v11$	$v12$	$v13$	$v14$	$v15$
$v0$	0	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
$v1$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v2$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v3$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v4$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v5$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v6$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v7$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v8$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v9$	α	α	α	α	α	α	α	α	α	α	123	α	α	α	α	α
$v10$	α	α	α	α	α	α	α	α	α	α	α	168	158	α	α	α
$v11$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v12$	α	α	α	α	α	α	α	α	α	α	α	α	α	214	198	α
$v13$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v14$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v15$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Quinto paso, al ser $(v13)$ el único vértice adyacente de $(v11)$ se realiza el cálculo y se verifica si el costo actual del vértice $(v13)$ es mayor que el calculado en la presente iteración, de ser cierto se actualiza el costo del vértice $(v13)$, de lo contrario se mantiene, en la presente iteración se actualiza el vértice $(v13)$ debido que el cálculo da como resultado un menor costo, luego de actualizar el costo del vértice $(v13)$, se evalúan los vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v14)$, con un costo de 198m, como se visualiza en la figura 27.

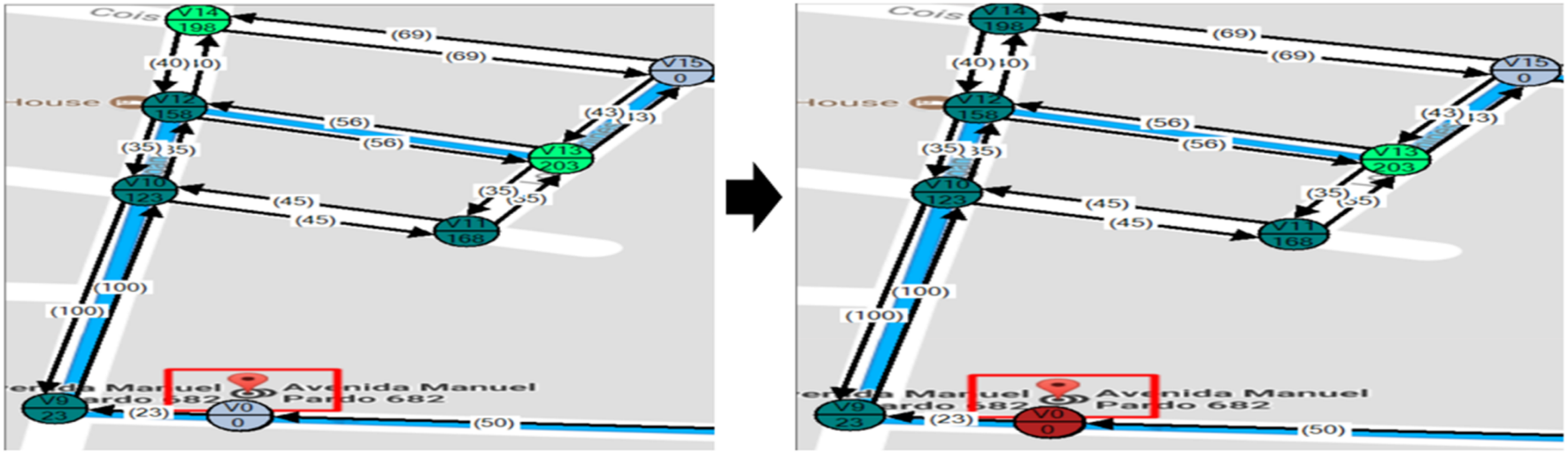


Figura 27: Obtener la ruta optima a partir del vértice v11, del Grafo dirigido de la red de recolección de residuos solidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $(v11, v13)$ y $v14$ como permanente.

Tabla 9

Matriz de adyacencia (v11, v13) y v14.

	$v0$	$v1$	$v2$	$v3$	$v4$	$v5$	$v6$	$v7$	$v8$	$v9$	$v10$	$v11$	$v12$	$v13$	$v14$	$v15$
$v0$	0	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
$v1$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v2$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v3$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v4$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v5$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v6$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v7$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v8$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v9$	α	α	α	α	α	α	α	α	α	α	123	α	α	α	α	α
$v10$	α	α	α	α	α	α	α	α	α	α	α	168	158	α	α	α
$v11$	α	α	α	α	α	α	α	α	α	α	α	α	α	203	α	α
$v12$	α	α	α	α	α	α	α	α	α	α	α	α	α	214	198	α
$v13$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v14$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v15$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Sexto paso, al ser $(v15)$ el único vértice adyacente de $(v14)$ y no tener un costo actual, se actualiza el costo del vértice $(v15)$, luego de actualizar el costo del vértice $(v15)$, se evalúan los vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v13)$, con un costo de 203m, como se visualiza en la figura 28.

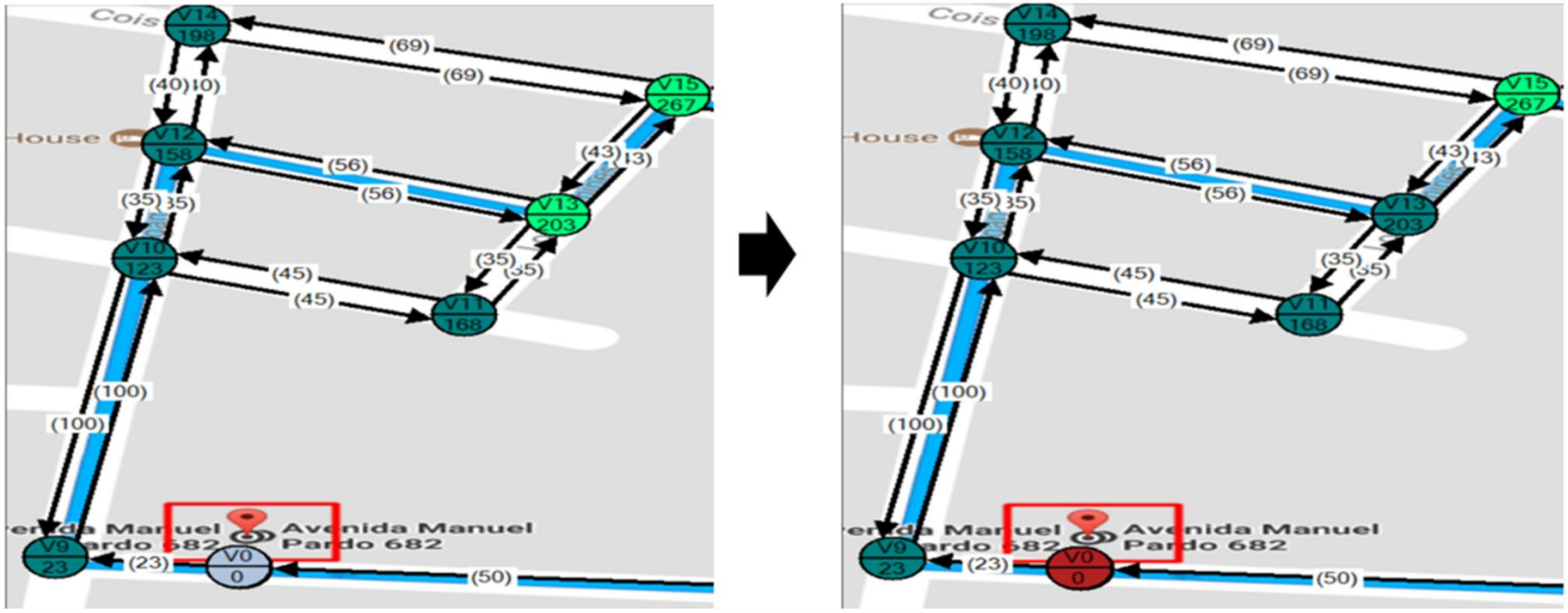


Figura 28: Obtener la ruta optima a partir del vértice v14, del Grafo dirigido de la red de recolección de residuos solidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $(v14, v15)$ y $v13$ como permanente.

Tabla 10

Matriz de adyacencia (v14, v15) y v13

	$v0$	$v1$	$v2$	$v3$	$v4$	$v5$	$v6$	$v7$	$v8$	$v9$	$v10$	$v11$	$v12$	$v13$	$v14$	$v15$
$v0$	0	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
$v1$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v2$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v3$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v4$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v5$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v6$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v7$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v8$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v9$	α	α	α	α	α	α	α	α	α	α	123	α	α	α	α	α
$v10$	α	α	α	α	α	α	α	α	α	α	α	168	158	α	α	α
$v11$	α	α	α	α	α	α	α	α	α	α	α	α	α	203	α	α
$v12$	α	α	α	α	α	α	α	α	α	α	α	α	α	214	198	α
$v13$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v14$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	267
$v15$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Séptimo paso, al ser $(v15)$ el único vértice adyacente de $(v13)$, por lo tanto se realizó el cálculo y se verifica si el costo actual del vértice $(v15)$ es mayor que el calculado en la presente iteración, de ser cierto se actualiza el costo del vértice $(v15)$, de lo contrario se mantiene, en la presente iteración se actualiza el vértice $(v15)$, debido que el cálculo da como resultado un menor costo, como es el único vértice analizado se convierte en el de menor costo y se convierte en un vértice permanente como se visualiza en la figura 29.

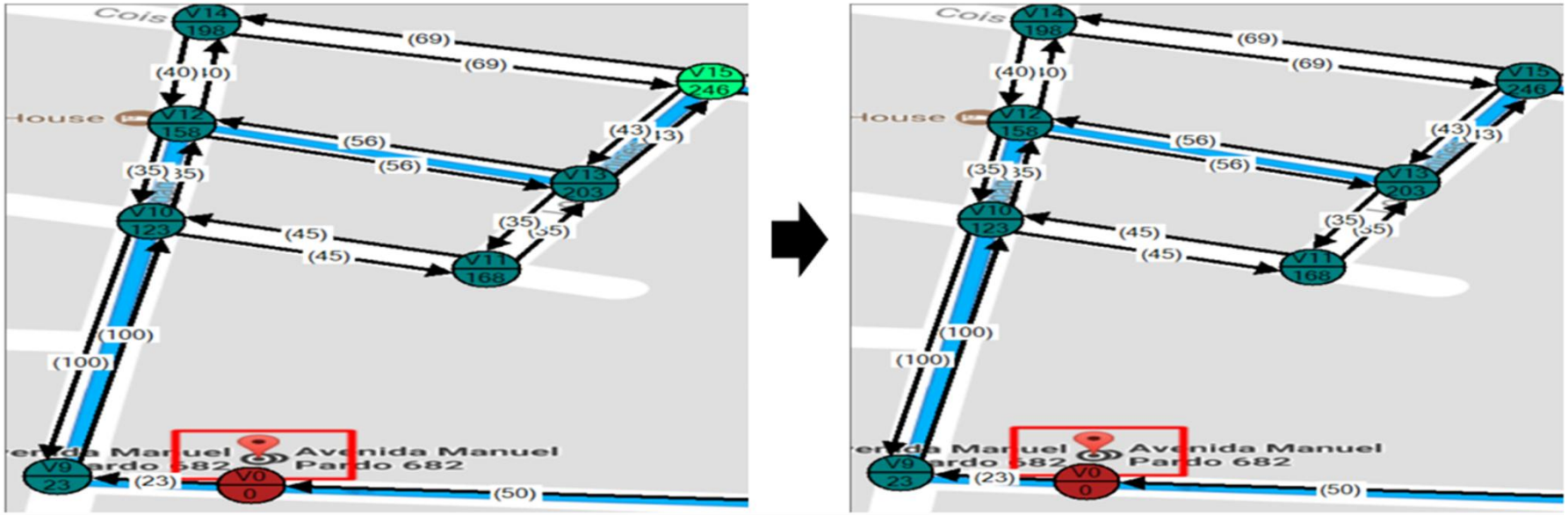


Figura 29: Obtener la ruta optima a partir del vértice v13, del Grafo dirigido de la red de recolección de residuos solidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración entre un par de vértices finaliza actualizando la matriz de adyacencia $(v13, v15)$.

Tabla 11:

Matriz de adyacencia (v13, v15).

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	0	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
v1	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v6	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	123	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	168	158	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	203	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	214	198	α
v13	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	246
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	267
v15	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Octavo paso, al ser $(v07)$ el único vértice adyacente de $(v15)$ se realizó el cálculo y se actualizó el vértice $(v07)$, como se visualiza en la figura 30.

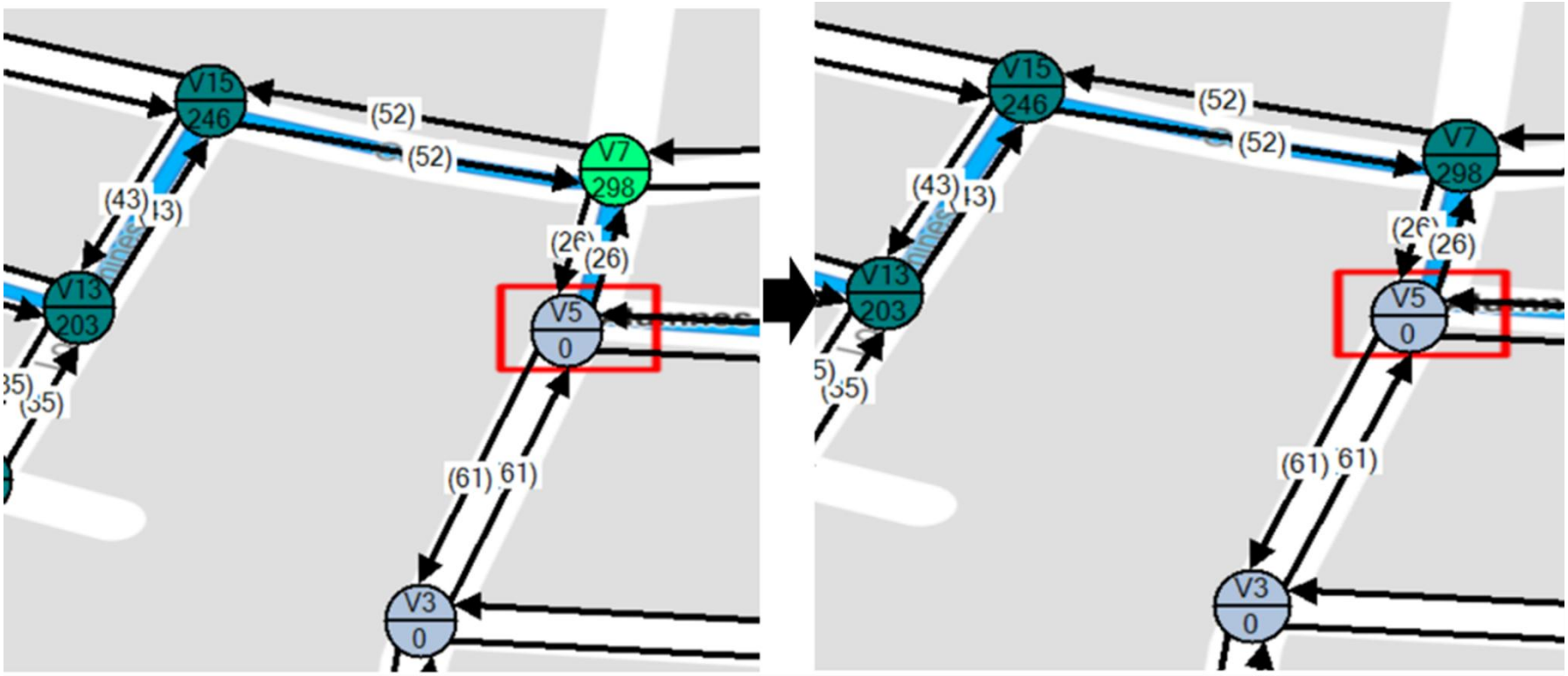


Figura 30: Obtener la ruta optima a partir del vértice v15, del Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración entre un par de vértices finaliza actualizando la matriz de adyacencia $(v15, v07)$.

Tabla 12:

Matriz de adyacencia (v15, v07).

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	0	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
v1	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v6	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	123	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	168	158	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	203	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	214	198	α
v13	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	246
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	267
v15	α	α	α	α	α	α	α	298	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Noveno paso, el vértice $(v07)$, tiene dos posibles caminos $(v08 \text{ y } v05)$, (este último siendo uno de los focos infecciosos) se debe tomar el camino de menor costo (distancia), el costo actual del vértice $(v07)$, tiene una distancia acumulada de 298m, por lo tanto el camino que debería seguir es el vértice $(v05)$, debido que tiene la menor distancia y a su vez se está llegando al primer foco infeccioso $(v05)$ de manera óptima, como se aprecia en la figura 31.

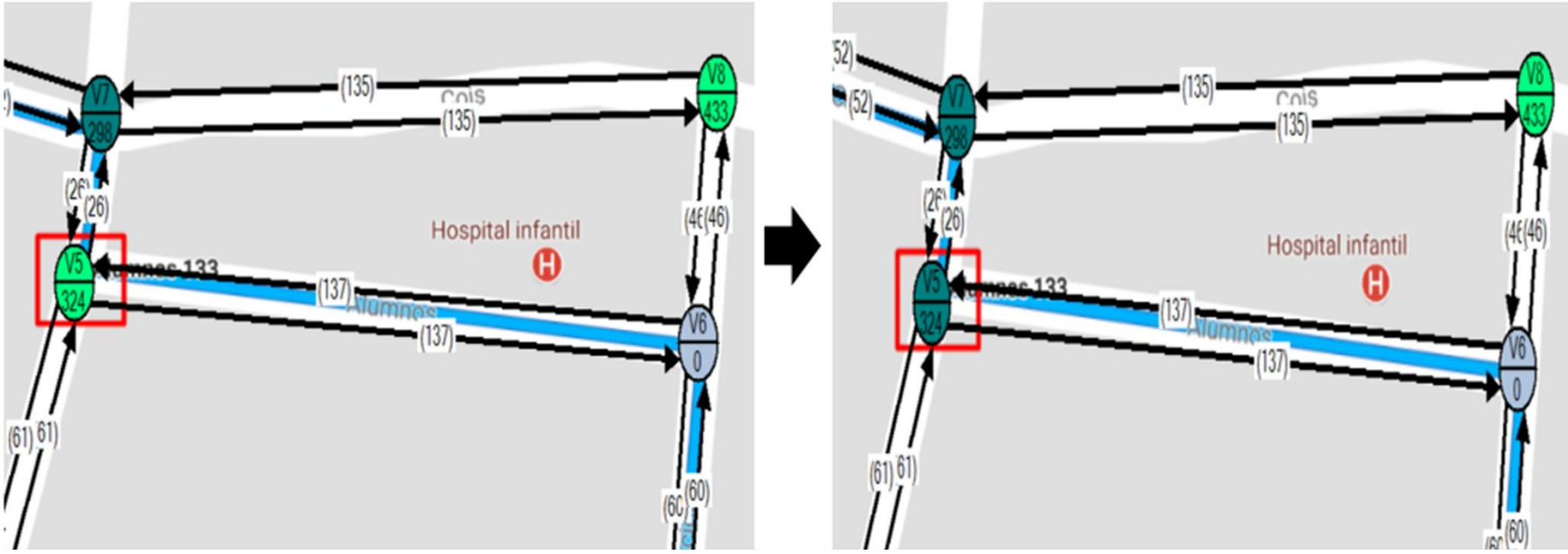


***Figura 31**:* Obtener la ruta optima a partir del vértice v7, del Grafo dirigido de la red de recolección de residuos sólidos.

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración entre un par de vértices finaliza actualizando la matriz de adyacencia $(v7, v5)$ y $(v7, v8)$.

Tabla 13:

Matriz de adyacencia (v7, v5) y (v7, v8).

