

Trabajo de Fin de Grado

Ingeniería en Organización Industrial

Desarrollo de una metodología para el diseño de una red de carriles para bicicletas considerando el nivel de satisfacción de los usuarios.

Autor: Cristina Isabel León Cubeiro

Tutor: José David Canca Ortiz

Dpto. Organización Industrial y Gestión de
Empresas I

Escuela Técnica Superior de Ingeniería

Sevilla, 2022



Trabajo de Fin de Grado
Ingeniería en Organización Industrial

**Desarrollo de una metodología para el diseño de
una red de carriles para bicicletas considerando el
nivel de satisfacción de los usuarios.**

Autor:

Cristina Isabel León Cubeiro

Tutor:

José David Canca Ortiz

Profesor titular

Dpto. de Organización Industrial y Gestión de Empresas I

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2022

Trabajo de Fin de Grado: Desarrollo de una metodología para el diseño de una red de carriles para bicicletas considerando el nivel de satisfacción de los usuarios.

Autor: Cristina Isabel León Cubeiro

Tutor: José David Canca Ortiz

El tribunal nombrado para juzgar el Trabajo de Fin de Grado arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2022

El Secretario del Tribunal

AGRADECIMIENTOS

En este apartado quiero agradecer primeramente a mis padres, hermana y abuela por siempre brindarme su amor y apoyo incondicional en todos los momentos importantes y no tan importantes de mi vida. Especialmente a mis padres por darme la educación y herramientas necesarias para ser la persona que soy.

A Marcel Martinez por estar siempre presente no solo en esta etapa de mi vida si no en todo momento y brindarme su apoyo incondicional.

A la Sra. Mercedes y al Sr. Marcel por el apoyo y cariño que he recibido en todo momento de esta etapa de mi vida.

A Lola por su apoyo y amistad durante esta etapa de mi vida.

Finalmente, a mi tutor por darme las herramientas necesarias para la realización del presente trabajo.

Muchas gracias a todos.

Cristina Isabel León Cubeiro

Sevilla, 2022

RESUMEN

En los últimos años se han realizados cambios en la mentalidad urbanística de las diferentes ciudades [1], que en conjunto con la concienciación de la problemática ambiental, han impulsado a las personas a emplear sistemas de transportes alternativos a los convencionales.

Por ello, es necesario que las ciudades cuenten con una infraestructura que permita a los usuarios emplear medios de transportes alternativos, sin perjudicar la infraestructura de los sistemas de transportes actuales.

Estas infraestructuras (carriles para bicicletas) deben cumplir con las normativas de seguridad, las normativas de urbanismo y al mismo tiempo deben ser adecuados para la práctica del ciclismo [2], es por ello por lo que su diseño se ha convertido en una tarea importante. Desde un punto de vista científico, este diseño es posible gracias a la implementación de modelos matemáticos que permiten describir y analizar sistemas o situaciones reales mediante el modelado de las relaciones entre sus elementos.

En el presente Trabajo de Fin de Grado se propone una metodología que permite resolver el problema de diseño de una red de carriles de bici teniendo en cuenta el nivel de satisfacción de los usuarios, así como la implementación de múltiples tipos de infraestructuras, y se ilustra su aplicación usando la red de la ciudad de Sevilla, España.

ABSTRACT

In recent years, changes have been made in the urban mentality of the different cities [1], together with the awareness of environmental problems, have encouraged people to use alternative transport systems to conventional ones.

Therefore, it is necessary for cities to have an infrastructure that allows users to use alternative means of transport, without harming the infrastructure of current transport systems.

These infrastructures (bicycle lanes) must comply the safety regulations, urban planning regulations and at the same time must be suitable for the practice of cycling [2], which is why their design has become an important task. for government agencies. From a scientific point of view, this design is possible thanks to the implementation of mathematical models that allow describing and analyzing real systems or situations by modeling the relationships between their elements.

In this Final Degree Project, a methodology is proposed that allows solving the problem of designing a network of bike lanes considering the level of user satisfaction, as well as the implementation of multiple types of infrastructure, and it is illustrated your application using the network of the city of Seville, Spain.

ÍNDICE

AGRADECIMIENTOS	vii
RESUMEN	ix
ABSTRACT	xi
ÍNDICE	xiii
ÍNDICE DE TABLAS	xv
ÍNDICE DE ILUSTRACIONES	xvii
NOTACIÓN	xx
1 INTRODUCCIÓN	1
1.1 <i>ESTADO DEL ARTE</i>	1
2 OBJETIVOS	5
2.1 <i>OBJETIVO GENERAL</i>	5
2.2 <i>OBJETIVOS ESPECÍFICOS</i>	5
3 FUNDAMENTOS TEÓRICOS	6
3.1 <i>TEORÍA DE GRAFOS</i>	6
3.2 <i>DEFINICIÓN DE UN GRAFO</i>	7
3.3 <i>GRAFO PONDERADO</i>	8
3.4 <i>SUBGRAFO</i>	8
3.5 <i>MATRIZ DE ADYACENCIA</i>	9
3.6 <i>GRAFO MULTICAPA O RED MULTICAPA</i>	10
4 DESARROLLO	19
4.1 <i>ANÁLISIS DEL PROBLEMA DE DISEÑO DE LA RED DE CARRILES PARA BICICLETAS.</i>	19
4.2 <i>DESARROLLO DE LA METODOLOGÍA</i>	25
4.2.1 <i>ANÁLISIS DEL PROBLEMA</i>	25
4.2.2 <i>PLANTEAMIENTO DE LA METODOLOGÍA:</i>	25
4.3 <i>FUNCIONALIDAD DE LA METODOLOGÍA</i>	37
3.3.1 <i>PRUEBA 1: Presupuesto P=700.000 u.m.</i>	40
3.3.2 <i>PRUEBA 2: Presupuesto P=1.000.000 u.m.</i>	48
3.3.3 <i>PRUEBA 3: Presupuesto P=3.000.000 u.m.</i>	49
3.3.4 <i>PRUEBA 4. Presupuesto P=25.000.000 u.m.</i>	53
3.3.5 <i>PRUEBA 5: Presupuesto P=50.000.000 u.m.</i>	54
4.4 <i>CONCLUSIONES DE LAS PRUEBAS:</i>	55
5 CASO PRÁCTICO	57
5.1 <i>DATOS DEL PROBLEMA</i>	58
6 CONCLUSIÓN	67
REFERENCIAS	69
7 ANEXO	70
7.1 <i>ESTRUCTURA DEL MODELO GENERADO POR EL PROGRAMA</i>	70

ÍNDICE DE TABLAS

Tabla 1 Matriz de adyacencia del grafo G .	10
Tabla 2 Matriz de demanda D de viajes por los arcos de la red G	38
Tabla 3 Coste de construcción y usuario en función del tipo de infraestructura.	39
Tabla 4 Penalización por cambio de infraestructura	40
Tabla 5 Solución Prueba 1: Presupuesto $P = 700.000$ u.m.	42
Tabla 6 Solución para el viaje $W = (1,5)$	42
Tabla 7 Coste del usuario (en unidades de tiempo) para realizar el viaje $W = 1,5$	44
Tabla 8 Solución para el viaje $W = (1,3)$	44
Tabla 9 Coste del usuario (en unidades de tiempo) para realizar el viaje $W = 1,3$	45
Tabla 10 Solución para el viaje $W=(2,6)$	45
Tabla 11 Coste del usuario (en unidades de tiempo) para realizar el viaje $W = 1,3$	47
Tabla 12 Solución para el viaje $W = (4,6)$	47
Tabla 13 Coste del usuario (en unidades de tiempo) para realizar el viaje $W = 1,3$	48
Tabla 14 Solución Prueba 2: Presupuesto $P = 1.000.000$ u.m.	49
Tabla 15 Solución Prueba 3: Presupuesto $P = 3.000.000$ u.m.	50
Tabla 16 Matriz de demanda D de viajes por los arcos de la red H .	51
Tabla 17 Coste de construcción y usuario en función del tipo de infraestructura.	52
Tabla 18 Penalización por cambio de infraestructura	52
Tabla 19 Solución Prueba 4: Presupuesto $P = 25.000.000$ u.m.	53
Tabla 20 Solución Prueba 5: Presupuesto $P = 50.000.000$ u.m.	54
Tabla 21 Datos red Sevilla	60
Tabla 22 Solución Prueba 1: Presupuesto $P = 6.000.000$ u.m.	62
Tabla 23 Solución Prueba 2: Presupuesto $P = 15.000.000$ u.m.	64
Tabla 24 Solución Caso 2: Presupuesto $P = 15.000.000$ u.m.	66

ÍNDICE DE ILUSTRACIONES

Ilustración 1 Carril compuesto por paneles solares realizado por la empresa SOLAROAD. Fuente: https://www.solaroad.nl/	2
Ilustración 2 Carril compuesto de 1000 kg de plástico reciclado, realizado por la empresa PLASCTICROAD. Fuente: https://plasticroad.com/	3
Ilustración 3 Grafo con 5 vértices y 6 aristas	7
Ilustración 4 Grafo no dirigido (izquierda) y grafo dirigido (derecha)	7
Ilustración 5 Grafo no ponderado (izquierda) y grafo ponderado (derecha).	8
Ilustración 6 Grafo G (izquierda) y subgrafo de G (derecha).	9
Ilustración 7 Grafo dirigido G con 6 vértices y 5 aristas.	9
Ilustración 8 Capa 1: Servicio urbano en la ciudad de Lisboa. Fuente: https://www.turismoenportugal.org/tren-en-portugal	11
Ilustración 9 Capa 2: Servicio pendular. Fuente: https://www.eurail.com/es/get-inspired/trains-europe/high-speed-trains/alfa-pendular	11
Ilustración 10 Capa 3: Servicio Inter ciudades. Fuente: https://www.eurail.com/es/get-inspired/trains-europe/high-speed-trains/alfa-pendular	12
Ilustración 11. Redes temporales. Fuente: Boccaletti et al.,2014	13
Ilustración 12 Redes interconectadas .Fuente: Boccaletti et al.,2014	13
Ilustración 13 Redes multinivel. Fuente: Hammoud y Kramer (2020).	15
Ilustración 14. Hipernetes. Fuente: Hammoud y Kramer (2020)	15
Ilustración 15. Nodos coloreados. Fuente: Hammoud y Kramer (2020)	16
Ilustración 16. Arcos coloreados. Fuente: Hammoud y Kramer (2020).	17
Ilustración 17. Red base	26
Ilustración 18 Tipos de infraestructura	26
Ilustración 19 Soluciones alternativas	27
Ilustración 20 Estructura multicapa (nodos)	32
Ilustración 21. Estructura multicapa (arcos capas).	33
Ilustración 22 Estructura multicapa (arcos de cambio).	34
Ilustración 23. Red G base.	38
Ilustración 24 Conjunto de pares OD para un conjunto $W = \{1,3,1,5,2,6,4,6\}$.	39
Ilustración 25 Solución Prueba 1: Presupuesto $P = 700.000$ u.m.	41
Ilustración 26 Solución gráfica del viaje $W = (1,5)$ en la estructura multicapa	43
Ilustración 27 Solución grafo final para el viaje $W = 1,5$	43
Ilustración 28 Solución gráfica del viaje $W = (1,3)$ en la estructura multicapa	44
Ilustración 29 Solución grafo final para el viaje $W = (1,3)$	45
Ilustración 30 Solución grafo final para el viaje $W = (2,6)$	46
Ilustración 31 Solución grafo final para el viaje $W = (2,6)$	46
Ilustración 32 Solución grafo final para el viaje $W = (4,6)$	47
Ilustración 33 Solución grafo final para el viaje $W = (4,6)$	48

Ilustración 34 Solución Prueba 2: Presupuesto $P = 1.000.000$ u.m.	49
Ilustración 35 Solución Prueba 3: Presupuesto $P = 3.000.000$ u.m.	50
Ilustración 36 Red H.	51
Ilustración 37 Conjunto de pares OD para un conjunto $W = \{(2,6), (7,9), (4,6), (1,7)\}$.	52
Ilustración 38 Solución Prueba 4: Presupuesto $P = 25.000.000$ u.m	53
Ilustración 39 Prueba 5: Presupuesto $P = 50.000.000$ u.m	54
Ilustración 40 Red Bicicleta Sevilla	57
Ilustración 41 Red de las principales avenidas y calles de Sevilla	58
Ilustración 42 Red de las principales avistas y calles de Sevilla	59
Ilustración 43 Red de bicicleta para la ciudad de Sevilla. Prueba 1, Caso 1	61
Ilustración 44 Red de bicicleta para la ciudad de Sevilla. Prueba 2, Caso 1.	63
Ilustración 45 Red de bicicletas para la ciudad de Sevilla. Prueba 1, Caso 2.	65

NOTACIÓN

\in	Pertenciente
Σ	Sumatoria
	Tal que
\forall	Para todo
\cap	Intersección
\cup	Unión
\leq	Menor o igual que
\geq	Mayor o igual que
\neq	Diferente de

1 INTRODUCCIÓN

“La flor que florece en la adversidad es la más rara y hermosa de todas.”

- Mulán -

En muchas ciudades se está promoviendo la bicicleta como medio de transporte para entre otras cosas, enfrentar la problemática ambiental [3], debido a esto, cada vez son más las personas que deciden emplear bicicletas frente al transporte público u otros vehículos de motor. Poco a poco la bicicleta ha ido ganando protagonismo como medio de transporte alternativo, ya que es un medio beneficioso para la salud de los usuarios, es económico y se trata de un medio de transporte sostenible libre de emisiones de CO_2 al ambiente [4].

Las personas son más conscientes de la importancia de ser protagonista y agente de cambio en materia ambiental, por tanto, la bicicleta y recientemente los monopatines se están convirtiendo en actores cada vez más populares en la rutina de las personas, principalmente para desplazarse al trabajo, universidad, etc.

Los organismos gubernamentales reconocen la importancia de una movilidad más sostenible [5], para ello deben proporcionar a las ciudades, las infraestructuras necesarias que permitan el desplazamiento de los usuarios y que representen una alternativa frente al uso del transporte público u otros vehículos de motor. Estas infraestructuras (carriles para bicicletas) deben cumplir con las normativas de seguridad, las normativas de urbanismo y al mismo tiempo deben ser adecuados para la práctica del ciclismo [2]. Desde un punto de vista científico, este diseño es posible gracias a la implementación de modelos matemáticos que permiten describir y analizar sistemas o situaciones reales mediante el modelado de las relaciones entre sus elementos.

1.1 ESTADO DEL ARTE

Un sistema de transporte sostenible se basa en la eficiencia energética y en la falta de efectos nocivos para el medioambiente y los seres humanos [6]. Para fomentar una movilidad sostenible es necesario que las ciudades cuenten con una infraestructura que permita a los usuarios utilizar medios de transporte alternativos a los convencionales.

El diseño y estudio de dichas infraestructuras forman parte del estudio de sistemas abstractos. Las redes o grafos

pueden facilitar la comprensión de dichos sistemas, por ejemplo, pueden facilitar el estudio del comportamiento de la red de personas que forman la sociedad, las neuronas, los transportes urbanos, la ruta de barcos o aviones, etc. Todo ello mediante la implementación de herramientas o teorías matemáticas.

Los modelos de diseño de redes tienen numerosas aplicaciones en varios campos, tales como: telecomunicaciones [7], planificación de la distribución [8] y transporte [9]. Aunque en cada una de estas aplicaciones el planteamiento del problema puede ser diferente, los modelos matemáticos subyacentes pueden ser muy similares entre si [9].

En cuanto al campo de transporte existen numerosos estudios que no solo se enfocan al diseño de una red para bicicletas, sino que también se enfocan en la optimización de estas. Una mejora puede ser por ejemplo la que describe Murawsky y Church [10], que proponen un modelo que mejore los enlaces de la red para garantizar una máxima accesibilidad de las ubicaciones (centros de salud). Otro ejemplo en cuanto a la optimización de las redes se debe a Leblanc [11], que trata de determinar que enlaces deben mejorarse en una red vial urbana para minimizar la congestión total.

El diseño de carriles para bicicletas en una determinada ciudad debe satisfacer la demanda de viajes diarios de los usuarios y de igual forma no perjudicar a los sistemas de transportes existentes, para ello, las ciudades deben contar con infraestructura para la práctica del ciclismo de la misma forma que existen las infraestructuras especializadas para peatones y automóviles.

Hoy en día existen numerosos proyectos para llevar a cabo la construcción de infraestructuras de carriles para bicicletas que empleen materiales que sean amigables para el medio ambiente, como lo es por ejemplo el proyecto que llevó a cabo la empresa SOLAROAD en la ciudad de Krommenie (Países bajos) donde instalaron 90 m de un carril compuesto por paneles solares capaces de generar energía renovable.



Ilustración 1 Carril compuesto por paneles solares realizado por la empresa SOLAROAD. Fuente: <https://www.solaroad.nl/>

Otros proyectos los ha llevado a cabo la empresa PLASTICROAD, en el bosque de Chapultepec, México, consiste en un carril de bicicleta fabricados con 1000 kilos de residuos plásticos, que equivalen a medio millón de tapas de botellas. Además de ser un carril para bicicletas, este proyecto ayuda a enfrentar la problemática de las inundaciones que tiene la zona ya que el carril se utiliza para almacenar agua de lluvia de forma rápida para hacer frente a las lluvias extremas.



Ilustración 2 Carril compuesto de 1000 kg de plástico reciclado, realizado por la empresa PLASCTICROAD. Fuente: <https://plasticroad.com/>

Por otro lado, también existen las infraestructuras convencionales como lo pueden ser: (i) Vías específicamente acondicionada para el tráfico de bicicletas, con la señalización horizontal y vertical correspondiente, y cuyo ancho permite el paso seguro de estos vehículos, (ii) Vía ciclista que discurre adosada a la calzada, en un solo sentido o en doble sentido, (iii) Carril-bici provisto de elementos laterales que lo separan físicamente del resto de la calzada, así como de la acera, etc.

Como se puede observar el diseño de carriles de bicicletas puede abarcar muchas casuísticas que incluyen considerar el coste de construcción, la demanda de los usuarios y hasta seleccionar el tipo de infraestructura a utilizar, en este caso la elección del tipo de material a utilizar determinará o no el coste total de construcción.

2 OBJETIVOS

*“Un solo granito de arroz puede inclinar la balanza.
Un solo hombre puede ser la diferencia entre una
victoria y una derrota.”*

-Mulán-

2.1 OBJETIVO GENERAL

El presente Trabajo de Fin de Grado tiene como objetivo desarrollar una metodología que permita resolver el problema de diseño de una red de carriles para bicicletas e ilustrar su aplicación utilizando la red de la ciudad de Sevilla, España.

2.2 OBJETIVOS ESPECÍFICOS

- Construir un modelo matemático para el diseño de una red de carriles para bicicletas.
- Demostrar el funcionamiento de la metodología propuesta en redes pequeñas.
- Aplicar la metodología a una red de mayor tamaño y analizar las soluciones obtenidas.

3 FUNDAMENTOS TEÓRICOS

“He oído hablar mucho de ti, Fa Mulán. Robaste la armadura de tu padre, escapaste de casa, te hiciste pasar por soldado, engañaste a tu superior, deshonraste al ejército chino, destruiste mi palacio... y nos has salvado a todos.”

-Mulán-

3.1 TEORÍA DE GRAFOS

La Teoría de Grafos es una de las áreas de la matemática cuyo desarrollo ha estado motivado por sus diferentes aplicaciones.

Desde sus orígenes la Teoría de Grafos se ha utilizado para diferentes áreas de aplicación, por ejemplo, la resolución de juegos matemáticos, para el estudio de circuitos eléctricos y diversas aplicaciones en una multitud de campos tan diferentes como lo puede ser la biología, psicología, economía, entre otros.

Con la Teoría de Grafos se estudia las propiedades de estos problemas para dar respuesta a múltiples situaciones cotidianas, desde buscar el camino más corto para que una persona se desplace de un lugar a otro, o por ejemplo para desarrollar una estrategia comercial de una empresa a la que le interesa conocer las interacciones de sus clientes con su entorno para poder ofrecerles productos que satisfagan sus necesidades.

Como se ha comentado anteriormente, la teoría de grafos se aplica dentro y fuera de la matemática y es una rama de investigación muy activa. En el campo de la informática se utiliza para la representación y almacenamiento de datos, así como para diseñar algoritmos adecuados para la resolución de diversos problemas que se benefician de la estructura de la red.

A continuación, se realizará un breve repaso sobre los conceptos fundamentales de la teoría de grafos y especialmente se menciona aquellos conceptos que se han empleado en la realización del presente trabajo de fin de grado.

3.2 DEFINICIÓN DE UN GRAFO

Un grafo G es un par (V, A) , donde V es un conjunto cuyos elementos son denominados vértices o nodos y A es un subconjunto de pares no ordenados de vértices que reciben el nombre de aristas o arcos.

Si $V = (v_1, \dots, v_n)$, los elementos de A se representan de la forma (v_i, v_j) , donde v_i, v_j se denominan extremos de dicha arista. Dos vértices se consideran adyacentes si están conectados por una arista. Por ejemplo, la Ilustración 3 muestra un grafo con 5 nodos:

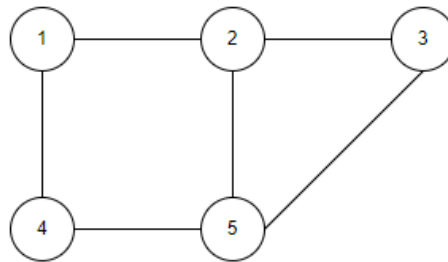


Ilustración 3 Grafo con 5 vértices y 6 aristas

Siendo:

El conjunto de vértices: $V = (1,2,3,4,5)$.

El conjunto de arcos: $A = \{(1,2), (2,1), (1,4), (4,1), (4,5), (5,4), (5,2), (2,5), (5,3), (3,5), (2,3), (3,2)\}$.

Si se considera A como un conjunto de pares ordenados, es decir, si se les asigna un sentido a todas las aristas, entonces se tiene un grafo dirigido. En este caso las aristas se suelen denominar arcos.

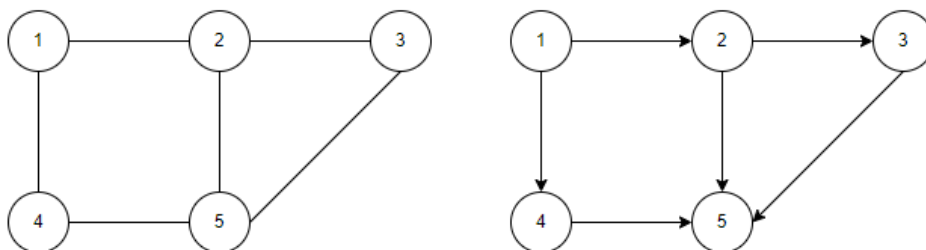


Ilustración 4 Grafo no dirigido (izquierda) y grafo dirigido (derecha)

Así, por ejemplo, para el grafo de la parte derecha de la Ilustración 4, se tiene:

El conjunto de vértices del grafo dirigido: $V = (1,2,3,4,5)$.

El conjunto de aristas $A = \{(1,2), (1,4), (4,5), (2,5), (5,3), (2,3)\}$.

3.3 GRAFO PONDERADO

En la teoría de grafos, un grafo que sea ponderado o con pesos es un grafo donde las aristas tienen un valor asociado.

Un grafo ponderado se define como un trio ordenado $G = (V, A, W)$, donde $V = (v_1, \dots, v_n)$ es un conjunto de vértices, $A = (a_1, \dots, a_n)$ es un conjunto de aristas y $W = (w_1, \dots, w_n)$ es el conjunto de pesos asociados a cada arco.

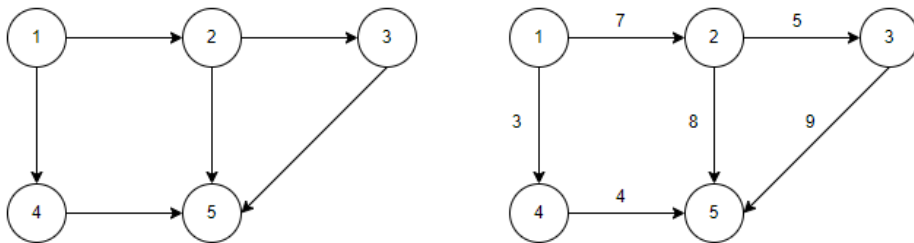


Ilustración 5 Grafo no ponderado (izquierda) y grafo ponderado (derecha).

Para este ejemplo, se tiene:

El conjunto de vértices del grafo ponderado $V = (1,2,3,4,5)$.

El conjunto de arcos $A = \{(1,2), (1,4), (4,5), (2,5), (5,3), (2,3)\}$.

El conjunto de pesos $W = \{(7), (3), (4), (8), (9), (5)\}$.

El grafo G queda definido de la siguiente forma:

$$G = \{(1, 2, 7), (1, 4, 3), (4, 5, 4), (2, 5, 8), (5, 3, 9), (2, 3, 5)\}$$

3.4 SUBGRAFO

Un subgrafo de un grafo G , es un grafo cuyos conjuntos de vértices y aristas son subconjuntos de G .

Sea $G = (V, A)$ un grafo con un conjunto de V vértices y un conjunto de aristas o arcos A . Un subgrafo de G es un grafo $G' = (V', A')$ tal que:

1. V' es un subconjunto de V .
2. A' es un subconjunto de A .

Para cada elemento que aparece en arcos de A' sus extremos pertenecen a V' .

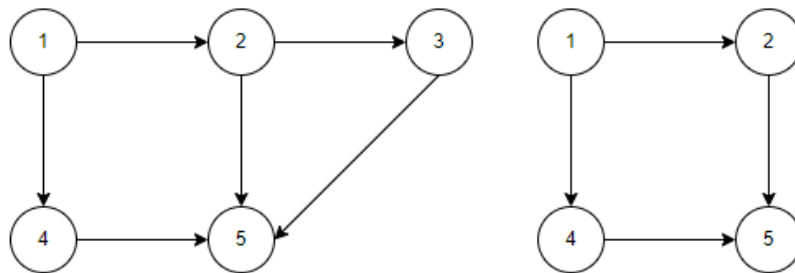


Ilustración 6 Grafo G (izquierda) y subgrafo de G (derecha).

Siendo:

El conjunto de vértices del grafo dirigido: $V = (1,2,3,4,5)$.

El conjunto de arcos: $A = \{(1,2), (1,4), (4,5), (2,5), (5,3), (2,3)\}$.

El conjunto de vértices del subgrafo de $G: V' = (1,2,4,5)$.

El conjunto de arcos $A' = \{(1,2), (1,4), (4,5), (2,5)\}$.

3.5 MATRIZ DE ADYACENCIA

Todo grafo simple puede ser representado por una matriz, denominada matriz de adyacencia. Se trata de una matriz cuadrada de n filas por n columnas (donde n es el número de vértices o nodos del grafo).

Para construir la matriz de adyacencia, cada elemento a_{ij} vale 1 si existe una arista que une a los vértices v_i y v_j . En caso contrario el elemento a_{ij} vale 0. Por lo tanto, la matriz de adyacencia estará formada por ceros y unos.

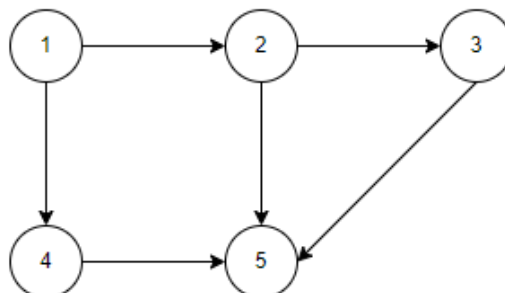


Ilustración 7 Grafo dirigido G con 6 vértices y 5 aristas.

A continuación, se muestra la matriz de adyacencia del grafo G :

Tabla 1 Matriz de adyacencia del grafo G .

V	1	2	3	4	5
1	0	1	0	1	0
2	0	0	1	0	1
3	0	0	0	0	1
4	0	0	0	0	1
5	0	0	0	0	0

3.6 GRAFO MULTICAPA O RED MULTICAPA

La Teoría de Grafos ha demostrado ser una herramienta eficaz para describir y analizar sistemas mediante el modelado de las relaciones de sus elementos [12], tal y como se ha descrito anteriormente, los elementos que conforman a un grafo o una red son un conjunto de nodos y un conjunto de arcos, por ejemplo, los nodos pueden representar ubicaciones y las aristas la relación existente entre cada subconjunto de nodos o en este caso la relación existente entre dos o más ubicaciones.

Con los conceptos descritos anteriormente sobre la Teoría de Grafos, se pueden representar grafos donde se analiza la relación que tienen los nodos a través de las aristas, sin embargo, esa teoría no es suficiente para el estudio de problemas más complejos, puesto que la representación de los elementos del sistema se realiza de forma equivalente lo que conlleva a la pérdida de información adicional sobre las características de los nodos o arcos, es decir, características relacionadas con el tiempo o el contexto de los elementos que forman parte del caso de estudio [13], para evitar la pérdida de información dichos grafos deben adquirir el carácter multicapa.

A continuación, se describe un ejemplo para ilustrar lo comentado anteriormente, el servicio de trenes de Portugal realiza enlaces entre diferentes ciudades o regiones del país y también conexiones dentro de las ciudades, los servicios más utilizados por los portugueses se describen a continuación:

1. Urbano/Regional: permite la conexión de localidades y pueblos dentro una ciudad.
2. Pendulares (Faro-Lisboa-Coimbra-Oporto-Braga): permite la conexión de varios puntos del norte y sur del país en menos de 6 horas.
3. Inter ciudades: permite la conexión entre las principales ciudades del país.

Una representación simple sería, por ejemplo, una red de una única capa donde los nodos son las estaciones de trenes, mientras que los arcos representan los enlaces entre dos o más estaciones diferentes, al definir a todos los nodos como estaciones y representarlos en una misma capa se pierde el contexto de los diferentes servicios que presta la red de trenes en Portugal. Para evitar esta situación, la red debe adquirir el carácter multicapa y para este caso en concreto la representación se realiza mediante 3 capas diferentes, es decir, una capa correspondiente a la red de trenes urbanos, otra para representar la red de transporte pendulares y otra para modelar la red de transporte inter-ciudades, cada capa tiene sus propias características como, por ejemplo, tiempo de

desplazamiento y coste.

Para el caso de una persona que vive en una localidad de la ciudad de Lisboa y se quiere trasladar a la ciudad de Oporto, en primer lugar, tendría que desplazarse por la capa de trenes urbanos y hacer conexión con cualquiera de las dos capas restantes ya que todas tienen la conexión Lisboa-Oporto. La decisión se tomaría en función del tiempo y del coste que tenga una capa versus otra.



Ilustración 8 Capa 1: Servicio urbano en la ciudad de Lisboa. Fuente: <https://www.turismoenportugal.org/tren-en-portugal>



Ilustración 9 Capa 2: Servicio pendular. Fuente: <https://www.eurail.com/es/get-inspired/trains-europe/high-speed-trains/alfa-pendular>



Ilustración 10 Capa 3: Servicio Inter ciudades. Fuente: <https://www.eurail.com/es/get-inspired/trains-europe/high-speed-trains/alfapendular>

Gracias al carácter multiplex de las redes multicapa se pueden representar sistemas considerando todas las interacciones que tienen cada una de las redes y a su vez las interacciones entre redes. A continuación, se describe la definición formal básica de una red o grafo multicapa:

Una red o grafo multicapa es un par $M = (G, C)$ donde $G = \{G_\alpha; \alpha \in \{1, \dots, M\}\}$ es una familia de grafos (direccionados o no, ponderados o no) $G_\alpha = (V_\alpha, A_\alpha)$, que contiene las capas de M , siendo: $C = \{A_{\alpha\beta} \subseteq V_\alpha \times V_\beta; \alpha, \beta \in M, \alpha \neq \beta\}$

El conjunto de interconexiones entre nodos de diferentes capas G_α y G_β con $\alpha \neq \beta$. Los elementos de C se denominarán capas cruzadas. Los elementos de A_α se denominarán conexiones intracapas de M , mientras que los elementos de cada $A_{\alpha\beta}$ con $\alpha \neq \beta$ se denominarán conexiones Inter capa.