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	0	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
v1	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v6	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	324	α	α	433	α	α	α	α	α	α	α
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	123	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	168	158	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	203	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	214	198	α
v13	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	246
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	267
v15	α	α	α	α	α	α	α	298	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

El proceso de encontrar la ruta óptima de un vértice inicial (punto de partida del camión) hasta un vértice final (primer foco infeccioso), se llegó a encontrar siguiendo la trayectoria mostrada a continuación: $(v0 - v9 - v10 - v11 - v13 - v15 - v7 - v5)$ con un costo total de 324 en este caso metros. Una vez encontrado el primer foco infeccioso se reinicia la matriz y a su vez el grafo para repetir el proceso pero esta vez desde el vértice v5 el cual es ahora el vértice inicial, ver figura 32.

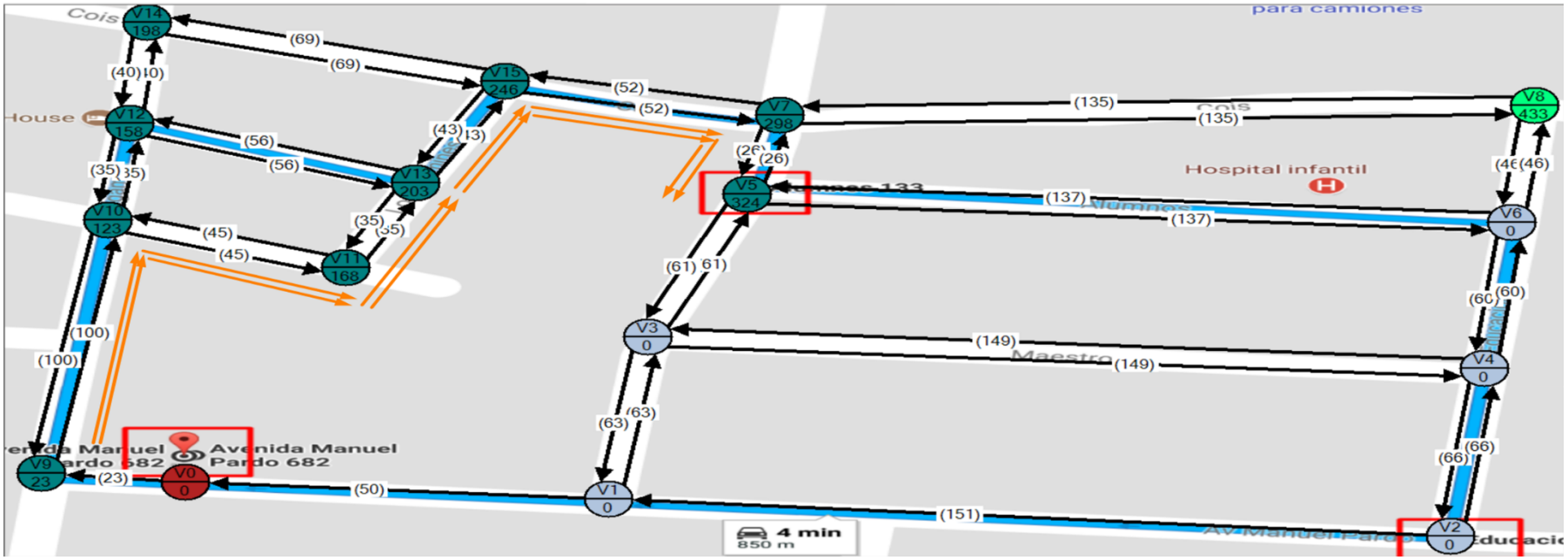


Figura 32: Trayectoria de la ruta optima de v0-v5, del Grafo dirigido de la red de recolección de residuos solidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Tabla 14:

Ruta optima Matriz de adyacencia v0-v5.

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	0	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
v1	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v6	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	324	α	α	α	α	α	α	α	α	α	α
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	123	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	168	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	203	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	246
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	298	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Primer paso, buscar los vértices adyacentes del vértice inicial $(v5)$, el vértice $(v5)$ tiene 3 posibles caminos $(v3, v7, v8)$, se debe tomar el camino de menor costo, por lo tanto el camino que debe seguir es el vértice $(v7)$, debido que tiene la menor distancia, ver figura 33.

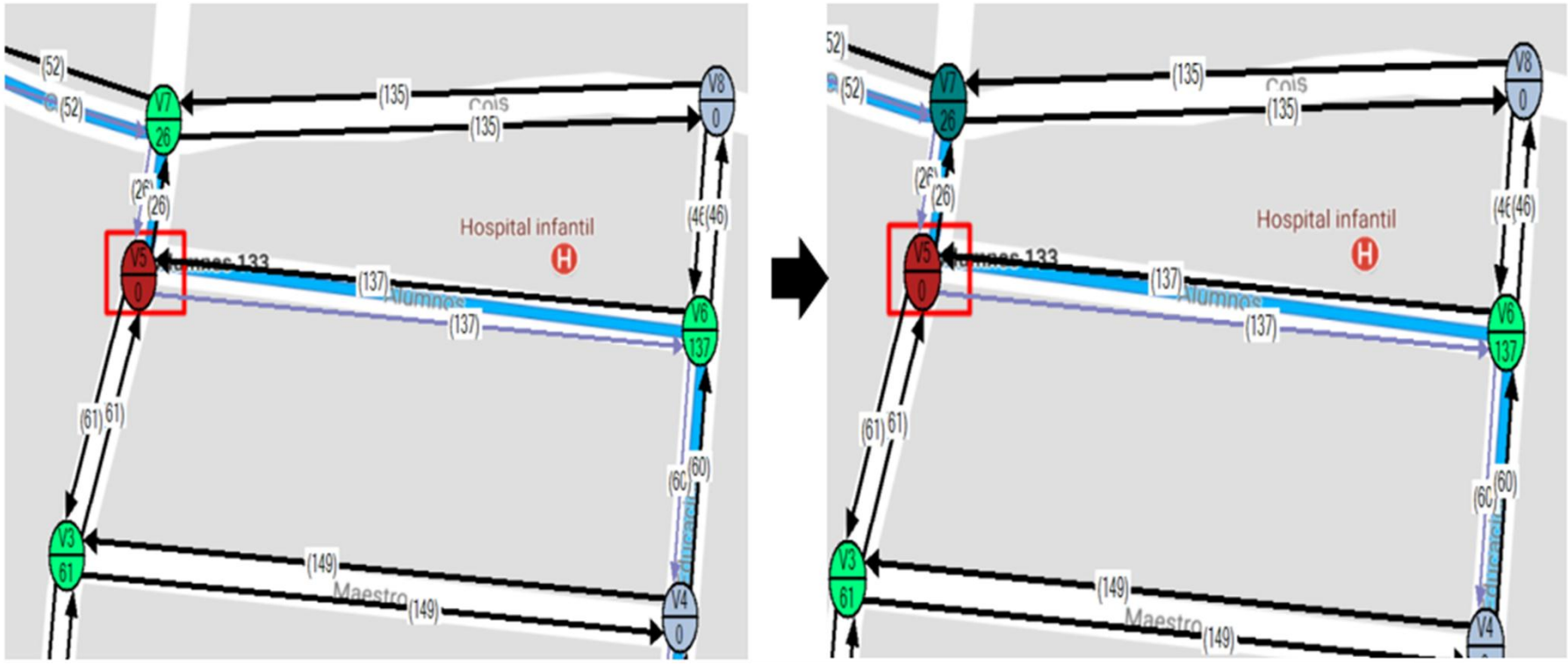


Figura 33: Obtener la ruta optima a partir del vértice v5, del Grafo dirigido de la red de recolección de residuos solidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración entre un par de vértices finaliza actualizando la matriz de adyacencia $(v5, v3)$ y $(v5, v7), (v5. v8)$.

Tabla 15

Matriz de adyacencia (v5, v3), (v5, v7) y (v5, v8).

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v1	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
v6	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Segundo paso, buscar los vértices adyacentes del vértice $(v7)$, el vértice $(v7)$ tiene 2 posibles caminos $(v8, v15)$, se debe tomar el camino de menor costo, por lo tanto el camino que debe seguir es el vértice $(v15)$, debido que tiene la menor distancia, sin embargo al actualizar el costo de los vértices $(v8, v15)$, estos tienen un costo de $(161\ y\ 78)$ metros respectivamente, entonces ahora existen cuatro vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v3)$, con un costo de 61m, como se visualiza en la figura 34:

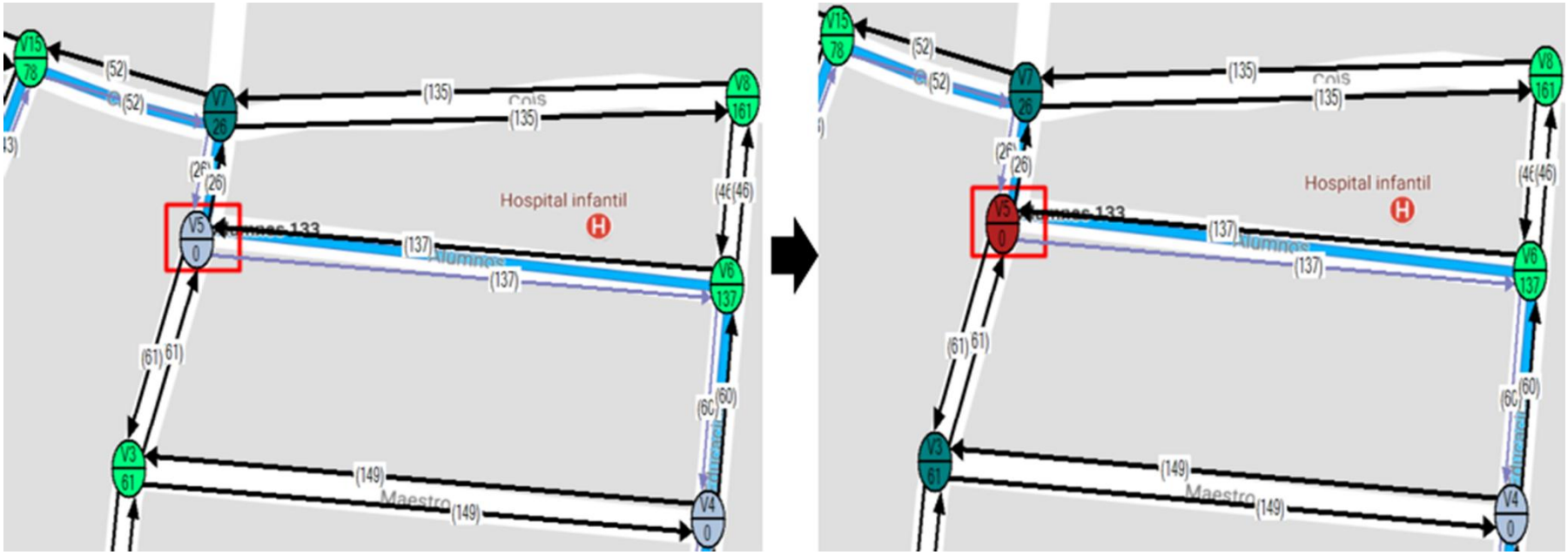


Figura 34: Obtener la ruta optima a partir del vértice v7, del Grafo dirigido de la red de recolección de residuos solidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $(v7, v8)$ y $(v7, v15)$ y $(v3)$ como permanente.

Tabla 16:

Matriz de adyacencia (v7, v8), (v7, v15) y (v3).

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v1	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
v6	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	161	α	α	α	α	α	α	78
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Tercer paso, buscar los vértices adyacentes del vértice $(v3)$, el vértice $(v3)$ tiene 2 posibles caminos $(v1\ y\ v4)$, se debe tomar el camino de menor costo, por lo tanto el camino que debe seguir es el vértice $(v1)$, debido que tiene la menor distancia, sin embargo al actualizar el costo de los vértices $(v1\ y\ \ v4)$, estos tienen un costo de $(124\ y\ 210)$ metros respectivamente, entonces ahora existen cinco vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v15)$, con un costo de 78m, como se visualizó en la figura 35:

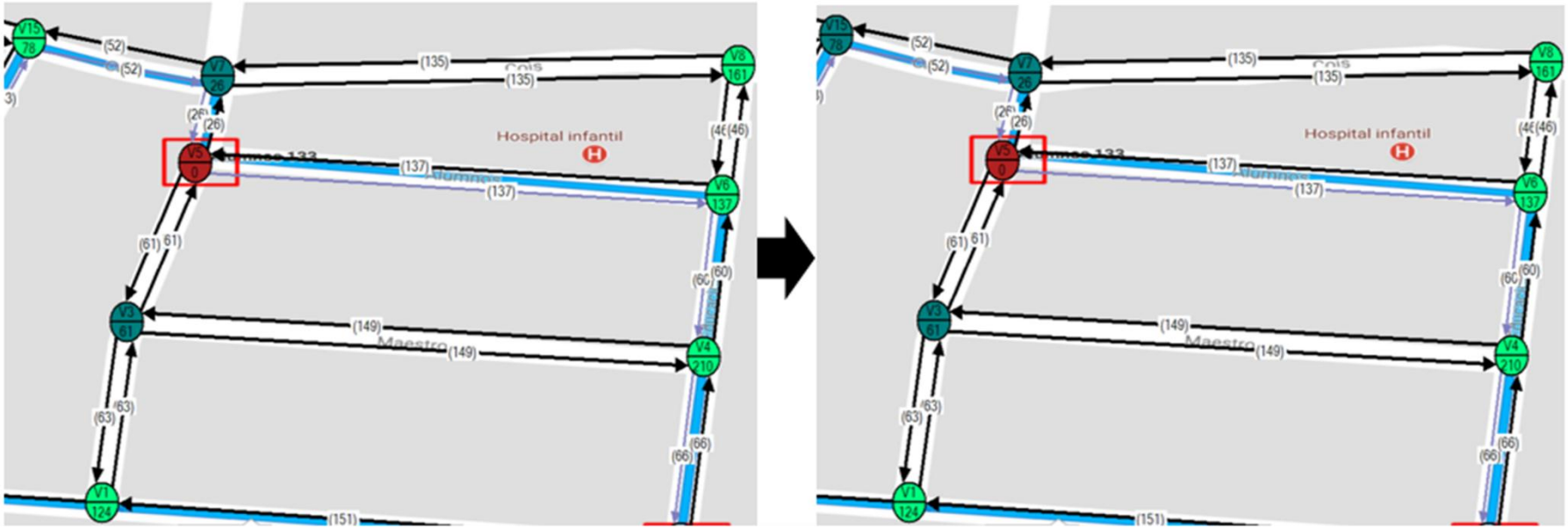


Figura 35: Obtener la ruta optima a partir del vértice v3, del Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $(v3, v1)$ y $(v3, v4)$ y $(v15)$ como permanente.

Tabla 17:

Matriz de adyacencia (v3, v1), (v3, v4) y (v15).

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v1	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	124	α	α	210	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
v6	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	161	α	α	α	α	α	α	78
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Cuarto paso, buscar los vértices adyacentes del vértice $(v15)$, el vértice $(v15)$ tiene 2 posibles caminos $(v13\ y\ v14)$, se debe tomar el camino de menor costo, por lo tanto el camino que debe seguir es el vértice $(v13)$, debido que tiene la menor distancia, sin embargo al actualizar el costo de los vértices $(v13\ y\ \ v14)$, estos tienen un costo de $(121\ y\ 147)$ metros respectivamente entonces ahora existen seis vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v13)$, con un costo de 121m, como se visualizó en la figura 36:

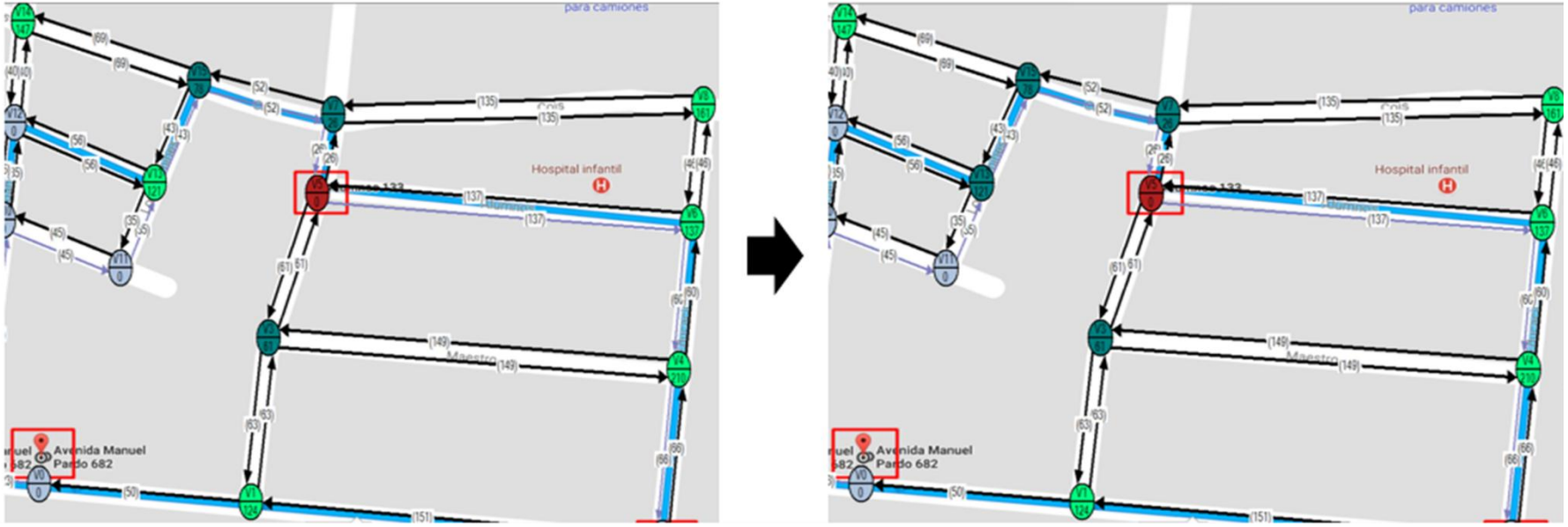


***Figura 36**:* Obtener la ruta optima a partir del vértice v15, del Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $(v15, v13)$ y $(v15, v14)$ y $(v13)$ como permanente.

Tabla 18:

Matriz de adyacencia (v15, v13), (v15, v14) y (v13).