El conjunto de vértices en cada capa se denota de la siguiente forma: $V_\alpha = \{v_1^\alpha, \dots, v_{V_\alpha}^\alpha\}$ y la matriz de adyacencia de cada capa se denota con $A^{[\alpha]}$ cuyos elementos son:

$$a_{ij}^\alpha = \begin{cases} 1 & \text{si la arista } (v_i^\alpha, v_j^\alpha) \in A_\alpha \\ 0 & \text{en caso contrario} \end{cases}$$

La matriz de adyacencia entre capas corresponde al conjunto $A_{\alpha\beta}$ de las interconexiones entre las capas α y β es $A^{[\alpha,\beta]}$ cuyos elementos son:

$$a_{ij}^{\alpha\beta} = \begin{cases} 1 & \text{si la arista } (v_i^\alpha, v_j^\beta) \in A_{\alpha\beta} \\ 0 & \text{en caso contrario} \end{cases}$$

A continuación, se presenta los principales tipos e implementaciones del marco multicapa:

1. Redes multiplex: Una red multiplex M , es un conjunto de capas donde cada capa es un grafo $G = \{G_\alpha; \alpha \in \{1, \dots, M\}\}$, puede ser dirigido o no, puede estar ponderado o no, denominado grafo $G_\alpha = (V_\alpha, A_\alpha)$, con $V_\alpha = \{v_1, \dots, v_N\}$. Todas las capas contienen los mismos nodos, es por ello por lo que las redes multiplex se pueden considerar como una red multicapa ya que en todas las capas se dispone del mismo conjunto de nodos $V_1 = \dots = V_M = V$ y $A_{\alpha\beta} = \{(x, x); x \in X\}$ para cada $1 \leq \alpha \neq \beta \leq M$.
2. Redes temporales: una red temporal $(G(t))_{t=1}^T$ se puede representar como una red multicapa con un conjunto de capas $\{G_1, \dots, G_T\}$ donde $G_t = G(t)$, $A_{\alpha\beta} = \emptyset$ si $\beta \neq \alpha + 1$, mientras $A_{\alpha, \alpha+1} = \{(v, v); v \in V_\alpha \cap V_{\alpha+1}\}$.

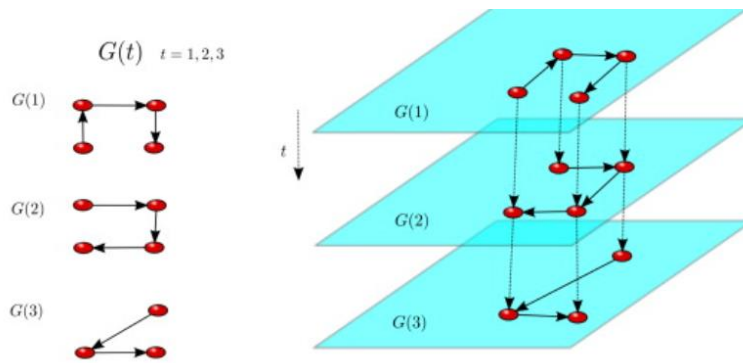


Ilustración 11. Redes temporales. Fuente: Boccaletti et al.,2014

3. Redes interconectadas: Si se considera una familia de redes $\{G_1, \dots, G_L\}$ que interactúan, se puede modelar como una red multicapa de capas $\{G_1, \dots, G_L\}$ y cuyas capas cruzadas $A_{\alpha\beta}$ corresponde a las interacciones entre la red G_α y G_β .

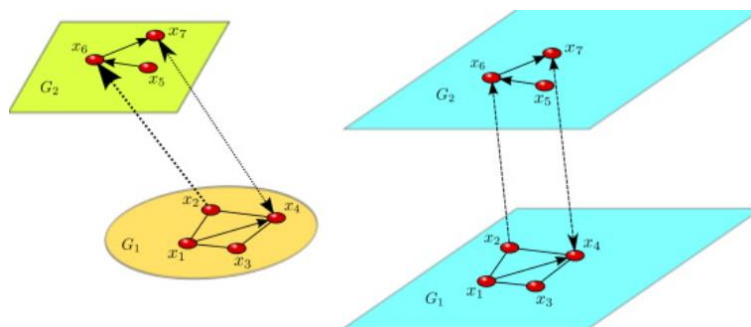


Ilustración 12 Redes interconectadas. Fuente: Boccaletti et al.,2014

4. Redes multidimensionales: un multigrafo con aristas o arcos etiquetados, $G = (V, A, D)$, donde V es el conjunto de nodos o vértices, D es un conjunto de etiquetas que representan dimensiones y A es un conjunto de arcos etiquetados, es decir un conjunto triple $A = \{(u, v, d); u, v \in V, d \in D\}$.

Se puede suponer que dado un par de nodos $u, v \in V$ y una etiqueta $d \in D$, puede existir solo un arco (u, v, d) . Además, si el grafo es considerado un grafo dirigido, los arcos (u, v, d) y (v, u, d) son distintos. Así, dado $|D| = m$, cada par de nodos en G se puede conectar como máximo a m arcos posibles.

Cuando sea necesario, es posible considerar pesos, de modo que los arcos ya no sean triples, sino que vienen representados por tuplas de 4 elementos, (u, v, d, w) , donde w es un número real que representa el peso de la relación entre $u, v \in V$ y etiquetado con $d \in D$. Una red multidimensional $G = (V, A, D)$ se puede modelar como una red multicapa asignando cada etiqueta a una capa. Específicamente, G se puede asociar a una red multicapa de capas $\{G_1, \dots, G_{|D|}\}$ donde por cada $\alpha \neq D, G_\alpha = (X_\alpha, A_\alpha), X_\alpha = V, A_\alpha = \{(x, x); x \in V \times V; (u, v, d) \in A \text{ y } d = \alpha\}$ y $A_{\alpha\beta} = \{(x, x); x \in V \text{ para cada } 1 \leq \alpha \neq \beta \leq |D|\}$.

5. Redes interdependientes: Es una colección de diferentes redes, las capas, cuyos nodos son interdependientes entre sí. En la práctica, los nodos de una capa de la red dependen de los nodos de control en una capa diferente. En este tipo de representación, las dependencias son aristas adicionales que conectan las diferentes capas. Esta estructura, entre las capas de la red, a menudo se denomina meso estructura.
6. Redes multinivel: Son similares a las redes multiplex, con la diferencia de que las capas pueden contener no solo un subconjunto de aristas sino también un subconjunto de nodos. El grafo M es una tupla definida por conjuntos de nodos V , aristas A y capas S con longitud p .

$$M = (V, A, S)$$

$$S = \{S_1, \dots, S_p\} \text{ subgrafos}$$

$$\text{con } S_j = (V_j, A_j), j = 1, \dots, p$$

$$V = \bigcup_{j=1}^p V_j$$

$$A = \bigcup_{j=1}^p A_j$$

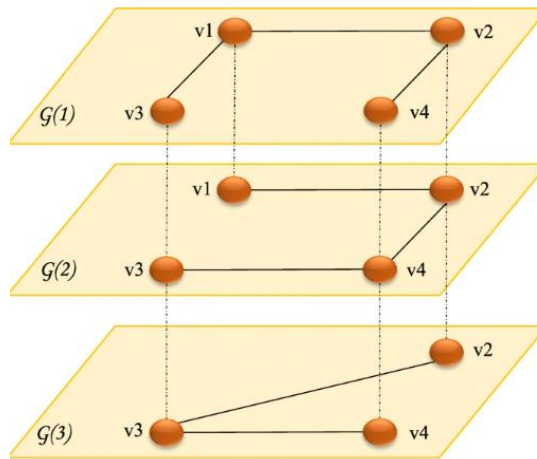


Ilustración 13 Redes multinivel. Fuente: Hammoud y Kramer (2020).

7. Hiperredes: Son grafos basados en nodos formados por diferentes comunidades o subredes que se cruzan. Un hipergrafo está definido por un par (V, H) , siendo V el conjunto de nodos y H una familia de subconjunto de V , cada uno llamado hiperaristas de H . Estos grafos se centran en los nodos que pertenecen al mismo grupo en lugar de las conexiones entre ellos. Los nodos del mismo subconjunto se asignan a una capa con nodos completamente conectados. La noción de las aristas aquí difiere de las redes normales; las aristas en estas redes se denominan hiperaristas y puede conectar múltiples nodos al mismo tiempo. Luego, cada hiperarista se asigna a una sola capa. Las hiperredes múltiples no están separadas por capas, sino alineadas con los nodos, ya que un nodo se puede asignar a varias capas, en función de la intersección de los subconjuntos.

$$\varphi = (V, h)$$

$$H = \{H_1, \dots, H_p\} \text{ subconjuntos}$$

$$A_{\alpha\beta} = \{(x, x); x \in V_\alpha \cap V_\beta\} \text{ donde } \alpha, \beta \in H$$

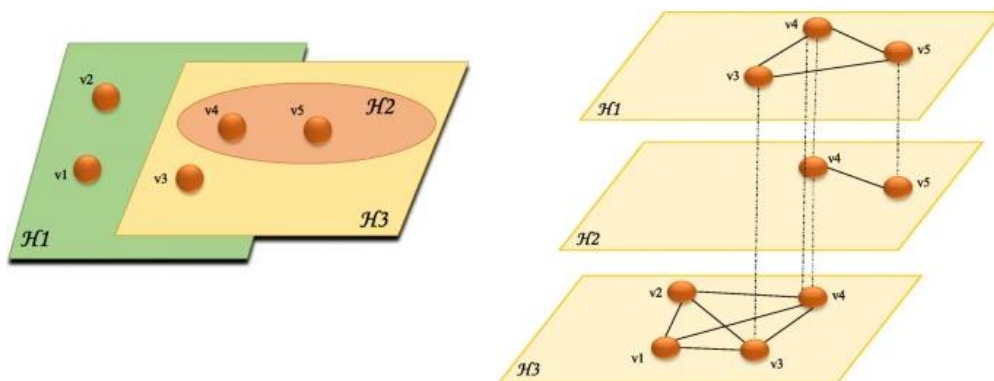
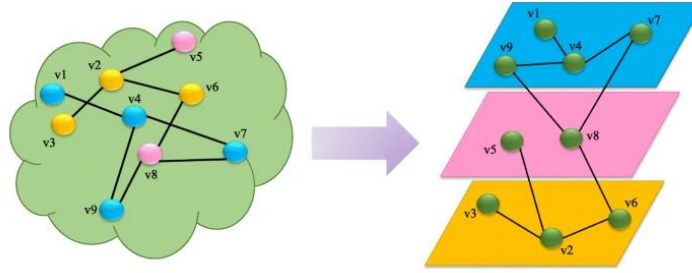


Ilustración 14. Hiperredes. Fuente: Hammoud y Kramer (2020)

Otro enfoque sobre las redes multicapa se explica en [14] que presenta una formulación abstracta sobre los grafos multicapas y los diferentes criterios para elegir capas.

El enfoque que presentan Hammoud y Kramer [14], se basa en realizar una diferenciación entre los tipos de nodos y los tipos de arcos. Para el primer caso, especifican que los nodos tienen diferentes aspectos o tipos (para diferenciarlos gráficamente se emplea el uso de colores). El grafo G_N es una tupla definida por un conjunto de nodos V , un conjunto de arcos A y un conjunto de capas C , donde las capas representan el color de los nodos. Dos nodos v_α y v_β están conectados por un arco $A_{\alpha\beta}$.

$$G_N = (V, A, C), V_\alpha \in V \times C, A_{\alpha\beta} \in V_\alpha \times V_\beta, \text{ donde } \alpha, \beta \in C$$



Node-colored Graphs. Nodes having similar colors are grouped in the same layer. The edges are the same as in the original network

Ilustración 15. Nodos coloreados. Fuente: Hammoud y Kramer (2020)

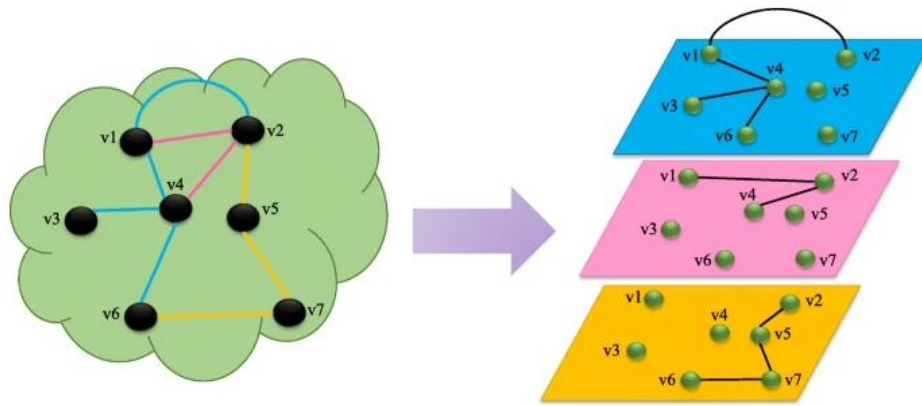
En dichos gráficos, el enfoque es explotar la topología del sistema desde una perspectiva basada en nodos. En la mayoría de los casos, el gráfico es disjuncto en capas, es decir, el nodo solo puede pertenecer a una capa y tener un color.

Para el segundo caso, especifican que el sistema se representa con conexiones (arcos) heterogéneas entre los nodos. El grafo G_A es una tupla definida por un conjunto de nodos V , un conjunto de colores C con longitud b y un conjunto de aristas A , cada uno asignado a un tipo o un aspecto definido por el color. Los arcos del mismo color pertenecen a una misma capa. Cada capa puede contener un subconjunto o el conjunto completo de nodos.

$$G_A = (V, A, C)$$

$$\{G_\alpha\}_{\alpha=1}^b = \{(V_\alpha, A_\alpha)\}_{\alpha=1}^b, \forall \alpha, \beta \in C \quad V_\alpha = V_\beta$$

$$A_\alpha \in V_\alpha \times V_\beta \times C$$



Edge-colored Graphs. The nodes of the original networks are replicated over the different layers. The layers represent the different aspects of relationships in the network. Edges are divided among the layers based on their colors

Ilustración 16. Arcos coloreados. Fuente: Hammoud y Kramer (2020).

El gráfico está alineado con los nodos, es decir, el mismo nodo puede pertenecer a diferentes capas simultáneamente. En dichos gráficos, la atención se centra en la dinámica de la red y las interacciones entre sus componentes. Resulta muy eficiente cuando se estudian diferentes aspectos del mismo sistema. Dos nodos solo pueden conectarse usando una arista de cada color. En caso de que se deba establecer el mismo tipo de conexión entre dos nodos, este marco se puede combinar con otro para lograr este propósito.

4 DESARROLLO

“El viento puede soplar fuerte, pero la montaña no lo reverencia.”

- Mulán -

En la primera parte de este capítulo se realiza un análisis del problema de flujo sobre un grafo sin limitaciones de capacidad, definiendo las variables que conforman al modelo. Posteriormente, se explica su implementación para el diseño de la red de carriles para bicicletas.

4.1 ANÁLISIS DEL PROBLEMA DE DISEÑO DE LA RED DE CARRILES PARA BICICLETAS.

El problema general de flujo a coste mínimo sobre un grafo sin limitación de capacidad [9] es un problema de la teoría de grafos que considera diferentes productos (demanda de flujo) que deben ser transportados entre un conjunto de nodos del grafo que actúan como orígenes y destinos de los envíos. Cada envío se origina en un nodo y su destino es otro nodo del grafo. Estos dos nodos definen un par origen-destino.

Los principales conjuntos necesarios para modelar el problema son: (i) un conjunto de nodos N y (ii) un conjunto de arcos A (direccionados o no). El modelo busca la inclusión de los arcos a la red en función de los costes de construcción de estos.

A continuación, se describe los *inputs* del modelo:

- i. P representa el presupuesto para la construcción de la red.
- ii. C_a representa el coste de construcción de un arco $a \in A$.
- iii. θ_{iw} representa el termino de la derecha de la restricción de equilibrio de flujo, es igual D_w si i es el origen del par OD w , es igual a $-D_w$ si i es el destino del par w , e igual a 0 en caso contrario. D_w representa el elemento de la matriz OD de demanda correspondiente al par w –ésimo, cuyo origen es

o_w y cuyo destino es d_w . El conjunto completo de pares OD se representará mediante la letra W .

- iv. y_a es una variable binaria que se utiliza para representar la construcción ($y_a = 1$) o no del arco ($y_a = 0$).
- v. x_{aw} es una variable real no negativa que representa el flujo del par OD $w \in W$ sobre el arco $a \in A$.
- vi. Una matriz cuadrada D que representa la demanda de viajes en los nodos del conjunto N en un horizonte temporal dado. Cada elemento de la matriz, D_{ij} , representa el número de desplazamientos desde el nodo i al nodo j .
- vii. $u_a \geq 0$ es el coste de desplazamiento del usuario al usar el arco a .

El modelo matemático general que permite diseñar la red corresponde a la siguiente formulación:

$$\min \sum_{a \in A^T} c_a * y_a + \alpha \sum_{w \in W} \sum_{a \in A} u_a * x_{aw} \quad (1)$$

$$s. a. \quad \sum_{a \in A} c_a * y_a \leq P \quad (2)$$

$$\sum_{a \in A_i^+} x_{aw} - \sum_{a \in A_i^-} x_{aw} = \theta_{iw}; \forall i \in N, w \in W, \quad (3)$$

$$x_{aw} \leq D_w * y_a; \forall a \in A, w \in W, \quad (4)$$

$$y_a \in \{0,1\}; \forall a \in A, \quad (5)$$

$$x_{aw} \geq 0; \forall a \in A, w \in W \quad (6)$$

La función objetivo (1) minimiza el coste total de construcción de la red y el coste de viaje de los usuarios, ponderando ambos términos adecuadamente mediante el peso α . La restricción (2) impone el límite presupuestario para la construcción de la red. El conjunto de restricciones (3) impone el equilibrio de flujo en cada nodo del grafo, para cada par OD. El conjunto de restricciones (4) establece que el flujo sobre cada arco de la red está sujeto a la existencia de la infraestructura adecuada sobre ese arco, es decir, si existe flujo sobre un arco, ese arco debe ser construido. Finalmente, las restricciones (5) y (6) establecen los dominios de las variables de decisión.

En el contexto del diseño de redes, el modelo descrito anteriormente busca el diseño de una red que por un lado minimice el coste de viaje en los arcos y por otro que el coste de construcción sea el menor posible.

En lo que sigue se amplía el modelo anterior considerando la posibilidad de construir diferentes tipos de carriles

bici. Para cada arco del grafo se considerarán diferentes posibilidades o tecnologías de construcción, que hacen referencia a aspectos como la calidad del firme, la anchura del carril, la señalización y protecciones laterales. Se planteará pues el diseño de una red de carriles bici en la que diferentes tramos pueden utilizar una tecnología de construcción diferente, si es conveniente y se valorará también la repercusión que los usuarios pueden experimentar al cambiar entre tecnologías de construcción. Así, si un carril que transcurre por una avenida es cómodo y ancho, está protegido, la rodadura es suave como consecuencia del material usado en el firme del carril y al finalizar la calle el carril cambia a un carril estrecho o de peor firme, o con menos protecciones, se supondrá que el usuario experimenta una pérdida de confort que deberá ser tenida en cuenta. El objetivo del modelo que se plantea pretende:

1. Maximizar la satisfacción del usuario, es decir, minimizar del coste del viaje del usuario, que puede ser expresado como el tiempo de viaje necesario para realizar el trayecto. Este tiempo de viaje estará relacionado con la propia construcción del carril bici. Carriles con mejor firme, más amplios y adecuadamente protegidos reducirán el coste de usuario, aunque resultarán más caros.
2. Deberá seleccionar la tecnología de construcción que se usará en las diferentes calles, si es que se decide usar esa calle para construir un carril bici. El modelo expuesto con anterioridad simplemente incorpora la construcción o no de un arco de la red. En lo que sigue, no sólo se considerará si construir o no un carril bici sobre cada uno de los arcos del grafo, sino que también se seleccionará la tecnología de construcción de entre un conjunto de posibilidades disponibles. En la experimentación, a fin de simplificar, se considerarán dos tipos de carriles bici, un tipo básico y otro de mayor coste que permita un menor tiempo de viaje y una mayor seguridad en los desplazamientos. Aunque se consideren dos tipos, el modelo propuesto será general y extensible a un mayor número de posibilidades de construcción.
3. Cada arco tendrá por tanto un coste de construcción asociado a cada uno de los tipos de carril bici que puedan ser construidos sobre él, además, cada tecnología de construcción implicará un coste o tiempo de viaje diferente para los usuarios. Si se opta por un tipo de carril bici más caro (más ancho, con mejor firme y protecciones), la velocidad de desplazamiento podrá ser en promedio mayor y por lo tanto, desde el punto de vista del usuario, su coste (tiempo de viaje) será menor. Es decir, mayores costes de construcción implicarán menores costes de usuario.

Para ampliar el modelo anterior de forma que se consideren diferentes posibilidades de construcción del carril bici sobre los arcos del grafo, se procederá de la siguiente manera:

- Se considerará un grafo dirigido $\hat{G} = (\hat{V}, \hat{A})$, que representa la red base disponible para construir la red de carriles bici. Cada arco $a \in \hat{A}$ es vendrá caracterizado por su longitud l_a .

- Puesto que se consideran diferentes tipos de infraestructura para construir la red de carriles bici, se enumerarán desde la más económica a la más cara, siendo la infraestructura 1 la opción más barata en término de costes de construcción, por lo tanto, la peor en términos del coste o tiempo de viaje de los usuarios. El orden creciente en el tipo de infraestructura o tecnología de construcción implicará un mayor coste de construcción y, por el contrario, una disminución en el tiempo de viaje de los usuarios. Para cada arco de \hat{A} se debe seleccionar una sola alternativa de infraestructura, incluida la alternativa de no hacer nada que no consume presupuesto de construcción pero que no mejora el nivel de satisfacción del usuario.
- Supondremos conocida la matriz D que representa la demanda de viajes sobre el grafo, entre cada par de nodos del conjunto N para un horizonte temporal dado.
- A fin de poder modelar las diferentes posibilidades de construcción, cada arco de la red se representará mediante un conjunto de arcos paralelos, cada uno de ellos en una capa diferente. Es decir, cada arco se replica tantas veces como tipos posibles de infraestructuras se consideren. Desde un punto de vista práctico, lo que sucede es que la red base se replica tantas veces como tecnologías de construcción, y cada una de estas copias se almacena en una capa diferente en una estructura multicapa. Para cada arco $a \in \hat{A}$ se tendrá un conjunto de “copias” o réplicas, cada una de ellas en una capa diferente de una red multicapa. El conjunto de réplicas del arco a se representará mediante $M(a)$. Este conjunto tiene una cardinalidad igual al número de tecnologías de construcción disponibles más uno, pues incluye también al propio arco a en la capa base. La capa base está compuesta por los arcos de \hat{A} , en estos arcos, la tecnología de construcción es la número 0, que es equivalente a no construir el carril bici. Su coste de construcción es por tanto nulo mientras que su coste de usuario o tiempo de viaje es el mayor de todos (sería el tiempo de desplazamiento en bicicleta sobre un arco en el que no existe carril bici. Nótese que el tiempo de viaje que se usa en el modelo depende de la longitud del arco. Es decir, se puede representar como una la longitud del arco dividido por la velocidad de recorrido. Precisamente esta velocidad es la que depende de la tecnología de construcción utilizada).

Para facilitar la comprensión del modelo se definirán los siguientes conjuntos. El conjunto completo de arcos del grafo multicapa se denominará A . Cada tecnología de construcción dispone de un grafo que resulta de replicar la red base descrita por \hat{G} . El grafo de cada tipo de infraestructura de construcción se denominará G^k , donde el índice $k = \{1, 2, \dots, K\}$ hace referencia a cada una de las tecnologías de construcción, ordenadas según coste de construcción creciente, como se comentó anteriormente. El conjunto de arcos y nodos de cada una de las capas se denotará como A^k y V^k , respectivamente. De la misma manera que a cada arco del grafo base se le hace corresponder un conjunto $M(a)$ que incluye los arcos replicados en las diferentes capas, a cada nodo del grafo base, esto es $i \in \hat{V}$ se le hace corresponder un conjunto que contiene los nodos replicados en las capas correspondientes a las diferentes tecnologías. Este conjunto se denominará $N(i)$. Para cada nodo i del grafo base, los nodos pertenecientes a $N(i)$ en dos capas consecutivas estarán unidos por arcos de transferencia (de cambio de tecnología), que permiten saltar entre capas en cualquier dirección, mejorando la tecnología de

construcción (movimientos hacia arriba en la red multicapa) o empeorando la tecnología (movimientos hacia abajo en la red multicapa). Estos arcos de cambio de tecnología que unen entre si los nodos de la capa base y los nodos de las diferentes capas (usando sólo arcos dentro de cada conjunto $N(i)$) forman un conjunto que representaremos como A^C , o arcos de cambio. Es importante señalar que estos arcos unen nodos consecutivos en $N(i)$, es decir, si i está en la capa k e i' en la capa $k + 1$, existen dos arcos de cambio, uniendo i e i' como arco de subida e i' con i como arco de bajada o de disminución de la calidad de construcción.

A modo de resumen:

\hat{G} es el grafo de la capa base, cuyos arcos y nodos están en \hat{A} y \hat{V} respectivamente.

Cada capa correspondiente a una tecnología de construcción viene representada por un grafo G^k , compuesto por nodos en V^k y arcos en A^k .

A cada nodo $i \in \hat{V}$ (capa base) se asocia el conjunto de nodos $N(i)$ que lo contiene, junto al resto de réplicas del nodo en las diferentes capas. Cada arco $a \in \hat{A}$ genera un conjunto de arcos $M(a)$ que contiene al arco a y a sus diferentes réplicas en las diferentes capas.

Los nodos de cada conjunto $N(i)$ están unidos mediante arcos dirigidos que conectan todos los nodos en ambas direcciones, permitiendo cambiar de capa, mejorando o empeorando la tecnología usada.

Finalmente, el grafo de la red multicapa, $G = \{V, A\}$, estará compuesto por todos los nodos de las distintas capas, $V = \hat{V} \cup_{k=1, \dots, K} V^k$ y por los arcos de las distintas capas y los arcos de cambio, $A = \hat{A} \cup_{k=1, \dots, K} A^k \cup A^C$.

Dentro del conjunto A^C hay dos tipos de arcos, de subida, que mejoran la construcción del carril bici y de bajada, que empeoran la tecnología de construcción empleada. Un usuario que atraviesa un arco hacia abajo estará cambiando de tecnología hacia una de peor calidad. Estos arcos, que representaremos mediante A^{CB} , implicarán una penalización o pérdida de confort en los usuarios de bicicleta, y su utilización será penalizada en la función objetivo del problema de diseño de la red utilizando un factor β . Los arcos de A^C orientados desde una capa inferior a una superior formarán el conjunto A^{CA} . De forma que $A^C = A^{CA} \cup A^{CB}$.

El modelo que se propone a continuación está pensado para su utilización desde el punto de vista de un planificador central, encargado de decidir la construcción de la red de carriles bici con un presupuesto limitado. El modelo tratará de maximizar el nivel de servicio de los usuarios, agotando el presupuesto en la construcción del mayor número de carriles bici posibles. Para ello tratará de enrutar a los usuarios desde sus orígenes a sus destinos de forma que puedan realizar sus desplazamientos en el menor tiempo posible. Para ello será conveniente construir carriles bici, a mejor calidad de construcción, menor el tiempo necesario para recorrerlos. Dado que existe un presupuesto global, el modelo deberá decidir el tipo de construcción que se seguirá en cada uno de los carriles seleccionados.

El modelo resultante, se muestra a continuación:

$$\text{Min} \sum_{w \in W} \left(\sum_{a \in A \setminus A^C} u_a x_{aw} + \beta \sum_{a \in A^{CB}} x_{aw} \right) \quad (1)$$

$$\text{s. a.} \quad \sum_{a \in A \setminus A^C} c_a y_a \leq P \quad (2)$$

$$\sum_{a \in A_i^+} x_{aw} - \sum_{a \in A_i^-} x_{aw} = \theta_{iw} \quad \forall i \in V, w \in W, \quad (3)$$

$$x_{aw} \leq D_w * y_a \quad \forall a \in A \setminus A^C, w \in W, \quad (4)$$

$$\sum_{s \in M(a)} y_s = 1 \quad \forall a \in A \setminus A^C, \quad (5)$$

$$y_a \in \{0,1\} \quad \forall a \in A \setminus A^C, \quad (6)$$

$$x_{aw} \geq 0 \quad \forall a \in A, w \in W \quad (7)$$

La función objetivo (1) minimiza el coste total de desplazamientos de los usuarios en bicicleta, incluyendo una penalización como consecuencia de que en su desplazamiento se empeore de tipo de infraestructura de construcción (es decir, si en un tramo se usa la tecnología k y en el siguiente la tecnología $k - 1$, se generará una penalización que aparecerá multiplicada por el volumen de pasajeros afectados. Los arcos que implican cambios descendientes de calidad en el carril bici son los pertenecientes al conjunto A^{CB} .

La restricción (2) impone el límite de presupuesto para la construcción de la red de carriles bici. Por otro lado, el conjunto de restricciones (3) impone el equilibrio de flujo en cada nodo del grafo, para cada par OD $w \in W$. Las restricciones (4) establecen que el flujo sobre un arco dado está sujeto a la existencia de la infraestructura en ese arco, si un arco no se incluye en la red de carriles bici, no podrá circular flujo de usuarios sobre él. El conjunto de restricciones (5) fuerza que en cada uno de los arcos del grafo se seleccione exclusivamente un tipo de construcción, incluyendo la opción de no hacer nada. Por este motivo, para cada arco a del grafo multicapa (sin contar los arcos de cambio) se suman las variables y_s pertenecientes al conjunto $M(a)$, que incluye el arco de la capa base y sus respectivas réplicas en las diferentes capas. Finalmente, las restricciones (6) y (7) establecen los dominios de las variables de decisión.

4.2 DESARROLLO DE LA METODOLOGÍA

Para la formulación de la metodología se utilizará las librerías de optimización del solver Gurobi® en conjunto con el lenguaje de programación Python.

4.2.1 ANÁLISIS DEL PROBLEMA

El problema que describe la metodología representa el desarrollo de una red para bicicletas que busca maximizar la satisfacción del usuario, para ello, el modelo debe seleccionar una combinación de infraestructuras para la construcción de los arcos de dicha red, teniendo en cuenta que cada arco tiene un coste de construcción y un coste de viaje del usuario, los cuales se especifican para cada una de las infraestructuras. Cuanto mayor sea el coste de construcción menor será el coste del viaje del usuario, en otras palabras, cuanto mejor sea el tipo de infraestructura mayor será el coste de construcción y menor el coste de viaje del usuario.

El problema a tratar en el presente Trabajo de Fin de Grado es considerado un problema NP-HARD, lo que quiere decir que para obtener el óptimo, el número de iteraciones necesarias para garantizar una solución óptima es de orden exponencial.

Existe diversas formas para la resolución de estos problemas, una de ella es emplear métodos aproximados (metaheurísticas) y de esta forma obtener una solución aproximada del problema. Por otro lado, se pueden aplicar métodos exactos siempre que el problema corresponda a un escenario de tamaño mediano.

4.2.2 PLANTEAMIENTO DE LA METODOLOGÍA:

El desarrollo de esta metodología se realizó en 3 fases:

- Fase 1: Interpretación del modelo mediante un fundamento teórico.
- Fase 2: Análisis de estructura de las ecuaciones en el modelo.
- Fase 3: Escritura de la metodología.

4.2.2.1 Fase 1: Interpretación del modelo mediante un fundamento teórico: En esta fase se buscó un fundamento teórico que ayude a la correcta interpretación del modelo.

Para traducir el modelo matemático al lenguaje de programación en primer lugar, se utilizó el primer enfoque de redes multicapa explicado en el apartado de “Fundamentos Teóricos”. En dicho enfoque se explica que el sistema a representar es un sistema con arcos heterogéneos entre los nodos, es decir, el grafo está alineado con los nodos por lo que un mismo nodo puede pertenecer a diferentes capas simultáneamente, mientras que los arcos con la misma naturaleza pertenecen a una misma capa.

Dicho lo anterior, para el desarrollo de la metodología se usa red inicial (red base) que posee un conjunto de nodos y arcos.

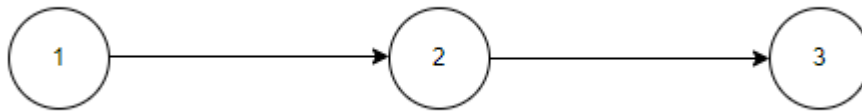


Ilustración 17. Red base

Sobre los arcos de dicha red, la metodología tomará la decisión de que tipo de infraestructura utilizar en base a las restricciones dispuestas en el modelo.