	$v0$	$v1$	$v2$	$v3$	$v4$	$v5$	$v6$	$v7$	$v8$	$v9$	$v10$	$v11$	$v12$	$v13$	$v14$	$v15$
$v0$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v1$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v2$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v3$	α	124	α	α	210	α	α	α	α	α	α	α	α	α	α	α
$v4$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v5$	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
$v6$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v7$	α	α	α	α	α	α	α	α	161	α	α	α	α	α	α	78
$v8$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v9$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v10$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v11$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v12$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v13$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v14$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v15$	α	α	α	α	α	α	α	α	α	α	α	α	α	121	147	α

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Quinto paso, buscar los vértices adyacentes del vértice $(v13)$, el vértice $(v13)$ tiene 2 posibles caminos $(v11\ y\ v12)$, se debe tomar el camino de menor costo, por lo tanto el camino que debe seguir es el vértice $(v11)$, debido que tiene la menor distancia, sin embargo al actualizar el costo de los vértices $(v11\ y\ \ v12)$, estos tienen un costo de $(156\ y\ 177)$ metros respectivamente entonces ahora existen siete vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v1)$, con un costo de 124m, como se visualizó en la figura 37:

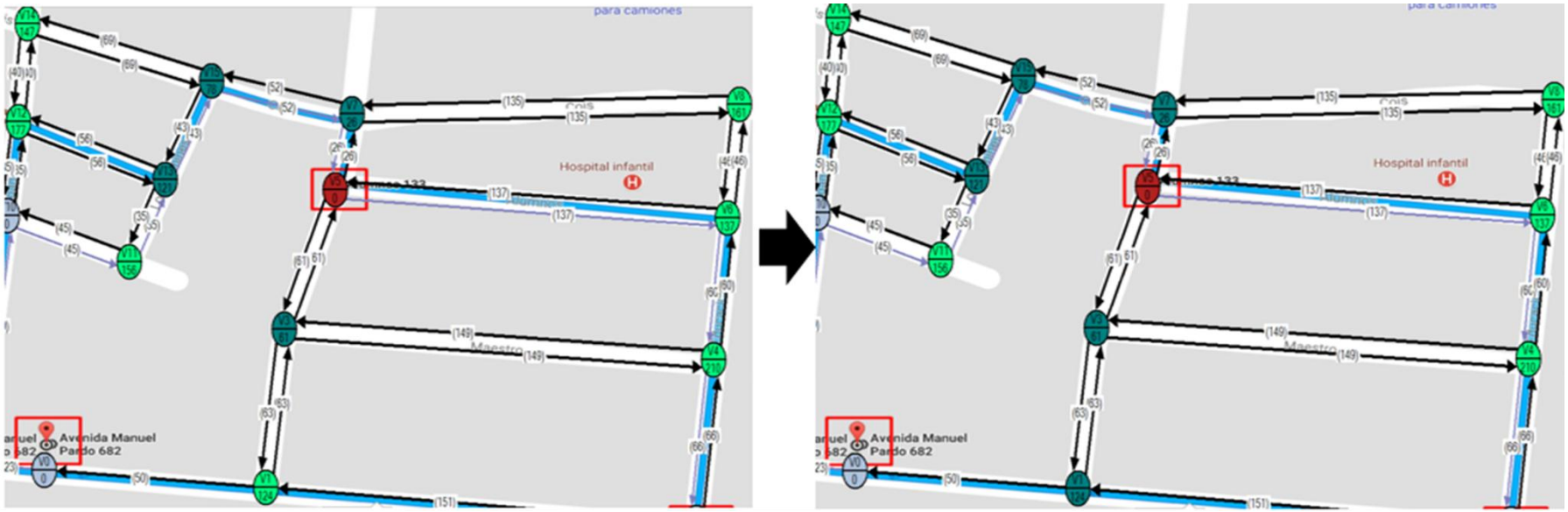


Figura 37: Obtener la ruta optima a partir del vértice v13, del Grafo dirigido de la red de recolección de residuos solidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $(v13, v11)$ y $(v13, v12)$ y $(v1)$ como permanente.

Tabla 19:

Matriz de adyacencia (v13, v11), (v13, v12) y (v1).

	$v0$	$v1$	$v2$	$v3$	$v4$	$v5$	$v6$	$v7$	$v8$	$v9$	$v10$	$v11$	$v12$	$v13$	$v14$	$v15$
$v0$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v1$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v2$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v3$	α	124	α	α	210	α	α	α	α	α	α	α	α	α	α	α
$v4$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v5$	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
$v6$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v7$	α	α	α	α	α	α	α	α	161	α	α	α	α	α	α	78
$v8$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v9$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v10$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v11$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v12$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v13$	α	α	α	α	α	α	α	α	α	α	α	156	177	α	α	α
$v14$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v15$	α	α	α	α	α	α	α	α	α	α	α	α	α	121	147	α

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Sexto paso, buscar los vértices adyacentes del vértice $(v1)$, el vértice $(v1)$ tiene 2 posibles caminos $(v0\ y\ v2)$, se debe tomar el camino de menor costo, por lo tanto el camino que debe seguir es el vértice $(v0)$, debido que tiene la menor distancia, sin embargo al actualizar el costo de los vértices $(v0\ y\ \ v2)$, estos tienen un costo de $(174\ y\ 275)$ metros respectivamente entonces ahora existen ocho vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v6)$, con un costo de 137m, ver figura 38:

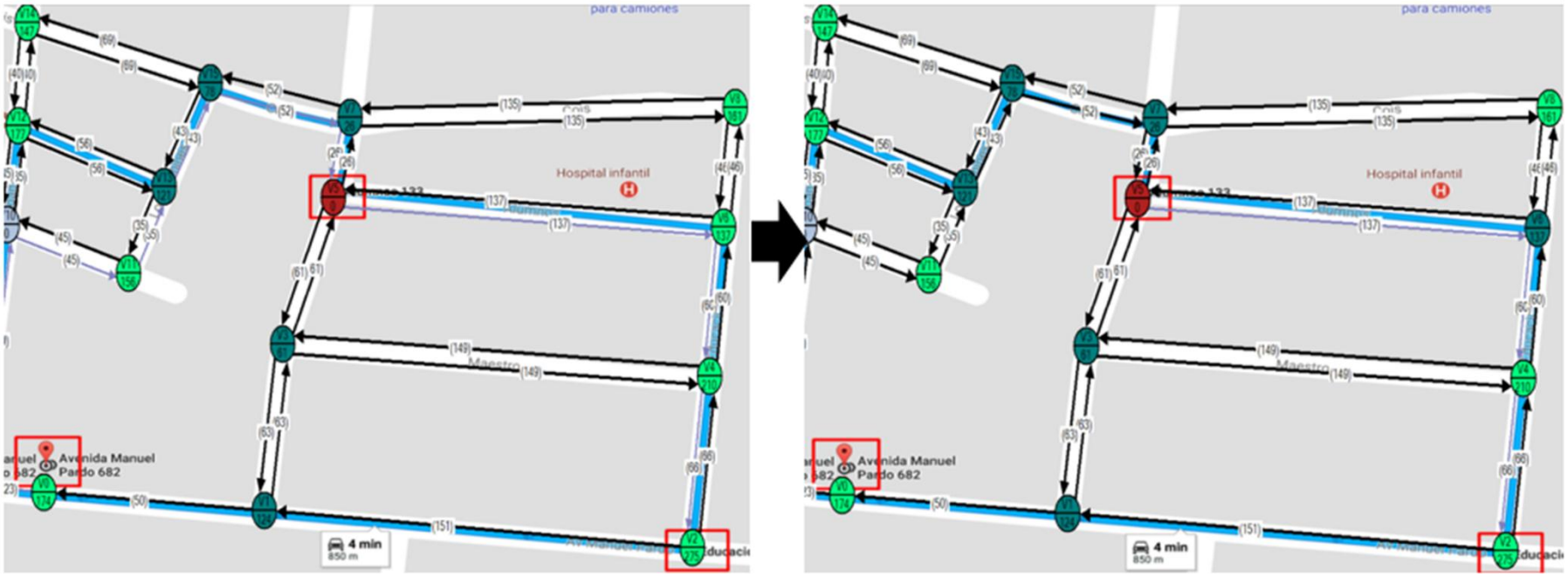


Figura 38: Obtener la ruta optima a partir del vértice v1, del Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $(v1, v0)$ y $(v1, v2)$ y $(v6)$ como permanente.

Tabla 20:

Matriz de adyacencia (v1, v0), (v1, v2) y (v6).

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v1	174	α	275	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	124	α	α	210	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
v6	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	161	α	α	α	α	α	α	78
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	156	177	α	α	α
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	α	α	α	α	α	α	121	147	α

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Séptimo paso, buscar los vértices adyacentes del vértice $(v6)$, el vértice $(v6)$ tiene 2 posibles caminos $(v4\ y\ v8)$, (cabe recalcar que se realizó una previa validación debido que $(v4\ y\ v8)$ ya tenían un costo previo) se debe tomar el camino de menor costo, por lo tanto el camino que debe seguir es el vértice $(v8)$, debido que tiene la menor distancia, sin embargo al actualizar el costo de los vértices $(v4\ y\ \ v8)$ estos tienen un costo de $(197\ y\ 161)$ metros respectivamente entonces ahora existen siete vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v14)$, con un costo de 147m, ver figura 39:

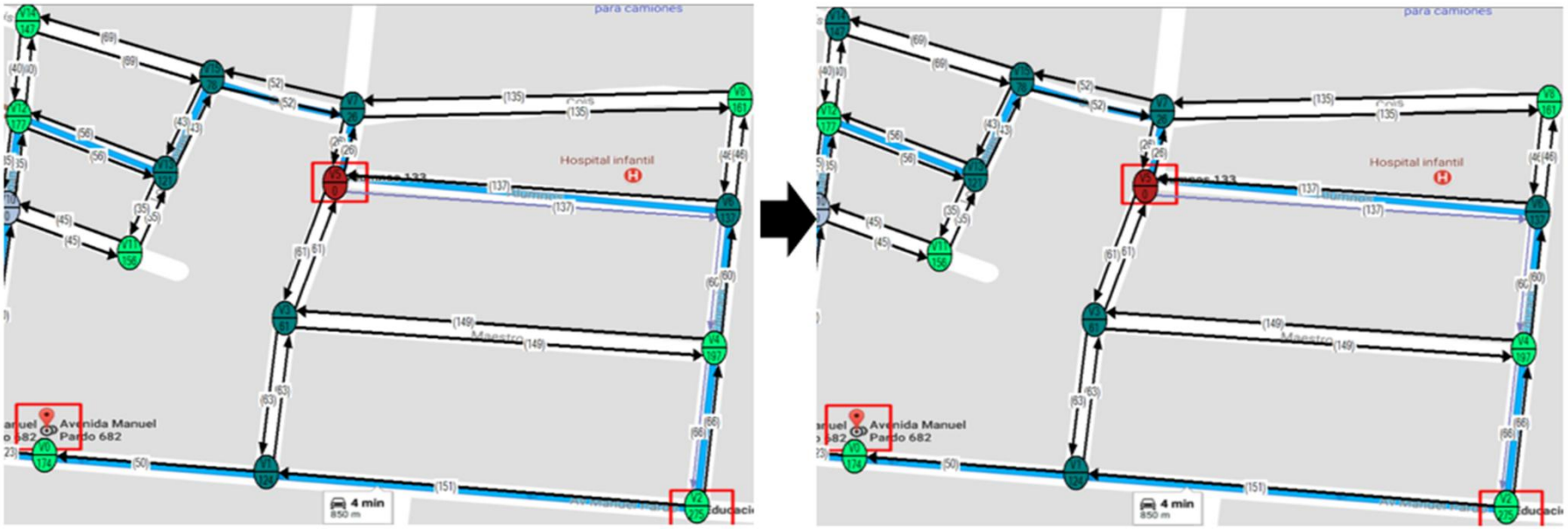


Figura 39: Obtener la ruta optima a partir del vértice v6, del Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $(v6, v4)$ y $(v14)$ como permanente.

Tabla 21:

Matriz de adyacencia (v6, v4) y (v14).

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v1	174	α	275	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	124	α	α	210	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
v6	α	α	α	61	197	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	161	α	α	α	α	α	α	78
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	156	177	α	α	α
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	α	α	α	α	α	α	121	147	α

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Octavo paso, buscar los vértices adyacentes del vértice $(v14)$, el vértice $(v14)$ tiene 1 solo camino $(v12)$, (cabe recalcar que se realizó una previa validación debido que $(v12)$ ya tenía un costo previo), el costo del vértice $(v12)$ se mantiene, entonces ahora existen seis vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v11)$, con un costo de 156m, ver figura 40:

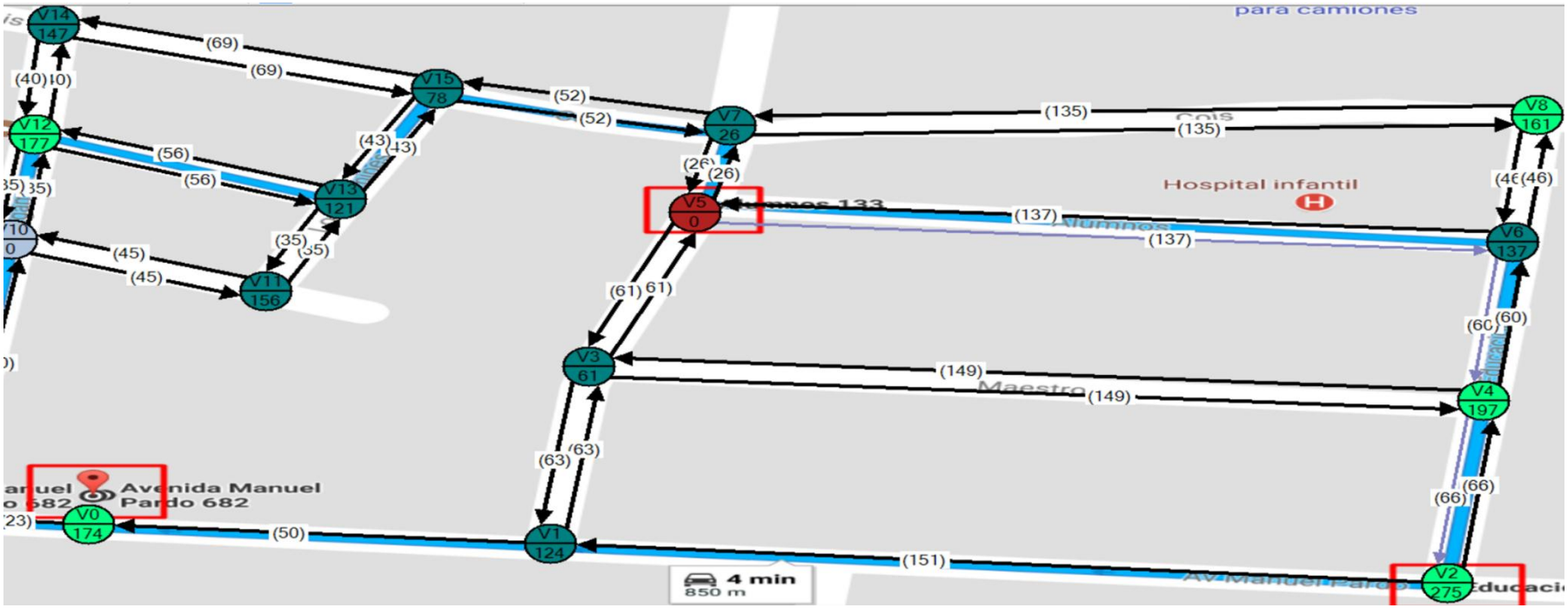


Figura 40: Obtener la ruta optima a partir del vértice v14, del Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $(v11)$ como permanente.

Tabla 22:

Matriz de adyacencia (v11).

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v1	174	α	275	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	124	α	α	210	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
v6	α	α	α	α	197	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	161	α	α	α	α	α	α	78
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	156	177	α	α	α
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	α	α	α	α	α	α	121	147	α

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Noveno paso, buscar los vértices adyacentes del vértice $(v11)$, el vértice $(v11)$ tiene 1 solo camino $(v10)$, entonces ahora existen seis vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v8)$, con un costo de 161m, ver figura 41:

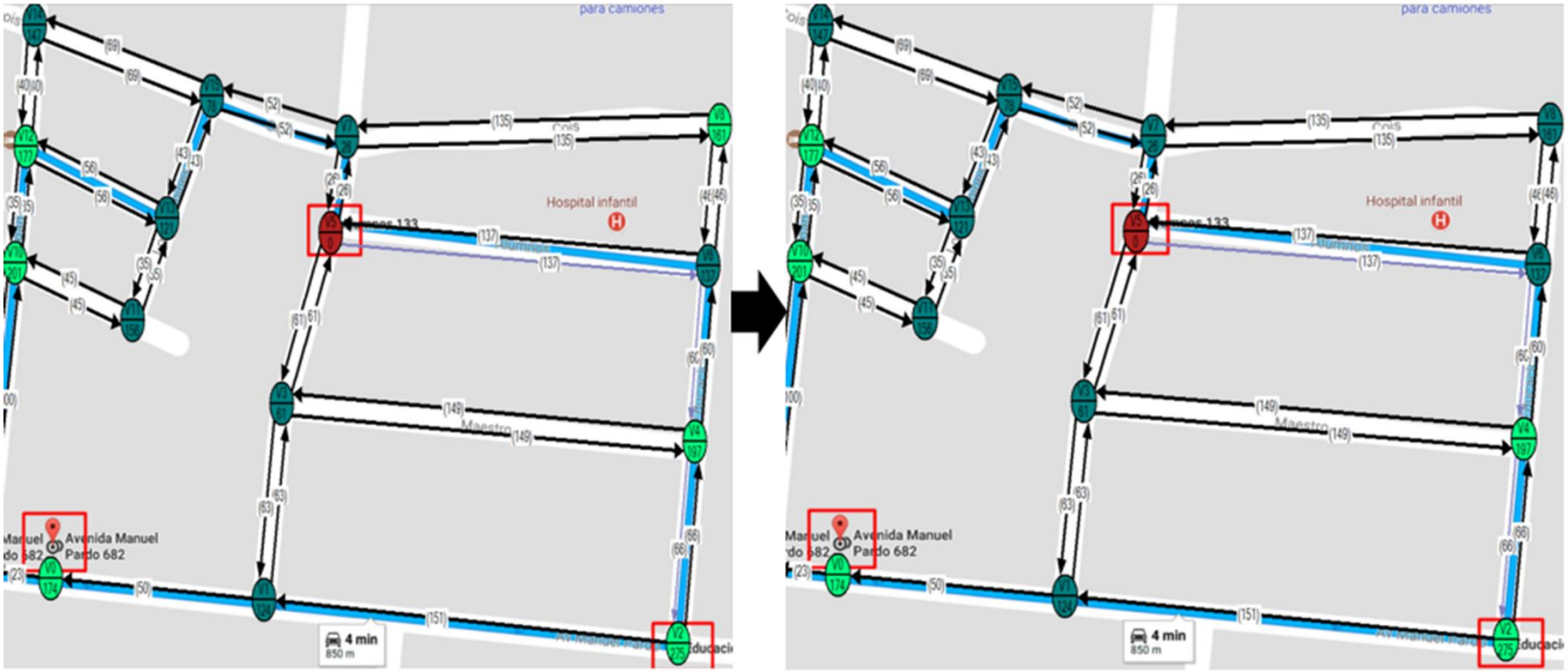


Figura 41: Obtener la ruta optima a partir del vértice v11, del Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $(v11, v10)$ y $v8$ como permanente.

Tabla 23:

Matriz de adyacencia (v11, v10) y v8.