Como se ha comentado anteriormente, el planificador central puede disponer de diferentes tipos de infraestructuras a considerar, la representación gráfica sería que por cada infraestructura se genera una capa de la red manteniendo el conjunto de nodos de la red base en cada capa.

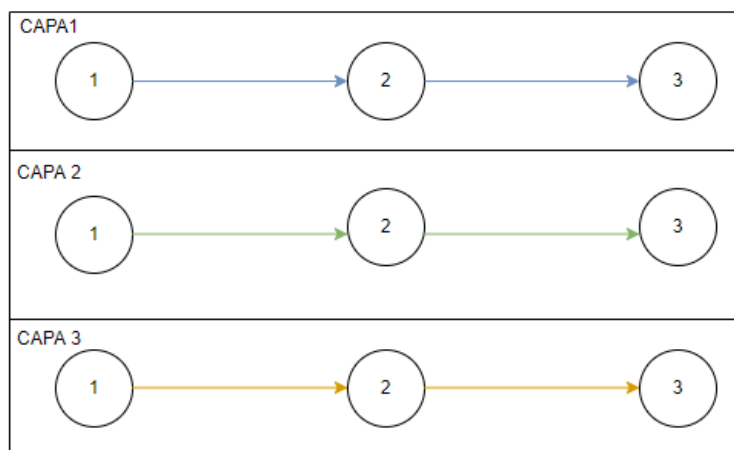


Ilustración 18 Tipos de infraestructura

En la ilustración anterior se observa que se disponen de tres tipos de infraestructuras:

1. Capa 1: infraestructura azul
2. Capa 2: infraestructura verde
3. Capa 3: infraestructura amarilla

Como se puede observar el conjunto de nodos de la red base pertenecen a las diferentes capas y la solución que se quiere conseguir es una red con una combinación de infraestructuras, a continuación, se muestra tres tipos de soluciones que pudiese aportar la metodología:

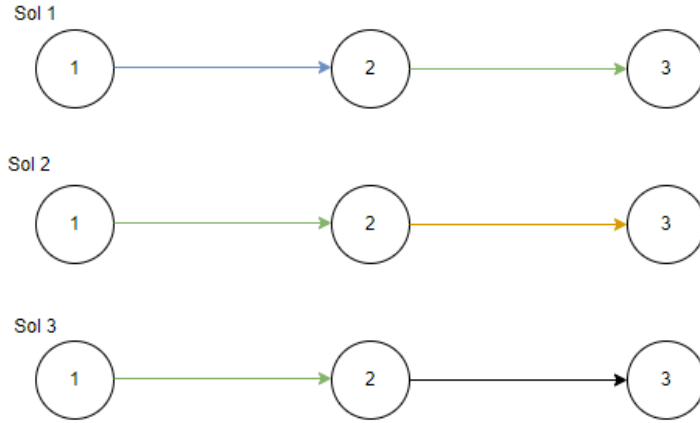


Ilustración 19 Soluciones alternativas

En primer lugar, la solución 1, consta de una red que emplea el uso de la infraestructura azul y verde. Para la solución 2, la red consta de la utilización de la infraestructura verde y amarilla, y finalmente la solución 3 consta de la utilización de la infraestructura verde y por otro lado la alternativa de no emplear ningún tipo de infraestructura.

4.2.2.2 Fase 2: Análisis de la estructura de las ecuaciones del modelo: El objetivo de esta fase es preparar de forma conveniente las ecuaciones del modelo para la fase 3.

Para encontrar la solución óptima al modelo descrito anteriormente se empleó el uso de un solver comercial (Gurobi). Dicho solver, requiere la traducción del modelo matemático a un lenguaje de programación.

La restricción (3) que trata del equilibrio de flujo en cada nodo, el valor de θ_{iw} cambia en función del par OD, es decir, θ_{iw} es igual D_w si $i = o_w$, es igual a $-D_w$ si $i = d_w$ e igual a 0 en caso contrario. Es por ello, por lo que la preparación de la restricción (3) para la fase 3, queda de la siguiente forma:

$$\sum_{a \in A_i^+} x_{aw} = D_w ; \forall i = o_w, w \in W,$$

$$\sum_{a \in A_i^-} x_{aw} = -D_w ; \forall i = d_w, w \in W,$$

$$\sum_{a \in A_i^+} x_{aw} - \sum_{a \in A_i^-} x_{aw} = 0 ; \forall i \in V \setminus \{o_w, d_w\}, w \in W,$$

El modelo para la fase 3, se estructura de la siguiente forma:

$$\text{Min} \sum_{w \in W} \left(\sum_{a \in A \setminus A^C} u_a x_{aw} + \beta \sum_{a \in A^{CB}} x_{aw} \right) \quad (1)$$

$$\text{s. a.} \quad \sum_{a \in A \setminus A^C} c_a y_a \leq P \quad (2)$$

$$\sum_{a \in A_i^+} x_{aw} = D_w; \quad \forall i = o_w, w \in W, \quad (3)$$

$$\sum_{a \in A_i^-} x_{aw} = -D_w; \quad \forall i = d_w, w \in W, \quad (4)$$

$$\sum_{a \in A_i^+} x_{aw} - \sum_{a \in A_i^-} x_{aw} = 0 \quad \forall i \in V \setminus \{o_w, d_w\}, w \in W, \quad (5)$$

$$x_{aw} \leq D_w * y_a \quad \forall a \in A \setminus A^C, w \in W, \quad (6)$$

$$\sum_{s \in M(a)} y_s = 1 \quad \forall a \in A \setminus A^C, \quad (7)$$

$$y_a \in \{0,1\} \quad \forall a \in A \setminus A^C, \quad (8)$$

$$x_{aw} \geq 0 \quad \forall a \in A, w \in W \quad (9)$$

.2.2.3 Fase 3: Escritura de la metodología y red multicapa:

En este apartado constará de dos partes. En la primera se explica el código utilizado para la metodología y en la segunda parte se explica la creación de la estructura multicapa.

.2.2.3.1 Escritura de la metodología:

1. Importar la librería y creación del modelo en Gurobi:

Para importar toda la librería de Gurobi se debe escribir la siguiente línea de código:

```
1. from gurobipy import *
```

Para crear el modelo, se utiliza una función constructora (*Model*) tal y como se muestra a continuación:


```
2. m = Model("Network_City")
```

El argumento de la función es el nombre del modelo en el que se está trabajando (Network_City).

2. Introducir las variables de decisión:

Las variables de decisión en el modelo son:

- i. y_a es una variable binaria que representa la construcción o no de la infraestructura del arco $a \in A \setminus A^C$.
- ii. x_{aw} es una variable real no negativa que representa el flujo del par OD $w \in W$ sobre el arco $a \in A$.

En el modelo matemático, las variables de decisión se estructuran de la siguiente forma:

$$y_a \in \{0,1\}; \forall a \in A \setminus A^C,$$

$$x_{aw} \geq 0; \forall a \in A, w \in W$$

Para definir las variables de decisión del modelo en lenguaje de programación se utiliza la función *addVars*, tal y como se muestra a continuación:

```
3. x = m.addVars(A,W, vtype=GRB.INTEGER, name='x')
4. y = m.addVars(A1, vtype=GRB.BINARY, name='y')
```

Mediante la función *addVars* se agregan las variables de decisión al modelo, la función recibe como argumento los conjuntos a los que pertenecen dichas variables, por otro lado, también se debe especificar el tipo de variable (binaria, continua, etc.) y finalmente el nombre de dicha variable.

3. Creación de la función objetivo:

Para definir la función objetivo se emplea el uso de la función *setObjective*, para los operadores de sumatorias se emplea el uso de la función *quicksum*.

$$\text{Min} \sum_{w \in W} \left(\sum_{a \in A \setminus A^C} c_a * x_{ak} + \beta \sum_{a \in A^{CB}} x_{ak} \right)$$

A continuación, se muestra la forma que toma la función objetivo en lenguaje de programación:

```
1. m.setObjective((quicksum((U[i,j]*x[i,j,t,v] for i,j in A1 for t,v in W)))+(quicksum(B[i,j]*x[i,j,t,v] for i,j in arcCB for t,v in W)))
```

4. Creación de las restricciones:

Restricciones de balance de flujo:

$$\sum_{a \in A_i^+} x_{aw} = D_w; \quad i = o_w, w \in W,$$

$$\sum_{a \in A_i^-} x_{aw} = -D_w; \quad i = d_w, w \in W,$$

$$\sum_{a \in A_i^+} x_{aw} - \sum_{a \in A_i^-} x_{aw} = 0; \quad \forall i \in V \setminus \{o_w, d_w\}, w \in W,$$

Para crear las restricciones, se debe emplear el uso de la función `addConstrs`, las tres líneas de código que se muestra a continuación representan la restricción de balance de flujo a través de los arcos de la red:

```
1. m.addConstrs(((quicksum(x[i,j,t,v] for i,j in arcos_0[t,v]))==D[t,v] for t,v in W),
name='Flujo_Norigen')
2. m.addConstrs(((quicksum(x[i,j,t,v] for i,j in arcos_D[t,v]))==D[t,v] for t,v in W),
name='Flujo_Ndestino')
3. m.addConstrs((-quicksum(x[n,j,t,v] for i,j in Aii[t,v] if (n==i)) + quicksum(x[i,n,t,v] for
i,j in Aii[t,v] if (n==j)))==0 for n in nodosf for t,v in W), name='Flujo_Nintermedios')
```

A continuación, se muestra la restricción, que representa la limitación de presupuesto:

$$\sum_{a \in A \setminus A^C} c_a y_a \leq P$$

La forma que toma la restricción en lenguaje de programación se detalla a continuación;

```
1. m.addConstr(quicksum((y[i,j]*C[i,j] for i,j in arcI ) <= P),
name='Limitación_Presupuesto_Construcción')
```

Seguidamente, se muestra la restricción que especifica que se debe asignar un solo tipo de infraestructura a cada arco:

$$\sum_{s \in M(a)} y_s = 1; \quad \forall a \in A \setminus A^C,$$

A continuación, se muestra la forma que toma la restricción en lenguaje de programación:

```
1. m.addConstrs((quicksum(y[i,j] for i,j in M[l,m] ) == 1 for l,m in arc),
name='Limitación_Tecnologíaporarcos')
```

Finalmente, se muestra la restricción que establece que el flujo sobre un arco dado está sujeto a la existencia de la infraestructura en ese arco:

$$x_{aw} \leq D_w * y_a ; \forall a \in A \setminus A^C, w \in W,$$

A continuación, se muestra la forma que toma la restricción en lenguaje de programación:

```
1. m.addConstrs((x[i,j,t,v]<=y[i,j]*D[t,v] for i,j in A1 for t,v in W), name='Existencia
Infraestructura_Sujeta a flujo')
```

Para buscar la solución óptima se debe emplear el uso de la función *optimize*, la función *display* para que imprima los resultados y la función *getAttr* muestra los valores que toma las variables de decisión del modelo.

```
2. m.optimize()
3. m.display()
4. sol = m.getAttr('x', y)
```

La metodología recibe los siguientes parámetros de entrada:

1. Una red base, es decir, un grafo dirigido con sus respectivos conjuntos de nodos y arcos, siendo que la estructura del conjunto de los arcos debe tener la forma de un diccionario mientras que la estructura de los nodos debe tener la forma de una lista.
2. Una matriz D que representa la demanda de viajes entre nodos del grafo en un horizonte temporal dado. Debe tener la forma de un diccionario, tiene como “clave” el par OD y como “valor” la demanda de ese viaje OD.
3. El conjunto de los pares origen-destino del horizonte temporal W , es decir, de los escenarios que se van a estudiar. Debe tener la forma de una lista.
4. El importe del presupuesto P para la construcción de la red.
5. Un conjunto K de diferentes infraestructuras para construir la red de bicicletas. Debe tener la forma de una lista.
6. Las características propias de los arcos, es decir, el coste de construcción, el coste de usuario y penalización diferenciado por infraestructuras. Todos deben tener la forma de un diccionario, es decir, como “clave” el arco y como “valor” el valor respectivo de cada característica.

Debido a que el coste de construcción va en función de la longitud del arco, al programa se le indicó los precios por metros, por lo que para calcular el coste de todos los arcos debe multiplicar dicho coste por la longitud del arco tal y como se muestra a continuación:

```

1. #Coste de contruccion
2.
3. CB=arc.copy()
4. for i in CB:
5.     CB[i]=0 # Red Base no consume coste de construcción
6. #Coste de construcción K1
7. CT1=arcT1.copy()
8. for i in CT1:
9.     CT1[i]=c_T1*long_dT1[i[0],i[1]] #Coste por longitud K=1
10. #Coste de construcción K2
11. CT2=arcT2.copy()
12. for i in CT2:
13.     CT2[i]=c_T2*long_dT2[i[0],i[1]] #Coste por longitud K=2
14. ##### Coste de construccion #####
15. C={} # coste total de construcción
16. C.update(CB)
17. C.update(CT1)
18. C.update(CT2)

```

.2.2.3.2 Red multicapa:

A partir del conjunto de nodos y arcos de la red base se construye la red multicapa.

Para conformar el conjunto de nodos de la red multicapa se tomó el conjunto de nodos de la red base y se “copió” tantas veces como tecnologías de construcción se tuviera. Por ejemplo: si se tiene una red base con un conjunto de nodos $\hat{V} = (1,2,3)$ y un conjunto de tecnología $K = (1,2)$, se crea para cada tecnología un conjunto de nodos V_1 y V_2 . Siendo V_1 el conjunto de nodos que pertenece a la tecnología 1 o lo que es lo mismo a la capa 1 y V_2 el conjunto de nodos que pertenece a la capa o tecnología 2, estos nuevos conjuntos tienen la misma forma y tamaño que \hat{V} . Finalmente, el conjunto de nodos de la red multicapa queda de la siguiente forma: $V = (\hat{V} \cup V_1 \cup V_2)$.

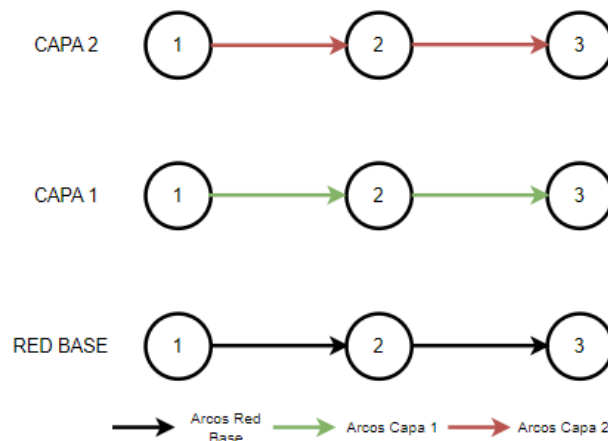


Ilustración 20 Estructura multicapa (nodos)

Para poder diferenciar el nodo 1 de la red base respecto al nodo 1 de la capa 1 y al nodo 1 de la capa 2. Se llevó a cabo un tratamiento para renombrar los nodos de V_1 y V_2 respectivamente. El tratamiento consistió en que los nodos de los conjuntos nuevos debían continuar la numeración de los nodos de la red base. De esta forma si el último nodo de la red se denomina “m” entonces el primer nodo de la red de la capa 1 se debía renombrar por $m+1$. El primer nodo de la capa 2 se debía renombrar por $2m+1$.

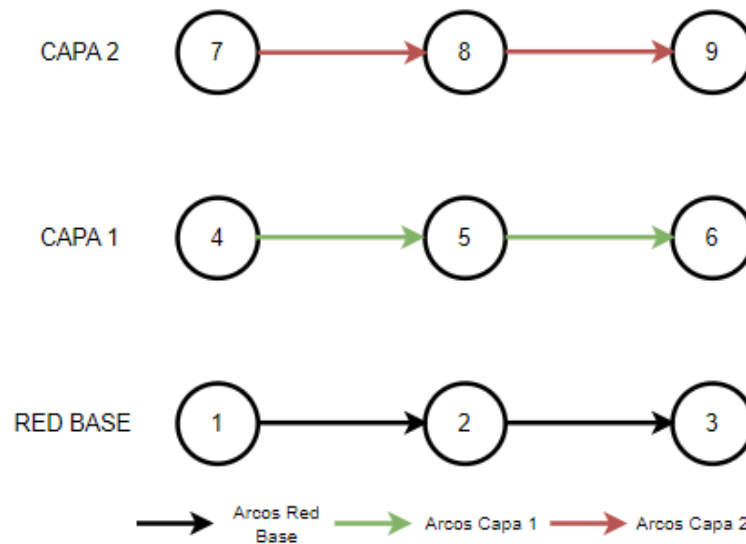


Ilustración 21. Estructura multicapa (arcos capas).

De esta forma ya queda definido el conjunto de nodos de la red multicapa $V = (1,2,3,4,5,6,7,8,9)$. Por otro lado, también se define el conjunto de arcos pertenecientes a cada tecnología, por ejemplo, para la tecnología 1 el conjunto de arcos queda de la siguiente forma $A^1 = \{(4,5), (5,6)\}$ y para la tecnología 2 el conjunto de arcos queda de la siguiente forma $A^2 = \{(7,8), (8,9)\}$. A continuación, se muestra las líneas de código utilizada para llevar a cabo lo descrito anteriormente:

```

1. ##### DEFINIMOS NODOS #####
2. nodosT11=[]
3. nodosT11.extend(nodos)
4. nodosT22=[]
5. nodosT22.extend(nodos) ### REPETIR TANTAS TECNOLOGÍAS SE TENGA
6. ##### Para poder diferenciar los nodos de las diferentes tecnologías le damos continuidad
7. # a la numeración.
8. nodosT1=list(range(len(nodos)+1,(2*(len(nodos))+1)))
9.
10. nodosf11= nodos+ nodosT1 # Replicar en caso de tener más tecnologías
11. nodosT2=list(range(len(nodosf11)+1,(3*(len(nodos))+1)))
    
```

Tal y como se ha comentado en apartados anteriores, en la red multicapa existe un conjunto de arcos denominados arcos de cambio de tecnología. Estos arcos unen entre si a los nodos de la capa base y los nodos de las diferentes capas, formando un conjunto representado como A^C , o arcos de cambio. Es importante señalar que estos arcos unen nodos consecutivos en $N(i)$, es decir, si i está en la capa k e i' en la capa $k + 1$, existen dos arcos de cambio, uniendo i e i' como arco de subida e i' con i como arco de bajada o de disminución de la

calidad de construcción.

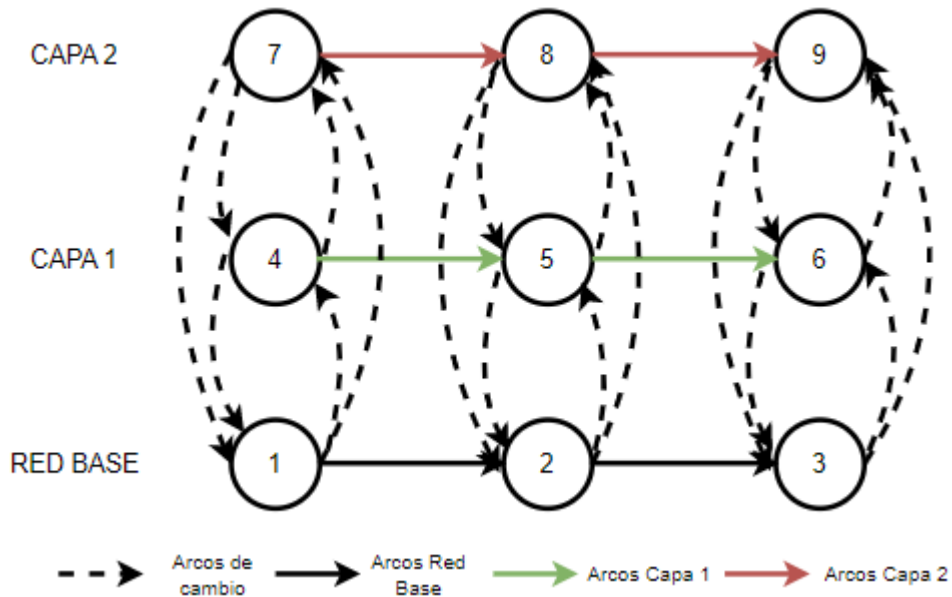


Ilustración 22 Estructura multicapa (arcos de cambio).

De esta forma queda definido el conjunto de arcos de cambio de la red multicapa. Los arcos de subida, es decir los que cambian a una tecnología $k+1$ se definen dentro del conjunto A^{CA} y por el contrario los arcos que cambian a una tecnología $k-1$ se definen dentro del conjunto A^{CB} . Finalmente, el conjunto que contiene todos los arcos de cambio se define como A^C .

A continuación, se muestra las líneas de código utilizada para llevar a cabo lo descrito anteriormente:

```

1.  ### Aquí se crea los arcos de cambio ####
2.
3.  ### UNIMOS LOS NODOS DEL GRAFO BASE CON LOS NODOS DE K1
4.  arcosB_T1={}
5.  arcosT1_B={}
6.  for i in enumerate(nodos):
7.      for j in enumerate(nodosT1):
8.          if i[0]==j[0]:
9.              arcosB_T1[i[1],j[1]] = {}
10.             arcosT1_B[j[1],i[1]] = {}
11. #print(arcosB_T1)
12.
13. ### UNIMOS LOS NODOS DEL GRAFO BASE CON LOS NODOS DE K2
14. arcosB_T2={}
15. arcosT2_B={}
16. for i in enumerate(nodos):
17.     for j in enumerate(nodosT2):
18.         if i[0]==j[0]:
19.             arcosB_T2[i[1], j[1]] = {}
20.             arcosT2_B[j[1], i[1]] = {}
21. #print(arcosB_T2)
22.
23. ### UNIMOS LOS NODOS DEL GRAFO K1 CON LOS NODOS DE K2
24. arcosT1_T2={}
25. arcosT2_T1={}
26. for i in enumerate(nodosT1):
27.     for j in enumerate(nodosT2):

```

```

28.     if i[0]==j[0]:
29.         arcosT1_T2[i[1], j[1]] = {}
30.         arcosT2_T1[j[1], i[1]] = {}
31. #print(arcosT1_T2)
32.
33. ### CONJUNTO DE ARCOS QUE CONECTA A LAS TECNOLOGÍAS
34.
35. arcC={}
36. arcC.update(arcosB_T1)
37. arcC.update(arcosT1_T2)
38. arcC.update(arcosB_T2)
39. arcC.update(arcosT1_B)
40. arcC.update(arcosT2_T1)
41. arcC.update(arcosT2_B)
42.
43. arcCB={} #PENALIZADOS
44. arcC.update(arcosT1_B)
45. arcC.update(arcosT2_T1)
46. arcC.update(arcosT2_B)
47.

```

Por otro lado, una vez que se resuelve el modelo, los resultados deben tratarse para obtener una representación gráfica de la red tal que se pueda observar la red base con los caminos señalados por el tipo de infraestructura a construir. Para realizar esta operación se emplea el uso de las siguientes líneas de códigos:

```

1. #####SE NECESITA SABER CUALES SON LOS ARCOS QUE PERTENECEN A LA SOLUCIÓN
2. arcos_activos1=[]
3. for i,j in arcos_activos:
4.     for l in arcos_sol1:
5.         if ((i,j)==(l[0],l[1])):
6.             arcos_activos1.append([i,j])
7.
8. ##### GRAFICA DE LA RED #####
9. #####
10.
11. print('>>PREPARACIÓN DE DATOS PARA GRAFICAR')
12. #####
13. # diccionario nodos
14. n_grafo = {}
15. for i in nodos:
16.     n_grafo = nodos.copy()
17.
18. e = {}
19. for i in range(1, len(n_grafo)+1):
20.     e[i] = str(i)
21.
22. arc_grafo={}
23. for i in arcos_activos1:
24.     for j in M:
25.         if (i[0],i[1]) == M[j][0]:
26.             arc_grafo.setdefault(j,1)
27.         if (i[0],i[1]) == M[j][1]:
28.             arc_grafo.setdefault(j,2)
29.         if (i[0], i[1]) == M[j][2]:
30.             arc_grafo.setdefault(j, 0)
31.
32. ##### COORDENADAS DE LOS NODOS #####
33.
34. for i in nodosXY:
35.     nodosXY[i]=(D_CoordenadasX[i-1],D_CoordenadasY[i-1])
36. ##### REALIZACIÓN DE LA GRAFICA ####
37.
38. print('>>INICIO DE GRAFICA')
39.
40. G = nx.Graph()
41. nx.draw_networkx_nodes(G,nodosXY, n_grafo,node_size=1,node_color="black") ###RECIBE
    DICIONARIO NODOXY Y LA LISTA DE NODOS
42. nx.draw_networkx_labels(G, pos=nodosXY, labels=e, font_color="black",font_size=7)
43. nx.draw_networkx_edges(G, nodosXY, arc, edge_color="silver",arrowstyle="-")

```

```

44. #print(nodosXY, '<-')
45. k=0
46.
47. for i in arc_grafo:
48.
49.     if arc_grafo[i]==1:
50.         arc_grafo_list=list(arc_grafo.keys())
51.
52.         nx.draw_networkx_edges(G,nodosXY,[arc_grafo_list[k]],edge_color="green",arrowstyle="-
53.         |>",arrows=False)
54.
55.     if arc_grafo[i]==2:
56.
57.         arc_grafo_list = list(arc_grafo.keys())
58.         nx.draw_networkx_edges(G, nodosXY,[arc_grafo_list[k]],
59.         edge_color="red",arrowstyle="-|>",arrows=False)
60.
61.     if arc_grafo[i]==0:
62.
63.         arc_grafo_list = list(arc_grafo.keys())
64.         nx.draw_networkx_edges(G, nodosXY,[arc_grafo_list[k]],
65.         edge_color="black",arrowstyle="-|>",arrows=False)
66.
67.     k=k+1
68. if DIBUJARED == 1:
69.     plt.show()

```

En primer lugar, se determina cuáles son los arcos que pertenecen a la solución, a continuación, para realizar la representación gráfica de la red. Para ello se define el grafo empleando la función *Graph*, a continuación, con la función *draw_networkx_nodes* el programa grafica los nodos. Esta función recibe como argumentos: el grafo definido y las coordenadas de los nodos. La función *draw_networkx_labels* asigna la enumeración a los nodos, para ello recibe el grafo definido y un diccionario que contine como “clave” a los nodos del grafo y como “valor” la etiqueta o numeración correspondiente del nodo.

Para graficar los arcos se emplea la función *draw_networkx_edge* que recibe como argumentos el grafo definido y el conjunto total de arcos, en esta función se especifica que todos estos arcos deben pintarse en la solución gráfica de color negro.

Una vez que el programa tiene generado el grafo original, mediante la implementación de un bucle *for* repasa el listado de arcos que pertenecen a la solución y le cambia el color si dicho arco pertenece al conjunto solución. Para diferenciar los arcos por el tipo de infraestructura, se le asigna un color u otro dependiendo de la infraestructura de dicho arco.

Finalmente, se emplearon las siguientes líneas de códigos para imprimir los resultados:

```

1. ##### IMPRIME RESULTADOS #####
2. arcos_activos2=[]
3. for i,j in arcos_activos1:
4.     if (i<500 ):
5.         if (j<600):
6.             arcos_activos2.append([i,j])
7.
8. Cont=0

```



```
9. for i in arcos_activos2:
10.     Cont=C[tuple(i)]+Cont
11. print('El presupuesto disponible ----->',P,'u.m.')
12. print('Coste total de construcción de la red fue ----->', Cont , 'u.m.')
13. #print("Coste total de viaje del usuario")
14.
15. Cont1=0
16. for i in arcos_activos2:
17.     Cont1=U[tuple(i)]+Cont1
18.     #print("U" + str(i) + " =" + str(U[tuple(i)]))
19. print('Coste total de viaje del usuario:----->', Cont1 , 'u.m.')
20.
21. #print('Utilizacion del presupuesto')
22. Cont6=(Cont/P)*100
23. print('Se ha utilizado----->-----', Cont6, '% del presupuesto')
24. print('La red cuenta con', Cont10 , 'carriles con infraestructura 0')
25. print('La red cuenta con', Cont4 , 'carriles con infraestructura 1')
26. print('La red cuenta con', Cont5 , 'carriles con infraestructura 2')
27.
```

Mediante el código anterior se puede conocer el coste total de construcción de la red, el coste total de los viajes de usuario, el número de carriles construidos y la diferenciación por infraestructura, finalmente se imprime el porcentaje de presupuesto utilizado.

De esta forma, se concluye la fase de desarrollo de la metodología. En los próximos apartados se explica las pruebas llevadas a cabo.

4.3 FUNCIONALIDAD DE LA METODOLOGÍA

Para testar la funcionalidad de la metodología se realizaron dos pruebas: Caso 1 y Caso 2, en ambos casos se emplearon redes diferentes.

Caso 1: Red 6 nodos: Con esta red, se llevaron a cabo 3 pruebas diferentes variando el presupuesto:

A continuación, se indican los parámetros de entrada para el programa:

Como se ha mencionado anteriormente, la metodología se alimenta de una red base, para ello se definió una red G que contiene un conjunto de 6 nodos y un conjunto de 7 arcos, dicha red tiene la siguiente forma:

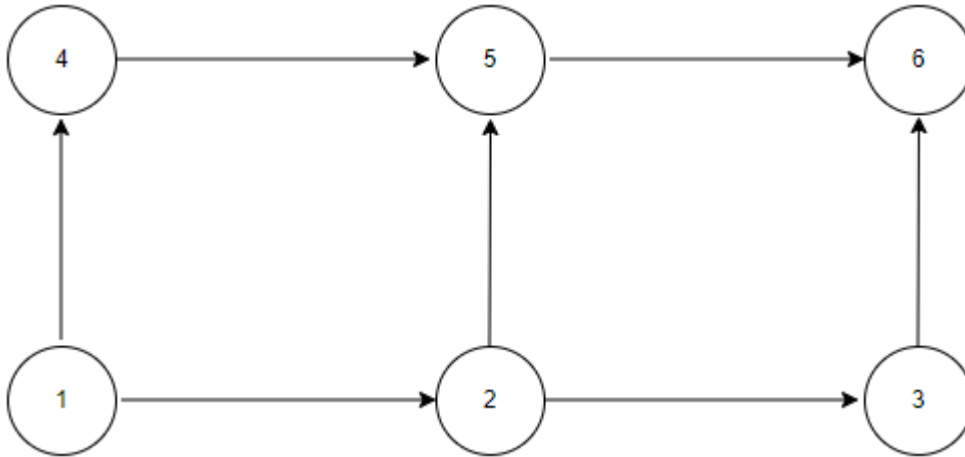


Ilustración 23. Red G base.

De la red \hat{G} se obtienen los siguientes conjuntos: un conjunto de nodos $\hat{V} = (1, 2, 3, 4, 5, 6)$ y un conjunto de arcos $\hat{A} = \{(1,2), (2,3), (4,5), (5,6), (1,4), (2,5), (3,6)\}$.

Por otra parte, el programa precisa de una matriz D que representa la demanda de viaje por los arcos, para este problema los valores de la demanda son los siguientes:

Tabla 2 Matriz de demanda D de viajes por los arcos de la red G

NODOS	1	2	3	4	5	6
1	0	100	700	300	300	200
2	900	0	100	100	900	100
3	700	800	0	300	500	200
4	300	200	400	0	300	400
5	300	900	300	200	0	100
6	200	600	500	400	100	0

En la red \hat{G} , el planificador central puede decidir cuáles son el conjunto de pares origen-destino del viaje W . En este caso, para una mayor cobertura de la red se empleó el siguiente conjunto: $W = \{(1,5), (1,3), (2,6), (4,6)\}$, con este conjunto lo que se quiere conseguir es una red satisfaga la demanda de estos caminos, es decir, buscar el camino o los caminos para conectar el nodo 1 con los nodos 3, 5, conectar los nodos 2 y 4 con el nodo 6 en función de las restricciones del modelo, tal y como se muestra a continuación:

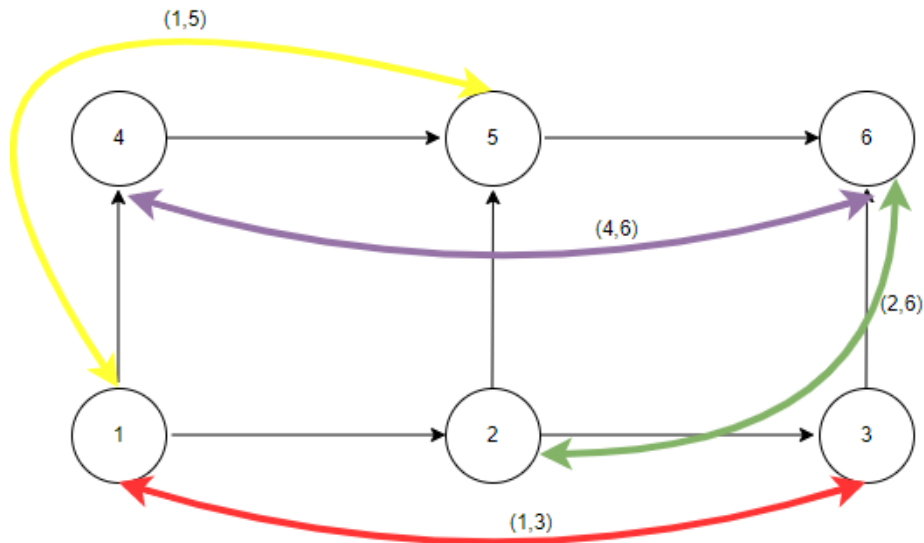


Ilustración 24 Conjunto de pares OD para un conjunto $W = \{(1,3), (1,5), (2,6), (4,6)\}$.