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v1	174	α	275	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	124	α	α	210	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
v6	α	α	α	α	197	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	161	α	α	α	α	α	α	78
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	201	α	α	α	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	156	177	α	α	α
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	α	α	α	α	α	α	121	147	α

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Decimo paso, buscar los vértices adyacentes del vértice $(v8)$, tiene 2 caminos $(v6, v7)$, sin embargo no se modifica sus costos de los vértices $(v6, v7)$, debido que sus costos actuales tienen menor costo que el cálculo por $(v8)$, entonces ahora existen cinco vértices explorados y se deberá elegir el vértice de menor distancia el cual es el vértice $(v0)$, con un costo de 174m, ver figura 42:

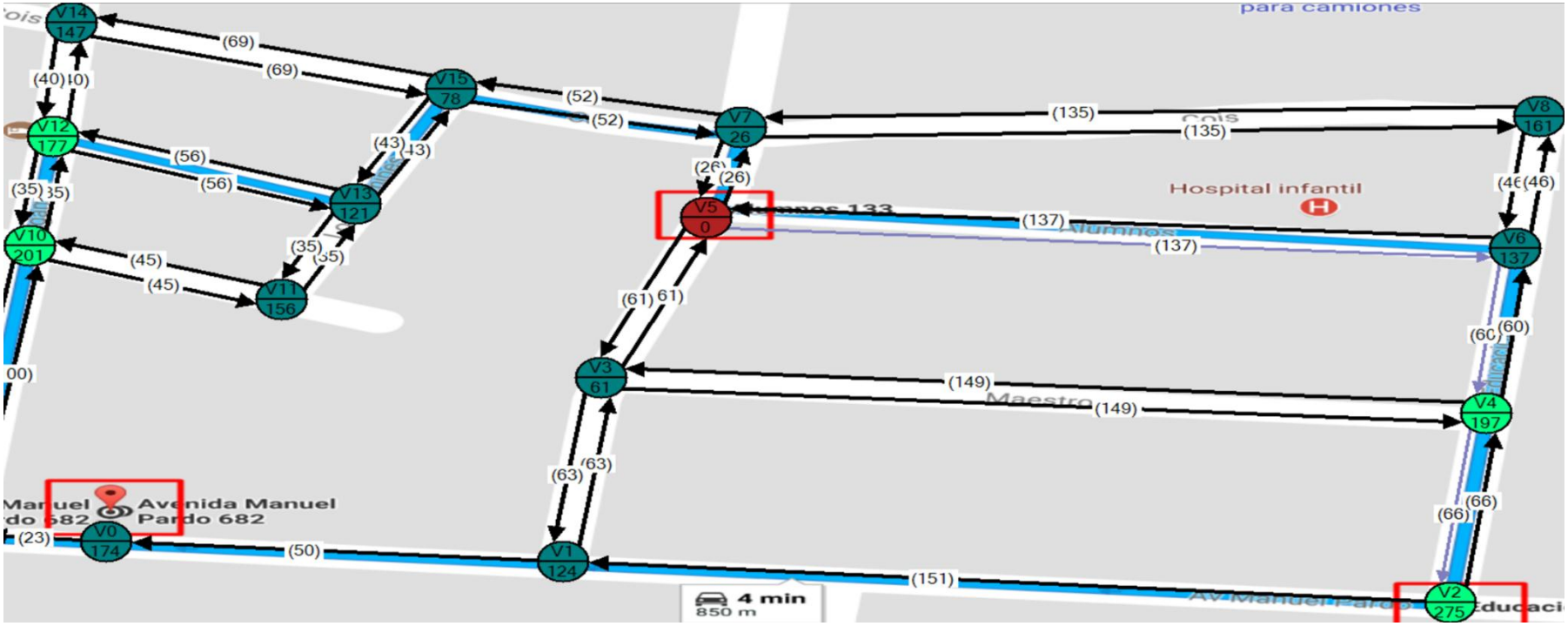


Figura 42: Obtener la ruta óptima a partir del vértice v8, del Grafo dirigido de la red de recolección de residuos sólidos.

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $v0$ como permanente.

Tabla 24:

Matriz de adyacencia v0.

	$v0$	$v1$	$v2$	$v3$	$v4$	$v5$	$v6$	$v7$	$v8$	$v9$	$v10$	$v11$	$v12$	$v13$	$v14$	$v15$
$v0$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v1$	174	α	275	α	α	α	α	α	α	α	α	α	α	α	α	α
$v2$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v3$	α	124	α	α	210	α	α	α	α	α	α	α	α	α	α	α
$v4$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v5$	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
$v6$	α	α	α	α	197	α	α	α	α	α	α	α	α	α	α	α
$v7$	α	α	α	α	α	α	α	α	161	α	α	α	α	α	α	78
$v8$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v9$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v10$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v11$	α	α	α	α	α	α	α	α	α	α	201	α	α	α	α	α
$v12$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v13$	α	α	α	α	α	α	α	α	α	α	α	156	177	α	α	α
$v14$	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
$v15$	α	α	α	α	α	α	α	α	α	α	α	α	α	121	147	α

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Décimo primer paso, buscar los vértices adyacentes del vértice $(v0)$, tiene solo un camino $(v9)$, se actualiza el costo del vértice $(v9)$, entonces ahora existen cinco vértices explorados, los vértices de menor costo son $(v4\ y\ v9)$ con un costo de 197m, el algoritmo de Dijkstra al encontrar dos vértices con un mismo costo toma cualquiera de ellos, para el presente grafo se tomara el vértice $(v4)$ debido que este se encuentra más próximo al vértice $(v2)$, el cual es el próximo foco infeccioso, ver figura 43.

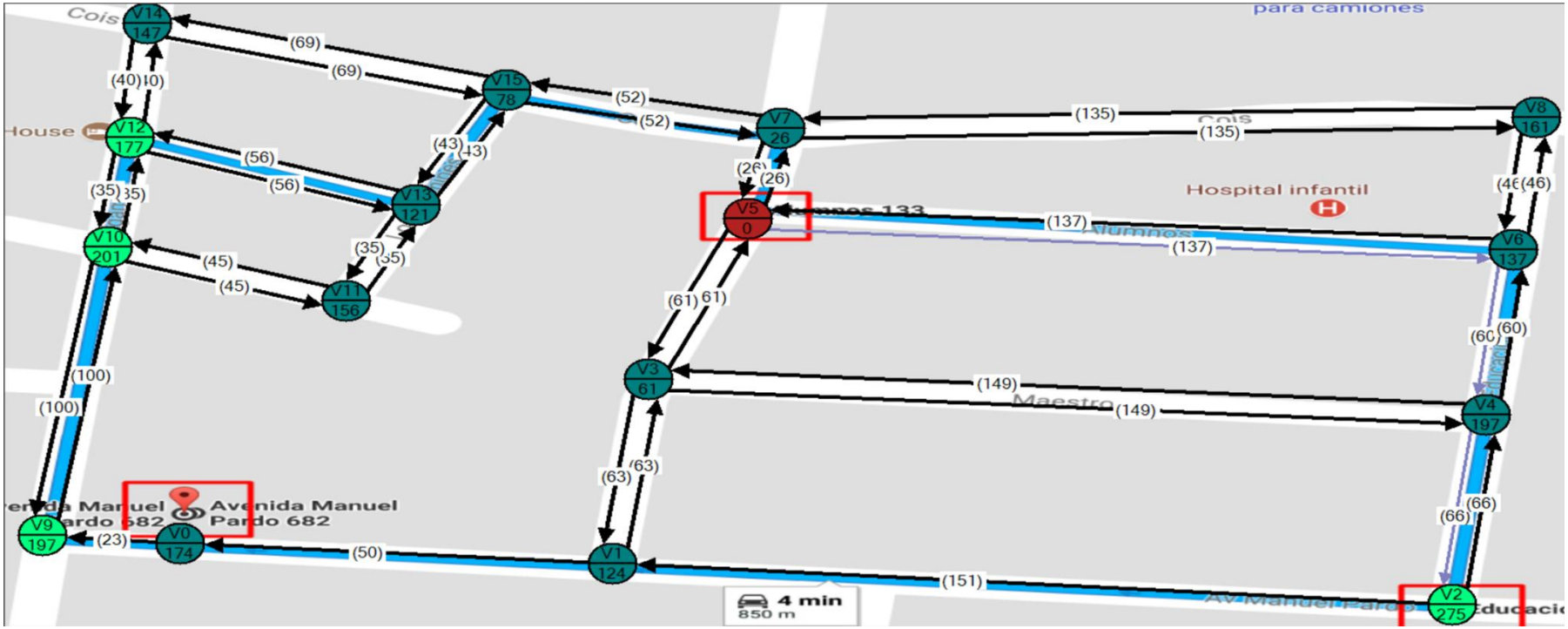


Figura 43: Obtener la ruta optima a partir del vértice v0, del Grafo dirigido de la red de recolección de residuos sólidos.

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

La iteración finaliza actualizando la matriz de adyacencia $v9$ y $v4$ esta última como permanente.

Tabla 25:

Matriz de adyacencia v4.

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	α	α	α	α	α	α	α	α	α	197	α	α	α	α	α	α
v1	174	α	275	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	124	α	α	210	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
v6	α	α	α	α	197	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	161	α	α	α	α	α	α	78
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	201	α	α	α	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	156	177	α	α	α
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	α	α	α	α	α	α	121	147	α

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Décimo segundo paso, buscar los vértices adyacentes del vértice $(v4)$, tiene solo un camino $(v2)$, (cabe recalcar que se realizó una previa validación debido que $(v2)$ ya tenía un costo previo), el costo del vértice $(v2)$ se actualiza, ver figura 44.

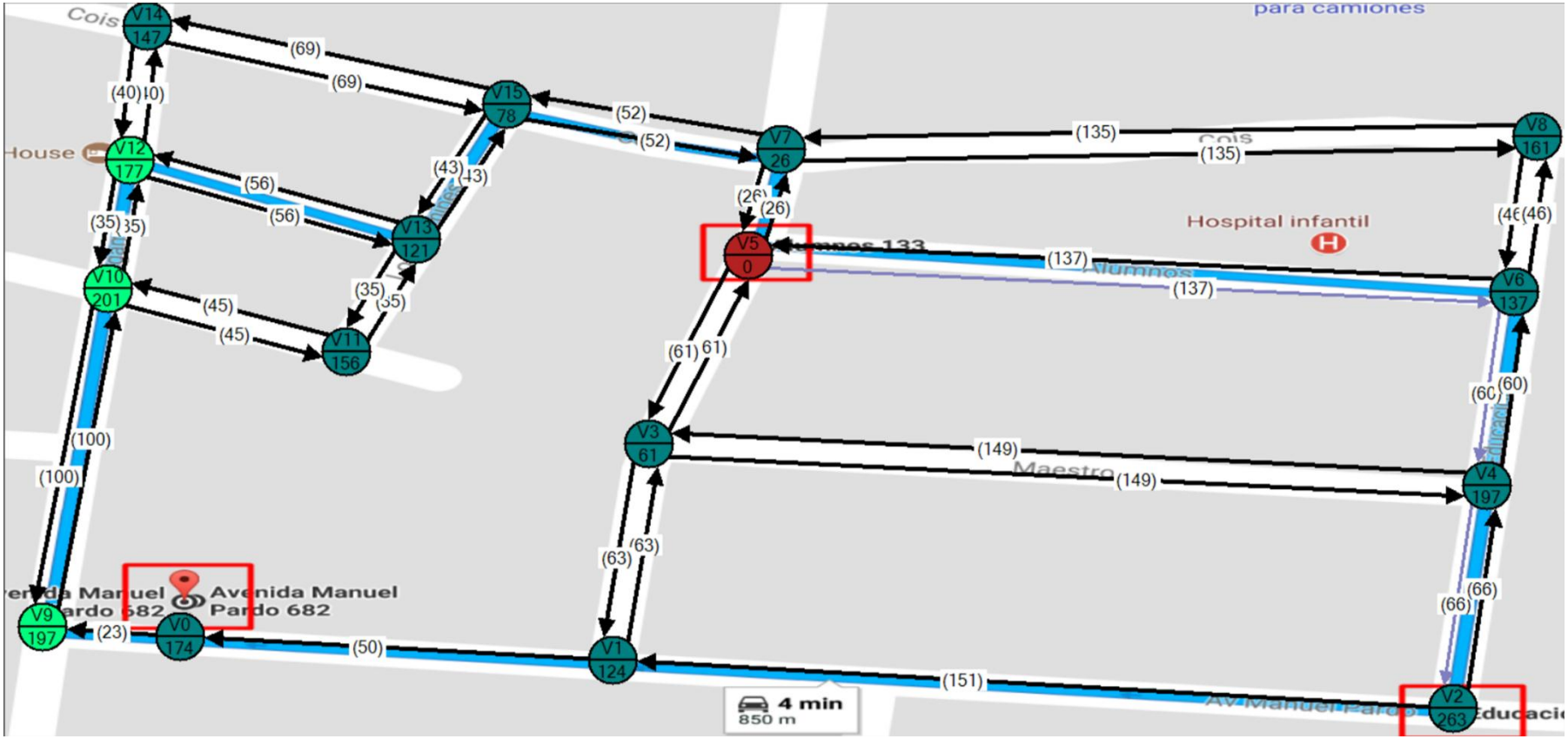


Figura 44: Trayectoria de la ruta optima de v4, del Grafo dirigido de la red de recolección de residuos sólidos.

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Tabla 26:

Matriz de adyacencia v4 y v9.

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	α	α	α	α	α	α	α	α	α	197	α	α	α	α	α	α
v1	174	α	275	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	124	α	α	210	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	263	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	61	α	α	137	26	α	α	α	α	α	α	α	α
v6	α	α	α	α	197	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	161	α	α	α	α	α	α	78
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	201	α	α	α	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	156	177	α	α	α
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	α	α	α	α	α	α	121	147	α

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

El proceso de encontrar la ruta óptima de un vértice inicial (primer foco infeccioso) hasta un vértice final (segundo foco infeccioso), se llegó a encontrar siguiendo la trayectoria mostrada a continuación: $(v5 - v6 - v4 - v2)$ con un costo total de 263 en este caso metros, ver figura 45.

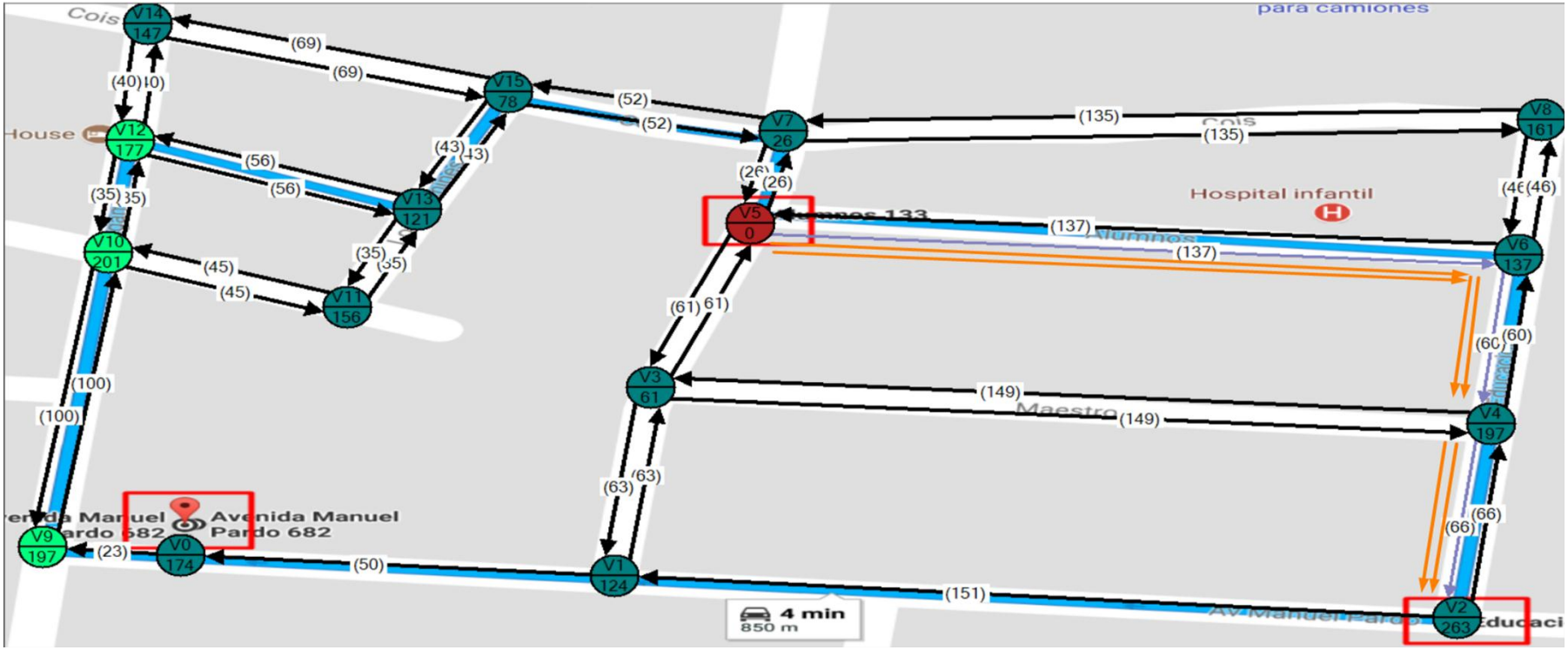


Figura 45: Trayectoria de la ruta optima de v5-v2, del Grafo dirigido de la red de recolección de residuos solidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Tabla 27:

Ruta optima Matriz de adyacencia v5-v2.

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v1	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	263	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	α	α	α	α	α	α	137	α	α	α	α	α	α	α	α	α
v6	α	α	α	α	197	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Lo próximo a realizar es juntar las dos matrices de adyacencia resultantes, es decir: $(v0 - v5\ y\ v5 - v2)$ con un costo total de 324m y 263m respectivamente, ver tabla 28.

Tabla 28:

Ruta óptima; Matriz de adyacencia v0-v5-v2.

	v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15
v0	0	α	α	α	α	α	α	α	α	23	α	α	α	α	α	α
v1	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v2	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v3	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v4	α	α	587	α	α	α	α	α	α	α	α	α	α	α	α	α
v5	324	α	α	α	α	α	461	α	α	α	α	α	α	α	α	α
v6	α	α	α	α	521	α	α	α	α	α	α	α	α	α	α	α
v7	α	α	α	α	α	324	α	α	α	α	α	α	α	α	α	α
v8	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v9	α	α	α	α	α	α	α	α	α	α	123	α	α	α	α	α
v10	α	α	α	α	α	α	α	α	α	α	α	168	α	α	α	α
v11	α	α	α	α	α	α	α	α	α	α	α	α	α	203	α	α
v12	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v13	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	246
v14	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α	α
v15	α	α	α	α	α	α	α	298	α	α	α	α	α	α	α	α

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

A partir de esta nueva matriz se generó el grafo con la ruta óptima de toda la trayectoria es decir $(v0 - v5 - v2)$, ver figura 46.

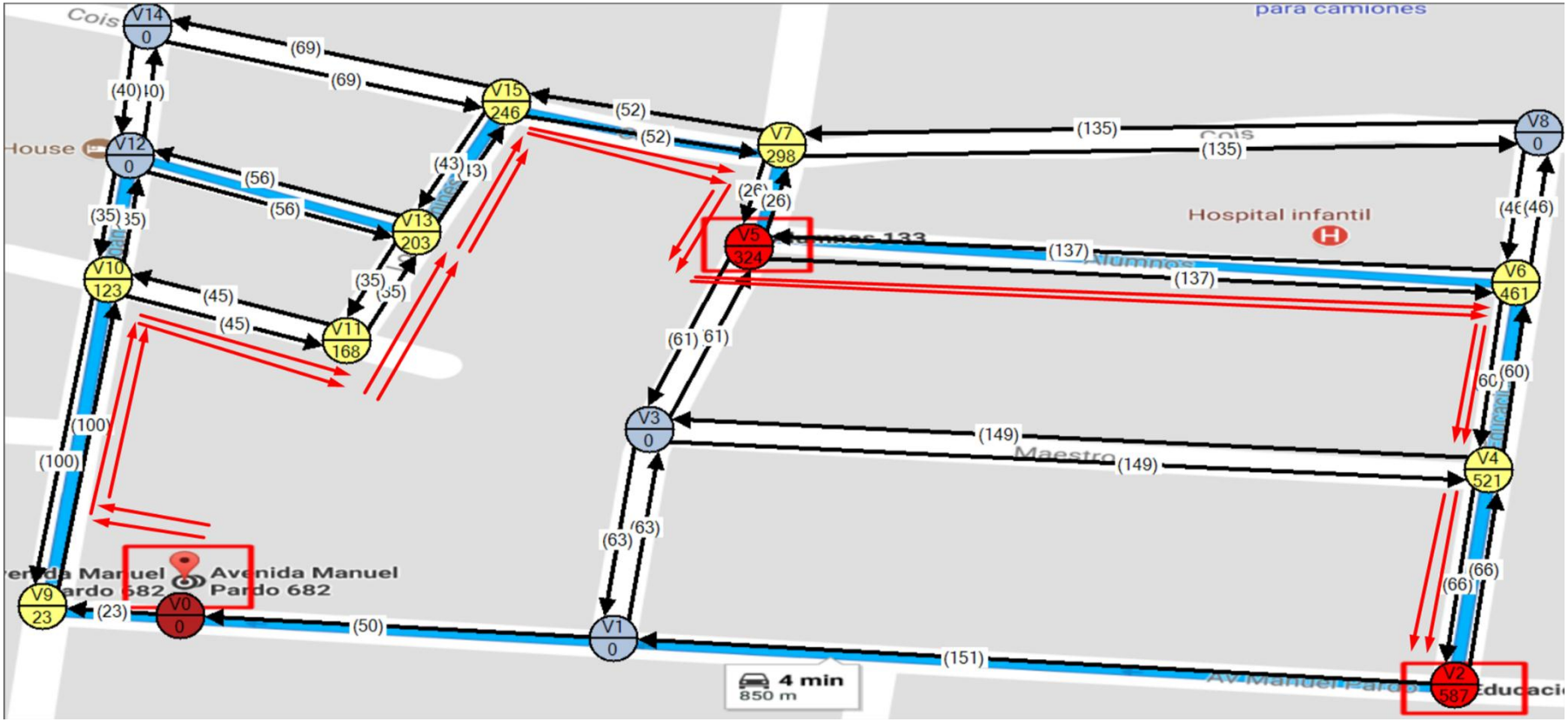


Figura 46: Trayectoria de la ruta optima de v0-v5-v2, del Grafo dirigido de la red de recolección de residuos sólidos

Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Código del sistema.

Código 1.0: Acceso al API V3 de Google Maps desde la aplicación.

```
1.  <!DOCTYPE HTML>
2.  <html lang="es">
3.
4.  <head>
5.      <meta charset="UTF-8">
6.      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7.      <meta http-equiv="X-UA-Compatible" content="ie=edge">
8.      <title>Dijkstra Algorithm</title>
9.  </head>
10. <body>
11.     <!--- Codigo... -->
12.     <div id="map"></div>
13.     <!--- Codigo... -->
14.     <script type="text/javascript" src="https://maps.googleapis.com/maps/api/js?key=AP
    YSECRET"></script>
15. </body>
16.  </html>
17.
18. //Carga el Mapa sobre un elemento DOM
19. createMap() {
20.     let me = this;
21.     //Centrar Mapa en un punto sobre el Distrito de Chiclayo
22.     var positionsMap = {
23.         lat: -6.7646274,
24.         lng: -79.8429839
25.     };
26.     //Configurar opciones del mapa
27.     var optionsMap = {
28.         center: positionsMap,
29.         zoom: 17,
30.         mapTypeId: google.maps.MapTypeId.ROADMAP
31.     };
32.     me.map = new google.maps.Map(document.getElementById('map'), optionsMap);
33. },
```

Al codificar la etiqueta de Google Maps mas el API Key de seguridad proporcionado por Google Maps Developer, se pudo acceder a los servicios del API V3 de Google Maps como los son sus bibliotecas y métodos.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Código 1.1: Cargar los marcadores en el selector que determina el punto inicial y un listado que determina los marcadores para la recolección de desechos.

```
1.  //Carga los select que contendrán los marcadores
2.  loadSelects() {
3.      let me = this;
4.       axios.get("controller/MarkerController.php?op=listar_marker").then(function(respons
    e) {
5.          let answer = response.data;
6.          me.direccionesPlace1 = answer.direcciones;
7.      }).catch(function(error) {
8.          console.log(error);
9.      });
10.
11.     axios.get("controller/MarkerController.php?op=listarAleatorio").then(function(respons
    e) {
12.         let answer = response.data;
13.         me.listMarcadores = answer.listado;
14.     }).catch(function(error) {
15.         console.log(error);
16.     });
17. },
```

Código que me permite cargar un selector con todas las calles que están mapeadas en el Grafo Dirigido y un listado donde simule los focos infecciosos que se localizaran y ubicaran sobre el mapa geográfico.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Código 1.2: Optimización de rutas utilizando el Algoritmo de Dijkstra.

Para la optimización de rutas en una red de transporte, se siguió la lógica del algoritmo de Dijkstra así como también las restricciones que tiene el Algoritmo sobre un Grafo y restricciones del propio Grafo Dirigido a través de su Matriz de Adyacencia.

Pseudocodigo del Algoritmo de Dijkstra.

```
1. Data: G: a grafo dirigido ponderado, w: todos los pesos
   del borde,
2. s: fuente o estado inicial.
3. Result: El camino más corto de una sola fuente.
4. for each vertex v ∈ V [G]:
5.     d[v] ← ∞
6.     π[v] ← NIL
7.
8. d[s] ← 0
9. S ← ∅
10.      Q ← V [G]
11.
12.      while Q ≠ ∅:
13.          u ← EXTRACT-MIN(Q)
14.          S ← S ∪ [u]
15.
16.          for each vertex v ∈ Adj[u]:
17.              if d[u] > d[u] + w(u,v):
18.              d[u] ← d[u] + w(u,v)
19.              π[u] ← u
```

En la línea 1 se especifica la fuente de datos, el cual recibe como argumentos el Grafo dirigido ponderado, el vértice inicial y vértice de destino, el presente código es equivalente a la línea 1 y 2 del pseudocodigo del Algoritmo de Dijkstra.

```
1. public function getShortestPath($graphs, $vinicial, $vdestino)
2. {
```

En la línea 1,2 y 3 se inicializan los arreglos que almacenaran el listado de vértices del Grafo, la Matriz de Adyacencia y la ruta más corta, se procede a recorrer el listado de vértices del Grafo para almacenar los vértices en el arreglo de vértices, asimismo se procede a construir la matriz de adyacencia siguiendo las condiciones establecidas a continuación.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

$$A[i][j] = \begin{cases} Wij, cuando < vi, vj > \epsilon E \\ \alpha, de\ lo\ contrario \end{cases}$$

La matriz de adyacencia deberá cumplir dos condiciones:

1: Debe existir un arco W_{ij}, siempre y cuando $vi\ y\ vj$, estén conectados por una arista $< vi, vj > \epsilon E$, en donde E representa el conjunto de aristas de un grafo $G = (V, E)$.

2: De lo contrario, no se considera adyacente por lo tanto se interpretara como α.

```
1. $vertices = [];
2. $neighbours = [];
3. $path = [];
4.
5. foreach($graphs as $edge)
6. {
7.   array_push($vertices, $edge->punto1, $edge->punto2);
8.
9.   $neighbours[$edge->punto1][] = array(
10.        'endEdge' => $edge->punto2,
11.        'cost' => $edge->costo
12.      );
13.      }
```

El algoritmo itera todos los vértices del Grafo e inicializa todos los costos de la matriz a 0 asimismo todos los vértices previos del actual los inicializa a NULL, debido que aún no han sido recorridos por lo tanto no se sabe cuál es el vértice previo del analizado. Sin embargo en la línea 9 al ser el vértice inicial el comienzo de la ruta, se asume que empieza su costo en 0, así que una vez iterado todos vértices e inicializado los costos a INF, se inicializa el costo del vértice inicial a 0, el presente código es equivalente a las líneas 4 - 9 del pseudocodigo del Algoritmo de Dijkstra.

```
1. $vertices = array_unique($vertices);
2.
3. foreach($vertices as $vertex)
4. {
5.    $dist[$vertex] = INF;
6.    $previous[$vertex] = NULL;
7. }
8.
9. $dist[$vinicial] = 0;
10.      $g = $vertices;
```


Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

El siguiente código el Algoritmo de Dijkstra realiza una validación previa, verifica que el número de vértices que conforman la matriz de adyacencia sean mayores que cero es decir que existan vértices a recorrer. Lo siguiente que realiza es declarar una variable bandera e inicializar a INF, que será útil más adelante, luego se realiza una iteración de todos los vértices y se realiza una validación, verificando la distancia del vértice iterado es menor que INF, entonces la distancia de dicho vértice se almacena en la variable bandera y el vértice iterado en una nueva variable por ende obliga a que realice el cálculo de la ruta más corta a partir de vértice inicial debido que es el único inicializado en 0 hasta el momento y cumple con la condición.

$$-\alpha < D[i,j] < \alpha, \forall i,j \in C := \{1, \dots, n\}$$

$$with\ D[i,j] = \alpha, \forall i \in C.$$

Se cumplen dos condiciones:

3: La matriz de adyacencia debe ser mayor que menos infinito y menor que más infinito para todo i, j perteneciente al conjunto de vértices. 4: Con la matriz de adyacencia igual que infinito, se interpreta como una indicación que no hay un enlace directo del nodo i al nodo j, para toda i perteneciente al conjunto C.

En la línea 12 realiza una comprobación de la restricción del Algoritmo de Dijkstra sobre el Grafo la cual es verificar si la distancia del vértice iterado perteneciente a la matriz de adyacencia es Infinito, si lo fuese terminaría el proceso, el presente código es equivalente a las líneas 12 - 14 del pseudocodigo del Algoritmo de Dijkstra.

```
1. while (count($g) > 0) {
2.         $min = INF;
3.         foreach($g as $vertex) {
4.             if ($dist[$vertex] < $min) {
5.                 $min = $dist[$vertex];
6.                 $u = $vertex;
7.             }
8.         }
9.
10.        $g = array_diff($g, array($u));
11.
12.        if ($dist[$u] == INF or $u == $vdestino) {
13.             break;
14.        }
```


Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

En la línea 15 en adelante verifica si está definida la matriz de adyacencia si lo está empieza a almacenar el costo actual del vértice iterado más el costo del arco hacia su destino y procede a realizar la verificación si el costo acumulado es menor que el costo del vértice destino entonces el costo acumulado se almacena en el costo del vértice destino, además se procede asignar el vértice actual como vértice procedente o previo del vértice destino. Equivalentes a las líneas 16 - 19 del pseudocodigo del Algoritmo de Dijkstra.

```
15.
16.         if (isset($neighbours[$u])) {
17.             foreach($neighbours[$u] as $arr)
18.             {
19.                 $alt = $dist[$u] + $arr["cost"];
20.
21.                 if ($alt < $dist[$arr["endEdge"]]) {
22.                     $dist[$arr["endEdge"]] = $alt;

23.                     $previous[$arr["endEdge"]] = $u;
24.                 }
25.             }
26.         }
27.     }
```

En la línea 1 se almacena el vértice destino en una variable, luego se verifica (línea 2 – 5) si está definido el vértice previo del vértice destino. Si lo está el vértice destino se agrega al inicio del arreglo que contendrá la ruta más corta, enseguida se almacena el vértice previo del vértice destino para iterar nuevamente para así poder obtener toda ruta de la trayectoria. Luego de ello (línea 10 - 14) se realiza una iteración para obtener las coordenadas de cada vértice que conforma la ruta más corta, posteriormente en la línea 17-21 se almacena en el arreglo dos cabeceras la primera es “paths” el cual es un arreglo con todos los vértices con sus respectivas coordenadas de la ruta más corta. Y la segunda cabecera es “cost” el cual contiene el costo total de la trayectoria.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

```
1. $u = $vdestino;
2. while (isset($previous[$u])) {
3.     array_unshift($path, $u);
4.     $u = $previous[$u];
5. }
6.
7. array_unshift($path, $u);
8. $pathcomplete = [];
9.
10.      foreach($path as $data){
11.          $stmt = $this->objPdo->prepare("SELECT * FROM marker
   WHERE alias = '$data'");
12.          $stmt->execute();
13.          $pathcomplete[] = $stmt->fetchAll(PDO::FETCH_OBJ);
14.      }
15.
16.      $resultado = [];
17.      $resultado = [
18.         'paths' => $pathcomplete,
19.         'cost' => $dist[end($path)]
20.      ];
21.      return $resultado;
22.      }
```


Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

```
1. public function getShortestPath($graphs, $vinicial, $vdestino)
2. {
3.     $vertices = [];
4.     $neighbours = [];
5.     $path = [];
6.
7.     foreach($graphs as $edge)
8.     {
9.         array_push($vertices, $edge->punto1, $edge->punto2);
10.
11.        $neighbours[$edge->punto1][] = array(
12.            'endEdge' => $edge->punto2,
13.            'cost' => $edge->costo
14.        );
15.      }
16.
17.      $vertices = array_unique($vertices);
18.
19.      foreach($vertices as $vertex)
20.      {
21.          $dist[$vertex] = INF;
22.          $previous[$vertex] = NULL;
23.      }
24.
25.      $dist[$vinicial] = 0;
26.      $g = $vertices;
27.
28.      while (count($g) > 0) {
29.          $min = INF;
30.          foreach($g as $vertex) {
31.             if ($dist[$vertex] < $min) {
32.                 $min = $dist[$vertex];
33.                 $u = $vertex;
34.             }
35.          }
36.
37.          $g = array_diff($g, array($u));
38.
39.          if ($dist[$u] == INF or $u == $vdestino) {
40.              break;
41.          }
42.
```


Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

```
1. if (isset($neighbours[$u])) {
2.    foreach($neighbours[$u] as $arr)
3.    {
4.      $alt = $dist[$u] + $arr["cost"];
5.
6.      if ($alt < $dist[$arr["endEdge"]]) {
7.         $dist[$arr["endEdge"]] = $alt;
8.         $previous[$arr["endEdge"]] = $u;
9.      }
10.      }
11.       }
12.        }
13.      $u = $vdestino;
14.
15.      while (isset($previous[$u])) {
16.          array_unshift($path, $u);
17.          $u = $previous[$u];
18.      }
19.
20.      array_unshift($path, $u);
21.      $pathcomplete = [];
22.
23.      foreach($path as $data){
24.         $stmt = $this->objPdo->prepare("SELECT * FROM marker
   WHERE alias = '$data'");
25.         $stmt->execute();
26.         $pathcomplete[] = $stmt->fetchAll(PDO::FETCH_OBJ);
27.      }
28.
29.      $resultado = [];
30.      $resultado = [
31.         'paths' => $pathcomplete,
32.         'cost' => $dist[end($path)]
33.      ];
34.      return $resultado;
35.      }
```

El algoritmo de Dijkstra retorna un listado de vértices que deben ser recorridos para llegar desde un vértice inicial hasta un vértice final y a su vez también retorna el costo total de la trayectoria más corta.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Código 1.3: Retorna los datos obtenidos del Algoritmo de Dijkstra hacia la aplicación.

```
1.  //Tiempo que inicia el Proceso de cálculo del Algoritmo de Dijkstra
2.  $time_start = microtime(true);
3.  //Instancia la clase Grafo
4.  $grafo = new Grafo();
5.  //Invoca al metodo listar() el cual contiene vértices y arcos del Grafo
6.  $graphs = $grafo->listar();
7.  //Instancia la clase Dijkstra.
8.  $dijkstra = new Dijkstra();
9.
10. //Obtiene el array de alias que alimentaran el algoritmo
11. $array = $_GET['list'];
12. //Inicializa el array que contendrá las cabeceras con la info del costo, destinos, coordenadas y
    tiempo de ejecución
13. $respuesta = [];
14. //Inicializa el costo a 0;
15. $costo = 0;
16. //Inicializa el array que contendrá todos los destinos por donde debe ir el Camión
17. $destinos = [];
18. //Incializa el array de coordenadas
19. $coordenadas = [];
20. for ($c=0; $c < (count($arraynewverticesalias)-1); $c++) {
21.     //$x esta variable servirá para obtener el segundo vértice( vértice destino)
22.     $csgte=$c+1;
23.     //Invoca al método getShortestPath() recibiendo 03 args (grafo, Vinicio, Vdestino)
24.     $path2 = $dijkstra-
    >getShortestPath($graphs, $arraynewverticesalias[$c], $arraynewverticesalias[$csgte]);
25.     //Obtener el costo e ir autoincrementando el costo de la ruta más corta
26.     $costo = $costo + $path2['cost'];
27.     //Itero la ruta mas corta para extraer su latitud, longitud y destino
28.     for($i=0; $i < count($path2['paths']); $i++) {
29.         for($j=0; $j < count($path2['paths'][$i]); $j++)  {
30.             for ($k=0; $k < count($path2['paths'][$i][$j]); $k++) {
31.                 //Obtener todo los destino de la ruta mas corta
32.                 $destinos[] = $path2['paths'][$i][$j]->destino;
33.
34.                 //Guardamos todas las coordenadas de la rutas para el polígono
35.                 $coordenadas[] = [
36.                     'longitud' => $path2['paths'][$i][$j]->longitud,
37.                     'latitud' => $path2['paths'][$i][$j]->latitud
38.                 ];
39.             }
40.         }
41.     }
42. }
43.
44. $time_end = microtime(true);
45. //Resta el tiempo para obtener el tiempo de ejecución
46. $time = $time_end - $time_start;
47. //Array que almacena cabeceras e información de cada dato obtenido
48. $respuesta = [
49.     'costo' => $costo,
50.     'destinos' => $destinos,
51.     'coordenadas' => $coordenadas,
52.     'tiempo' => round($time, 3),
53. ];
54.
55. //Echo en los datos de la respuesta
56. echo json_encode($respuesta);
```


Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Código 1.4: Recibe la respuesta del Algoritmo, dibuja la ruta optima en el Mapa y realiza la animación.

```
1.  //Realiza la extracción de la ruta más corta
2.  getShortestPathIn() {
3.      let me = this;
4.      axios.get('controller/ShortestPathController.php', {
5.          params: {
6.              'list': me.arrayAlias,
7.          }
8.      }).then(function(response) {
9.          //El costo lo muestro en una etiqueta HTML
10.         me.costodijkstra = response.data.costo;
11.         //Las direcciones lo muestro en una etiqueta HTML
12.         me.destinosdijkstra = response.data.destinos;
13.         //Coordenadas se agregan en un array de coordenadas
14.         me.coordenadasdijkstra = response.data.coordenadas;
15.         //Tiempo de ejecución del algoritmo de Dijkstra
16.         me.tiempodijkstra = response.data.tiempo;
17.         //Inmediatamente se invoca al método que dibuja la ruta en el Mapa
18.         me.drawShortestPath();
19.     }).catch(function(error) {
20.         console.log(error);
21.     });
22. },
23. //Dibuja la ruta más corta sobre el Mapa
24. drawShortestPath() {
25.     let me = this;
26.     //Se realiza un for de todas las coordenadas para obtener solo la lat, lng
27.     me.coordenadasdijkstra.map(function(coordenadas, i) {
28.         //Array que captura información de las coordenadas y se parseas a float
29.         var locationsPolyline = {
30.             lat: parseFloat(coordenadas.latitud),
31.             lng: parseFloat(coordenadas.longitud),
32.         };
33.         //Agrego los datos de las coordenadas en un nuevo array
34.         me.coordenadasdijkstraclean.push(locationsPolyline);
35.     });
36.
37.     // Defina el símbolo, utilizando una de las rutas predefinidas ('CIRCULO')
38.     // suministrado por la API de JavaScript de Google Maps.
39.     me.lineSymbol = {
40.         path: google.maps.SymbolPath.FORWARD_CLOSED_ARROW,
41.         scale: 6,
42.         strokeColor: '#4db3b8'
43.     };
44.
```


Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

```
1.  //Método que me permite dibujar polígonos en el mapa
2.  //Cree la poli línea y agregue el símbolo a la misma a través de la propiedad
    'iconos'.
3.      me.flightPath = new google.maps.Polyline({
4.          path: me.coordenadasdijkstraclean,
5.          icons: [{
6.              icon: me.lineSymbol,
7.              offset: '100%'
8.          }],
9.          geodesic: true,
10.         strokeColor: '#FF0000',
11.         strokeOpacity: 1.0,
12.         strokeWeight: 3,
13.         fillColor: '#FFC107',
14.         fillOpacity: 0.35
15.     });
16.
17.     me.animateCircle(me.flightPath);
18.     //Los polígonos dibujados los muestro en el mapa
19.     me.flightPath.setMap(me.map);
20. },
21. //Utilice la función DOM setInterval() para cambiar el desplazamiento
22. //del símbolo a intervalos fijos.
23. animateCircle(line) {
24.     var count = 0;
25.     window.setInterval(function() {
26.         count = (count + 1) % 200;
27.
28.         var icons = line.get('icons');
29.         icons[0].offset = (count / 2) + '%';
30.         line.set('icons', icons);
31.     }, 28);
32. },
```


Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

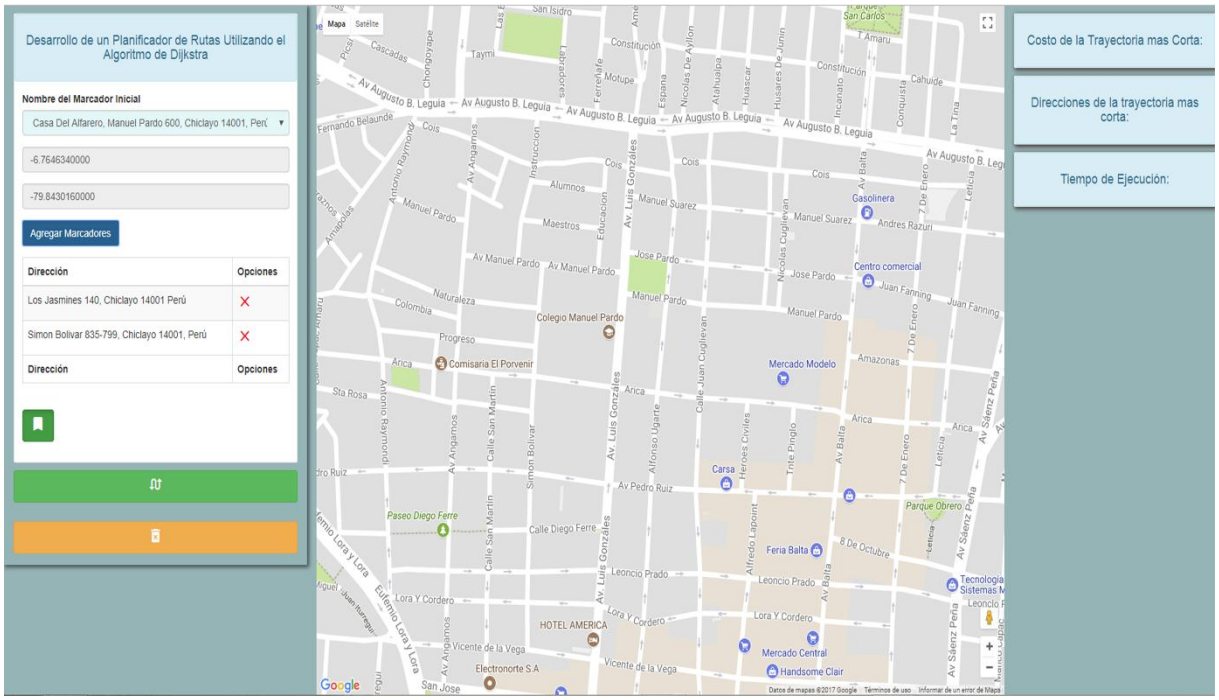


Figura 47: GUI del sistema informático.

Fuente: Elaboración propia

Interfaz gráfica del sistema informático donde se visualiza el planificador de rutas sobre el cual se ejecutara el Algoritmo de Dijkstra, que permitirá seleccionar el punto de partida es decir donde empieza el camión y los puntos de acción que vienen a ser los focos infecciosos, estos parámetros permitirán obtener datos como el costo total de la trayectoria expresados en metros, el número de vértices recorridos y las calles a recorrer asimismo el tiempo de ejecución del Algoritmo, además permitirá visualizar en el mapa del sistema, el camión y los puntos de acción así como también la trayectoria de la ruta más corta.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

CAPITULO VI: CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones

a) El grafo se construyó recopilando información de las calles del distrito de Chiclayo tales como direcciones, coordenadas, intersección de calles, mediante el proveedor de mapas Google Maps a través de una Matriz de Adyacencia, para posteriormente aplicar el Algoritmo de Dijkstra.

b) Se aplicó El algoritmo de Dijkstra para resolver el problema de la planificación de rutas de recojo de desechos sólidos, utilizando como fuente de datos un Grafo dirigido ponderado mediante una matriz de adyacencia, tomando en cuenta los focos infecciosos y el punto de partida, combinado con el lenguaje de programación PHP para darle funcionalidad.

c) Se concluye que el algoritmo de Dijkstra sobre un planificador de rutas con un computador con procesador Core i7 a partir de 20 focos infecciosos, comienza a mostrar una tendencia de mejora con un tiempo de ejecución de 1.518 seg en comparación al computador con procesador Core i5 con un tiempo de ejecución de 1.663 seg. Sin embargo el costo total de la trayectoria en cada caso es invariable para los 7 casos de prueba mostrando 799m, 1922, 2949, 4572, 5133, 5044 y 5462m para ambos procesadores, por lo que el costo de la trayectoria es indiferente del procesador en el que se ejecute, observando que lo único que varía es el tiempo de ejecución, siendo así confiable el planificador de rutas.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

6.2. Recomendaciones

a) Se recomienda a los futuros investigadores implementar mejoras en el algoritmo de Dijkstra para poder observar de cerca el comportamiento del algoritmo respecto a la planificación de rutas, incorporando diferentes factores como la congestión del tráfico vehicular, modos de transporte, número de semáforos, impedancia en la intersección de calles, estado de las calles, consumo de combustible, entre otras.

b) Se recomienda a los futuros investigadores implementar diferentes algoritmos de búsqueda de rutas cortas, bajo el mismo escenario del problema expresado en la presente investigación y que realicen comparaciones entre ellas.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

REFERENCIAS

Alonso Revenga, J. M. (2008). *Flujo en redes y gestión de proyectos. Teoría y ejercicios resueltos.* España: Netbiblo.

Barrantes Ríos, W. A. (2015). *Implementación de un algoritmo recocidos simulado para el diseño de rutas de transporte público para Lima Centro.* Lima, Perú: Pontificia Universidad Católica del Perú. Obtenido de http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/5961/BARRANTES_WALTER_IMPLEMENTACION_ALGORITMO_RECOGIDO_SIMULADO.pdf?sequence=1&isAllowed=y

Caicedo Barrero, A., Wagner de García, G., & Méndez Parra, R. (2010). *Introducción a la teoría de grafos.* Armenia, Colombia: Ediciones Elizcom.

Chen, Y.-Z., Shen, S.-f., Chen, T., & Rui, Y. (2014). *Path optimization study for vehicles evacuation based on dijkstra algorithm.* Beijing, China. Obtenido de https://doi.org/10.1016/j.proeng.2014.04.023

Contreras Santos, M. J. (2008). *Evaluación de experiencias locales urbanasdesde el concepto de sostenibilidad: el caso de los desechos sólidos del municipio de Los Patios (Norte de Santander. Colombia).* Bogotá, Colombia: Trabajo Social.

Coto, E. (2003). *Algoritmos básicos de grafos.* Caracas, Venezuela: Universidad Central de Venezuela. Obtenido de http://lcg.ciens.ucv.ve/~ernesto/nds/CotoND200302.pdf

Cusco Tenesaca, J. W., & Picón Aguirre, K. E. (2015). *Optimización de rutas de recolección de desechos sólidos Domiciliarios mediante uso de herramientas SIG.* Cuenca, Ecuador. Obtenido de http://dspace.ucuenca.edu.ec/handle/123456789/21304

De la Cruz, J. C., Magwili, G. V., Mundo, J. P., Gregorio, G. P., Lamoca, M. L., & Villaseñor, J. A. (2016). *Items-mapping and route optimization in a grocery store using Dijkstra's, Bellman-Ford and Floyd-Warshall Algorithms.* Obtenido de https://doi.org/10.1016/j.proeng.2014.04.023

Fernández-Peteiro Belmonte, M. (2016). *Desarrollo de algoritmos de planificación de caminos en 3D para Blender y V-REP.* Sevilla, España: Universidad de Sevilla. Obtenido de https://idus.us.es/xmlui/bitstream/handle/11441/53158/TFG_Miguel_Fern%C3%83%C2%A1ndez_Peteiro_Belmonte.pdf?sequence=1

Gary W., H., & J.Glynn, H. (1999). *Ingeniería Ambiental.* México: Prentice Hall.

Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, M. D. (2014). *Metodología de la Investigación.* Mexico: McGRAW-HILL .

Indrajaya, M. A., Affandi, A., & Pratomo, I. (2015). Design of geographic information system for tracking and routing using Dijkstra algorithm for public transportation. *Wireless and Telematics (ICWT), 1st international conference on.* Manado.

Instituto Nacional de Estadística y Geografía. (2014). *Sistema de información geografica.* Obtenido de http://www.inegi.org.mx/inegi/SPC/doc/internet/sistemainformaciongeografica.pdf

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Joyanes Aguilar, L., Rodríguez Baena, L., & Fernández Azuela, M. (2000). *Fundamentos de programación, libro de problemas.* Madrid, España: S.A. Mc Graw-Hill. Obtenido de http://www.compuclasico.com/pl/McGraw_Hill-Luis_Joyanes_Aguilar-Fundamentos_de_Programacion_Libro_de_Problemas.pdf

Lucero Pastás, J. E., & Viñamagua Carrión, J. B. (2016). *Diseño de un sistema de recolección de residuos sólidos en el Cantón Cayambe.* Quito, Ecuador: Universidad Central del Ecuador.

Marchena Pejerrey, D. J. (2015). *Desarrollo de un sistema para la optimización de rutas de trabajo utilizando el algoritmo de Dijkstra y diagramas de Voronoi.* Pimentel, Perú: Universidad Señor de Sipán. Obtenido de http://repositorio.uss.edu.pe/handle/uss/157

Meza, O., & Ortega, M. (2006). *Grafos y algoritmos.* Caracas, Venezuela: Equinoccio.

Mingjun , W., & Meng, Y. (2014). *Research On The Optimal Route Choice Based On Improved Dijkstra.* Ottawa, Canada: IEEE. doi:Research On The Optimal Route Choice Based On Improved Dijkstra

Moreno, E., & Ramirez, H. (2011). *Grafos:Fundamentos y Algoritmos* . JC Saez Editor.

OEFA. (2014). *Fiscalización ambiental en residuos sólidos de gestión municipal provincial.* Organismos de Evaluación y fiscalización ambiental. Lima: Deposito Legal del Perú.

Organización de las Naciones Unidas. (1992). Conferencia mundial sobre el medio ambiente y el desarrollo. *Agenda 21.*

Organización de las Naciones Unidas para el Desarrollo Industrial. (2007). *Guia para la gestión integral de los residuos sólidos urbanos.* ONUDI. Obtenido de https://www.unido.org/fileadmin/import/72852_Gua_Gestin_Integral_de_RSU.pdf

Organizacion para el desarrollo sostenible, & Ambiente y desarrollo sostenible S.A.C. (2013). *Plan integral de gestión ambiental de residuos sólidos de la provincia de Chiclayo, departamento de Lambayeque.* Perú, Perú: Universidad Nacional de Ingeniería. Obtenido de https://www.munichiclayo.gob.pe/Documentos/PIGARSChiclayo.pdf

Pérez Navarro, A. (2011). *Introducción a los sistemas de información geográfica y geotelemática.* Barcelona, España: UOC. Obtenido de https://books.google.com.pe/books?id=xip1wtr8k58C&printsec=frontcover&dq=Sistema+de+informacion+geografica+(SIG)&hl=es-419&sa=X&redir_esc=y#v=onepage&q=Sistema de informacion geografica (SIG)&f=false

Real academia española. (2014). *RAE.es*. Obtenido de Algoritmo: http://dle.rae.es/?id=1nmLTsh

Rodríguez Salas, J. J., Santamaría Arana, L., Rabasa Dolado, A., & Martínez Bonastre, O. (2003). *Introducción a la programación. Teoría y práctica.* Alicante, España: Club Universitario. Obtenido de https://s3.amazonaws.com/academia.edu.documents/38609102/introduccion_a_la_programacion_teoria_y_practica.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1505184289&Signature=aXxL4%2FKrmlyBLmPbaudbEq3z5hw%3D&response-content-disposition=inline%3B%20filena

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Rosyidi, L., Pram Pradityo, H., & Gunawan, D. (2014). *Timebase dynamic weight for Dijkstra Algorithm implementation in route planning software.* Taiwan, China: IEEE. doi:10.1109/IGBSG.2014.6835261

Ruiz Liza, I. J., & Vidal, U. W. (2016). *Modelo de optimización del sistema de recojo de residuos sólidos en el distrito de Reque para mejorar la eficiencia de operaciones.* Pimentel, Perú: Universidad Señor de Sipán.

Ruohonen, K. (Marzo de 2013). *Graph Theory.* Obtenido de http://math.tut.fi/~ruohonen/GT_English.pdf

Sato Yamada, J. M. (2017). *Implementación de algoritmos meta heurísticos para la definición de frecuencias y horarios para rutas de transporte público.* Lima, Perú: Pontificia Universidad Católica del Perú. Obtenido de http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/9244/SATO_JOS%C3%89_ALGORITMOS_HEURISTICOS_FRECUENCIA_TRANSPORTE_PUBLICO.pdf?sequence=1&isAllowed=y

Say Chamán, A. B. (2005). *Manejo de la basura y su clasifiación.* Guatemala. Obtenido de http://biblioteca.usac.edu.gt/EPS/07/07_1989.pdf

Taquía Valdivia, J. A. (2013). *Optimización de rutas en una empresa de recojo de residuos sólidos en el distrito de los Olivos.* Lima, Perú: Pontificia Universidad Católica del Perú. Obtenido de http://tesis.pucp.edu.pe/repositorio/handle/123456789/4603

Zhang , X., Chen, Y., & Li, T. (2015). Optimization of logistics route based on Dijkstra. *Software Engineering and Service Science (ICSESS), 2015 6th IEEE International Conference on.* Beijing. doi:10.1109/ICSESS.2015.7339063

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

ANEXOS

Anexo 1 - Guía de Observación científica.

Resultados obtenidos del planificador de rutas, tales como el vértice inicial, los vértices destino, el costo de la trayectoria y el tiempo de ejecución del algoritmo.

Computador: *Computador con procesador Corei7.*	***Responsable:*** *Anton Bernal Juan Manuel* ***Fecha:*** *26 de septiembre de 2017* ***Institución:*** *Universidad Señor de Sipán.*

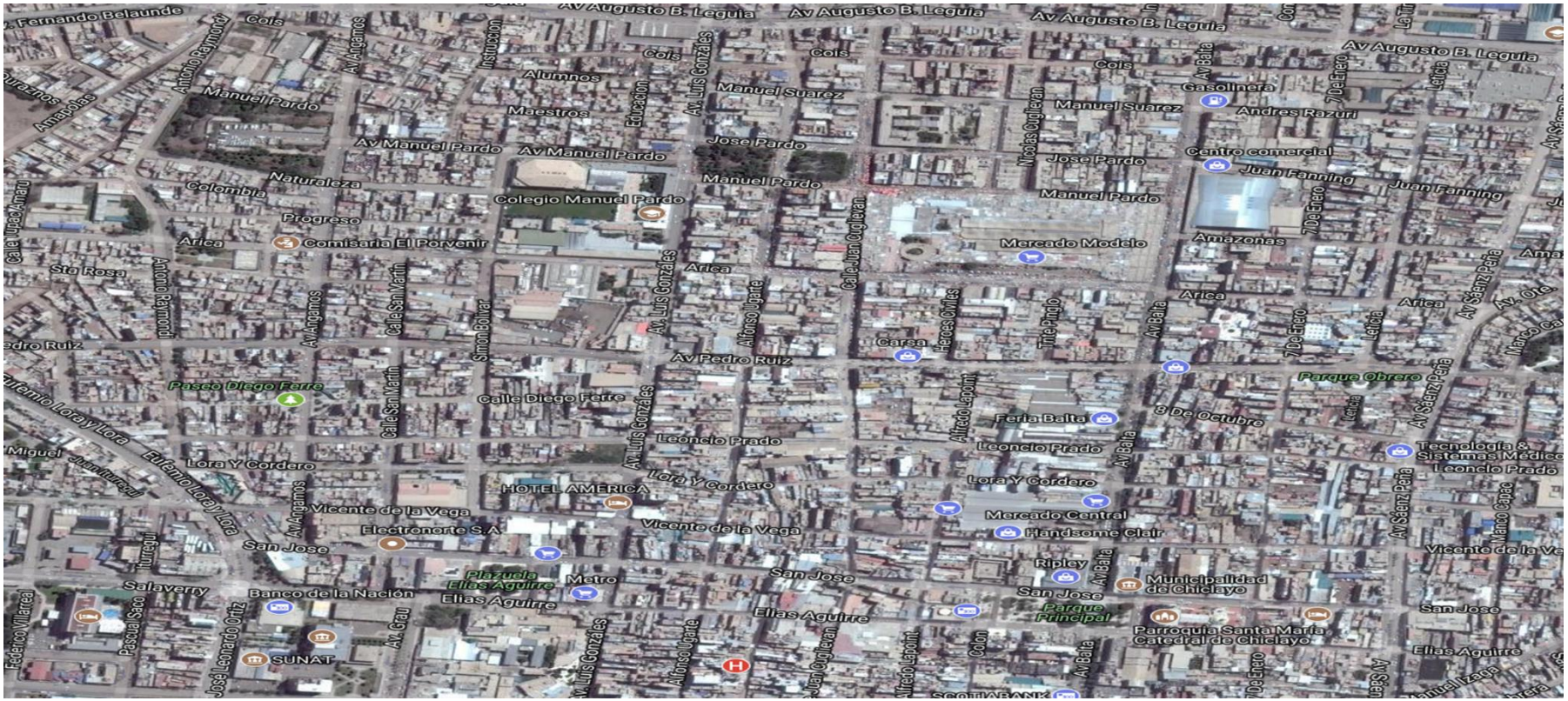
Código	***Vértice Inicial***	***Vértices Destino.***	***Costo (m.)***	***Tiempo Ejecución (s)***
001	*v16*	*v73*	*799 m.*	*0.052 s.*
002	*v18*	*v13 – v4 – v82 – v22- v85*	*1922 m.*	*0.221 s.*
003	*v50*	*v79 – v31 – v49 – v41 – v57 – v64 – v71 – v85 – v29 – v4*	*2949 m.*	*0.529 s.*
004	*v64*	*v79 - v15 – v39 – v87 – v94 – v85 – v67 – v27 – v80 – v60 – v13 – v89 – v41 – v30 –v53*	*4572 m.*	*1.030 s*
005	*v55*	*v57 – v63 – v58 – v78 – v2 – v3 – v51 – v99 – v20 – v32 – v8 – v75 – v94 – v4 – v59 – v83 – v85 – v88 – v35 – v11*	*5133 m.*	*1.518 s.*
006	*v10*	*v21 – v28 – v17 – v11 – v8 – v70 – v44 – v87 – v27 – v56 – v97 – v52 – v93 – v64 – v94 – v20 – v73 – v92 – v23 – v78 – v98 – v39 – v35 – v36 – v41*	*5044 m,*	*2.282 s.*
007	*v87*	*v32 – v99 – v11 – v47 – v21 – v66 – v40 – v39 – v82 – v22 – v48 – v98 – v38 – v50 – v19 – v57 – v9 – v96 – v77 – v52 – v94 – v13 – v41 – v4 – v25 – v12 – v64 – v1 – v10 – v70*	*5462 m.*	*3.158 s.*