Por otro lado, el programa requiere de un presupuesto de construcción, para la red se emplearon tres tipos de presupuesto P : uno “ajustado”, uno “intermedio” y uno “generoso”. El importe del presupuesto ajustado fue de 700.000 u.m., el presupuesto intermedio fue de 1.000.000 u.m. y por último el presupuesto generoso fue de 3.000.000 u.m. Con esto se quiere probar el comportamiento del modelo en función de la variación del presupuesto.

En referencia al conjunto de infraestructuras k , el planificador central es quien decide la cantidad y la tipología de estas. Para la red \hat{G} el conjunto de infraestructuras a analizar es el siguiente: $k = \{1, 2\}$, siendo la infraestructura 1 más barata en término de construcción, pero más cara en cuanto al coste de viaje del usuario respecto a la infraestructura 2, en otras palabras, la infraestructura 2 es mejor que la infraestructura 1 en término de coste de viaje del usuario. Como se ha comentado anteriormente, el modelo podrá incorporar la opción de no realizar ninguna construcción de infraestructura. Esta decisión no consume presupuesto, pero si repercute en el tiempo de viaje del usuario. En adelante esta la opción de no construir se denominará infraestructura 0.

Finalmente se indican el coste de usuario y coste de construcción en función de la infraestructura:

Tabla 3 Coste de construcción y usuario en función del tipo de infraestructura.

INFRAESTRUCTURA	COSTE DE CONSTRUCCIÓN ARCOS (U. MONETARIA/U. LONGITUD)	COSTE DE USUARIO (TIEMPO)
0	0	50
1	100	20
2	250	5

De la tabla anterior se obtiene la siguiente información:

- i. La infraestructura 2 es mejor que la 1 en términos de costes de viaje de usuario puesto el coste es menor

para esta infraestructura que para la 1. Por otro lado, se observa que la infraestructura 0 es la peor opción para el usuario puesto que tiene un coste mayor que la infraestructura 1 y 2.

- ii. En términos de coste de construcción, la infraestructura 1 es mejor que la 2 puesto que la infraestructura 1 tiene un coste menor que en la infraestructura 2. Por otro lado, la infraestructura 0 es la mejor opción ya que no consume presupuesto.

Para cuantificar la repercusión que los usuarios de bicicletas puedan experimentar durante el cambio de tecnología se empleará el término β . En particular este término penaliza el cambio de tecnología a peor. Esta penalización representaría la “incomodidad” causada al usuario de la red por el cambio de infraestructura en su trayecto. A continuación, se indica las penalizaciones utilizadas:

Tabla 4 Penalización por cambio de infraestructura

CAMBIO DE INFRAESTRUTURA	PENALIZACIÓN β
DE LA INFRAESTRUTURA 1 A LA 0	1
DE LA INFRAESTRUTURA 2 A LA 0	3
DE LA INFRAESTRUTURA 2 A LA 1	2

De acuerdo con lo expuesto anteriormente los conjuntos de la estructura multicapa serían los siguientes: $G = (V, A)$; siendo el conjunto de nodos $V = (\hat{V} \cup V^1 \cup V^2)$:

- $\hat{V} = (1,2,3,4,5,6)$; corresponde al conjunto de nodos de la red base.
- $V^1 = (7,8,9,10,11,12)$; corresponde al conjunto de nodos de la capa 1.
- $V^2 = (13,14,15,16,17,18)$; corresponde al conjunto de nodos de la capa 2.

Siendo $A = \{A^{RedBase} \cup A^{k=1} \cup A^{k=2} \cup A^C \cup A^{OD}\}$

- $A^{RedBase} = \{(1,2), (2,3), (4,5), (5,6), (1,4), (2,5), (3,6)\}$
- $A^1 = \{(7,8), (8,9), (10,11), (11,12), (7,10), (8,11), (9,12)\}$
- $A^2 = \{(13,14), (14,15), (16,17), (17,18), (13,16), (14,17), (15,18)\}$
- $A^C = \{A^{CA} \cup A^{CB}\}$
- A^{OD}

A continuación, se muestra los resultados de las pruebas realizadas con los diferentes presupuestos establecidos:

3.3.1 PRUEBA 1: Presupuesto P=700.000 u.m.

Para un presupuesto de red de 700.000 u.m., se genera una red para los viajes $W = \{(1,5), (1,3), (2,6), (4,6)\}$ con las siguientes características:

- La red cuenta con 4 carriles con infraestructura 0. Estos carriles no consumen presupuesto de

construcción, pero si incrementan el coste del usuario. En la solución gráfica, estos carriles se representan con el color negro.

- La red cuenta con 2 carriles con infraestructura 1. Estos carriles consumen presupuesto de construcción y son la mejor alternativa desde el punto de vista del usuario. En la solución gráfica estos carriles se representan con el color verde.
- El coste de construcción fue de 600.000 u.m. lo que representa un 85.71% del presupuesto total.
- El coste total de para los usuarios de esta red fue: 240 unidades de tiempo.

A continuación, se muestra el resultado gráfico de la prueba 1:

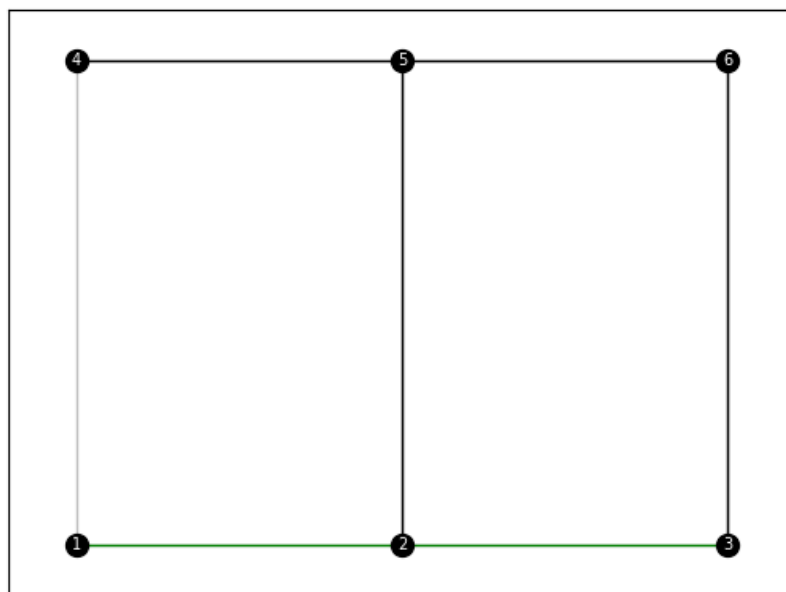


Ilustración 25 Solución Prueba 1: Presupuesto $P = 700.000$ u.m.

En la ilustración anterior se observa que para los arcos (2,5),(3,6),(4,5) y (5,6) se utiliza la infraestructura 0 o lo que es lo mismo no se construye nada sobre esos arcos. Esta infraestructura no consume presupuesto, pero es la peor opción desde el punto de vista del usuario. En relación con los arcos (1,2) y (3,4) se utiliza la infraestructura 1, estos arcos consumen presupuesto, pero beneficia al usuario ya que el coste de este a través de estos arcos es menor que para los arcos de la infraestructura 0. Nótese que existe un arco (1,4) de color gris, esto se refiere a que el modelo no ha contemplado este arco para la solución del problema. En la tabla que se muestra a continuación se detalla los arcos pertenecientes a la solución y la infraestructura a la cual pertenecen:

Tabla 5 Solución Prueba 1: Presupuesto $P = 700.000$ u.m.

ARCOS SOLUCIÓN	INFRAESTRUCTURA
$\gamma(4,5)$	0
$\gamma(5,6)$	0
$\gamma(2,5)$	0
$\gamma(3,6)$	0
$\gamma(1,2)$	1
$\gamma(2,3)$	1

A continuación, se muestra un desglose de la solución generada para cada uno de los viajes del conjunto W :

1- Viaje $W = (1,5)$ con una demanda $D_{1,5} = 300$:

Tabla 6 Solución para el viaje $W = (1,5)$

VARIABLE FLUJO	VALOR DEL FLUJO	TIPO DE ARCO
$x[501,7,1,5]$	300	OD
$x[7,8,1,5]$	300	Tecnología 1
$x[8,2,1,5]$	300	Transferencia
$x[2,5,1,5]$	300	Red Base
$x[5,11,1,5]$	300	Transferencia
$x[11,601,1,5]$	300	OD

A continuación, se muestra el recorrido del flujo a través de la estructura multicapa:

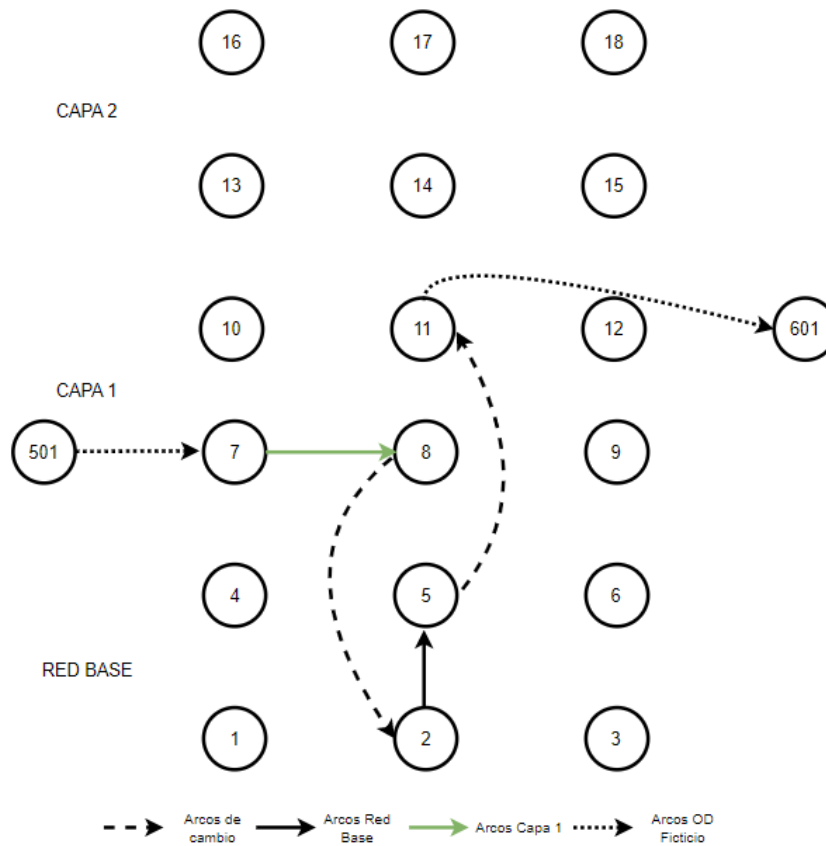


Ilustración 26 Solución gráfica del viaje $W = (1,5)$ en la estructura multicapa

De la ilustración anterior, se observa que para el viaje $W = (1,5)$ se utiliza un arco con la infraestructura 1 y uno con la infraestructura cero. El arco (7,8) que pertenece a la infraestructura 1 es equivalente arco (1,2) en la red base. La solución final para este viaje se representaría de la siguiente forma:

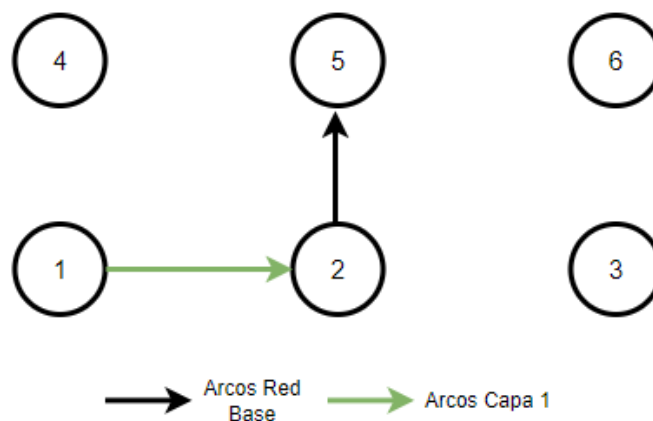


Ilustración 27 Solución grafo final para el viaje $W = (1,5)$

El coste del usuario para realizar este viaje se muestra a continuación:

Tabla 7 Coste del usuario (en unidades de tiempo) para realizar el viaje $W = (1,5)$

COSTE DE USUARIO	
U[2, 5]	50
U[7, 8]	20
TOTAL	70

2. Viaje $W = (1,3)$ con una demanda $D_{1,3} = 700$:

Tabla 8 Solución para el viaje $W = (1,3)$

VARIABLE FLUJO	VALOR DEL FLUJO	TIPO DE ARCO
$x[503,1,1,3]$	700	OD
$x[1,7,1,3]$	700	Transferencia
$x[7,8,1,3]$	700	Infraestructura 1
$x[8,9,1,3]$	700	Infraestructura 1
$x[9,3,1,3]$	700	Transferencia
$x[3,603,1,3]$	700	OD

A continuación, se muestra el recorrido del flujo a través de la estructura multicapa:

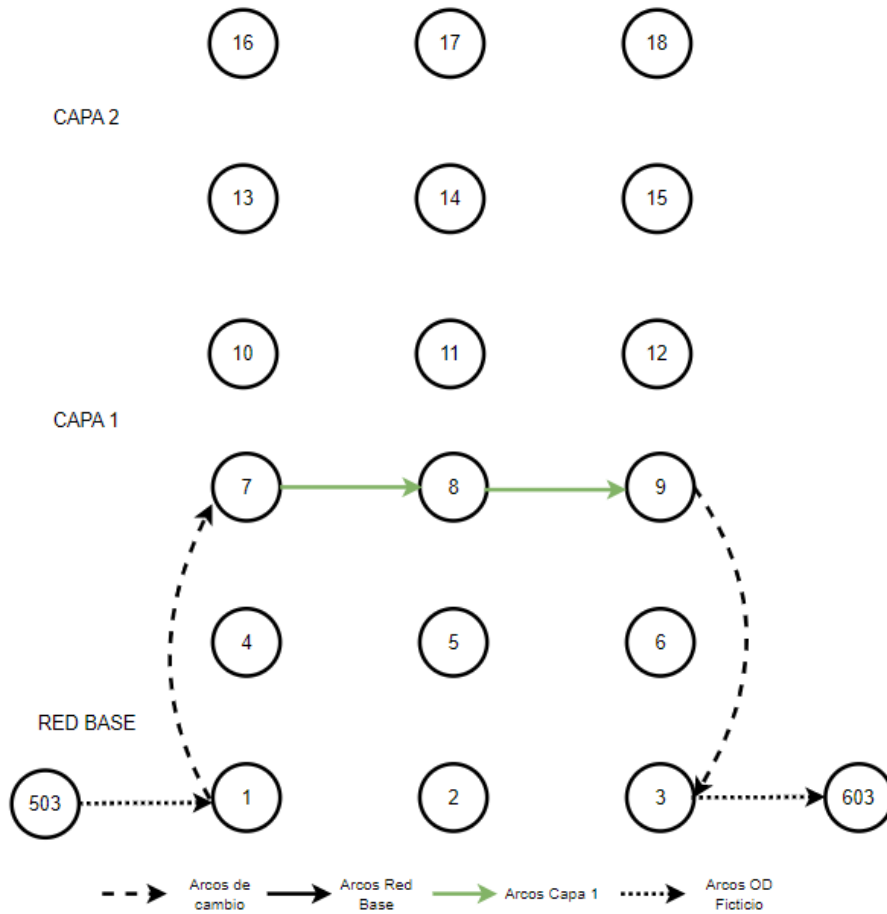


Ilustración 28 Solución gráfica del viaje $W = (1,3)$ en la estructura multicapa

De la ilustración anterior, se observa que para el viaje $W = (1,3)$ se utiliza dos arcos con la infraestructura 1. El arco (7,8) que pertenece a la infraestructura 1 es equivalente arco (1,2) en la red base, de igual forma ocurre con el arco (8,9) que es equivalente al arco (2,3) en la red base. La solución final para este viaje se representaría de la siguiente forma:

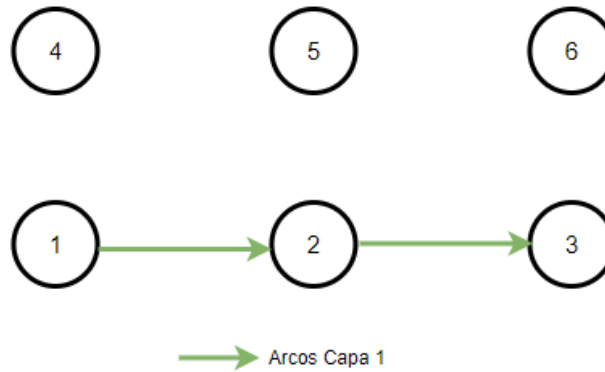


Ilustración 29 Solución grafo final para el viaje $W = (1,3)$

El coste del usuario para realizar este viaje se muestra a continuación:

Tabla 9 Coste del usuario (en unidades de tiempo) para realizar el viaje $W = (1,3)$

COSTE DE USUARIO	
U[8, 9]	20
U[7, 8]	20
TOTAL	40

3. Viaje $W = (2,6)$ con una demanda $D_{2,6} = 100$:

Tabla 10 Solución para el viaje $W=(2,6)$

VARIABLE FLUJO	VALOR DEL FLUJO	TIPO DE ARCO
x[508,14,2,6]	100	OD
x[14,8,2,6]	100	Transferencia
x[8,9,2,6]	100	Infraestructura 1
x[9,3,2,6]	100	Transferencia
x[3,6,2,6]	100	Red base
x[6,12,2,6]	100	Transferencia
x[12,18,2,6]	100	Transferencia
x[18,608,2,6]	100	OD

A continuación, se muestra el recorrido del flujo a través de la estructura multicapa:

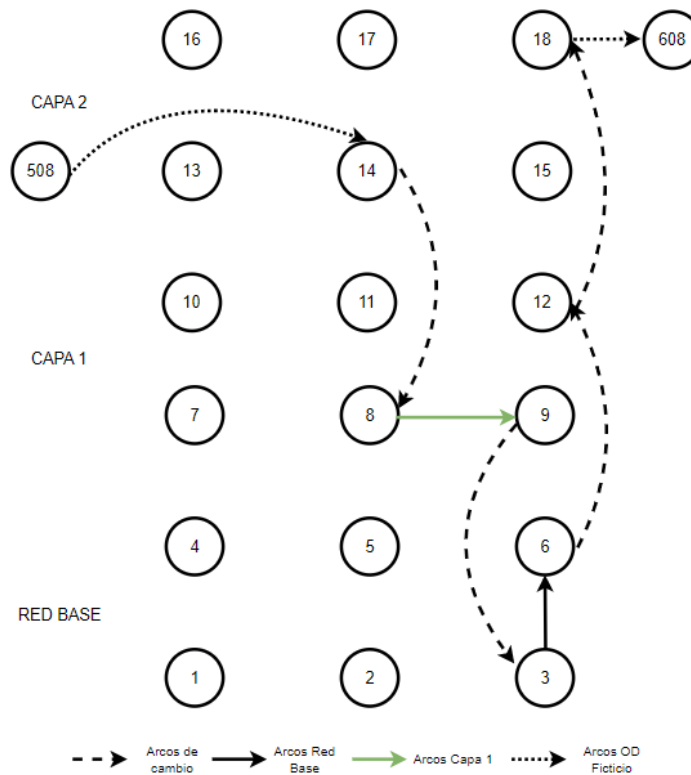


Ilustración 30 Solución grafo final para el viaje $W = (2,6)$

De la ilustración anterior, se observa que para el viaje $W = (2,6)$ se utiliza dos arcos con la infraestructura 1 y uno con la infraestructura cero. El arco (8,9) que pertenece a la infraestructura 1 es equivalente arco (2,3) en la red base. La solución final para este viaje se representaría de la siguiente forma:

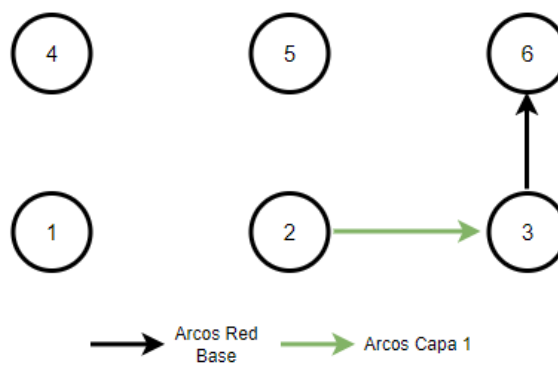


Ilustración 31 Solución grafo final para el viaje $W = (2,6)$

El coste del usuario para realizar este viaje se muestra a continuación:

Tabla 11 Coste del usuario (en unidades de tiempo) para realizar el viaje $W = (1,3)$

COSTE DE USUARIO	
U[3,6]	50
U[8,9]	20
TOTAL	70

4. Viaje $W = (4,6)$ con una demanda $D_{4,6} = 400$:

Tabla 12 Solución para el viaje $W = (4,6)$

VARIABLE FLUJO	VALOR DEL FLUJO	TIPO DE ARCO
$x[509,4,4,6]$	-400	OD
$x[4,5,4,6]$	+400	Red Base
$x[5,6,4,6]$	-400	Red Base
$x[6,609,4,6]$	+400	OD

A continuación, se muestra el recorrido del flujo a través de la estructura multicapa:

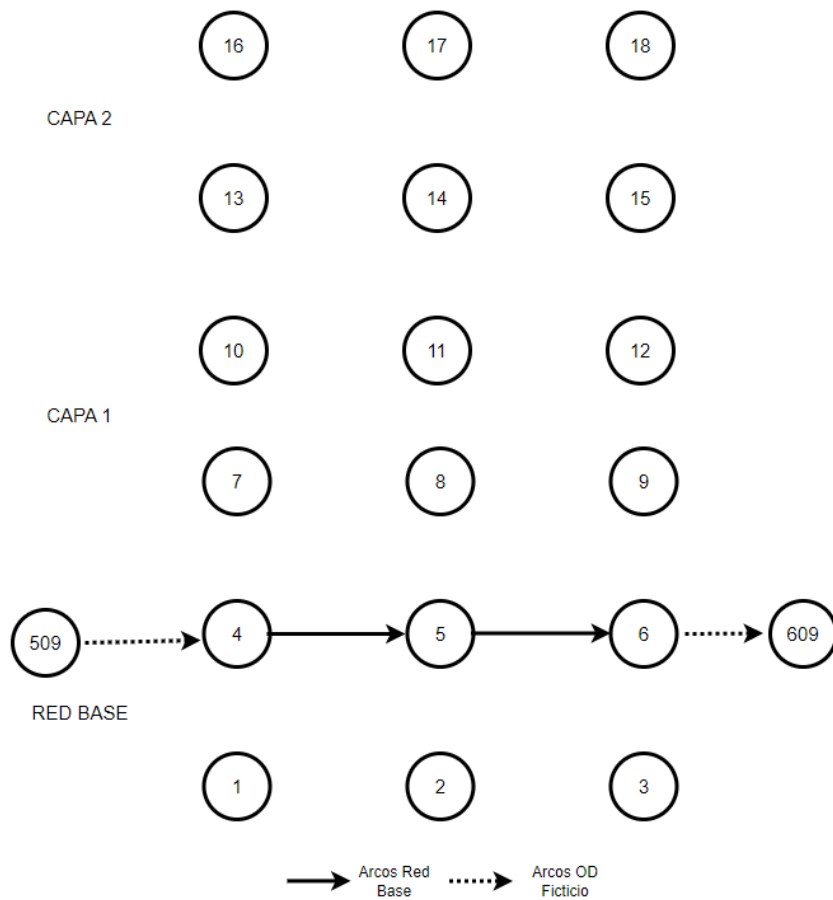


Ilustración 32 Solución grafo final para el viaje $W = (4,6)$

De la ilustración anterior, se observa que para el viaje $W = (4,6)$ se utiliza dos arcos con la infraestructura 0.

La solución final para este viaje se representaría de la siguiente forma:

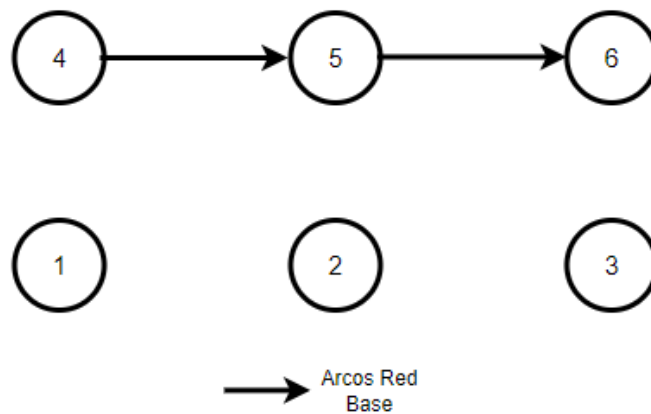


Ilustración 33 Solución grafo final para el viaje $W = (4,6)$

El coste del usuario para realizar este viaje se muestra a continuación:

Tabla 13 Coste del usuario (en unidades de tiempo) para realizar el viaje $W = (1,3)$

COSTE DE USUARIO	
U[4,5]	50
U[5,6]	50
TOTAL	100

Con lo expuesto anteriormente, se demuestra el cumplimiento de la demanda para cada viaje del conjunto W . En adelante para cada prueba sólo se indicará la tabla que contiene los arcos pertenecientes a la solución y su representación gráfica.

3.3.2 PRUEBA 2: Presupuesto $P=1.000.000$ u.m.

Para un presupuesto de red de $P = 1.000.000$ u.m., se genera una red para los viajes $W = \{(1,5), (1,3), (2,6), (4,6)\}$ con las siguientes características:

- La red cuenta con 2 carriles con infraestructura 0. Estos carriles no consumen presupuesto de construcción, pero si incrementan el coste del usuario. En la solución gráfica, estos carriles se representan con el color negro.
- La red cuenta con 3 carriles con infraestructura 1. Estos carriles consumen presupuesto de construcción y son la mejor alternativa desde el punto de vista del usuario. En la solución gráfica estos carriles se representan con el color verde.
- El coste de construcción fue de 900.000 u.m. lo que representa un 90% del presupuesto total.

- El coste total de para los usuarios de esta red fue: 160 unidades de tiempo.

A continuación, se muestra el resultado gráfico de la prueba 1:

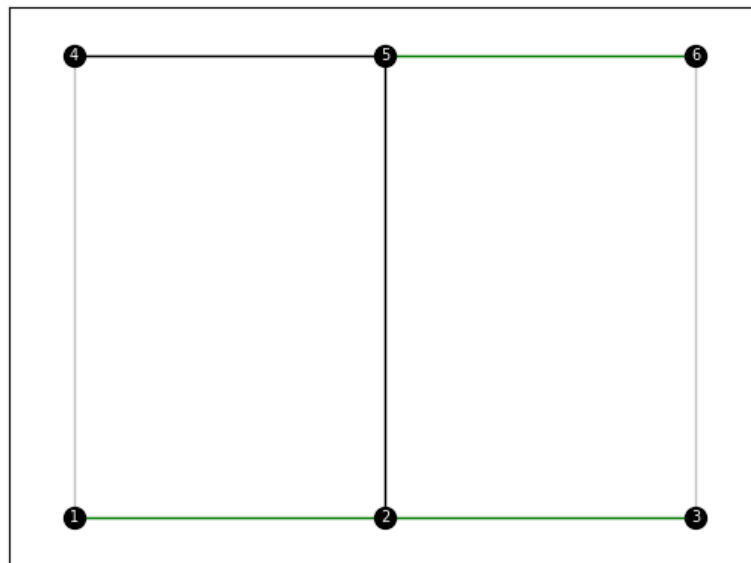


Ilustración 34 Solución Prueba 2: Presupuesto $P = 1.000.000$ u.m.

De la ilustración anterior, se observa que los arcos (1,2),(2,3) y (5,6) se construirán con la infraestructura 1 y los arcos (4,5) y (2,5) con la infraestructura 0.

Tabla 14 Solución Prueba 2: Presupuesto $P = 1.000.000$ u.m.

ARCOS SOLUCIÓN	INFRAESTRUCTURA
$\gamma(4, 5)$	0
$\gamma(2, 5)$	0
$\gamma(1, 2)$	1
$\gamma(2, 3)$	1
$\gamma(5, 6)$	1

3.3.3 PRUEBA 3: Presupuesto $P=3.000.000$ u.m.

Para un presupuesto de red de $P = 3.000.000$ u.m., se genera una red para los viajes $W = \{(1,5), (1,3), (2,6), (4,6)\}$ con las siguientes características:

- La red cuenta con 2 carriles con infraestructura 1.
- La red cuenta con 3 carriles con infraestructura 2.
- El coste de construcción fue de 2.850.000 u.m. lo que representa un 95% del presupuesto total.
- El coste total de para los usuarios de esta red fue: 55 unidades de tiempo.

A continuación, se muestra el resultado gráfico de la prueba 1:

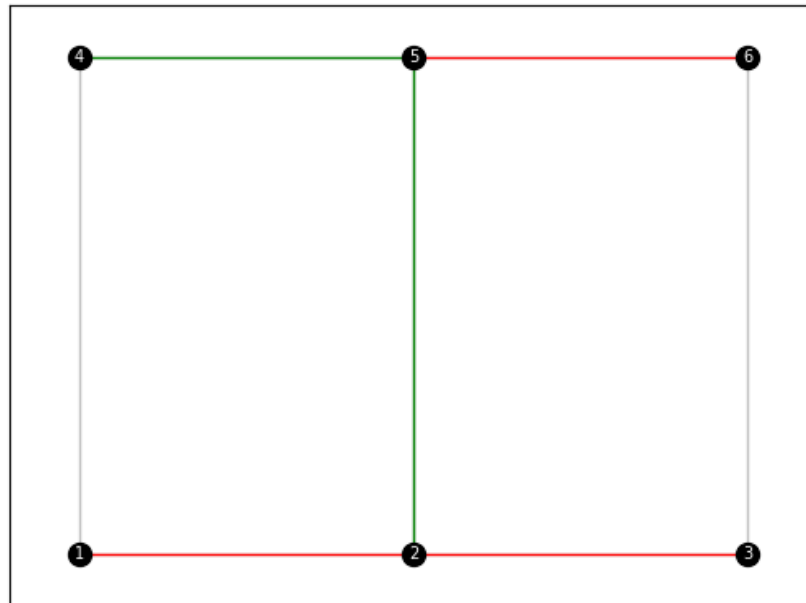


Ilustración 35 Solución Prueba 3: Presupuesto $P = 3.000.000$ u.m.

De la ilustración anterior, se observa que los arcos (1,2),(2,3) y (5,6) se construirán con la infraestructura 2 y los arcos (4,5) y (2,5) con la infraestructura 1.

Tabla 15 Solución Prueba 3: Presupuesto $P = 3.000.000$ u.m.