Computador: *Computador con procesador Corei5.*	***Responsable:*** *Anton Bernal Juan Manuel* ***Fecha:*** *26 de septiembre de 2017* ***Institución:*** *Universidad Señor de Sipán.*

Código	***Vértice Inicial***	***Vértices Destino.***	***Costo (m.)***	***Tiempo Ejecución (s)***
001	*v16*	*v73*	*799 m.*	*0.048 s.*
002	*v18*	*v13 – v4 – v82 – v22- v85*	*1922 m.*	*0.219 s.*
003	*v50*	*v79 – v31 – v49 – v41 – v57 – v64 – v71 – v85 – v29 – v4*	*2949 m.*	*0.535 s.*
004	*v64*	*v79 - v15 – v39 – v87 – v94 – v85 – v67 – v27 – v80 – v60 – v13 – v89 – v41 – v30 –v53*	*4572 m.*	*1.038 s*
005	*v55*	*v57 – v63 – v58 – v78 – v2 – v3 – v51 – v99 – v20 – v32 – v8 – v75 – v94 – v4 – v59 – v83 – v85 – v88 – v35 – v11*	*5133 m.*	*1.663 s.*
006	*v10*	*v21 – v28 – v17 – v11 – v8 – v70 – v44 – v87 – v27 – v56 – v97 – v52 – v93 – v64 – v94 – v20 – v73 – v92 – v23 – v78 – v98 – v39 – v35 – v36 – v41*	*5044 m,*	*2.680 s.*
007	*v87*	*v32 – v99 – v11 – v47 – v21 – v66 – v40 – v39 – v82 – v22 – v48 – v98 – v38 – v50 – v19 – v57 – v9 – v96 – v77 – v52 – v94 – v13 – v41 – v4 – v25 – v12 – v64 – v1 – v10 – v70*	*5462 m.*	*4.003 s.*

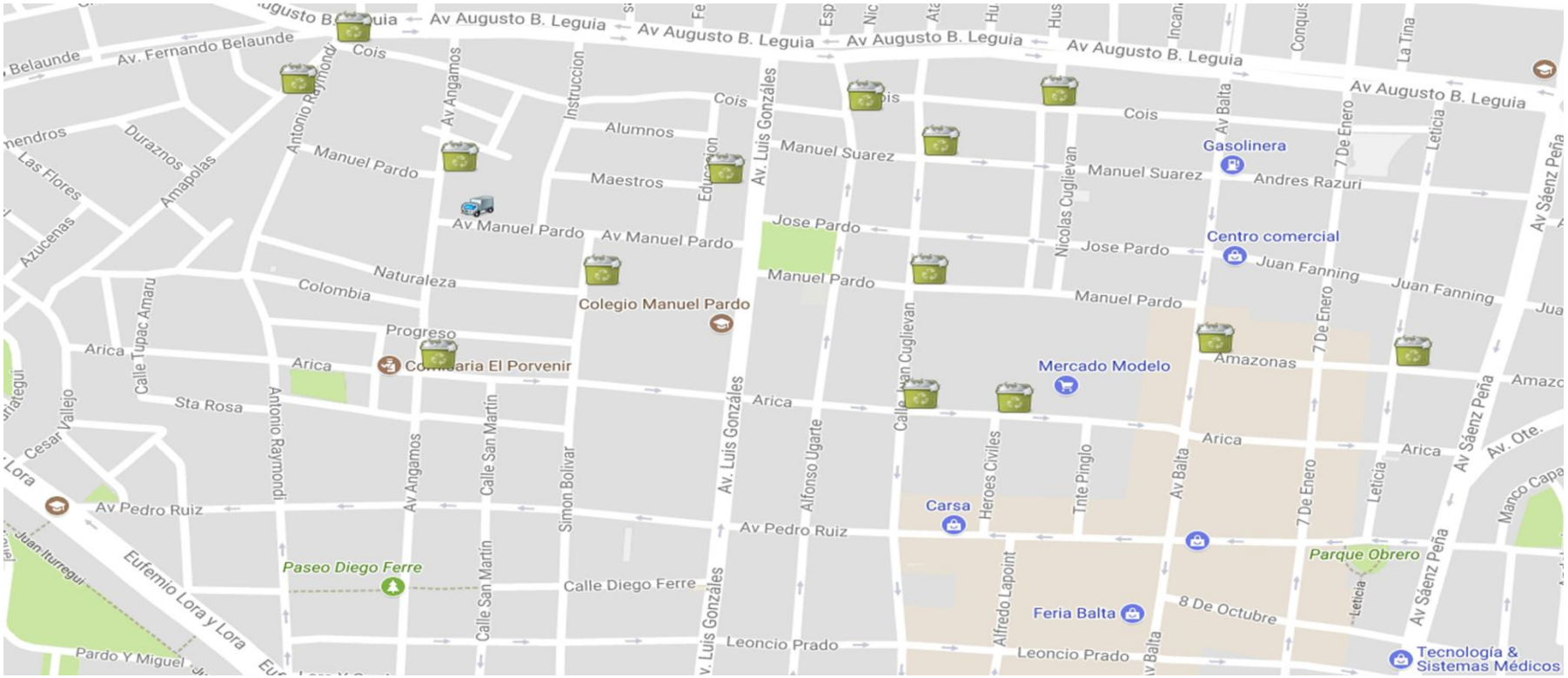
Anexo 2 - Sector del Mapa.

Sector del mapa extraído del distrito de Chiclayo, sobre el cual se realizara la simulación del Algoritmo de Dijkstra sobre un planificador de rutas.



Anexo 3 - Mapa con focos infecciosos.

Simulación del mapa del sistema informático con los puntos de acción o focos infecciosos sobre el cual se aplicara posteriormente el Algoritmo de Dijkstra haciendo uso del API V3 de Google Maps.



Anexo 4 - Arcos que conforman el Grafo.

Datos obtenidos del proveedor de aplicaciones de mapas Google Maps que conforman el Grafo Dirigido ponderado mediante una matriz de adyacencia.

ARCO	INICIO	FIN	COSTO (m.)
6	1	10	23
7	2	1	50
8	2	4	63
9	3	2	151
10	3	5	66
11	4	2	63
12	4	5	149
13	4	6	61
14	5	3	66
15	5	4	149
16	5	7	60
17	6	4	61
18	6	7	137
19	6	8	26
20	7	5	60
21	7	6	137
22	7	9	46
23	8	6	26
24	8	9	135
25	8	16	52
26	9	7	46
27	9	8	135
28	10	37	56
29	37	10	56
30	11	12	45
31	11	13	35
32	12	11	45
33	12	14	35
34	13	11	35
35	13	14	56
36	13	15	40
37	14	12	35
38	14	13	56
39	14	16	43
40	15	13	40
41	15	16	69
42	16	14	43
43	16	15	69
44	16	8	52
45	10	17	75
46	17	10	75
47	17	18	150
48	18	17	150
49	18	19	72
50	19	18	72

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

51	17	33	31
52	33	17	31
53	18	21	62
54	21	18	62
55	20	21	147
56	21	20	147
57	20	22	50
58	22	20	50
59	21	23	48
60	23	21	48
61	22	24	70
62	24	23	78
63	23	25	154
64	25	26	151
65	26	27	35
66	27	3	50
67	28	22	41
68	29	20	40
69	20	29	40
70	28	29	50
71	29	28	50
72	30	28	98
73	31	29	111
74	29	31	111
75	31	30	47
76	30	31	47

77	33	20	31
78	20	33	31
79	32	33	173
80	33	32	173
81	31	32	59
82	32	31	59
83	19	2	47
84	34	17	171
85	17	34	171
86	32	34	46
87	34	32	46
88	34	35	115
89	35	34	115
90	35	36	112
91	36	35	112
92	37	11	44
93	11	37	44
94	35	37	142
95	37	35	142
96	36	38	12
97	38	36	12
98	36	39	69
99	39	36	69
100	39	40	66
101	40	39	66
102	40	41	57

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

103	41	40	57
104	41	42	40
105	42	41	40
106	42	43	51
107	43	42	51
108	43	44	67
109	44	43	67
110	45	42	60
111	42	45	60
112	44	46	24
113	46	44	24
114	46	47	43
115	47	46	43
116	47	48	28
117	48	47	28
118	32	48	64
119	48	32	64
120	15	38	115
121	38	15	115
122	44	49	55
123	49	44	55
124	49	50	51
125	50	49	51
126	50	51	50
127	51	50	50
128	51	52	19

129	52	51	19
130	30	52	130
131	52	30	130
132	27	54	33
133	54	55	89
134	55	53	44
135	9	53	49
136	53	9	49
137	25	60	78
138	60	59	159
139	26	59	82
140	59	26	82
141	59	58	63
142	58	57	87
143	57	56	45
144	55	57	87
145	56	53	77
146	3	19	100
147	53	56	77
148	56	67	22
149	57	66	65
150	59	64	83
151	60	63	80
152	61	60	150
153	62	61	84
154	63	62	140

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

155	63	73	90
156	64	59	83
157	64	63	150
158	64	72	110
159	65	58	69
160	65	64	64
161	66	65	89
162	66	70	110
163	67	56	22
164	67	68	68
165	68	66	72
166	68	67	68
167	68	69	120
168	69	68	120
169	69	70	69
170	69	85	16
171	70	69	69
172	70	71	89
173	70	100	28
174	71	65	110
175	71	70	89
176	71	72	60
177	72	64	110
178	72	71	60
179	72	98	26
180	73	74	140

181	73	77	87
182	74	62	85
183	74	73	140
184	75	74	26
185	76	75	63
186	76	77	140
187	77	76	140
188	77	79	94
189	78	76	87
190	79	78	130
191	79	90	120
192	80	79	92
193	80	89	120
194	81	80	60
195	81	98	140
196	82	81	63
197	82	99	140
198	83	82	91
199	83	87	120
200	84	86	120
201	84	101	140
202	85	69	16
203	85	101	16
204	86	97	86
205	87	86	65
206	87	96	81

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

207	88	82	120
208	88	87	120
209	89	80	120
210	89	88	120
211	89	94	71
212	90	89	95
213	90	93	68
214	91	78	130
215	91	90	120
216	92	91	62
217	93	92	110
218	94	89	71
219	94	93	93
220	95	88	77
221	95	94	100
222	96	95	110
223	97	86	86
224	97	96	72
225	98	72	26
226	98	81	140
227	98	99	59
228	99	71	27
229	99	98	59
230	99	100	90
231	100	83	140
232	100	99	90

233	100	101	70
234	101	84	140
235	101	85	16
236	101	100	70

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Anexo 5 - Manual de Usuario.

La interfaz principal de la plataforma del sistema planificador de rutas, es la pantalla donde el administrador del sistema podrá seleccionar el punto de partida del camión así como también los puntos de acción o focos infecciosos por donde el camión debe realizar el recojo de los desechos sólidos.

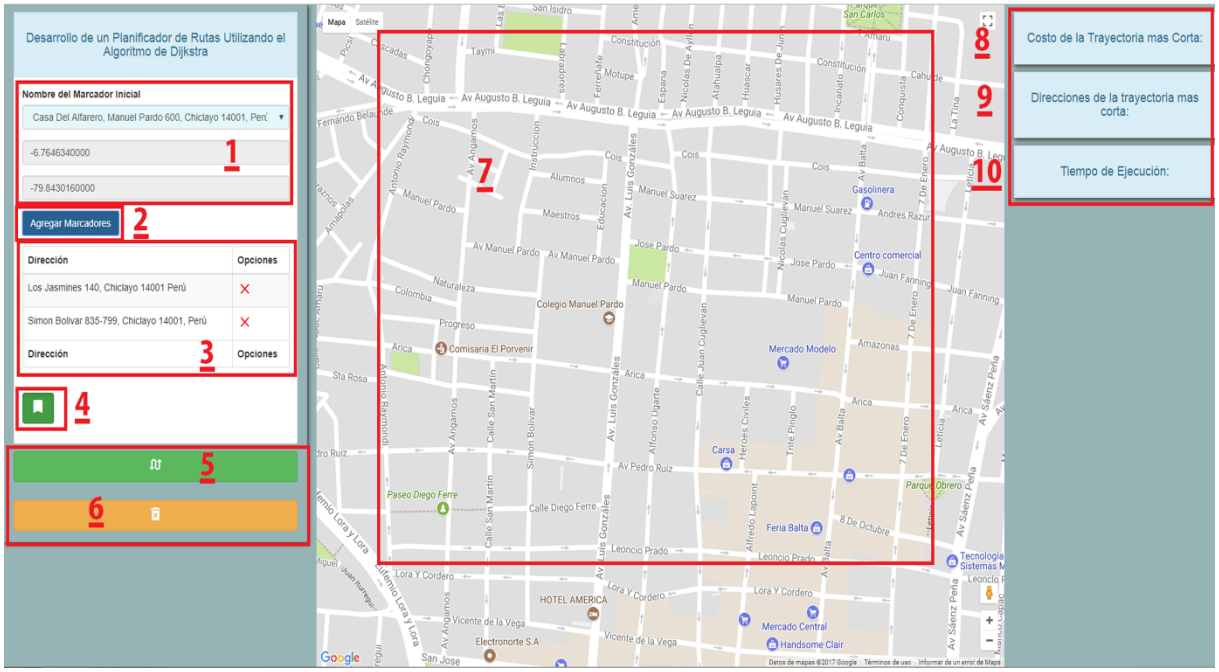


Figura 48: GUI del sistema informático.

Fuente: Elaboración propia

En esta pantalla puede observarse 10 secciones:

1. Sección 01:

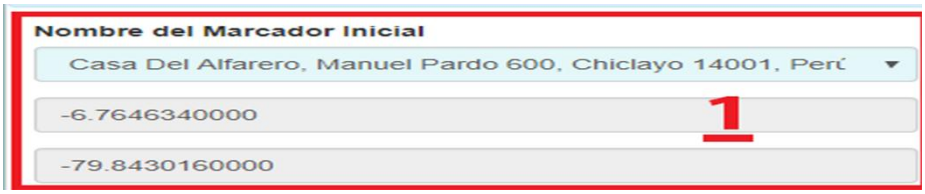


Figura 49: Selector del punto de partida**.**

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

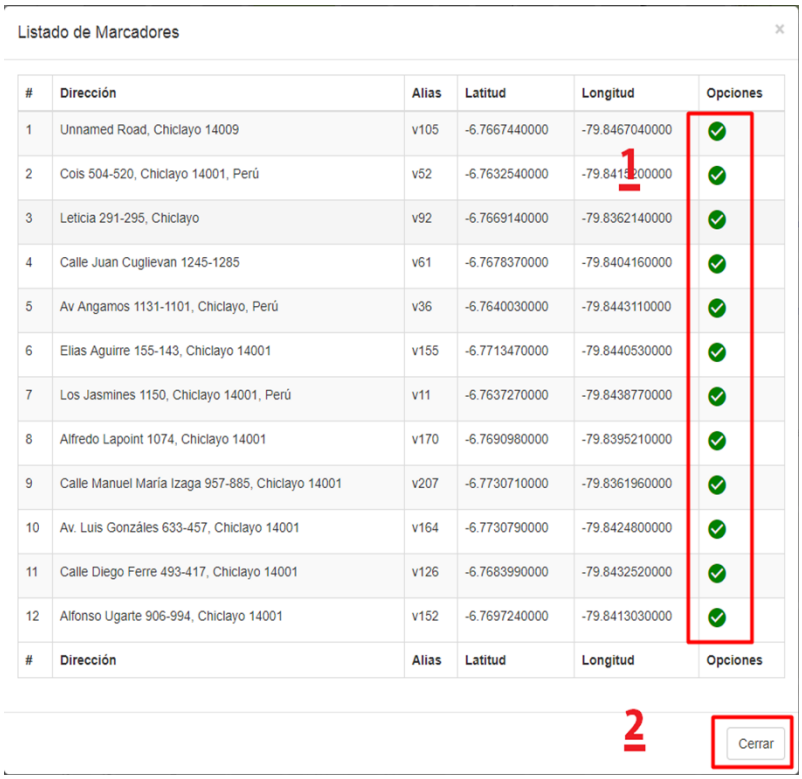
a. Desplegar el combo con el listado ubicaciones.
b. Seleccionar el lugar en donde será el punto de partida del camión.
c. Una vez seleccionado el lugar del punto de partida, automáticamente se cargaran las coordenadas de dicho lugar, para posteriormente realizar localización del camión sobre el mapa del sistema.

2. Sección 02:

Figura 50: Botón despliega focos infecciosos.

Fuente**:** Elaboración propia

a. Presionar en el botón, se abrirá una ventana emergente con un listado de puntos de acción.



#	Dirección	Alias	Latitud	Longitud	Opciones
1	Unnamed Road, Chiclayo 14009	v105	-6.7667440000	-79.8467040000	
2	Cois 504-520, Chiclayo 14001, Perú	v52	-6.7632540000	-79.8418200000	
3	Leticia 291-295, Chiclayo	v92	-6.7669140000	-79.8362140000	
4	Calle Juan Cuglievan 1245-1285	v61	-6.7678370000	-79.8404160000	
5	Av Angamos 1131-1101, Chiclayo, Perú	v36	-6.7640030000	-79.8443110000	
6	Elias Aguirre 155-143, Chiclayo 14001	v155	-6.7713470000	-79.8440530000	
7	Los Jasmines 1150, Chiclayo 14001, Perú	v11	-6.7637270000	-79.8438770000	
8	Alfredo Lapoint 1074, Chiclayo 14001	v170	-6.7690980000	-79.8395210000	
9	Calle Manuel María Izaga 957-885, Chiclayo 14001	v207	-6.7730710000	-79.8361960000	
10	Av. Luis Gonzáles 633-457, Chiclayo 14001	v164	-6.7730790000	-79.8424800000	
11	Calle Diego Ferre 493-417, Chiclayo 14001	v126	-6.7683990000	-79.8432520000	
12	Alfonso Ugarte 906-994, Chiclayo 14001	v152	-6.7697240000	-79.8413030000	
#	Dirección	Alias	Latitud	Longitud	Opciones

Figura 51: Ventana emergente con los focos infecciosos.

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

b. Se seleccionará el/los puntos de acción que desea recorrer el camión recolector de desechos sólidos.
c. Automáticamente se van agregando a la sección 03.

3. Sección 03:

Figura 52: Visor de focos infecciosos agregados

Fuente: Elaboración propia

a. Permite visualizar los puntos de acción seleccionados.

4. Sección 04:

Figura 53: Agregar marcadores sobre el mapa del sistema.

Fuente: Elaboración propia

a. Presionar botón, para agregar el punto de partida y los puntos de acción sobre el mapa del sistema.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

5. Sección 05:

Figura 54: Botón que genera la ruta más corta sobre el mapa del sistema.

Fuente: Elaboración propia

a. Botón que permite generar la ruta más corta sobre el planificador de rutas, visualizado en el mapa del sistema.

6. Sección 06:

Figura 55: Botón que borrar los marcadores del mapa del sistema

Fuente: Elaboración propia

a. Botón que permite borrar todos los marcadores visualizados en el mapa del sistema.