ARCOS SOLUCIÓN	INFRAESTRUCTURA
$\gamma(4, 5)$	1
$\gamma(2, 5)$	1
$\gamma(1, 2)$	2
$\gamma(2, 3)$	2
$\gamma(5, 6)$	2

Caso 2: red 9 nodos: Con esta red, se llevaron a cabo 2 pruebas diferentes variando el presupuesto y esta vez se emplearon arcos de diferentes longitudes:

Los parámetros de entrada se describen a continuación:

Para este caso, la red definida \hat{H} contiene un conjunto de 9 nodos y un conjunto de 12 arcos, dicha red tiene la siguiente forma:

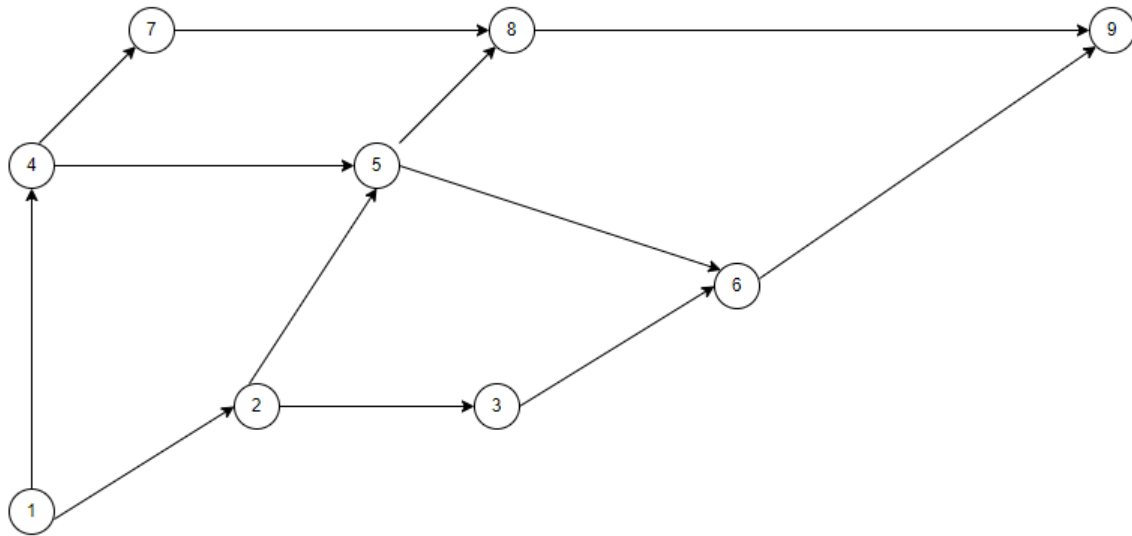


Ilustración 36 Red H.

De la red \hat{H} se obtienen los siguientes conjuntos: un conjunto de nodos $\hat{V} = (1,2,3,4,5,6,7,8,9)$ y un conjunto de arcos $\hat{A} = \{(1,2), (2,3), (4,5), (5,6), (1,4), (2,5), (3,6), (4,7), (5,8), (6,9), (7,8), (8,9)\}$.

El conjunto D que representa la demanda de viaje por los arcos, para este problema los valores de la demanda son los siguientes:

Tabla 16 Matriz de demanda D de viajes por los arcos de la red \hat{H} .

NODOS	1	2	3	4	5	6	7	8	9
1	0	500	300	100	700	100	100	700	200
2	600	0	200	500	200	800	100	600	300
3	400	600	0	100	100	200	100	400	800
4	300	500	200	0	200	100	200	500	500
5	200	100	500	600	0	200	100	200	100
6	500	900	800	900	400	0	100	100	300
7	600	700	800	900	500	300	0	200	100
8	500	300	100	700	100	100	700	0	300
9	500	300	100	700	100	100	700	200	0

En esta red \hat{H} , para una mayor cobertura de la red se empleó el siguiente conjunto: $W = \{(2,6), (7,9), (4,6), (1,7)\}$, con este conjunto lo que se quiere conseguir es una red satisfaga la demanda de estos caminos, tal y como se muestra a continuación:

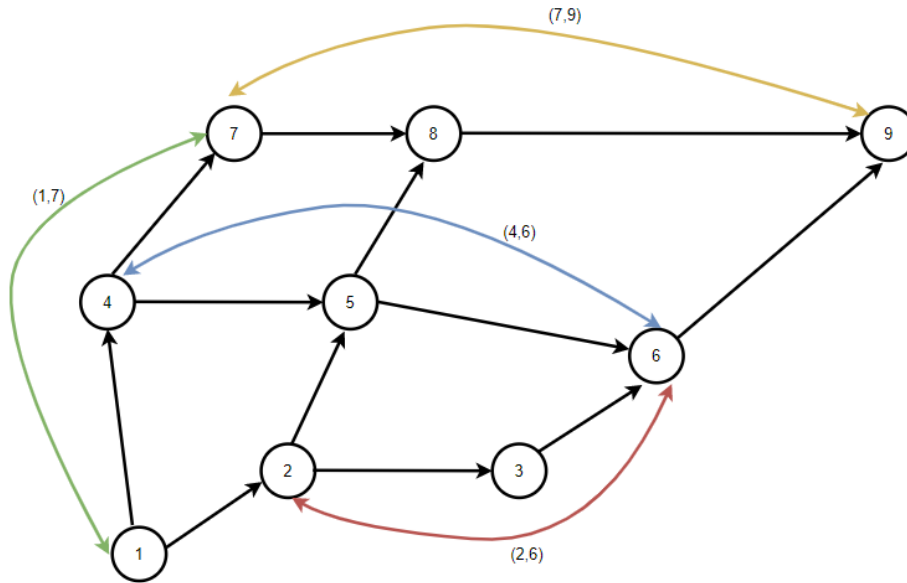


Ilustración 37 Conjunto de pares OD para un conjunto $W = \{(2,6), (7,9), (4,6), (1,7)\}$.

Por otro lado, los presupuestos de construcción para la red \hat{H} fueron los siguientes: un presupuesto ajustado de 25.000.000 u.m. y por último el presupuesto generoso fue de 50.000.000 u.m.

En referencia al conjunto de infraestructuras k , el planificador central es quien decide la cantidad y la tipología de estas. Para la red \hat{H} el conjunto de infraestructuras a analizar es el siguiente: $K = (1,2)$, siendo la infraestructura 1 más barata en término de construcción, pero más cara en cuanto al coste de viaje del usuario respecto a la infraestructura 2, en otras palabras, la infraestructura 2 es mejor que la infraestructura 1 en término de coste de viaje del usuario.

Finalmente se indican los conjuntos de coste de usuario, coste de construcción y penalización en función de los arcos y la infraestructura:

A continuación, se muestra los parámetros anteriormente mencionados en función de las infraestructuras:

Tabla 17 Coste de construcción y usuario en función del tipo de infraestructura.

INFRAESTRUTURA	COSTE DE CONSTRUCCIÓN ARCOS (U. MONETARIA/U. LONGITUD)	COSTE DE USUARIO (TIEMPO)
0	0	50
1	3000	20
2	6000	5

A continuación, se indica las penalizaciones utilizadas:

Tabla 18 Penalización por cambio de infraestructura

CAMBIO DE INFRAESTRUTURA	PENALIZACIÓN β
DE LA INFRAESTRUTURA 1 A LA 0	1
DE LA INFRAESTRUTURA 2 A LA 0	3
DE LA INFRAESTRUTURA 2 A LA 1	2

A continuación, se muestra los resultados de las pruebas realizadas con los diferentes presupuestos establecidos:

3.3.4 PRUEBA 4. Presupuesto $P=25.000.000$ u.m.

Para un presupuesto de red de $P = 25.000.000$ u.m., se genera una red para los viajes $W = \{(2,6), (7,9), (4,6), (1,7)\}$ con las siguientes características:

- La red cuenta con 5 carriles con infraestructura 0.
- La red cuenta con 2 carriles con infraestructura 1.
- El coste de construcción fue de 21.727.922,06 u.m. lo que representa un 86.91% del presupuesto total.
- El coste total de para los usuarios de esta red fue: 290 unidades de tiempo.

A continuación, se muestra el resultado gráfico del programa:

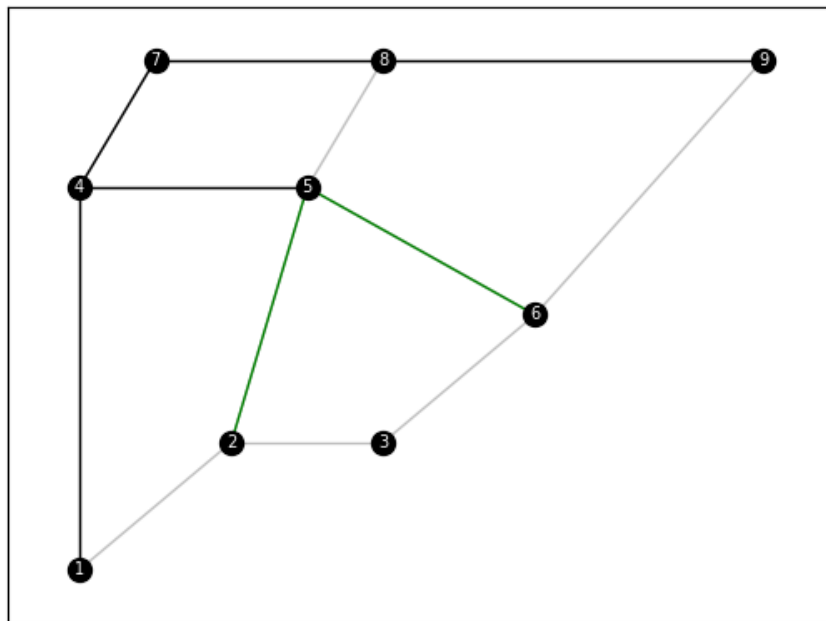


Ilustración 38 Solución Prueba 4: Presupuesto $P = 25.000.000$ u.m

De la ilustración anterior, se observa que los arcos (2,5) y (5,6) se construirán con la infraestructura 1 y los arcos (1,4), (4,7),(7,8),(8,9) y (4,5) con la infraestructura 0.

Tabla 19 Solución Prueba 4: Presupuesto $P = 25.000.000$ u.m.

ARCOS SOLUCIÓN	INFRAESTRUCTURA
$\gamma(4, 5)$	0
$\gamma(7, 8)$	0
$\gamma(8, 9)$	0
$\gamma(1, 4)$	0
$\gamma(4, 7)$	0

$\gamma(5, 6)$	1
$\gamma(2, 5)$	1

3.3.5 PRUEBA 5: Presupuesto $P=50.000.000$ u.m.

Para un presupuesto de red de $P = 50.000.000$ u.m., se genera una red para los viajes $W = \{(2,6), (7,9), (4,6), (1,7)\}$ con las siguientes características:

- La red cuenta con 4 carriles con infraestructura 0.
- La red cuenta con 1 carril con infraestructura 1.
- La red cuenta con 2 carriles con infraestructura 2.
- El coste de construcción fue de 49.147.943,91 u.m. lo que representa un 98.29% del presupuesto total.
- El coste total de para los usuarios de esta red fue: 230 unidades de tiempo.

A continuación, se muestra el resultado gráfico del programa:

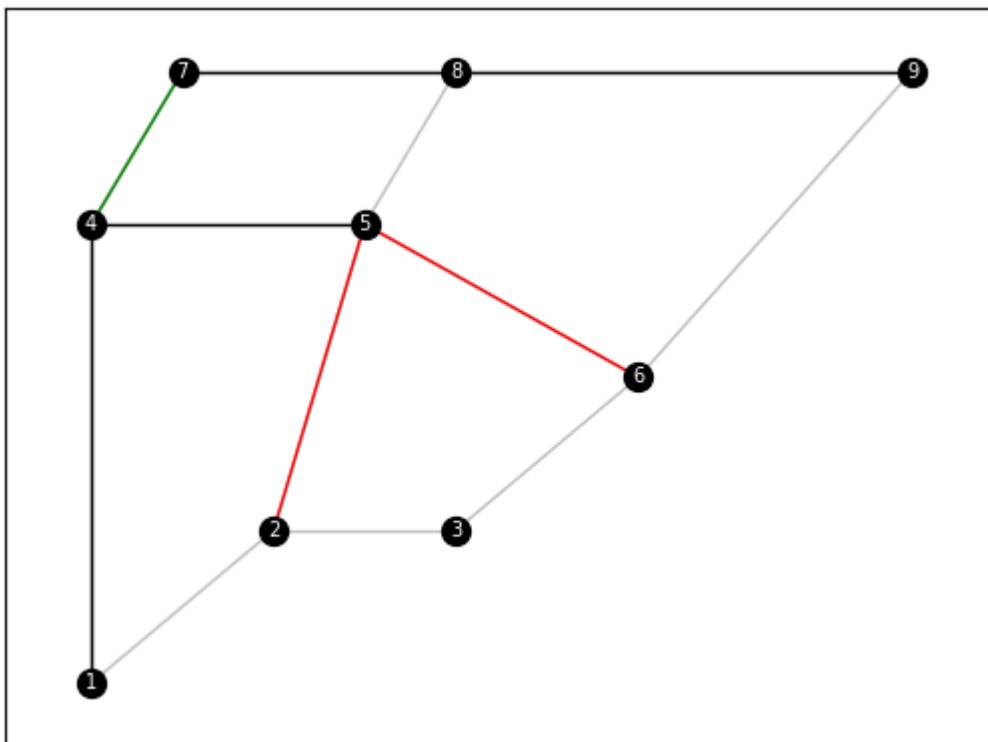


Ilustración 39 Prueba 5: Presupuesto $P = 50.000.000$ u.m

De la ilustración anterior, se observa el arco (4,7) se construirán con la infraestructura 1 y los arcos (2,5) y (5,6) con la infraestructura 2. Finalmente, los arcos (1,4),(4,5),(7,8) y (8,9) utilizarán la infraestructura 0.

Tabla 20 Solución Prueba 5: Presupuesto $P = 50.000.000$ u.m.

ARCOS SOLUCIÓN	INFRAESTRUCTURA
$\gamma(4, 5)$	0
$\gamma(7, 8)$	0
$\gamma(8, 9)$	0
$\gamma(1, 4)$	0
$\gamma(4, 7)$	1
$\gamma(5, 6)$	2
$\gamma(2, 5)$	2

4.4 CONCLUSIONES DE LAS PRUEBAS:

- Se realizó el seguimiento del flujo a través de los arcos para cada viaje. Con esto se demostró el cumplimiento de la demanda de cada viaje W .
- Las pruebas realizadas con elevado presupuesto incluían el mayor número de arcos con la mejor tecnología.
- Se observó que, al aumentar el presupuesto el coste del usuario disminuía. Esto se traduce en una relación inversa entre estos dos parámetros. Cuanto mayor presupuesto exista menor será el coste para el usuario o lo que es lo mismo mayor será su satisfacción.
- Se observó homogeneidad en las estructuras de los arcos para cada viaje de W . Esto es importante puesto que el cambio de tecnologías (a peor) para cada viaje de W es penalizado por el modelo, lo cual empeora la función objetivo del problema aumentando la insatisfacción del usuario por la “incomodidad” que representa el cambio de tecnología.

5 CASO PRÁCTICO

“Ahora demuestra todo lo que vales”.

-Mulán-

En este capítulo se presenta el diseño de la red de bicicletas para la ciudad de Sevilla, España, empleando el programa desarrollado en apartados anteriores.

Sevilla es un municipio y una ciudad de España, capital de la provincia de Andalucía, según datos del Instituto Nacional de Estadística para el año 2021 contaba con una población de 684 234 habitantes. El municipio tiene una extensión de 160 km^2 .



Ilustración 40 Red Bicicleta Sevilla

Es considerada una ciudad adecuada para el uso de bicicleta gracias a las condiciones climatológicas y del

terreno, es por ello por lo que actualmente cuenta con una amplia extensión de carril para bicicleta.

5.1 DATOS DEL PROBLEMA

Para realizar el diseño con datos urbanísticos de la ciudad, se empleó una base de datos utilizadas se compone de una red de 139 nodos y 228 arcos, dicha red se compone de las calles y avenidas principales de la ciudad de Sevilla.

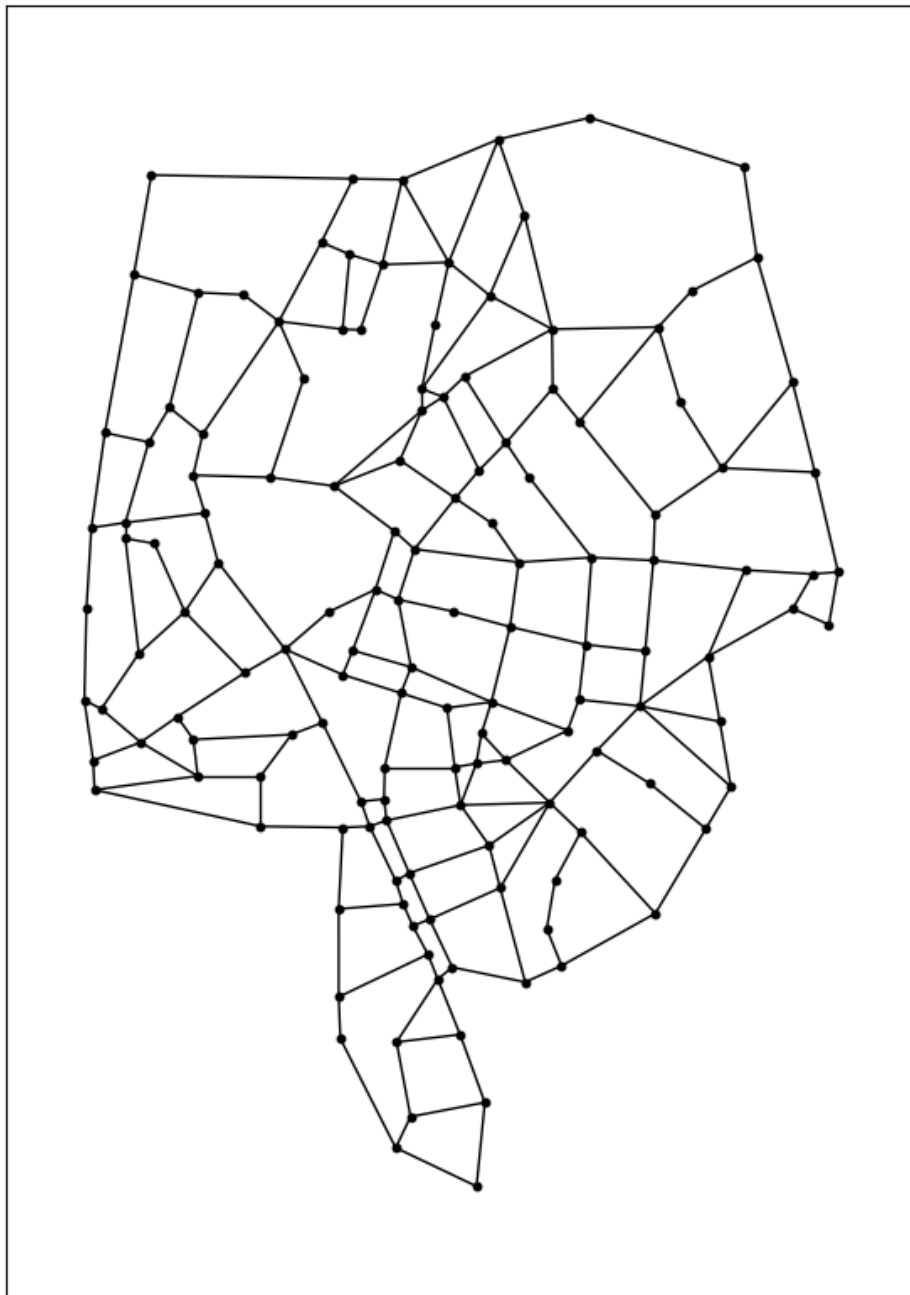


Ilustración 41 Red de las principales avenidas y calles de Sevilla

A continuación, se muestra la red de forma acotada:

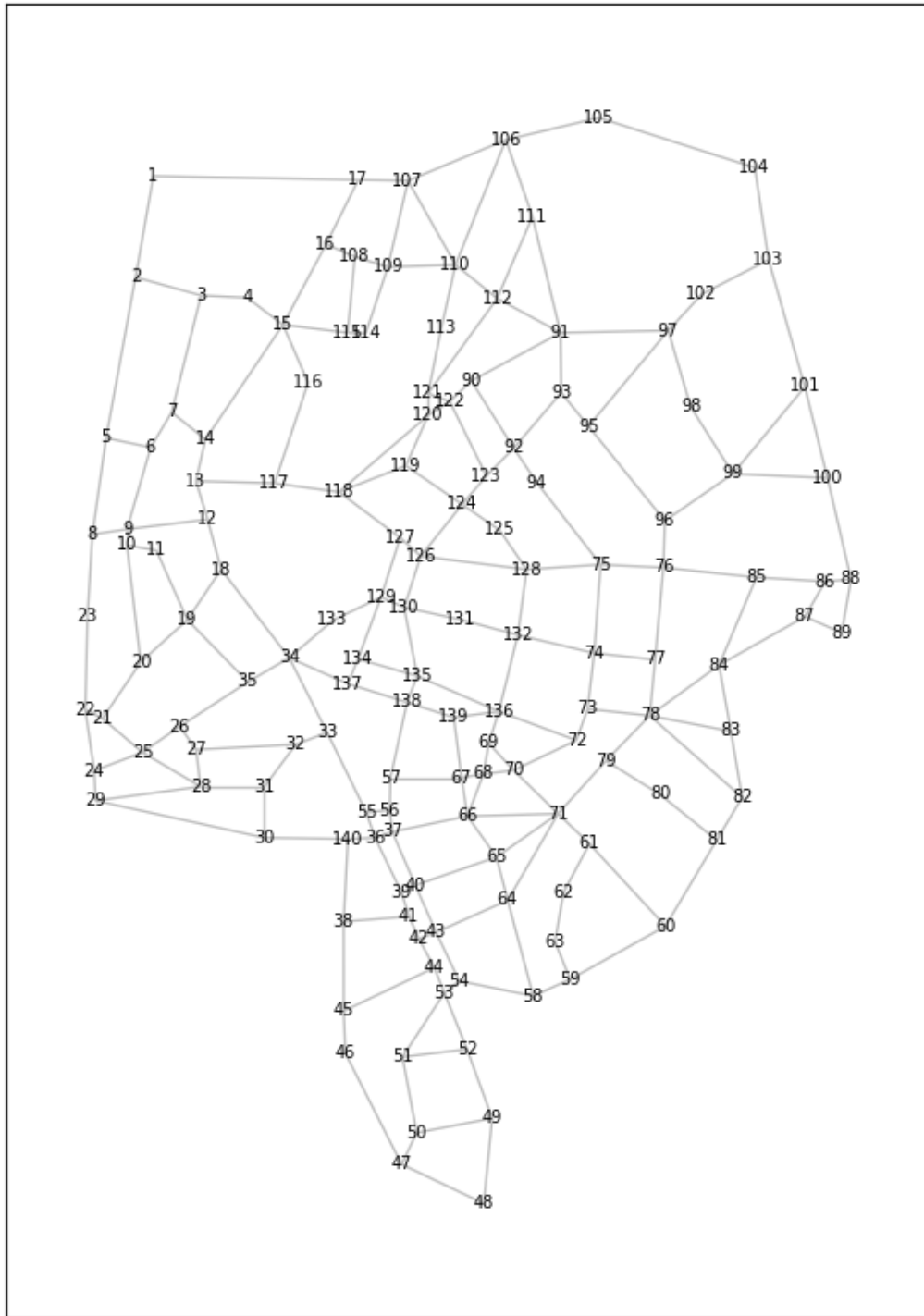


Ilustración 42 Red de las principales avistas y calles de Sevilla

De esta forma, la matriz de demanda para esta red es del orden de 139x139. Por otro lado, los datos de costes y son los siguientes:

Tabla 21 Datos red Sevilla

	COSTE DE CONSTRUCCIÓN	COSTE DE USUARIO (UNIDADES DE TIEMPO)
INFRAESTRUCTURA 0	0	50
INFRAESTRUCTURA 1	$400 \frac{u.m}{longitud}$	20
INFRAESTRUCTURA 2	$700 \frac{u.m}{longitud}$	5

Tal y como se observa en la Tabla 21 se emplearon dos tipos de infraestructuras (infraestructura 1,2) y la opción de no realizar ninguna construcción (infraestructura 0).

Al igual que en apartado anterior, se evaluaron diferentes casos variando el presupuesto. Para el diseño de esta red se llevó a cabo la realización de un primer caso donde se evaluaron dos tipos de presupuestos. Por otro lado, se llevó a cabo la ejecución de un segundo caso para el cual se evaluó el mismo presupuesto utilizado para la prueba 2 del caso 1 pero esta vez ampliando el conjunto de viajes W .

Caso 1:

Prueba 1: El presupuesto asignado es el siguiente: $P = 6.000.000$ um. El conjunto de viajes definidos para esta red es el siguiente:

$$W = \{(1,60), (37,107), (106,123), (124,139), (60,104), (102,139), (60,139), (1,107), (6,110), (128,139), (86,126), (54,60), (38,52), (91,128), (52,72), (20,38), (91,96)\}$$

El resultado gráfico generado por la metodología se muestra a continuación:

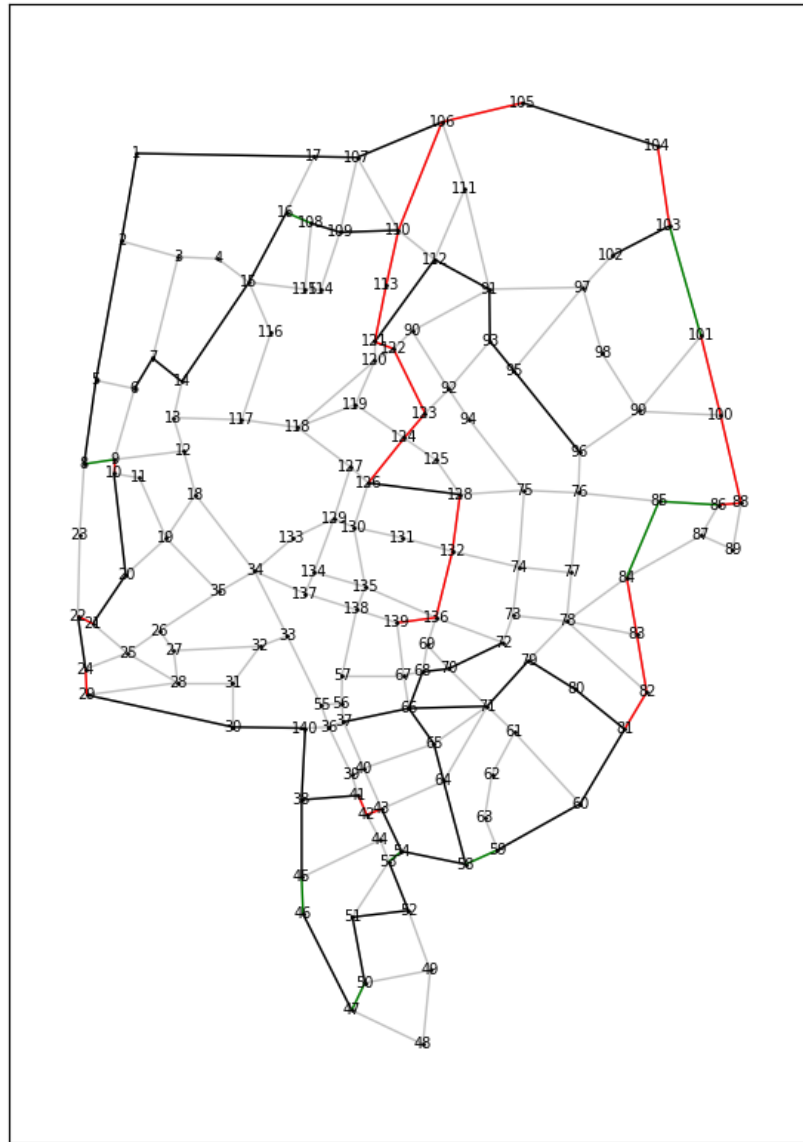


Ilustración 43 Red de bicicleta para la ciudad de Sevilla. Prueba 1, Caso 1

Para un presupuesto de red de 6.000.000 u.m., se genera una red para los viajes W con las siguientes características:

- La red cuenta con 47 carriles con infraestructura 0.
- La red cuenta con 9 carriles con infraestructura 1.
- La red cuenta con 23 carriles con infraestructura 2.
- El coste de construcción fue de 5.998.621.88 u.m. lo que representa un 99,71% del presupuesto total.
- El coste total de para los usuarios de esta red fue: 2645 unidades de tiempo.

Tabla 22 Solución Prueba 1: Presupuesto $P = 6.000.000$ u.m.

ARCOS SOLUCIÓN	CAPA	ARCOS SOLUCIÓN	CAPA	ARCOS SOLUCIÓN	CAPA
$\gamma(1, 2)$	0	$\gamma(70, 72)$	0	$\gamma(86, 88)$	2
$\gamma(2, 5)$	0	$\gamma(79, 80)$	0	$\gamma(100, 101)$	2
$\gamma(6, 7)$	0	$\gamma(80, 81)$	0	$\gamma(88, 100)$	2
$\gamma(5, 8)$	0	$\gamma(91, 93)$	0	$\gamma(103, 104)$	2
$\gamma(1, 17)$	0	$\gamma(93, 95)$	0	$\gamma(105, 106)$	2
$\gamma(15, 16)$	0	$\gamma(95, 96)$	0	$\gamma(106, 110)$	2
$\gamma(14, 15)$	0	$\gamma(102, 103)$	0	$\gamma(110, 113)$	2
$\gamma(7, 14)$	0	$\gamma(104, 105)$	0	$\gamma(113, 121)$	2
$\gamma(10, 20)$	0	$\gamma(17, 107)$	0	$\gamma(121, 122)$	2
$\gamma(20, 21)$	0	$\gamma(106, 107)$	0	$\gamma(122, 123)$	2
$\gamma(22, 24)$	0	$\gamma(108, 109)$	0	$\gamma(123, 124)$	2
$\gamma(29, 30)$	0	$\gamma(109, 110)$	0	$\gamma(124, 126)$	2
$\gamma(30, 140)$	0	$\gamma(91, 112)$	0	$\gamma(128, 132)$	2
$\gamma(140, 38)$	0	$\gamma(112, 121)$	0	$\gamma(132, 136)$	2
$\gamma(38, 45)$	0	$\gamma(126, 128)$	0	$\gamma(136, 139)$	2
$\gamma(38, 41)$	0	$\gamma(8, 9)$	1		
$\gamma(46, 47)$	0	$\gamma(45, 46)$	1		
$\gamma(43, 54)$	0	$\gamma(53, 54)$	1		
$\gamma(51, 52)$	0	$\gamma(47, 50)$	1		
$\gamma(52, 53)$	0	$\gamma(58, 59)$	1		
$\gamma(50, 51)$	0	$\gamma(84, 85)$	1		
$\gamma(37, 66)$	0	$\gamma(85, 86)$	1		
$\gamma(54, 58)$	0	$\gamma(101, 103)$	1		
$\gamma(59, 60)$	0	$\gamma(16, 108)$	1		
$\gamma(60, 81)$	0	$\gamma(9, 10)$	2		
$\gamma(58, 64)$	0	$\gamma(21, 22)$	2		
$\gamma(64, 65)$	0	$\gamma(24, 29)$	2		
$\gamma(65, 66)$	0	$\gamma(41, 42)$	2		
$\gamma(68, 70)$	0	$\gamma(42, 43)$	2		
$\gamma(66, 68)$	0	$\gamma(81, 82)$	2		
$\gamma(66, 71)$	0	$\gamma(82, 83)$	2		
$\gamma(71, 79)$	0	$\gamma(83, 84)$	2		

Prueba 2: El presupuesto asignado es el siguiente: $P = 15.000.000$ um. El conjunto de viajes definidos para esta red corresponde al mismo conjunto utilizado la prueba 1. A continuación se muestra el resultado gráfico generado por la metodología:

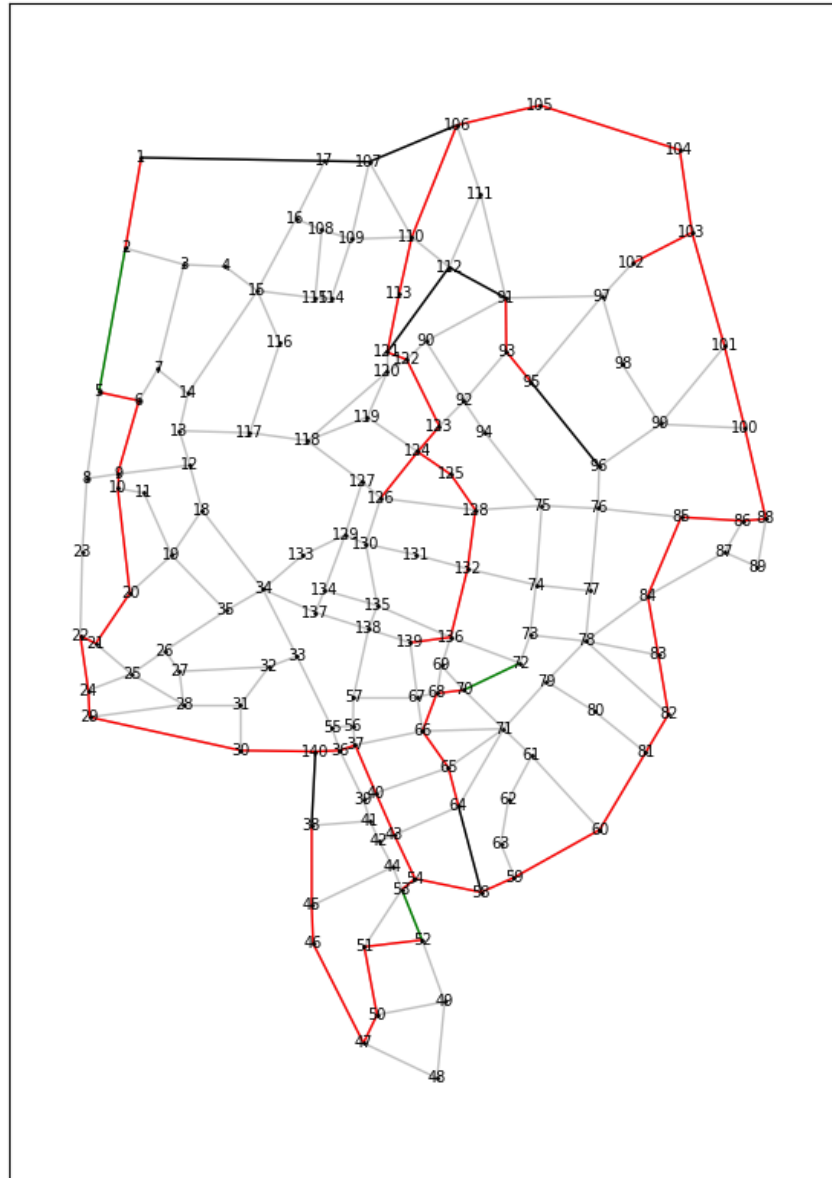


Ilustración 44 Red de bicicleta para la ciudad de Sevilla. Prueba 2, Caso 1.

Para un presupuesto de red de $P = 15.000.000$ u.m., se genera una red para los viajes W con las siguientes características:

- La red cuenta con 8 carriles con infraestructura 0.
- La red cuenta con 3 carriles con infraestructura 1.
- La red cuenta con 58 carriles con infraestructura 2.
- El coste de construcción fue de 14.944.217,42 u.m. lo que representa un 99,62% del presupuesto total.
- El coste total de para los usuarios de esta red fue: 750 unidades de tiempo.

Tabla 23 Solución Prueba 2: Presupuesto $P = 15.000.000$ u.m.

ARCOS SOLUCIÓN	CAPA	ARCOS SOLUCIÓN	CAPA	ARCOS SOLUCIÓN	CAPA
$\gamma(1, 17)$	0	$\gamma(50, 51)$	2	$\gamma(124, 125)$	2
$\gamma(140, 38)$	0	$\gamma(47, 50)$	2	$\gamma(125, 128)$	2
$\gamma(58, 64)$	0	$\gamma(54, 58)$	2	$\gamma(128, 132)$	2
$\gamma(95, 96)$	0	$\gamma(58, 59)$	2	$\gamma(132, 136)$	2
$\gamma(17, 107)$	0	$\gamma(59, 60)$	2	$\gamma(136, 139)$	2
$\gamma(106, 107)$	0	$\gamma(60, 81)$	2		
$\gamma(91, 112)$	0	$\gamma(64, 65)$	2		
$\gamma(112, 121)$	0	$\gamma(65, 66)$	2		
$\gamma(2, 5)$	1	$\gamma(68, 70)$	2		
$\gamma(52, 53)$	1	$\gamma(66, 68)$	2		
$\gamma(70, 72)$	1	$\gamma(81, 82)$	2		
$\gamma(1, 2)$	2	$\gamma(82, 83)$	2		
$\gamma(5, 6)$	2	$\gamma(83, 84)$	2		
$\gamma(6, 9)$	2	$\gamma(84, 85)$	2		
$\gamma(9, 10)$	2	$\gamma(85, 86)$	2		
$\gamma(10, 20)$	2	$\gamma(86, 88)$	2		
$\gamma(20, 21)$	2	$\gamma(91, 93)$	2		
$\gamma(21, 22)$	2	$\gamma(93, 95)$	2		
$\gamma(22, 24)$	2	$\gamma(100, 101)$	2		
$\gamma(24, 29)$	2	$\gamma(88, 100)$	2		
$\gamma(29, 30)$	2	$\gamma(101, 103)$	2		
$\gamma(30, 140)$	2	$\gamma(102, 103)$	2		
$\gamma(140, 36)$	2	$\gamma(103, 104)$	2		
$\gamma(38, 45)$	2	$\gamma(104, 105)$	2		
$\gamma(36, 37)$	2	$\gamma(105, 106)$	2		
$\gamma(37, 40)$	2	$\gamma(106, 110)$	2		
$\gamma(40, 43)$	2	$\gamma(110, 113)$	2		
$\gamma(45, 46)$	2	$\gamma(113, 121)$	2		
$\gamma(46, 47)$	2	$\gamma(121, 122)$	2		
$\gamma(43, 54)$	2	$\gamma(122, 123)$	2		
$\gamma(53, 54)$	2	$\gamma(123, 124)$	2		
$\gamma(51, 52)$	2	$\gamma(124, 126)$	2		

Caso 2:

Prueba 1: El presupuesto asignado es el siguiente: $P = 15.000.000$ um. Al conjunto de viajes W anterior se añadieron los siguientes viajes: $(6,117)$, $(117,133)$, $(37,133)$, $(126,133)$, $(24,133)$, $(2,117)$, $(6,119)$, $(58,61)$, $(84,130)$ y $(107,138)$. Por lo que el conjunto total de viajes W queda de la siguiente forma:

$$W=[(1,60), (37,107), (106,123), (124,139), (60,104), (102,139), (60,139), (1,107), (6,110), (128,139),$$

(86,126), (54,60), (38,52), (91,128), (52,72), (20,38), (91,96), **(6,117), (117,133), (37,133), (126,133), (24,133), (2,117), (6,119), (58,61),(84,130), (107,138)]**

A continuación, se muestra el resultado gráfico generado por la metodología:

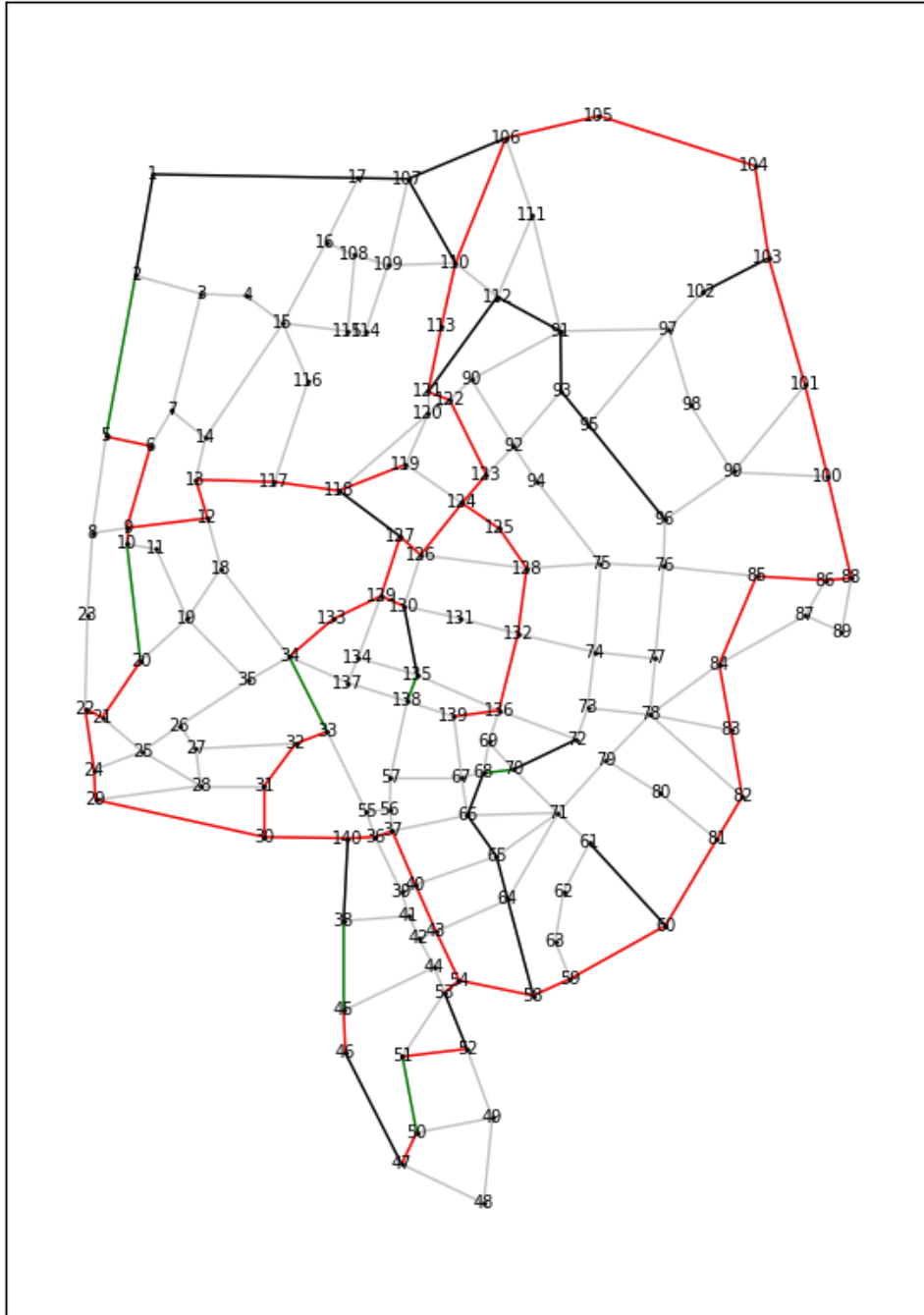


Ilustración 45 Red de bicicletas para la ciudad de Sevilla. Prueba 1, Caso 2.

Para un presupuesto de red de $P = 15.000.000$ u.m., se genera una red para los viajes W con las siguientes características:

- La red cuenta con 22 carriles con infraestructura 0.
- La red cuenta con 7 carriles con infraestructura 1.
- La red cuenta con 59 carriles con infraestructura 2.
- El coste de construcción fue de 14.997.760,39 u.m. lo que representa un 99,98% del presupuesto total.
- El coste total de para los usuarios de esta red fue: 1535 unidades de tiempo.

Tabla 24 Solución Caso 2: Presupuesto $P = 15.000.000$ u.m.

ARCOS SOLUCIÓN	CAPA	ARCOS SOLUCIÓN	CAPA	ARCOS SOLUCIÓN	CAPA
y(1, 2)	0	y(12, 13)	2	y(101, 103)	2
y(1, 17)	0	y(9, 10)	2	y(103, 104)	2
y(140, 38)	0	y(20, 21)	2	y(104, 105)	2
y(46, 47)	0	y(21, 22)	2	y(13, 117)	2
y(52, 53)	0	y(22, 24)	2	y(105, 106)	2
y(58, 64)	0	y(24, 29)	2	y(106, 110)	2
y(64, 65)	0	y(32, 33)	2	y(110, 113)	2
y(65, 66)	0	y(31, 32)	2	y(113, 121)	2
y(66, 68)	0	y(30, 31)	2	y(121, 122)	2
y(60, 61)	0	y(29, 30)	2	y(118, 119)	2
y(70, 72)	0	y(30, 140)	2	y(117, 118)	2
y(91, 93)	0	y(140, 36)	2	y(122, 123)	2
y(93, 95)	0	y(36, 37)	2	y(34, 133)	2
y(95, 96)	0	y(37, 40)	2	y(123, 124)	2
y(102, 103)	0	y(40, 43)	2	y(124, 126)	2
y(17, 107)	0	y(45, 46)	2	y(126, 127)	2
y(106, 107)	0	y(43, 54)	2	y(124, 125)	2
y(107, 110)	0	y(53, 54)	2	y(125, 128)	2
y(91, 112)	0	y(51, 52)	2	y(127, 129)	2
y(112, 121)	0	y(47, 50)	2	y(128, 132)	2
y(118, 127)	0	y(54, 58)	2	y(129, 130)	2
y(130, 135)	0	y(58, 59)	2	y(129, 133)	2
y(2, 5)	1	y(59, 60)	2	y(132, 136)	2
y(10, 20)	1	y(60, 81)	2	y(136, 139)	2
y(33, 34)	1	y(81, 82)	2		
y(38, 45)	1	y(82, 83)	2		
y(50, 51)	1	y(83, 84)	2		
y(68, 70)	1	y(84, 85)	2		
y(135, 138)	1	y(85, 86)	2		
y(5, 6)	2	y(86, 88)	2		
y(6, 9)	2	y(100, 101)	2		
y(9, 12)	2	y(88, 100)	2		

6 CONCLUSIÓN

Nuestra recompensa se encuentra en el esfuerzo y no en el resultado. Un esfuerzo total es una victoria completa.

-Mahatma Gandhi-

El presente Trabajo de Fin de Grado tuvo como objetivo el desarrollo de una metodología que permitiese resolver el problema de diseño de una red de carriles para bicicletas e ilustrar su aplicación utilizando la red de la ciudad de Sevilla.

Para el cumplimiento del objetivo se propuso una ampliación del problema general de flujo a coste mínimo sin limitación de capacidad. La característica principal del modelo ampliado fue que además de buscar la inclusión de los arcos en la red, este seleccionase la tecnología de construcción a utilizar en el arco. De esa forma el modelo ampliado permitía la construcción de una red con diferentes tipos de carriles de bici.

Otro aspecto importante que consideró en la ampliación del modelo es que debía valorar la repercusión que los usuarios podían experimentar por el cambio de tecnologías en su trayecto. Para minimizar dicha repercusión se empleó un parámetro que penalizaba los cambios de tecnologías, específicamente los cambios a peor. De esta forma se consiguió que, para un determinado viaje, la ruta fuera lo más homogénea posible en relación con las tecnologías de construcción.

Como objetivo, el modelo ampliado debía maximizar la satisfacción del usuario o lo que es lo mismo minimizar el coste de viaje del usuario a través de la red.

A fin de poder modelar las diferentes posibilidades de construcción, se empleó una estructura multicapa que permitía el modelado de cada arco de la red base mediante un conjunto de arcos paralelos, cada uno de ellos en una capa diferente.

Para de probar la aplicación del modelo ampliado, se utilizaron diferentes escenarios con 6 y 9 nodos a los que se les fue variando el presupuesto asignado. Se observó que cuanto más alto es el presupuesto el modelo

incorpora más arcos con la mejor tecnología esto provoca que el coste del usuario disminuya.

Finalmente se realizó un caso de aplicación para la ciudad de Sevilla donde se evaluaron diferentes casos con variación de presupuestos e incrementos de viajes OD. De este caso práctico se obtuvieron las siguientes conclusiones:

- Al igual que en los escenarios anteriores, se observó homogeneidad en las estructuras de los arcos para cada viaje de W . Lo cual garantiza que la satisfacción del usuario no se verá afectada por los cambios de tecnologías.
- Cuanto mayor sea el presupuesto menor será el coste para el usuario o lo que es lo mismo mayor será su satisfacción.
- El modelo no solo selecciona arcos para la red, sino que además selecciona el tipo de tecnología que se usará en la construcción de dicho arco.

Con esto se demuestra que el modelo ampliado permite el diseño de una red para carriles bici que maximiza el nivel de servicio de los usuarios, agotando el presupuesto en la construcción de los carriles bici. De esta forma se consiguió el enrutamiento de los usuarios desde sus orígenes a sus destinos de forma que puedan realizar sus desplazamientos en el menor tiempo posible.

REFERENCIAS

- [1] Liñán, R., Merino, S., Martínez, J. 2013. OPTIMIZACIÓN Y DISEÑO DE CARRILES BICI. *Greencities & Sostenibilidad: Inteligencia aplicada a la sostenibilidad urbana Convocatoria de Comunicaciones y 1ª Bienal de Edificación y Urbanismo Sostenible (Edición 2013)*
- [2] Zhuang, D., Jin, J. G., Shen, Y., Jiang, W. 2021. Understanding the bike sharing travel demand and cycle lane network: The case of Shanghai. *International Journal of Sustainable Transportation*, 15, 111–123.
- [3] Ospina, J., Duque, J., Botero-Fernández, V., Montoya, A. 2022. The maximal covering bicycle network design problem. *Transportation Research Part A: Policy and Practice*, 159, 222–236.
- [4] Saltykova, K., Ma, X., Yao, L., Kong, H. 2022 Environmental impact assessment of bike-sharing considering the modal shift from public transit. *Transportation Research Part D: Transport and Environment*, 105, 103238.
- [5] Danique, T., Dorine, D. 2021. Understanding long-term changes in commuter mode use of a pilot featuring free e-bike trials. *Transport Policy*, 105, 134–144.
- [6] Larrodé, E., Gallego, J., Fraile del Pozo, A. 2011. Optimización de redes de transporte. *LYCHNOS*, 6, 2011.
- [7] Minoux, M. 2001. Discrete Cost Multicommodity Network Optimization Problems and Exact Solution Methods. *Annals of Operations Research*, 106, 19–46.
- [8] Geoffrion, A. Graves, G. 1974. Multicommodity distribution system design by Benders decomposition. *Management science*, 20, 5, 822–844.
- [9] Magnanti, T., Wong, R. 1984. Network design and transportation planning: Models and algorithms. *Transportation science*, 18, 1, 1–55.
- [10] Murawski, L., Church, R. 2009. Improving accessibility to rural health services: The maximal covering network improvement problem. *Socio-Economic Planning Sciences*, 43, 102–110.
- [12] Leblanc, L. 1975. An Algorithm for the Discrete Network Design Problem. *Transportation Science*, 9, 3, 183–199.
- [13] Boccaletti, S., Bianconi, G., Criado, R., Del Genio, C., Gómez-Gardeñes, J., Romance, M., Sendiña-Nadal, I., Wang, Z., Zanin, M. 2014. The structure and dynamics of multilayer networks. *Phys Physics Reports*, 544, 1–122.
- [14] Hammoud, Z., Kramer, K. 2020. Multilayer networks: aspects, implementations, and application in biomedicine. *Big Data Anal*, 5, 2.
- [15] Mauttone, A., Mercadante, G., Rabaza, M., Toledo, F. 2017. Bicycle network design: model and solution algorithm. *Transportation Research Procedia*, 27, 969-976.

7 ANEXO

7.1 ESTRUCTURA DEL MODELO GENERADO POR EL PROGRAMA

Ejemplo de la estructura de las restricciones y función objetivo generada por el programa, para la prueba 1:

En primer lugar, se muestra la estructura de las restricciones y la función objetivo generada por el programa:

La función objetivo queda de la siguiente forma:

Minimize $50 x[1,2,1,5] + 50 x[1,2,1,3] + 50 x[1,2,2,6] + 50 x[1,2,4,6] + 50 x[2,3,1,5] + 50 x[2,3,1,3] + 50 x[2,3,2,6] + 50 x[2,3,4,6] + 50 x[4,5,1,5] + 50 x[4,5,1,3] + 50 x[4,5,2,6] + 50 x[4,5,4,6] + 50 x[5,6,1,5] + 50 x[5,6,1,3] + 50 x[5,6,2,6] + 50 x[5,6,4,6] + 50 x[1,4,1,5] + 50 x[1,4,1,3] + 50 x[1,4,2,6] + 50 x[1,4,4,6] + 50 x[2,5,1,5] + 50 x[2,5,1,3] + 50 x[2,5,2,6] + 50 x[2,5,4,6] + 50 x[3,6,1,5] + 50 x[3,6,1,3] + 50 x[3,6,2,6] + 50 x[3,6,4,6] + 20 x[7,8,1,5] + 20 x[7,8,1,3] + 20 x[7,8,2,6] + 20 x[7,8,4,6] + 20 x[8,9,1,5] + 20 x[8,9,1,3] + 20 x[8,9,2,6] + 20 x[8,9,4,6] + 20 x[10,11,1,5] + 20 x[10,11,1,3] + 20 x[10,11,2,6] + 20 x[10,11,4,6] + 20 x[11,12,1,5] + 20 x[11,12,1,3] + 20 x[11,12,2,6] + 20 x[11,12,4,6] + 20 x[7,10,1,5] + 20 x[7,10,1,3] + 20 x[7,10,2,6] + 20 x[7,10,4,6] + 20 x[8,11,1,5] + 20 x[8,11,1,3] + 20 x[8,11,2,6] + 20 x[8,11,4,6] + 20 x[9,12,1,5] + 20 x[9,12,1,3] + 20 x[9,12,2,6] + 20 x[9,12,4,6] + 5 x[13,14,1,5] + 5 x[13,14,1,3] + 5 x[13,14,2,6] + 5 x[13,14,4,6] + 5 x[14,15,1,5] + 5 x[14,15,1,3] + 5 x[14,15,2,6] + 5 x[14,15,4,6] + 5 x[16,17,1,5] + 5 x[16,17,1,3] + 5 x[16,17,2,6] + 5 x[16,17,4,6] + 5 x[17,18,1,5] + 5 x[17,18,1,3] + 5 x[17,18,2,6] + 5 x[17,18,4,6] + 5 x[13,16,1,5] + 5 x[13,16,1,3] + 5 x[13,16,2,6] + 5 x[13,16,4,6] + 5 x[14,17,1,5] + 5 x[14,17,1,3] + 5 x[14,17,2,6] + 5 x[14,17,4,6] + 5 x[15,18,1,5] + 5 x[15,18,1,3] + 5 x[15,18,2,6] + 5 x[15,18,4,6]$

El contexto de las variables es la siguiente: para el caso de la variable que representa el flujo de viaje en los arcos X, se lee de la siguiente forma: por ejemplo, $x[1,2,1,5]$ los primeros dos términos (1,2) representan el arco (1,2), los siguientes términos (1,6) representa el par OD del horizonte temporal. Se realiza de forma similar para el caso de la variable Y, que representa la construcción o no del arco. Esta variable se lee de la siguiente forma: por ejemplo: $y[1,2]$ el término (1,2) representa el arco.

A continuación, se muestran las restricciones del problema:

Subject To

Restricción 1: Caso nodo origen coincide con el nodo de origen del par OD:

Flujo_Norigen[1,5]: $x[500,1,1,5] + x[501,7,1,5] + x[502,13,1,5] = 300$

$$\text{Flujo_Norigen}[1,3]: x[503,1,1,3] + x[504,7,1,3] + x[505,13,1,3] = 700$$

$$\text{Flujo_Norigen}[2,6]: x[506,2,2,6] + x[507,8,2,6] + x[508,14,2,6] = 100$$

$$\text{Flujo_Norigen}[4,6]: x[509,4,4,6] + x[510,10,4,6] + x[511,16,4,6] = 400$$

Restricción 2: Caso nodo destino coincide con el nodo de destino del par OD:

$$\text{Flujo_Ndestino}[1,5]: x[5,600,1,5] + x[11,601,1,5] + x[17,602,1,5] = 300$$

$$\text{Flujo_Ndestino}[1,3]: x[3,603,1,3] + x[9,604,1,3] + x[15,605,1,3] = 700$$

$$\text{Flujo_Ndestino}[2,6]: x[6,606,2,6] + x[12,607,2,6] + x[18,608,2,6] = 100$$

$$\text{Flujo_Ndestino}[4,6]: x[6,609,4,6] + x[12,610,4,6] + x[18,611,4,6] = 400$$

Restricción 3: Balance de flujo nodos intermedios:

$$\text{Flujo_Nintermedios}[1,1,5]: -x[1,2,1,5] - x[1,4,1,5] + x[500,1,1,5] - x[1,7,1,5] - x[1,13,1,5] + x[7,1,1,5] + x[13,1,1,5] = 0$$

$$\text{Flujo_Nintermedios}[1,1,3]: -x[1,2,1,3] - x[1,4,1,3] + x[503,1,1,3] - x[1,7,1,3] - x[1,13,1,3] + x[7,1,1,3] + x[13,1,1,3] = 0$$

$$\text{Flujo_Nintermedios}[1,2,6]: -x[1,2,2,6] - x[1,4,2,6] - x[1,7,2,6] - x[1,13,2,6] + x[7,1,2,6] + x[13,1,2,6] = 0$$

$$\text{Flujo_Nintermedios}[1,4,6]: -x[1,2,4,6] - x[1,4,4,6] - x[1,7,4,6] - x[1,13,4,6] + x[7,1,4,6] + x[13,1,4,6] = 0$$

$$\text{Flujo_Nintermedios}[2,1,5]: x[1,2,1,5] - x[2,3,1,5] - x[2,5,1,5] - x[2,8,1,5] - x[2,14,1,5] + x[8,2,1,5] + x[14,2,1,5] = 0$$

$$\text{Flujo_Nintermedios}[2,1,3]: x[1,2,1,3] - x[2,3,1,3] - x[2,5,1,3] - x[2,8,1,3] - x[2,14,1,3] + x[8,2,1,3] + x[14,2,1,3] = 0$$

$$\text{Flujo_Nintermedios}[2,2,6]: x[1,2,2,6] - x[2,3,2,6] - x[2,5,2,6] + x[506,2,2,6] - x[2,8,2,6] - x[2,14,2,6] + x[8,2,2,6] + x[14,2,2,6] = 0$$

$$\text{Flujo_Nintermedios}[2,4,6]: x[1,2,4,6] - x[2,3,4,6] - x[2,5,4,6] - x[2,8,4,6] - x[2,14,4,6] + x[8,2,4,6] + x[14,2,4,6] = 0$$

$$\text{Flujo_Nintermedios}[3,1,5]: x[2,3,1,5] - x[3,6,1,5] - x[3,9,1,5] - x[3,15,1,5] + x[9,3,1,5] + x[15,3,1,5] = 0$$

$$\text{Flujo_Nintermedios}[3,1,3]: x[2,3,1,3] - x[3,6,1,3] - x[3,603,1,3] - x[3,9,1,3] - x[3,15,1,3] + x[9,3,1,3] + x[15,3,1,3] = 0$$

$$\text{Flujo_Nintermedios}[3,2,6]: x[2,3,2,6] - x[3,6,2,6] - x[3,9,2,6] - x[3,15,2,6] + x[9,3,2,6] + x[15,3,2,6] = 0$$

$$\text{Flujo_Nintermedios}[3,4,6]: x[2,3,4,6] - x[3,6,4,6] - x[3,9,4,6] - x[3,15,4,6] + x[9,3,4,6] + x[15,3,4,6] = 0$$

$$\text{Flujo_Nintermedios}[4,1,5]: -x[4,5,1,5] + x[1,4,1,5] - x[4,10,1,5] - x[4,16,1,5] + x[10,4,1,5] + x[16,4,1,5] = 0$$

$$\text{Flujo_Nintermedios}[4,1,3]: -x[4,5,1,3] + x[1,4,1,3] - x[4,10,1,3] - x[4,16,1,3] + x[10,4,1,3] + x[16,4,1,3] = 0$$

$$\text{Flujo_Nintermedios}[4,2,6]: -x[4,5,2,6] + x[1,4,2,6] - x[4,10,2,6] - x[4,16,2,6] + x[10,4,2,6] + x[16,4,2,6] = 0$$

$$\text{Flujo_Nintermedios}[4,4,6]: -x[4,5,4,6] + x[1,4,4,6] + x[509,4,4,6] - x[4,10,4,6] - x[4,16,4,6] + x[10,4,4,6] + x[16,4,4,6] = 0$$

$$\text{Flujo_Nintermedios}[5,1,5]: x[4,5,1,5] - x[5,6,1,5] + x[2,5,1,5] - x[5,600,1,5] - x[5,11,1,5] - x[5,17,1,5] + x[11,5,1,5] + x[17,5,1,5] = 0$$

$$\text{Flujo_Nintermedios}[5,1,3]: x[4,5,1,3] - x[5,6,1,3] + x[2,5,1,3] - x[5,11,1,3] - x[5,17,1,3] + x[11,5,1,3] + x[17,5,1,3] = 0$$

$$\text{Flujo_Nintermedios}[5,2,6]: x[4,5,2,6] - x[5,6,2,6] + x[2,5,2,6] - x[5,11,2,6] - x[5,17,2,6] + x[11,5,2,6] + x[17,5,2,6] = 0$$

$$\text{Flujo_Nintermedios}[5,4,6]: x[4,5,4,6] - x[5,6,4,6] + x[2,5,4,6] - x[5,11,4,6] - x[5,17,4,6] + x[11,5,4,6] + x[17,5,4,6] = 0$$

$$\text{Flujo_Nintermedios}[6,1,5]: x[5,6,1,5] + x[3,6,1,5] - x[6,12,1,5] - x[6,18,1,5] + x[12,6,1,5] + x[18,6,1,5] = 0$$

$$\text{Flujo_Nintermedios}[6,1,3]: x[5,6,1,3] + x[3,6,1,3] - x[6,12,1,3] - x[6,18,1,3] + x[12,6,1,3] + x[18,6,1,3] = 0$$

$$\text{Flujo_Nintermedios}[6,2,6]: x[5,6,2,6] + x[3,6,2,6] - x[6,606,2,6] - x[6,12,2,6] - x[6,18,2,6] + x[12,6,2,6] + x[18,6,2,6] = 0$$

$$\text{Flujo_Nintermedios}[6,4,6]: x[5,6,4,6] + x[3,6,4,6] - x[6,609,4,6] - x[6,12,4,6] - x[6,18,4,6] + x[12,6,4,6] + x[18,6,4,6] = 0$$

$$\text{Flujo_Nintermedios}[7,1,5]: -x[7,8,1,5] - x[7,10,1,5] + x[501,7,1,5] + x[1,7,1,5] - x[7,13,1,5] - x[7,1,1,5] + x[13,7,1,5] = 0$$

$$\text{Flujo_Nintermedios}[7,1,3]: -x[7,8,1,3] - x[7,10,1,3] + x[504,7,1,3] + x[1,7,1,3] - x[7,13,1,3] - x[7,1,1,3] + x[13,7,1,3] = 0$$