7. Sección 07:

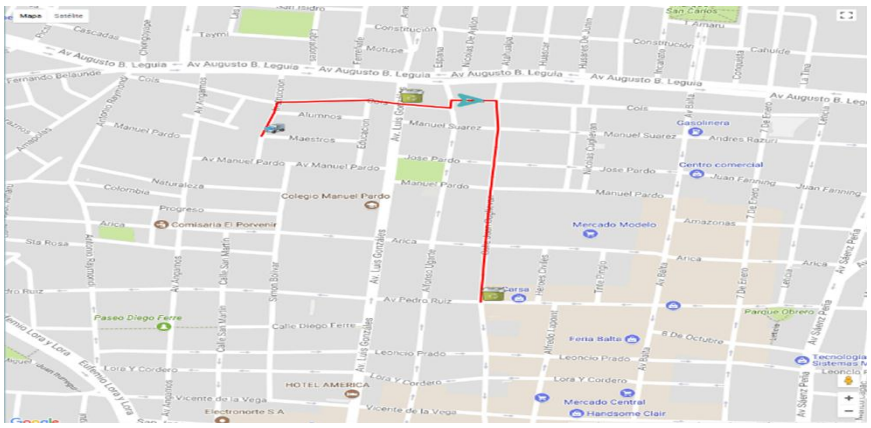


Figura 56: Mapa del sistema planificador de rutas.

Fuente: Elaboración propia

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

a. Mapa del sistema donde se visualiza el camión recolector, los puntos de acción y la simulación de la trayectoria óptima.

8. Sección 08:

Figura 57: Resultados del algoritmo de Dijkstra sobre el planificador.

Fuente: Elaboración propia

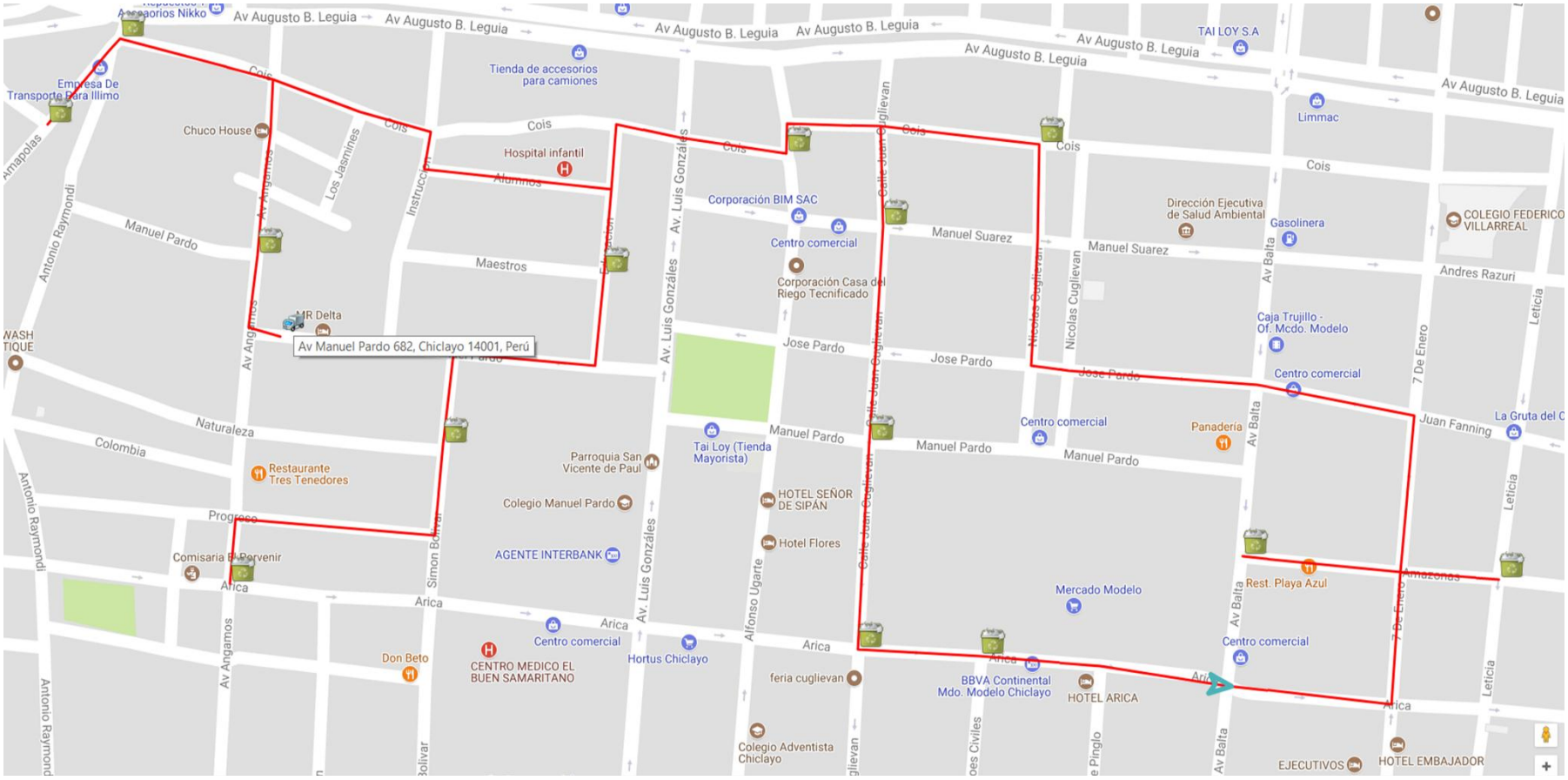
a. Permite visualizar el costo total de la trayectoria, los vértices por donde debe pasar, el número de vértices recorridos y el tiempo de ejecución del algoritmo.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Anexo 6 – Mapa del sistema con la planificación de la ruta más corta de 14 focos infecciosos.

Simulación del algoritmo de Dijkstra generando la ruta más corta, visualizado sobre el mapa del sistema haciendo uso del API V3 del Google Maps.



Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Anexo 7

Para el presente proyecto de investigación se tomaron en cuenta las características de los algoritmos de optimización de rutas aplicados sobre Grafos y se pondero los algoritmos de optimización de rutas. Los criterios de evaluación fueron los siguientes:

Nivel de Impacto	Descripción.
Cumple	Si
No cumple	No

Fuente: Elaboración propia.

Características	Algoritmos									
	A. Warshall	A. Búsqueda Profundidad-Primero	A. Búsqueda Amplitud-Primero	A. Dijkstra	A. Floyd	A. Kruskal	A. Prim	A. Recocido	A. Húngaro	A. Ford-Fulkerson
Evalúa los nodos una sola vez.	No	No	Si	No	No	No	No	No	No	Si
Grafos ponderados positivos.	Si	Si	Si	Si	Si	Si	Si	Si	Si	Si
Grafos ponderados negativos.	No	No	No	No	No	No	No	No	No	No
Comienza por cualquier nodo.	Si	Si	Si	Si	Si	Si	Si	Si	Si	Si
Recorre todos los nodos.	Si	No	Si	Si	Si	Si	Si	Si	Si	Si
Grafo dirigido.	Si	Si	Si	Si	Si	Si	Si	Si	Si	Si
Grafo no dirigido.	Si	Si	No	Si	No	Si	Si	No	Si	No
Grafo conexo	Si	Si	Si	Si	Si	Si	Si	Si	No	Si
Grafo no conexo	No	Si	No	No	No	No	No	No	Si	No

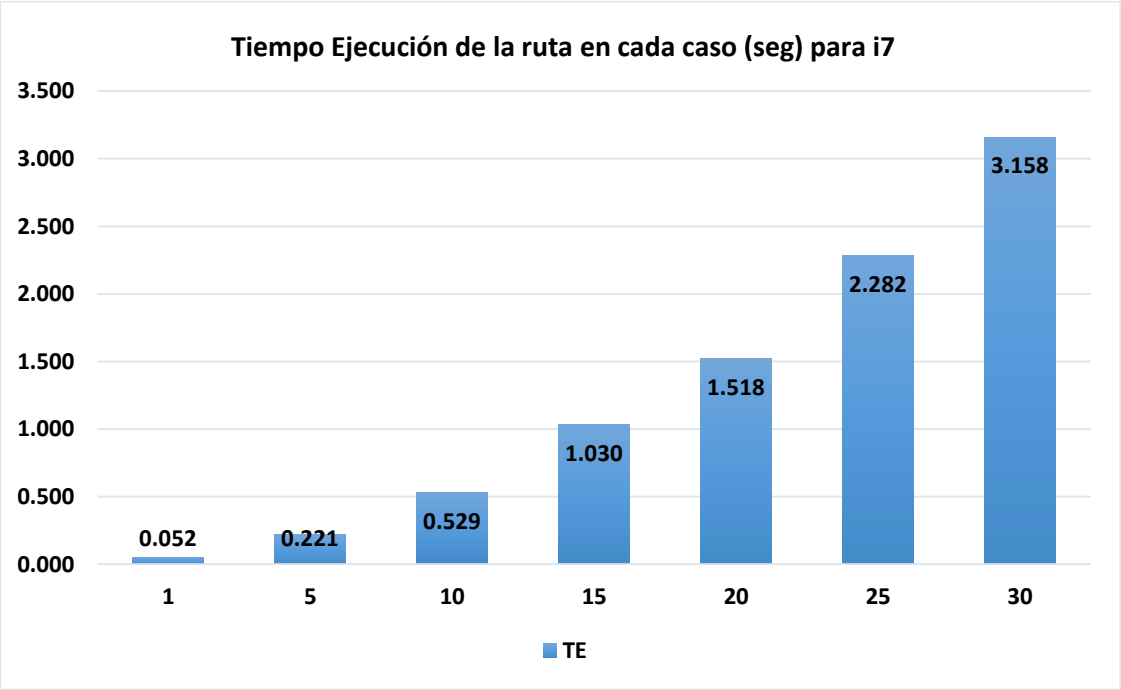
Fuente: Elaboración propia.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Anexo 8

Grafico 5: Tiempo de Ejecución de la ruta en cada caso (Tiempo en segundos) para un computador con procesador Corei7.



Fuente: Elaboración propia**.**

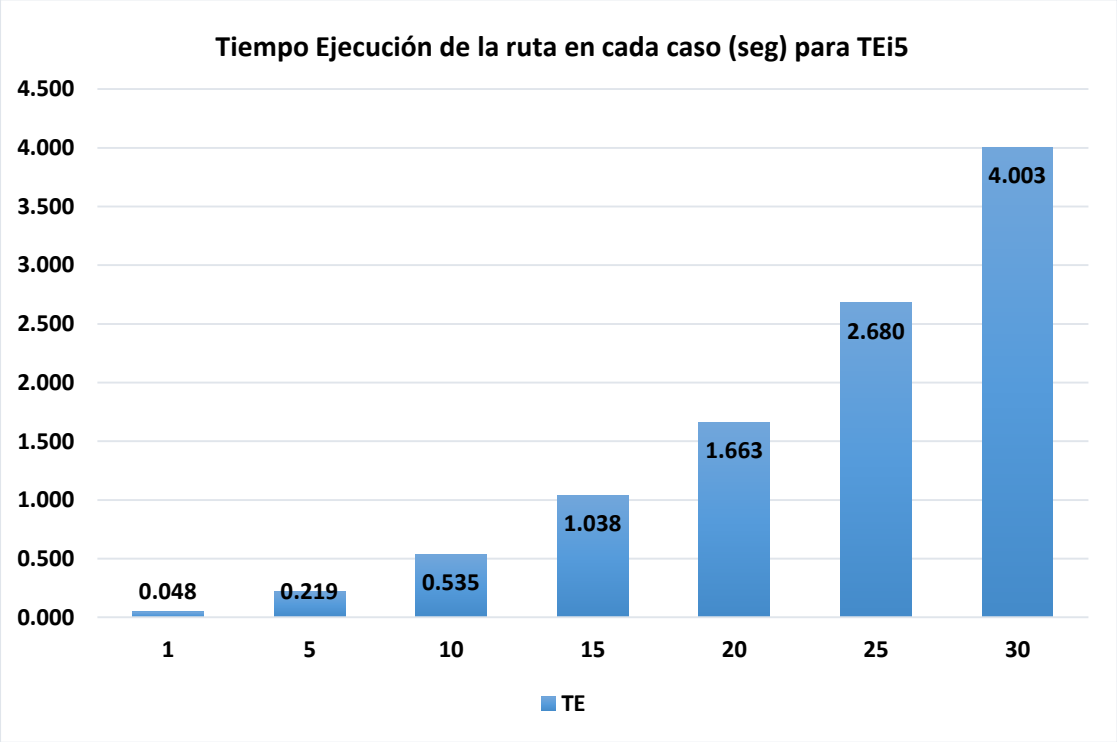
El tiempo de ejecución del algoritmo para cada caso se obtuvo midiendo el tiempo total del proceso de cómputo de la cadena de datos de la ruta más corta, es decir desde el punto de partida hasta encontrar la ruta más corta a cada punto de acción o foco infeccioso como indicia (De la Cruz, y otros, 2016).

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Anexo 9

Grafico 6: Tiempo de Ejecución de la ruta en cada caso (Tiempo en segundos) para un computador con procesador Corei5.



Fuente: Elaboración propia**.**

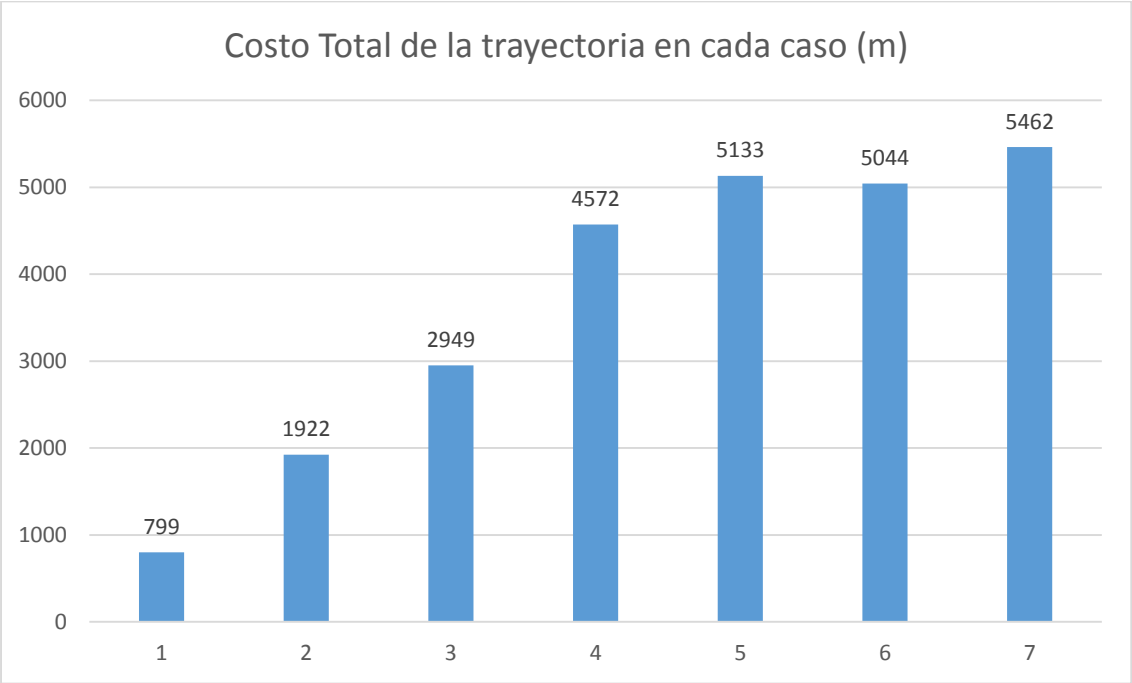
El tiempo de ejecución del algoritmo para cada caso se obtuvo midiendo el tiempo total del proceso de cómputo de la cadena de datos de la ruta más corta, es decir desde el punto de partida hasta encontrar la ruta más corta a cada punto de acción o foco infeccioso como indicia (De la Cruz, y otros, 2016).

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Anexo 10

Grafico 7: Costo Total (Distancia en metros) de la trayectoria en cada caso para un computador con procesador Corei7.



Fuente: Elaboración propia**.**

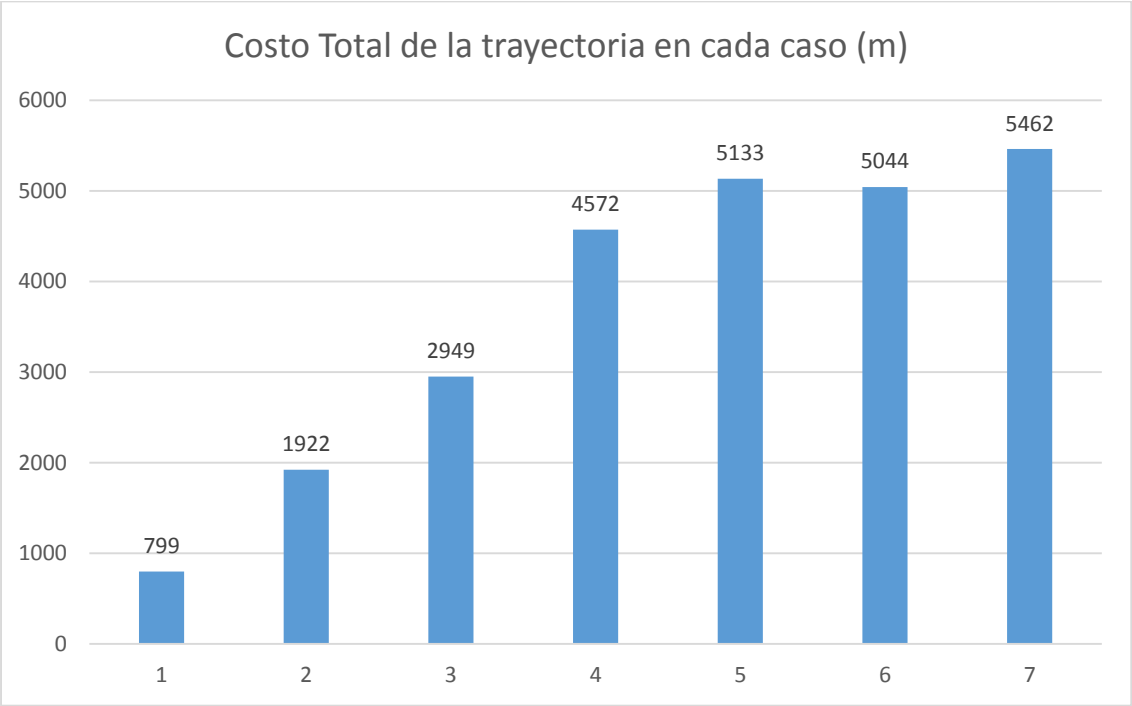
Como menciona De la Cruz, y otros (2016) El costo total de la trayectoria es la distancia total calculada por el algoritmo, es decir la distancia más corta para poder cubrir los puntos de acción para el recojo de desechos sólidos. Asimismo, Mingjun & Meng (2014) definen el costo, como la longitud de la ruta expresada $W_{Li} = L_i$, siendo L_i la longitud de la carretera y W_{Li} el arco total de la longitud de la carretera.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe

Anexo 11

Grafico 8: Costo Total (Distancia en metros) de la trayectoria en cada caso para un computador con procesador Corei5.



Fuente: Elaboración propia**.**

Como menciona De la Cruz, y otros (2016) El costo total de la trayectoria es la distancia total calculada por el algoritmo, es decir la distancia más corta para poder cubrir los puntos de acción para el recojo de desechos sólidos. Asimismo, Mingjun & Meng (2014) definen el costo, como la longitud de la ruta expresada $W_{Li} = L_i$, siendo L_i la longitud de la carretera y W_{Li} el arco total de la longitud de la carretera.

Tesis publicada con autorización del autor
Algunos Derechos Reservados. No olvide citar esta tesis

Campus Universitario Km. 5 Carretera a Pimentel - Chiclayo - Perú
Teléf: (+51)(74) 481610 / 481620 - Fax: 203861
www.uss.edu.pe