$$\begin{aligned}
&\text{Flujo_Nintermedios}[7,2,6]: -x[7,8,2,6] - x[7,10,2,6] + x[1,7,2,6] - x[7,13,2,6] - x[7,1,2,6] + x[13,7,2,6] = 0 \\
&\text{Flujo_Nintermedios}[7,4,6]: -x[7,8,4,6] - x[7,10,4,6] + x[1,7,4,6] - x[7,13,4,6] - x[7,1,4,6] + x[13,7,4,6] = 0 \\
&\text{Flujo_Nintermedios}[8,1,5]: x[7,8,1,5] - x[8,9,1,5] - x[8,11,1,5] + x[2,8,1,5] - x[8,14,1,5] - x[8,2,1,5] + x[14,8,1,5] = 0 \\
&\text{Flujo_Nintermedios}[8,1,3]: x[7,8,1,3] - x[8,9,1,3] - x[8,11,1,3] + x[2,8,1,3] - x[8,14,1,3] - x[8,2,1,3] + x[14,8,1,3] = 0 \\
&\text{Flujo_Nintermedios}[8,2,6]: x[7,8,2,6] - x[8,9,2,6] - x[8,11,2,6] + x[507,8,2,6] + x[2,8,2,6] - x[8,14,2,6] - x[8,2,2,6] + \\
&x[14,8,2,6] = 0 \\
&\text{Flujo_Nintermedios}[8,4,6]: x[7,8,4,6] - x[8,9,4,6] - x[8,11,4,6] + x[2,8,4,6] - x[8,14,4,6] - x[8,2,4,6] + x[14,8,4,6] = 0 \\
&\text{Flujo_Nintermedios}[9,1,5]: x[8,9,1,5] - x[9,12,1,5] + x[3,9,1,5] - x[9,15,1,5] - x[9,3,1,5] + x[15,9,1,5] = 0 \\
&\text{Flujo_Nintermedios}[9,1,3]: x[8,9,1,3] - x[9,12,1,3] - x[9,604,1,3] + x[3,9,1,3] - x[9,15,1,3] - x[9,3,1,3] + x[15,9,1,3] = 0 \\
&\text{Flujo_Nintermedios}[9,2,6]: x[8,9,2,6] - x[9,12,2,6] + x[3,9,2,6] - x[9,15,2,6] - x[9,3,2,6] + x[15,9,2,6] = 0 \\
&\text{Flujo_Nintermedios}[9,4,6]: x[8,9,4,6] - x[9,12,4,6] + x[3,9,4,6] - x[9,15,4,6] - x[9,3,4,6] + x[15,9,4,6] = 0 \\
&\text{Flujo_Nintermedios}[10,1,5]: -x[10,11,1,5] + x[7,10,1,5] + x[4,10,1,5] - x[10,16,1,5] - x[10,4,1,5] + x[16,10,1,5] = 0 \\
&\text{Flujo_Nintermedios}[10,1,3]: -x[10,11,1,3] + x[7,10,1,3] + x[4,10,1,3] - x[10,16,1,3] - x[10,4,1,3] + x[16,10,1,3] = 0 \\
&\text{Flujo_Nintermedios}[10,2,6]: -x[10,11,2,6] + x[7,10,2,6] + x[4,10,2,6] - x[10,16,2,6] - x[10,4,2,6] + x[16,10,2,6] = 0 \\
&\text{Flujo_Nintermedios}[10,4,6]: -x[10,11,4,6] + x[7,10,4,6] + x[510,10,4,6] + x[4,10,4,6] - x[10,16,4,6] - x[10,4,4,6] + \\
&x[16,10,4,6] = 0 \\
&\text{Flujo_Nintermedios}[11,1,5]: x[10,11,1,5] - x[11,12,1,5] + x[8,11,1,5] - x[11,601,1,5] + x[5,11,1,5] - x[11,17,1,5] - \\
&x[11,5,1,5] + x[17,11,1,5] = 0 \\
&\text{Flujo_Nintermedios}[11,1,3]: x[10,11,1,3] - x[11,12,1,3] + x[8,11,1,3] + x[5,11,1,3] - x[11,17,1,3] - x[11,5,1,3] + \\
&x[17,11,1,3] = 0 \\
&\text{Flujo_Nintermedios}[11,2,6]: x[10,11,2,6] - x[11,12,2,6] + x[8,11,2,6] + x[5,11,2,6] - x[11,17,2,6] - x[11,5,2,6] + \\
&x[17,11,2,6] = 0 \\
&\text{Flujo_Nintermedios}[11,4,6]: x[10,11,4,6] - x[11,12,4,6] + x[8,11,4,6] + x[5,11,4,6] - x[11,17,4,6] - x[11,5,4,6] + \\
&x[17,11,4,6] = 0 \\
&\text{Flujo_Nintermedios}[12,1,5]: x[11,12,1,5] + x[9,12,1,5] + x[6,12,1,5] - x[12,18,1,5] - x[12,6,1,5] + x[18,12,1,5] = 0 \\
&\text{Flujo_Nintermedios}[12,1,3]: x[11,12,1,3] + x[9,12,1,3] + x[6,12,1,3] - x[12,18,1,3] - x[12,6,1,3] + x[18,12,1,3] = 0 \\
&\text{Flujo_Nintermedios}[12,2,6]: x[11,12,2,6] + x[9,12,2,6] - x[12,607,2,6] + x[6,12,2,6] - x[12,18,2,6] - x[12,6,2,6] + \\
&x[18,12,2,6] = 0 \\
&\text{Flujo_Nintermedios}[12,4,6]: x[11,12,4,6] + x[9,12,4,6] - x[12,610,4,6] + x[6,12,4,6] - x[12,18,4,6] - x[12,6,4,6] + \\
&x[18,12,4,6] = 0 \\
&\text{Flujo_Nintermedios}[13,1,5]: -x[13,14,1,5] - x[13,16,1,5] + x[502,13,1,5] + x[7,13,1,5] + x[1,13,1,5] - x[13,7,1,5] - \\
&x[13,1,1,5] = 0 \\
&\text{Flujo_Nintermedios}[13,1,3]: -x[13,14,1,3] - x[13,16,1,3] + x[505,13,1,3] + x[7,13,1,3] + x[1,13,1,3] - x[13,7,1,3] - \\
&x[13,1,1,3] = 0 \\
&\text{Flujo_Nintermedios}[13,2,6]: -x[13,14,2,6] - x[13,16,2,6] + x[7,13,2,6] + x[1,13,2,6] - x[13,7,2,6] - x[13,1,2,6] = 0 \\
&\text{Flujo_Nintermedios}[13,4,6]: -x[13,14,4,6] - x[13,16,4,6] + x[7,13,4,6] + x[1,13,4,6] - x[13,7,4,6] - x[13,1,4,6] = 0 \\
&\text{Flujo_Nintermedios}[14,1,5]: x[13,14,1,5] - x[14,15,1,5] - x[14,17,1,5] + x[8,14,1,5] + x[2,14,1,5] - x[14,8,1,5] - x[14,2,1,5] \\
&= 0 \\
&\text{Flujo_Nintermedios}[14,1,3]: x[13,14,1,3] - x[14,15,1,3] - x[14,17,1,3] + x[8,14,1,3] + x[2,14,1,3] - x[14,8,1,3] - x[14,2,1,3] \\
&= 0 \\
&\text{Flujo_Nintermedios}[14,2,6]: x[13,14,2,6] - x[14,15,2,6] - x[14,17,2,6] + x[508,14,2,6] + x[8,14,2,6] + x[2,14,2,6] -
\end{aligned}$$

$$x[14,8,2,6] - x[14,2,2,6] = 0$$

$$\text{Flujo_Nintermedios}[14,4,6]: x[13,14,4,6] - x[14,15,4,6] - x[14,17,4,6] + x[8,14,4,6] + x[2,14,4,6] - x[14,8,4,6] - x[14,2,4,6] = 0$$

$$\text{Flujo_Nintermedios}[15,1,5]: x[14,15,1,5] - x[15,18,1,5] + x[9,15,1,5] + x[3,15,1,5] - x[15,9,1,5] - x[15,3,1,5] = 0$$

$$\text{Flujo_Nintermedios}[15,1,3]: x[14,15,1,3] - x[15,18,1,3] - x[15,605,1,3] + x[9,15,1,3] + x[3,15,1,3] - x[15,9,1,3] - x[15,3,1,3] = 0$$

$$\text{Flujo_Nintermedios}[15,2,6]: x[14,15,2,6] - x[15,18,2,6] + x[9,15,2,6] + x[3,15,2,6] - x[15,9,2,6] - x[15,3,2,6] = 0$$

$$\text{Flujo_Nintermedios}[15,4,6]: x[14,15,4,6] - x[15,18,4,6] + x[9,15,4,6] + x[3,15,4,6] - x[15,9,4,6] - x[15,3,4,6] = 0$$

$$\text{Flujo_Nintermedios}[16,1,5]: -x[16,17,1,5] + x[13,16,1,5] + x[10,16,1,5] + x[4,16,1,5] - x[16,10,1,5] - x[16,4,1,5] = 0$$

$$\text{Flujo_Nintermedios}[16,1,3]: -x[16,17,1,3] + x[13,16,1,3] + x[10,16,1,3] + x[4,16,1,3] - x[16,10,1,3] - x[16,4,1,3] = 0$$

$$\text{Flujo_Nintermedios}[16,2,6]: -x[16,17,2,6] + x[13,16,2,6] + x[10,16,2,6] + x[4,16,2,6] - x[16,10,2,6] - x[16,4,2,6] = 0$$

$$\text{Flujo_Nintermedios}[16,4,6]: -x[16,17,4,6] + x[13,16,4,6] + x[511,16,4,6] + x[10,16,4,6] + x[4,16,4,6] - x[16,10,4,6] - x[16,4,4,6] = 0$$

$$\text{Flujo_Nintermedios}[17,1,5]: x[16,17,1,5] - x[17,18,1,5] + x[14,17,1,5] - x[17,602,1,5] + x[11,17,1,5] + x[5,17,1,5] - x[17,11,1,5] - x[17,5,1,5] = 0$$

$$\text{Flujo_Nintermedios}[17,1,3]: x[16,17,1,3] - x[17,18,1,3] + x[14,17,1,3] + x[11,17,1,3] + x[5,17,1,3] - x[17,11,1,3] - x[17,5,1,3] = 0$$

$$\text{Flujo_Nintermedios}[17,2,6]: x[16,17,2,6] - x[17,18,2,6] + x[14,17,2,6] + x[11,17,2,6] + x[5,17,2,6] - x[17,11,2,6] - x[17,5,2,6] = 0$$

$$\text{Flujo_Nintermedios}[17,4,6]: x[16,17,4,6] - x[17,18,4,6] + x[14,17,4,6] + x[11,17,4,6] + x[5,17,4,6] - x[17,11,4,6] - x[17,5,4,6] = 0$$

$$\text{Flujo_Nintermedios}[18,1,5]: x[17,18,1,5] + x[15,18,1,5] + x[12,18,1,5] + x[6,18,1,5] - x[18,12,1,5] - x[18,6,1,5] = 0$$

$$\text{Flujo_Nintermedios}[18,1,3]: x[17,18,1,3] + x[15,18,1,3] + x[12,18,1,3] + x[6,18,1,3] - x[18,12,1,3] - x[18,6,1,3] = 0$$

$$\text{Flujo_Nintermedios}[18,2,6]: x[17,18,2,6] + x[15,18,2,6] - x[18,608,2,6] + x[12,18,2,6] + x[6,18,2,6] - x[18,12,2,6] - x[18,6,2,6] = 0$$

$$\text{Flujo_Nintermedios}[18,4,6]: x[17,18,4,6] + x[15,18,4,6] - x[18,611,4,6] + x[12,18,4,6] + x[6,18,4,6] - x[18,12,4,6] - x[18,6,4,6] = 0$$

Restricción 4: Limitación de presupuesto:

$$\text{Limitación_Presupuesto_Contrucción}[1,5]: 300000 y[7,8] + 300000 y[8,9] + 300000 y[10,11] + 300000 y[11,12] + 300000 y[7,10] + 300000 y[8,11] + 300000 y[9,12] + 750000 y[13,14] + 750000 y[14,15] + 750000 y[16,17] + 750000 y[17,18] + 750000 y[13,16] + 750000 y[14,17] + 750000 y[15,18] \leq 700000$$

Restricción 5: Garantizar que en un arco solo se pueda asignar una única tecnología:

$$\text{Limitación_Tecnologíaaporarcos}[1,2]: y[1,2] + y[7,8] + y[13,14] = 1$$

$$\text{Limitación_Tecnologíaaporarcos}[2,3]: y[2,3] + y[8,9] + y[14,15] = 1$$

$$\text{Limitación_Tecnologíaaporarcos}[4,5]: y[4,5] + y[10,11] + y[16,17] = 1$$

$$\text{Limitación_Tecnologíaaporarcos}[5,6]: y[5,6] + y[11,12] + y[17,18] = 1$$

$$\text{Limitación_Tecnologíaaporarcos}[1,4]: y[1,4] + y[7,10] + y[13,16] = 1$$

$$\text{Limitación_Tecnologíaaporarcos}[2,5]: y[2,5] + y[8,11] + y[14,17] = 1$$

$$\text{Limitación_Tecnología por arcos}[3,6]: y[3,6] + y[9,12] + y[15,18] = 1$$

Restricción 6: Garantiza que el flujo sobre un arco dado está sujeto a la existencia de la infraestructura en ese arco

$$\begin{aligned}
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[1,2,1,5]: x[1,2,1,5] - 300 y[1,2] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[1,2,1,3]: x[1,2,1,3] - 700 y[1,2] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[1,2,2,6]: x[1,2,2,6] - 100 y[1,2] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[1,2,4,6]: x[1,2,4,6] - 400 y[1,2] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[2,3,1,5]: x[2,3,1,5] - 300 y[2,3] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[2,3,1,3]: x[2,3,1,3] - 700 y[2,3] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[2,3,2,6]: x[2,3,2,6] - 100 y[2,3] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[2,3,4,6]: x[2,3,4,6] - 400 y[2,3] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[4,5,1,5]: x[4,5,1,5] - 300 y[4,5] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[4,5,1,3]: x[4,5,1,3] - 700 y[4,5] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[4,5,2,6]: x[4,5,2,6] - 100 y[4,5] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[4,5,4,6]: x[4,5,4,6] - 400 y[4,5] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[5,6,1,5]: x[5,6,1,5] - 300 y[5,6] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[5,6,1,3]: x[5,6,1,3] - 700 y[5,6] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[5,6,2,6]: x[5,6,2,6] - 100 y[5,6] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[5,6,4,6]: x[5,6,4,6] - 400 y[5,6] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[1,4,1,5]: x[1,4,1,5] - 300 y[1,4] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[1,4,1,3]: x[1,4,1,3] - 700 y[1,4] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[1,4,2,6]: x[1,4,2,6] - 100 y[1,4] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[1,4,4,6]: x[1,4,4,6] - 400 y[1,4] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[2,5,1,5]: x[2,5,1,5] - 300 y[2,5] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[2,5,1,3]: x[2,5,1,3] - 700 y[2,5] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[2,5,2,6]: x[2,5,2,6] - 100 y[2,5] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[2,5,4,6]: x[2,5,4,6] - 400 y[2,5] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[3,6,1,5]: x[3,6,1,5] - 300 y[3,6] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[3,6,1,3]: x[3,6,1,3] - 700 y[3,6] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[3,6,2,6]: x[3,6,2,6] - 100 y[3,6] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[3,6,4,6]: x[3,6,4,6] - 400 y[3,6] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[7,8,1,5]: x[7,8,1,5] - 300 y[7,8] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[7,8,1,3]: x[7,8,1,3] - 700 y[7,8] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[7,8,2,6]: x[7,8,2,6] - 100 y[7,8] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[7,8,4,6]: x[7,8,4,6] - 400 y[7,8] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[8,9,1,5]: x[8,9,1,5] - 300 y[8,9] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[8,9,1,3]: x[8,9,1,3] - 700 y[8,9] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[8,9,2,6]: x[8,9,2,6] - 100 y[8,9] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[8,9,4,6]: x[8,9,4,6] - 400 y[8,9] \leq 0 \\
&\text{Existencia_Infraestructura_Sujeta_a_flujo}[10,11,1,5]: x[10,11,1,5] - 300 y[10,11] \leq 0
\end{aligned}$$

Existencia_Infraestructura_Sujeta_a_flujo[10,11,1,3]: $x[10,11,1,3] - 700 y[10,11] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[10,11,2,6]: $x[10,11,2,6] - 100 y[10,11] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[10,11,4,6]: $x[10,11,4,6] - 400 y[10,11] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[11,12,1,5]: $x[11,12,1,5] - 300 y[11,12] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[11,12,1,3]: $x[11,12,1,3] - 700 y[11,12] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[11,12,2,6]: $x[11,12,2,6] - 100 y[11,12] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[11,12,4,6]: $x[11,12,4,6] - 400 y[11,12] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[7,10,1,5]: $x[7,10,1,5] - 300 y[7,10] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[7,10,1,3]: $x[7,10,1,3] - 700 y[7,10] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[7,10,2,6]: $x[7,10,2,6] - 100 y[7,10] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[7,10,4,6]: $x[7,10,4,6] - 400 y[7,10] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[8,11,1,5]: $x[8,11,1,5] - 300 y[8,11] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[8,11,1,3]: $x[8,11,1,3] - 700 y[8,11] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[8,11,2,6]: $x[8,11,2,6] - 100 y[8,11] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[8,11,4,6]: $x[8,11,4,6] - 400 y[8,11] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[9,12,1,5]: $x[9,12,1,5] - 300 y[9,12] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[9,12,1,3]: $x[9,12,1,3] - 700 y[9,12] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[9,12,2,6]: $x[9,12,2,6] - 100 y[9,12] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[9,12,4,6]: $x[9,12,4,6] - 400 y[9,12] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[13,14,1,5]: $x[13,14,1,5] - 300 y[13,14] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[13,14,1,3]: $x[13,14,1,3] - 700 y[13,14] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[13,14,2,6]: $x[13,14,2,6] - 100 y[13,14] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[13,14,4,6]: $x[13,14,4,6] - 400 y[13,14] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[14,15,1,5]: $x[14,15,1,5] - 300 y[14,15] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[14,15,1,3]: $x[14,15,1,3] - 700 y[14,15] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[14,15,2,6]: $x[14,15,2,6] - 100 y[14,15] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[14,15,4,6]: $x[14,15,4,6] - 400 y[14,15] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[16,17,1,5]: $x[16,17,1,5] - 300 y[16,17] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[16,17,1,3]: $x[16,17,1,3] - 700 y[16,17] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[16,17,2,6]: $x[16,17,2,6] - 100 y[16,17] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[16,17,4,6]: $x[16,17,4,6] - 400 y[16,17] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[17,18,1,5]: $x[17,18,1,5] - 300 y[17,18] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[17,18,1,3]: $x[17,18,1,3] - 700 y[17,18] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[17,18,2,6]: $x[17,18,2,6] - 100 y[17,18] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[17,18,4,6]: $x[17,18,4,6] - 400 y[17,18] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[13,16,1,5]: $x[13,16,1,5] - 300 y[13,16] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[13,16,1,3]: $x[13,16,1,3] - 700 y[13,16] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[13,16,2,6]: $x[13,16,2,6] - 100 y[13,16] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[13,16,4,6]: $x[13,16,4,6] - 400 y[13,16] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[14,17,1,5]: $x[14,17,1,5] - 300 y[14,17] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[14,17,1,3]: $x[14,17,1,3] - 700 y[14,17] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[14,17,2,6]: $x[14,17,2,6] - 100 y[14,17] \leq 0$

Existencia_Infraestructura_Sujeta_a_flujo[509,4,1,5]: $x[509,4,1,5] - 300 y[509,4] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[509,4,1,3]: $x[509,4,1,3] - 700 y[509,4] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[509,4,2,6]: $x[509,4,2,6] - 100 y[509,4] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[509,4,4,6]: $x[509,4,4,6] - 400 y[509,4] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[510,10,1,5]: $x[510,10,1,5] - 300 y[510,10] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[510,10,1,3]: $x[510,10,1,3] - 700 y[510,10] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[510,10,2,6]: $x[510,10,2,6] - 100 y[510,10] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[510,10,4,6]: $x[510,10,4,6] - 400 y[510,10] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[511,16,1,5]: $x[511,16,1,5] - 300 y[511,16] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[511,16,1,3]: $x[511,16,1,3] - 700 y[511,16] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[511,16,2,6]: $x[511,16,2,6] - 100 y[511,16] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[511,16,4,6]: $x[511,16,4,6] - 400 y[511,16] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[5,600,1,5]: $x[5,600,1,5] - 300 y[5,600] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[5,600,1,3]: $x[5,600,1,3] - 700 y[5,600] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[5,600,2,6]: $x[5,600,2,6] - 100 y[5,600] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[5,600,4,6]: $x[5,600,4,6] - 400 y[5,600] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[11,601,1,5]: $x[11,601,1,5] - 300 y[11,601] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[11,601,1,3]: $x[11,601,1,3] - 700 y[11,601] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[11,601,2,6]: $x[11,601,2,6] - 100 y[11,601] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[11,601,4,6]: $x[11,601,4,6] - 400 y[11,601] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[17,602,1,5]: $x[17,602,1,5] - 300 y[17,602] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[17,602,1,3]: $x[17,602,1,3] - 700 y[17,602] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[17,602,2,6]: $x[17,602,2,6] - 100 y[17,602] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[17,602,4,6]: $x[17,602,4,6] - 400 y[17,602] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[3,603,1,5]: $x[3,603,1,5] - 300 y[3,603] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[3,603,1,3]: $x[3,603,1,3] - 700 y[3,603] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[3,603,2,6]: $x[3,603,2,6] - 100 y[3,603] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[3,603,4,6]: $x[3,603,4,6] - 400 y[3,603] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[9,604,1,5]: $x[9,604,1,5] - 300 y[9,604] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[9,604,1,3]: $x[9,604,1,3] - 700 y[9,604] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[9,604,2,6]: $x[9,604,2,6] - 100 y[9,604] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[9,604,4,6]: $x[9,604,4,6] - 400 y[9,604] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[15,605,1,5]: $x[15,605,1,5] - 300 y[15,605] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[15,605,1,3]: $x[15,605,1,3] - 700 y[15,605] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[15,605,2,6]: $x[15,605,2,6] - 100 y[15,605] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[15,605,4,6]: $x[15,605,4,6] - 400 y[15,605] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[6,606,1,5]: $x[6,606,1,5] - 300 y[6,606] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[6,606,1,3]: $x[6,606,1,3] - 700 y[6,606] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[6,606,2,6]: $x[6,606,2,6] - 100 y[6,606] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[6,606,4,6]: $x[6,606,4,6] - 400 y[6,606] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[12,607,1,5]: $x[12,607,1,5] - 300 y[12,607] \leq 0$
Existencia_Infraestructura_Sujeta_a_flujo[12,607,1,3]: $x[12,607,1,3] - 700 y[12,607] \leq 0$

Existencia_Infraestructura_Sujeta_a_flujo[12,607,2,6]: $x[12,607,2,6] - 100 y[12,607] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[12,607,4,6]: $x[12,607,4,6] - 400 y[12,607] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[18,608,1,5]: $x[18,608,1,5] - 300 y[18,608] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[18,608,1,3]: $x[18,608,1,3] - 700 y[18,608] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[18,608,2,6]: $x[18,608,2,6] - 100 y[18,608] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[18,608,4,6]: $x[18,608,4,6] - 400 y[18,608] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[6,609,1,5]: $x[6,609,1,5] - 300 y[6,609] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[6,609,1,3]: $x[6,609,1,3] - 700 y[6,609] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[6,609,2,6]: $x[6,609,2,6] - 100 y[6,609] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[6,609,4,6]: $x[6,609,4,6] - 400 y[6,609] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[12,610,1,5]: $x[12,610,1,5] - 300 y[12,610] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[12,610,1,3]: $x[12,610,1,3] - 700 y[12,610] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[12,610,2,6]: $x[12,610,2,6] - 100 y[12,610] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[12,610,4,6]: $x[12,610,4,6] - 400 y[12,610] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[18,611,1,5]: $x[18,611,1,5] - 300 y[18,611] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[18,611,1,3]: $x[18,611,1,3] - 700 y[18,611] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[18,611,2,6]: $x[18,611,2,6] - 100 y[18,611] \leq 0$
 Existencia_Infraestructura_Sujeta_a_flujo[18,611,4,6]: $x[18,611,4,6] - 400 y[18,611] \leq 0$

Finalmente, muestra el listado de las variables generadas:

Variable Binaria

$y[1,2]$ $y[2,3]$ $y[4,5]$ $y[5,6]$ $y[1,4]$ $y[2,5]$ $y[3,6]$ $y[7,8]$ $y[8,9]$ $y[10,11]$ $y[11,12]$ $y[7,10]$ $y[8,11]$ $y[9,12]$ $y[13,14]$ $y[14,15]$
 $y[16,17]$ $y[17,18]$ $y[13,16]$ $y[14,17]$ $y[15,18]$ $y[500,1]$ $y[501,7]$ $y[502,13]$ $y[503,1]$ $y[504,7]$ $y[505,13]$ $y[506,2]$ $y[507,8]$
 $y[508,14]$ $y[509,4]$ $y[510,10]$ $y[511,16]$ $y[5,600]$ $y[11,601]$ $y[17,602]$ $y[3,603]$ $y[9,604]$ $y[15,605]$ $y[6,606]$ $y[12,607]$
 $y[18,608]$ $y[6,609]$ $y[12,610]$ $y[18,611]$

Generales

$x[1,2,1,5]$ $x[1,2,1,3]$ $x[1,2,2,6]$ $x[1,2,4,6]$ $x[2,3,1,5]$ $x[2,3,1,3]$ $x[2,3,2,6]$ $x[2,3,4,6]$ $x[4,5,1,5]$ $x[4,5,1,3]$ $x[4,5,2,6]$
 $x[4,5,4,6]$ $x[5,6,1,5]$ $x[5,6,1,3]$ $x[5,6,2,6]$ $x[5,6,4,6]$ $x[1,4,1,5]$ $x[1,4,1,3]$ $x[1,4,2,6]$ $x[1,4,4,6]$ $x[2,5,1,5]$ $x[2,5,1,3]$ $x[2,5,2,6]$
 $x[2,5,4,6]$ $x[3,6,1,5]$ $x[3,6,1,3]$ $x[3,6,2,6]$ $x[3,6,4,6]$ $x[7,8,1,5]$ $x[7,8,1,3]$ $x[7,8,2,6]$ $x[7,8,4,6]$ $x[8,9,1,5]$ $x[8,9,1,3]$ $x[8,9,2,6]$
 $x[8,9,4,6]$ $x[10,11,1,5]$ $x[10,11,1,3]$ $x[10,11,2,6]$ $x[10,11,4,6]$ $x[11,12,1,5]$ $x[11,12,1,3]$ $x[11,12,2,6]$ $x[11,12,4,6]$ $x[7,10,1,5]$
 $x[7,10,1,3]$ $x[7,10,2,6]$ $x[7,10,4,6]$ $x[8,11,1,5]$ $x[8,11,1,3]$ $x[8,11,2,6]$ $x[8,11,4,6]$ $x[9,12,1,5]$ $x[9,12,1,3]$ $x[9,12,2,6]$
 $x[9,12,4,6]$ $x[13,14,1,5]$ $x[13,14,1,3]$ $x[13,14,2,6]$ $x[13,14,4,6]$ $x[14,15,1,5]$ $x[14,15,1,3]$ $x[14,15,2,6]$
 $x[14,15,4,6]$ $x[16,17,1,5]$ $x[16,17,1,3]$ $x[16,17,2,6]$ $x[16,17,4,6]$ $x[17,18,1,5]$ $x[17,18,1,3]$ $x[17,18,2,6]$ $x[17,18,4,6]$
 $x[13,16,1,5]$ $x[13,16,1,3]$ $x[13,16,2,6]$ $x[13,16,4,6]$ $x[14,17,1,5]$ $x[14,17,1,3]$ $x[14,17,2,6]$ $x[14,17,4,6]$ $x[15,18,1,5]$
 $x[15,18,1,3]$ $x[15,18,2,6]$ $x[15,18,4,6]$ $x[500,1,1,5]$ $x[500,1,1,3]$ $x[500,1,2,6]$ $x[500,1,4,6]$ $x[501,7,1,5]$ $x[501,7,1,3]$
 $x[501,7,2,6]$ $x[501,7,4,6]$ $x[502,13,1,5]$ $x[502,13,1,3]$ $x[502,13,2,6]$ $x[502,13,4,6]$ $x[503,1,1,5]$ $x[503,1,1,3]$
 $x[503,1,2,6]$ $x[503,1,4,6]$ $x[504,7,1,5]$ $x[504,7,1,3]$ $x[504,7,2,6]$ $x[504,7,4,6]$ $x[505,13,1,5]$ $x[505,13,1,3]$ $x[505,13,2,6]$
 $x[505,13,4,6]$ $x[506,2,1,5]$ $x[506,2,1,3]$ $x[506,2,2,6]$ $x[506,2,4,6]$ $x[507,8,1,5]$ $x[507,8,1,3]$ $x[507,8,2,6]$ $x[507,8,4,6]$
 $x[508,14,1,5]$ $x[508,14,1,3]$ $x[508,14,2,6]$ $x[508,14,4,6]$ $x[509,4,1,5]$ $x[509,4,1,3]$ $x[509,4,2,6]$ $x[509,4,4,6]$ $x[510,10,1,5]$

x[510,10,1,3] x[510,10,2,6] x[510,10,4,6] x[511,16,1,5]x[511,16,1,3] x[511,16,2,6] x[511,16,4,6] x[5,600,1,5]
x[5,600,1,3]x[5,600,2,6] x[5,600,4,6] x[11,601,1,5] x[11,601,1,3] x[11,601,2,6]x[11,601,4,6] x[17,602,1,5] x[17,602,1,3]
x[17,602,2,6] x[17,602,4,6]x[3,603,1,5] x[3,603,1,3] x[3,603,2,6] x[3,603,4,6] x[9,604,1,5]x[9,604,1,3] x[9,604,2,6]
x[9,604,4,6] x[15,605,1,5] x[15,605,1,3]x[15,605,2,6] x[15,605,4,6] x[6,606,1,5] x[6,606,1,3] x[6,606,2,6]x[6,606,4,6]
x[12,607,1,5] x[12,607,1,3] x[12,607,2,6] x[12,607,4,6]x[18,608,1,5] x[18,608,1,3] x[18,608,2,6] x[18,608,4,6]
x[6,609,1,5]x[6,609,1,3] x[6,609,2,6] x[6,609,4,6] x[12,610,1,5] x[12,610,1,3]x[12,610,2,6] x[12,610,4,6] x[18,611,1,5]
x[18,611,1,3] x[18,611,2,6]x[18,611,4,6] x[1,7,1,5] x[1,7,1,3] x[1,7,2,6] x[1,7,4,6] x[2,8,1,5]x[2,8,1,3] x[2,8,2,6] x[2,8,4,6]
x[3,9,1,5] x[3,9,1,3] x[3,9,2,6]x[3,9,4,6] x[4,10,1,5] x[4,10,1,3] x[4,10,2,6] x[4,10,4,6] x[5,11,1,5]x[5,11,1,3] x[5,11,2,6]
x[5,11,4,6] x[6,12,1,5] x[6,12,1,3] x[6,12,2,6]x[6,12,4,6] x[7,13,1,5] x[7,13,1,3] x[7,13,2,6] x[7,13,4,6]
x[8,14,1,5]x[8,14,1,3] x[8,14,2,6] x[8,14,4,6] x[9,15,1,5] x[9,15,1,3] x[9,15,2,6]x[9,15,4,6] x[10,16,1,5] x[10,16,1,3]
x[10,16,2,6] x[10,16,4,6]x[11,17,1,5] x[11,17,1,3] x[11,17,2,6] x[11,17,4,6] x[12,18,1,5]x[12,18,1,3] x[12,18,2,6]
x[12,18,4,6] x[1,13,1,5] x[1,13,1,3] x[1,13,2,6]x[1,13,4,6] x[2,14,1,5] x[2,14,1,3] x[2,14,2,6] x[2,14,4,6]
x[3,15,1,5]x[3,15,1,3] x[3,15,2,6] x[3,15,4,6] x[4,16,1,5] x[4,16,1,3] x[4,16,2,6]x[4,16,4,6] x[5,17,1,5] x[5,17,1,3]
x[5,17,2,6] x[5,17,4,6] x[6,18,1,5] x[6,18,1,3] x[6,18,2,6] x[6,18,4,6] x[7,1,1,5] x[7,1,1,3] x[7,1,2,6]x[7,1,4,6] x[8,2,1,5]
x[8,2,1,3] x[8,2,2,6] x[8,2,4,6] x[9,3,1,5]x[9,3,1,3] x[9,3,2,6] x[9,3,4,6] x[10,4,1,5] x[10,4,1,3] x[10,4,2,6]x[10,4,4,6]
x[11,5,1,5] x[11,5,1,3] x[11,5,2,6] x[11,5,4,6] x[12,6,1,5]x[12,6,1,3] x[12,6,2,6] x[12,6,4,6] x[13,7,1,5] x[13,7,1,3]
x[13,7,2,6]x[13,7,4,6] x[14,8,1,5] x[14,8,1,3] x[14,8,2,6] x[14,8,4,6] x[15,9,1,5]x[15,9,1,3] x[15,9,2,6] x[15,9,4,6]
x[16,10,1,5] x[16,10,1,3] x[16,10,2,6]x[16,10,4,6] x[17,11,1,5] x[17,11,1,3] x[17,11,2,6] x[17,11,4,6] x[18,12,1,5]
x[18,12,1,3] x[18,12,2,6] x[18,12,4,6] x[13,1,1,5]x[13,1,1,3] x[13,1,2,6] x[13,1,4,6] x[14,2,1,5] x[14,2,1,3]
x[14,2,2,6]x[14,2,4,6] x[15,3,1,5] x[15,3,1,3] x[15,3,2,6] x[15,3,4,6] x[16,4,1,5]x[16,4,1,3] x[16,4,2,6] x[16,4,4,6]
x[17,5,1,5] x[17,5,1,3] x[17,5,2,6]x[17,5,4,6] x[18,6,1,5] x[18,6,1,3] x[18,6,2,6] x[18,6,4,6]

7.2 CÓDIGO DE LA METODOLOGÍA

```

1. #####LIBRERIAS#####
2. from typing import Set, Any, Tuple
3. import matplotlib.pyplot as plt
4. import networkx as nx
5. import numpy
6. import numpy as np
7. from numpy.lib import math
8. from IOfunctionsExcel5 import *
9. from MyPlotandNetworkFunctions5 import *
10. from gurobipy import *
11. from heapq import heappush, heappop
12. from itertools import count
13. import json as js
14. import io
15. import ast
16. import random
17. import gurobipy
18. #####
19. # #####
20. #
21. #
22. #
23. #
24. #
25. #
26. #
27. #
28. #
29. #
30. #
31. #
32. #
33. #
34. #
35. #
36. #
37. #
38. #
39. #
40. #
41. #
42. #
43. #
44. #
45. #
46. #
47. #
48. #
49. #
50. #
51. #
52. #
53. #
54. #
55. #
56. #
57. #
58. #
59. #
60. #
61. #
62. #
63. #
64. #
65. #
66. #
67. #

```

```

68.     ### PARAMETROS ###
69.     ## COSTE DE CONSTRCCUIÓN POR TECNOLOGÍA###
70.     c_T1 = 3000
71.     c_T2 = 6000
72.     ### PRESUPUESTO
73.     P=500000000
74.     ### COSTE DEL USUARIO POR TECNOLOGÍA###
75.     U_T1 = 20
76.     U_T2 = 5
77.     U_B = 50
78.     ### PENALIZAR CAMBIO DE TECNOLOGÍA###
79.     bT2_T1 = 2
80.     bT1_B = 1
81.     bT2_B = 3
82.     ##CONJUTNO OD
83.     W = [(2,6),(7,9),(4,6),(1,7)]
84.     exp1 = {
85.         'Nodos': ['A2', 'A10'],
86.         'Coordenadas': ['A1', 'C10'],
87.         'Arcos': ['I1', 'I13'],
88.         'DArcos': ['I1', 'L13'],
89.         'Escala': ['G2'],
90.         'Nodo1': ['G4'],
91.         'Nodo2': ['G6'],
92.         'CODE': ['A1', 'A10'],
93.         'CoordenadasX': ['B2', 'B10'],
94.         'CoordenadasY': ['C2', 'C10'],
95.     }
96. if ESCENARIO==3:
97.
98.     name1 = "NODOSSEV_MT.xlsx"
99.     sheet1 = 'DATOSPYTHON'
100.    ### PARAMETROS ###
101.    ## COSTE DE CONSTRCCUIÓN POR TECNOLOGÍA###
102.    c_T1 = 400
103.    c_T2 = 700
104.    ### PRESUPUESTO
105.    P=15000000
106.    ### COSTE DEL USUARIO POR TECNOLOGÍA###
107.    U_T1 = 20
108.    U_T2 = 5
109.    U_B = 50
110.    ### PENALIZAR CAMBIO DE TECNOLOGÍA###
111.    bT2_T1 = 2
112.    bT1_B = 1
113.    bT2_B = 3
114.    ##CONJUTNO OD
115.    #W = [(1, 60),(37,107),(106,123),(124,139),(60,104), (102,139),(60,139),
116.    #      (1,107),(6,110), (128,139),(86,126),(54,60),(38,52),(91,128),
117.    #      (52,72),(20,38),(91,96)]
118.    W = [(1, 60), (37, 107), (106, 123), (124, 139), (60, 104), (102, 139), (60, 139),
119.    (1, 107), (6, 110), (128, 139), (86, 126), (54, 60), (38, 52), (91, 128),
120.    (52, 72), (20, 38), (91, 96), (6, 117), (117, 133), (37, 133), (126, 133), (24,
121.    133),
122.    (2, 117),(6,119), (58,61),(84,130),(107,138)]
123.    exp1 = {
124.        'Nodos': ['A2', 'A141'],
125.        'Coordenadas': ['A1', 'C141'],
126.        'Arcos': ['I1', 'I228'],
127.        'DArcos': ['I1', 'L228'],
128.        'Escala': ['G2'],
129.        'Nodo1': ['G5'],
130.        'Nodo2': ['G7'],
131.        'CODE': ['A1', 'A141'],
132.        'CoordenadasX': ['B2', 'B141'],
133.        'CoordenadasY': ['C2', 'C141'],
134.    }
135. # *****
136. # *****
137. #
138. #

```

```

139. # *****
140.
141. ##### Lectura tecnologia 1#####
142. Escala = 300
143. print(">>> Lectura de datos T1")
144.
145. #LEER NODOS
146.
147. L_Nodos1 = Read_Excel_to_List(name1, sheet1, exp1['Nodos'][0], exp1['Nodos'][1])
148.
149. # LEER COORDENADAS X E Y
150.
151. D_Coordenadas1 = Read_Excel_to_NesteDic(name1, sheet1, exp1['Coordenadas'][0],
exp1['Coordenadas'][1])
152. #print(D_Coordenadas1)
153.
154. # LEER ARCOS
155.
156. D_Arcos1 = Read_Excel_to_NesteDic_TupleInt(name1, sheet1, exp1['DArcos'][0],
exp1['DArcos'][1])
157.
158. #LEER NODOS COMO DICCIONARIO
159.
160. #Necesito los nodos como un diccionario, para poder asignarle las coordenadas
161.
162. nodosXY = Read_Excel_to_NesteDic(name1, sheet1, exp1['CODE'][0], exp1['CODE'][1])
163.
164. # LEER COORDENADAS X E Y
165.
166. #Coordenadas X e Y separadas para poder agregarla al diccionario "nodosXY"
167.
168. D_CoordenadasX = Read_Excel_to_List(name1, sheet1, exp1['CoordenadasX'][0],
exp1['CoordenadasX'][1])
169. D_CoordenadasY = Read_Excel_to_List(name1, sheet1, exp1['CoordenadasY'][0],
exp1['CoordenadasY'][1])
170.
171. #LEER ARCOS COMO DICCIONARIO
172.
173. #Necesito los ARCOS como un diccionario, para poder asignarle
174.
175. L_ArcosT1=Read_Excel_to_NesteDic_TupleInt(name1, sheet1, exp1['Arcos'][0],
exp1['Arcos'][1])
176. #print(L_ArcosT1)
177.
178. #*****
179. # *****
180. # *****
181. #
182. #             PARTE III
183. #             Estructuras de datos para guardar escenarios.
184. #             Duplicar arcos. Cálculo de longitudes de arcos. Conversion a unidades
185. #             adecuadas aplicando factor de escala a cada red
186. # *****
187.
188. print(">>> ESTRUTURA DE DATOS")
189.
190. D_ArcosT1={} # Nivel1 D_Arcos_tecnologia 1
191. for (i,j) in D_Arcos1:
192.     D_ArcosT1[(i,j)]=D_Arcos1[(i,j)]
193.     D_ArcosT1[(j,i)]={}
194.
195. #Duplicate arcos (doble sentido)
196. for (i,j) in D_Arcos1:
197.     D_ArcosT1[(j,i)][ 'ORIGEN' ]=D_Arcos1[(i,j)][ 'DESTINO' ]
198.     D_ArcosT1[(j, i)][ 'DESTINO' ]=D_Arcos1[(i,j)][ 'ORIGEN' ]
199.     D_ArcosT1[(j,i)][ 'FACTOR' ]=D_Arcos1[(i,j)][ 'FACTOR' ]
200.
201. D_LongitudT1 = {} # Nivel1 D_Longitudes_Nivel1
202. for i in D_ArcosT1:
203.     a = D_ArcosT1[i][ 'ORIGEN' ]

```

```

204.     b = D_ArcosT1[i]['DESTINO']
205.     c = D_ArcosT1[i]['FACTOR']
206.     D_LongitudT1[(a, b)] = numpy.sqrt((D_Coordenadas1[a]['X'] - D_Coordenadas1[b]['X']) **
2 + (D_Coordenadas1[a]['Y'] - D_Coordenadas1[b]['Y']) ** 2) * c
207.     #print(D_LongitudT1)
208.     ##### Aplicando factor de escala a la red peatonal
#####
209.     # Para normalizar la distancia entre los nodos, tomamos la longitud calculada
anteriormente
210.     # y con la distancia real calculamos el factor de escala
211.     # La medida real que hemos tomado ha sido la distancia del nodo 1 al 2 (150 m)
212.
213.     numerador=Read_Excel_Cell(name1, sheet1, exp1['Escala'][0])
214.     nodo1=Read_Excel_Cell(name1, sheet1, exp1['Nodo1'][0])
215.     nodo2=Read_Excel_Cell(name1, sheet1, exp1['Nodo2'][0])
216.     denominador=numpy.sqrt((D_Coordenadas1[nodo1]['X'] - D_Coordenadas1[nodo2]['X']) ** 2 +
(D_Coordenadas1[nodo1]['Y'] - D_Coordenadas1[nodo2]['Y']) ** 2)
217.     factorescala1 = numerador / denominador
218.
219.     print("----- Factor Escala ->", factorescala1)
220.
221.     for i in D_ArcosT1:
222.         a = D_ArcosT1[i]['ORIGEN']
223.         b = D_ArcosT1[i]['DESTINO']
224.         c = D_ArcosT1[i]['FACTOR']
225.         D_LongitudT1[(a, b)] = D_LongitudT1[(a, b)] *c* factorescala1
226.
227.     #print( D_LongitudT1)
228.
229.     # *****
230.     # *****
231.     #
PARTE IV
232.     #
CARGA DE DATOS PARA GUROBI
233.     # #
*****Lectura*****
234.     # *****
235.     Escala1=100
236.     print(">>> Cargando y revisando Matriz OD del escenario")
237.
238.     #Lectura de la matriz de demanda y escalado según ESCENARIO
239.
240.     if ESCENARIO==1:
241.         sheet1 = 'MATOD'
242.         DD_MatOD = Read_Excel_to_NesteDic(name1, sheet1, 'A1', 'G7')
243.
244.     if ESCENARIO== 2:
245.         sheet1 = 'MATOD'
246.         DD_MatOD = Read_Excel_to_NesteDic(name1, sheet1, 'A1', 'J10')
247.
248.     if ESCENARIO== 3:
249.         sheet1 = 'MATOD'
250.         DD_MatOD = Read_Excel_to_NesteDic(name1, sheet1, 'A1', 'EK141')
251.
252.
253.     #print(DD_MatOD)
254.
255.     for i in L_Nodos1:
256.         for j in L_Nodos1:
257.             DD_MatOD[i][j] = DD_MatOD[i][j]*Escala1
258.
259.     #print (DD_MatOD)
260.
261.     #*****
262.     # *****
263.     #
PARTE V
264.     #
GUROBI
265.     # *****
266.     # *****
267.     print('>>>INICIO GUROBI')
268.
269.     #####INTRODUCCION DE LAS LIBRERIAS#####
270.

```

```

271. from gurobipy import GRB, quicksum, Model, multidict
272. import gurobipy as gp
273. from gurobipy import *
274. import gurobipy
275. from numpy import *
276.
277. ##### SET #####
278. #Conjunto de nodos del grafo base
279. nodos=L_Nodos1.copy()
280. #Conjunto de arcos del grafo base
281. #####GRAFO BASE= GRAFO EN EL QUE SE BASARÁ LAS RÉPLICAS EN FUNCIÓN DE LAS
TECNOLOGÍAS#####
282. arc={}#####CONJUNTO DE ARCOS#####
283.
284. for i,j in L_ArcosT1:
285.     arc.setdefault((i,j),{})
286.     #arc.setdefault((j, i), {})
287. #print(arc)
288. #Matriz de demanda
289.
290. D={}#####DEMANDA EN EL ARCO#####
291. for i in L_Nodos1:
292.     for j in L_Nodos1:
293.         if i!=j:
294.             D[i,j] = DD_MatOD[i][j]
295.
296. #print(sigma)
297.
298. ##### GENERAR GRAFOS EN FUNCIÓN DE TECNOLOGÍAS #####
299.
300. ##### DEFINIMOS NODOS ##### OJO hay que definir tantas listas de diccionarios de nodos
seg{un la tecnología a utilizar
301. nodosT1=[]
302. nodosT1.extend(nodos)
303. nodosT2=[]
304. nodosT2.extend(nodos) ### REPETIR TANTAS TECNOLOGÍAS SE TENGA
305. #print("Conjunto de nodos T1:",nodosT1)
306. #print("Conjunto de nodos T2:",nodosT2)
307.
308. ##### EL CONJUNTO DE NODOS TOTALES= sirve para saber los nodos del grafo expandido
309. ##### sirve para saber los nodos del grafo de transferencia
310. nodosf1= nodos+ nodosT1+ nodosT2 ##### OJO hay que actualizar segun listas de nodos por
tecnologías se tenga
311. nodosf=list(range(1,len(nodosf1)+1))
312. #print(nodosf)
313. ##### Para poder diferenciar los nodos de las diferentes tecnologías le damos continuidad
314. # a la numeración.
315.
316. ## OJO HAY QUE REPETIR ESTE PASO DEPENDIENDO DEL NUMERO DE TENCOLÓGÍAS QUE SE TENGA
317.
318. nodosT1=list(range(len(nodos)+1,(2*(len(nodos))+1)))
319. #print("Conjunto de nodos T1:",nodosT1)
320.
321. nodosf11= nodos+ nodosT1 # Replicar en caso de tener más tecnologías
322. nodosT2=list(range(len(nodosf11)+1,(3*(len(nodos))+1)))
323. #print("Conjunto de nodos T2:",nodosT2)
324.
325.
326. # ##### DEFINIMOS ARCOS QUE CONECTAN GRAFOS #####
327. ### Aquí se crea los arcos de cambio ###
328.
329. ### UNIMOS LOS NODOS DEL GRAFO BASE CON LOS NODOS DE T1
330. arcosB_T1={}
331. arcosT1_B={}
332. for i in enumerate(nodos):
333.     for j in enumerate(nodosT1):
334.         if i[0]==j[0]:
335.             arcosB_T1[i[1],j[1]] = {}
336.             arcosT1_B[j[1],i[1]] = {}
337. #print(arcosB_T1)

```



```

338.
339. ### UNIMOS LOS NODOS DEL GRAFO BASE CON LOS NODOS DE T2
340. arcosB_T2={}
341. arcosT2_B={}
342. for i in enumerate(nodos):
343.     for j in enumerate(nodosT2):
344.         if i[0]==j[0]:
345.             arcosB_T2[i[1], j[1]] = {}
346.             arcosT2_B[j[1], i[1]] = {}
347. #print(arcosB_T2)
348.
349. ### UNIMOS LOS NODOS DEL GRAFO T1 CON LOS NODOS DE T2
350. arcosT1_T2={}
351. arcosT2_T1={}
352. for i in enumerate(nodosT1):
353.     for j in enumerate(nodosT2):
354.         if i[0]==j[0]:
355.             arcosT1_T2[i[1], j[1]] = {}
356.             arcosT2_T1[j[1], i[1]] = {}
357. #print(arcosT1_T2)
358.
359. ### CONJUNTO DE ARCOS QUE CONECTA A LAS TECNOLOGÍAS
360.
361. arcC={}
362. arcC.update(arcosB_T1)
363. arcC.update(arcosT1_T2)
364. arcC.update(arcosB_T2)
365. arcC.update(arcosT1_B)
366. arcC.update(arcosT2_T1)
367. arcC.update(arcosT2_B)
368. arcCB={}
369. arcC.update(arcosT1_B)
370. arcC.update(arcosT2_T1)
371. arcC.update(arcosT2_B)
372. #print(arcT)
373.
374. ### Se define el conjunto de arcos de cada tecnología
375. ### en este caso las conexiones permanecen iguales para cada tecnología
376. ### lo que cambia es la nomenclatura del nodo
377. #print(L_ArcosT1)
378.
379. ##### Grafo tecnología 1
380.
381. #El conjunto de nodos es nodosT1
382. arcT1_o={}
383. arcT1_d={}
384. destino1={}
385. origen1={}
386.
387. for i in arc:
388.     origen1[i]=i[0]
389.     arcT1_o.update(origen1)
390.     destino1[i] = i[1]
391.     arcT1_d.update(destino1)
392. #print(arcT1_o)
393. #print(arcT1_d)
394.
395. for i in arc:
396.     for j in nodos:
397.         if arcT1_o[i]==nodos[j-1]:
398.             arcT1_o[i]=nodosT1[j-1]
399.         if arcT1_d[i]==nodos[j-1]:
400.             arcT1_d[i]=nodosT1[j-1]
401. #print(arc)
402. #print(arcT1_d)
403. ##### Aqui se crea el conjunto de arcos de la tecnología 1
404. arcT1={}
405. arcT11={}
406. for i,j in arcT1_o and arcT1_d:
407.     arcT11=arcT1_o[i,j],arcT1_d[i,j]
408.     arcT1.setdefault(arcT11, {})
409. #print(arcT1)

```

```

410.
411. ##### Grafo tecnología 2
412.
413. #El conjunto de nodos es nodosT2
414. arcT2_o={}
415. arcT2_d={}
416. origen2={}
417. destino2={}
418. for i in arc:
419.     origen2[i]=i[0]
420.     arcT2_o.update(origen2)
421.     destino2[i] = i[1]
422.     arcT2_d.update(destino2)
423. #print(arcT2_o)
424. #print(arcT2_d)
425.
426. for i in arc:
427.     for j in nodos:
428.         if arcT2_o[i]==nodos[j-1]:
429.             arcT2_o[i]=nodosT2[j-1]
430.         if arcT2_d[i]==nodos[j-1]:
431.             arcT2_d[i]=nodosT2[j-1]
432. #print(arcT2_o)
433. #print(arcT2_d)
434. ##### Aqui se crea el conjunto de arcos de la tecnología 2
435. arcT2={}
436. arcT22={}
437.
438. for i,j in arcT2_o and arcT2_d:
439.     arcT22=arcT2_o[i,j],arcT2_d[i,j]
440.     arcT2.setdefault(arcT22,{})
441. #print(arcT2)
442.
443. ##### Conjunto arcos que pertenecen a las tecnologías
444. arcI={}
445. arcI.update(arcT1)
446. arcI.update(arcT2)
447.
448.
449. Origenes=[]
450. Destinos=[]
451. M=500
452. N=600
453.
454. Infraestructuras=[0]
455. for i in T:#####OJO con T
456.     Infraestructuras.append(i)
457. dimension=np.zeros(len(Infraestructuras)*len(W))
458.
459. for i in dimension:
460.     Origenes.append(M)
461.     Destinos.append(N)
462.     M=M+1
463.     N=N+1
464.
465.
466. arcos_0 = {}
467. j=0
468. for i in W:
469.     arcos_0[i]= tuple([Origenes[j], nodos[i[0]-1]]), tuple([Origenes[j+1], nodosT1[i[0]-
470.     1]]), tuple([Origenes[j+2], nodosT2[i[0]-1]])
471.     j+=len(Infraestructuras)
472. #print(arcos_0, '<-')
473.
474. arcos_D = {}
475.
476. j=0
477. for i in W:

```

```

478.     arcos_D[i]= tuple([nodos[i[1]-1],Destinos[j]]), tuple([nodosT1[i[1]-
179.     ],Destinos[j+1]]), tuple([nodosT2[i[1]-1],Destinos[j+2]])
480.     #print(arcos_D, '<-')
481.
482.     arcos_OD= {}
483.
484.     for i,j in arcos_0:
485.         for l in range(0, len(arcos_0[i,j])):
486.             arcos_OD[arcos_0[i,j][l]] = {}
487.
488.     #print(arcos_D, '<----')
489.     for i,j in arcos_D:
490.         for l in range(0, len(arcos_D[i,j])):
491.             arcos_OD[arcos_D[i,j][l]] = {}
492.
493.     #print(arcos_OD, "<--")
494.     ##### Conjunto de arcos totales del grafo expandido
495.     ##### Contiene el conjunto de arcos OD + Conjunto de arcos de las transferencias
entre tecnologías T+ Conjunto d arcos de las tecnologías I
496.     #Para ec de balance
497.     A={}
498.     A.update(arc)
499.     A.update(arcI)# rcos de tTecnologías
500.     A.update(arcos_OD) # arcos OD
501.     A.update(arcC)#rcos de transferencia
502.     #print (A)
503.     #cONJUNTO DE ARCOS INCLUYE: ARCOS BASE, ARCOS DE TECNOLOGÍAS, ARCOS OD
504.     A1={}
505.     A1.update(arc)
506.     A1.update(arcI)# rcos de tTecnologías
507.     A1.update(arcos_OD)#rcos de transferencia
508.
509.     #cONJUNTO DE ARCOS INCLUYE: ARCOS BASE, ARCOS DE TECNOLOGÍAS, ARCOS TRANSFERENCIA
510.     Ai={}
511.     Aii={}
512.     Ai.update(arc)
513.     Ai.update(arcI)# rcos de tTecnologías
514.     Ai.update(arcC)#rcos de transferencia
515.
516.     arcos_O2 ={}
517.     arcos_D2 ={}
518.     for i,j in W:
519.         arcos_O2[i, j] = {}
520.         for m,l in arcos_0[i,j]:
521.             arcos_O2[i,j].update({(m,l):{}})
522.
523.     for i,j in W:
524.         arcos_D2[i, j] = {}
525.         for m,l in arcos_D[i,j]:
526.             arcos_D2[i,j].update({(m,l):{}})
527.
528.
529.     for i,j in W:
530.         Aii[i,j]=Ai.copy()
531.         Aii[i,j].update(arcos_O2[i,j])
532.         Aii[i,j].update(arcos_D2[i,j])
533.
534.
535.     #print(arcos_O2)
536.     #print(arcos_OD)
537.     #A.update(arcT)#rcos de transferencia
538.     ##### Conjunto M(a) es un dic que toma como valor el arco de grafo base y como
clave los arcos de la contraparte del arco base en T1 y T2
539.
540.     M= arc.copy()
541.
542.     j=0
543.     for val, i in enumerate(arc):
544.         #print(i)
545.         #H[i] = list(arcT1)[j], list(arcT2)[j]## Solo los arcos tecnologías
546.         M[i] = list(arcT1)[j], list(arcT2)[j],i

```

```

547.     j+=1
548. #print(M)
549. h1={}
550. h1.update(arcT1)
551. h1.update(arcT2)
552. h1.update(arc)
553.
554. #print(h1, '<-')
555.
556.
557. #####Conjuntos Costes de usuario, construccion y penalización
558.
559.
560. long_dB={}#####LONGITUD  Base#####
561.
562. for i,j in list(arc.keys()):
563.     long_dB.setdefault((i,j),D_LongitudT1[(i,j)])
564. #print(long_dB)
565.
566. long_dT1={}#####LONGITUD  T1#####
567.
568. for i in enumerate(arc):
569.     for j in enumerate(arcT1):
570.         if i[0]==j[0]:
571.             long_dT1.setdefault(j[1],long_dB[i[1]] )
572.             #print(i[1],j[1],long_dB[i[1]],"<-----")
573. #print(long_dT1, "<-----")
574.
575. long_dT2={}#####LONGITUD  T1#####
576.
577. for i in enumerate(arc):
578.     for j in enumerate(arcT2):
579.         if i[0]==j[0]:
580.             long_dT2.setdefault(j[1],long_dB[i[1]] )
581. #print(long_dT2, "<-----")
582.
583.
584. #Coste de usuario == Coste
585. #Coste de usuario grafo base
586. UB=arc.copy()
587. for i in UB:
588.     UB[i]=U_B#*long_dB[i[0],i[1]]
589. #print(CosteB)
590. #Coste de usuario grafo T1
591. UT1=arcT1.copy()
592. for i in UT1:
593.     UT1[i]=U_T1#*long_dT1[i[0],i[1]]
594. #print(CosteT1)
595. #Coste de usuario grafo T2
596. UT2=arcT2.copy()
597. for i in UT2:
598.     UT2[i]=U_T2#*long_dT2[i[0],i[1]]
599. #print(CosteT2)
600. #Coste de usuario en grafo OD
601. UOD=arcos_OD.copy()
602. for i in arcos_OD:
603.     UOD[i]=0
604. #print(CosteT2)
605.
606. #Coste de usuario en grafo transferencia
607. UT=arcC.copy()
608. for i in arcC:
609.     UT[i]=0
610. #print(arcT)
611.
612. ##### Coste de usuario #####
613. U={}
614. U.update(UB)
615. U.update(UT1)
616. U.update(UT2)

```

```

617. U.update(UOD)
618. U.update(UT)
619. #print(Coste)
620.
621. #Coste de contruccion
622.
623. CB=arc.copy()
624. for i in CB:
625.     CB[i]=0
626. #print(fB)
627. #Coste de usuario grafo T1
628. CT1=arcT1.copy()
629. for i in CT1:
630.     CT1[i]=c_T1*long_dT1[i[0],i[1]]
631. #print(fT1)
632. #Coste de usuario grafo T2
633. CT2=arcT2.copy()
634. for i in CT2:
635.     CT2[i]=c_T2*long_dT2[i[0],i[1]]
636. #print(fT2)
637. ##### Coste de construccion #####
638. C={}
639. C.update(CB)
640. C.update(CT1)
641. C.update(CT2)
642. #print(f)
643.
644. #penalización arcos de transferencia == ro
645. #penalización por cambiar del grafo base al grafo T1
646. ""##### de peor a mejor #####
647. roB_T1=arcosB_T1.copy()
648. for i in arcosB_T1:
649.     roB_T1[i]=5000
650. #print(roB_T1)
651.
652. ##penalización por cambiar del grafo base al grafo T2
653. roB_T2=arcosB_T2.copy()
654. for i in arcosB_T2:
655.     roB_T2[i]=5000
656. #print(roB_T2)
657.
658. #penalización por cambiar del grafo T1 al grafo T2
659. roT1_T2=arcosT1_T2.copy()
660. for i in arcosT1_T2:
661.     roT1_T2[i]=5000
662. #print(roT1_T1)"""
663.
664. ##### de mejor a peor #####
665. #penalización por cambiar del grafo T1 al grafo base
666. BT1_B=arcosT1_B.copy()
667. for i in arcosT1_B:
668.     BT1_B[i]=bT1_B
669. #print(roT1_B)
670.
671. ##penalización por cambiar del grafo T2 al grafo base
672. BT2_B=arcosT2_B.copy()
673. for i in arcosT2_B:
674.     BT2_B[i]=bT2_B
675. #print(roB_T2)
676.
677. #penalización por cambiar del grafo T1 al grafo T2
678. BT2_T1=arcosT2_T1.copy()
679. for i in arcosT2_T1:
680.     BT2_T1[i]=bT2_T1
681. #print(roT2_T1)
682.
683. ##### Penalización #####
684. B={}
685. #ro.update(roB_T1)
686. #ro.update(roB_T2)
687. #ro.update(roT1_T2)
688. B.update(BT1_B)

```

```

689. B.update(BT2_B)
690. B.update(BT2_T1)
691. #print(ro)
692.
693. ##### INICIALIZAR MODELO #####
694. #print(A)
695.
696. m = Model("modelo")
697.
698. ##### INTRODUCCION DE LAS VARIABLES DE DECISION #####
699. x={}
700. y={}
701.
702. x = m.addVars(A,W, vtype=GRB.INTEGER, name='x')
703. y = m.addVars(A1, vtype=GRB.BINARY, name='y')
704.
705. m.update()
706.
707.
708. ##### INTRODUCCION FUNCION OBJETIVO #####
709.
710.
711. m.setObjective((quicksum((U[i,j]*x[i,j,t,v] for i,j in A1 for t,v in
W)))+(quicksum(B[i,j]*x[i,j,t,v] for i,j in arcCB for t,v in W)))
712. # Restricciones
713.
714. #####
715. m.addConstrs(((quicksum(x[i,j,t,v] for i,j in arcos_0[t,v]))==D[t,v] for t,v in W),
name='Flujo_Norigen')
716.
717. m.addConstrs(((quicksum(x[i,j,t,v] for i,j in arcos_D[t,v]))==D[t,v] for t,v in W),
name='Flujo_Ndestino')
718.
719. m.addConstrs((-quicksum(x[n,j,t,v] for i,j in Aii[t,v] if (n==i)) + quicksum(x[i,n,t,v]
for i,j in Aii[t,v] if (n==j))==0 for n in nodosf for t,v in W), name='Flujo_Nintermedios')
720.
721. m.addConstr((quicksum(y[i,j]*C[i,j] for i,j in arcI) <= P),
name='Limitación_Presupuesto_Construcción')
722.
723. m.addConstrs((quicksum(y[i,j] for i,j in M[l,m]) == 1 for l,m in arc),
name='Limitación_Tecnologíaaporarcos')
724.
725. m.addConstrs((x[i,j,t,v]<=y[i,j]*D[t,v] for i,j in A1 for t,v in W), name='Existencia
Infraestructura_Sujeta a flujo')
726. #####
727.
728. m.write('modelo.lp')
729. #m.computeIIS()
730. #m.write("model1.ilp")
731. m.update()
732. m.optimize()
733. #m.printStats()
734.
735. sol = m.getAttr('x',y)#SI HAY SOLUCIÓN IMPRIME LOS VALORES QUE TOMA X
736. #print(sol)
737. sol1 = m.getAttr('x',x)
738.
739. print("RESULTADOS MODELO")
740.
741. #VALOR FUNCIÓN OBJETIVO
742. #print("El resultado de la función objetivo es:", m.ObjVal)
743.
744. print("Variables de decision")
745. #VALOR DE LAS VARIABLES DE DECISIÓN
746. for v in m.getVars():#IMPRIME LOS VALORES QUE TOMA TANTO X COMO Y
747.     print(str(v.varName)+"="+str(v.x))
748.
749.
750. arcos_activos = [] #####SE NECESITA SABER CUALES SON LOS ARCOS QUE PERTENECEN A LA
SOLUCIÓN

```

```

751. arcos_sol=[]
752. for i, l in sol:
753.     if sol.get((i,l)) == 1 :
754.         arcos_activos.append((i, l))
755. #print(arcos_activos)
756.
757. for i,l,j,v in sol1:
758.     if (sol1[i,l,j,v]>=1) :
759.         arcos_sol.append([i,l])
760. arcos_sol1=removeDuplicates(arcos_sol)
761. #print(arcos_sol1)
762. arcos_activos1=[]
763. for i,j in arcos_activos:
764.     for l in arcos_sol1:
765.         if ((i,j)==(l[0],l[1])):
766.             arcos_activos1.append([i,j])
767.
768. #print(arcos_OD)
769.
770. ##### GRAFICA DE LA RED #####
771. #####
772.
773. print('>>PREPARACIÓN DE DATOS PARA GRAFICAR')
774. #####
775. # diccionario nodos
776. n_grafo = {}
777. for i in nodos:
778.     n_grafo = nodos.copy()
779.     #print(n_grafo)
780. e = {}
781. for i in range(1, len(n_grafo)+1):
782.     e[i] = str(i)
783.     #print (e) #Etiqueta del nodo = nombre del nodo que aparece en el grafo
784.
785. #Se debe buscar los arcos equivalentes al grafo base de las tecnologías
786. arc_grafo={}
787. for i in arcos_activos1:
788.     for j in M:
789.         if (i[0],i[1]) == M[j][0]:
790.             arc_grafo.setdefault(j,1) #### CUANDO SE DEFINIÓ H LA TUPLA DE LA POSICION 0
CORRESPONDE A T1 Y LA POSICION 1 A T2
791.                 #print(j,"primer B")
792.         if (i[0],i[1]) == M[j][1]:
793.             arc_grafo.setdefault(j,2)
794.         if (i[0], i[1]) == M[j][2]:
795.             #print((i[0], i[1]),M[j][2])
796.             arc_grafo.setdefault(j, 0)
797.
798. print("ARCOS QUE PERTENCEN A LA SOLUCIÓN, (INDICA LA TECNOLOGÍA QUE USA EL ARCO ")
799. for v in arc_grafo:#IMPRIEME LOS VALORES QUE TOMA TANTO X COMO Y
800.     print("y"+str(v)+" "+str(arc_grafo[v]))
801.     ##### COORDENADAS DE LOS NODOS #####
802.
803. for i in nodosXY:
804.     nodosXY[i]=(D_CoordenadasX[i-1],D_CoordenadasY[i-1])
805.     #print(nodosXY)
806.
807. ##### REALIZACIÓN DE LA GRAFICA #####
808. if ESCENARIO==3:
809.     figure = plt.figure(figsize=(7, 10))
810.
811. print('>>INICIO DE GRAFICA')
812.
813. G = nx.Graph()
814. nx.draw_networkx_nodes(G,nodosXY, n_grafo,node_size=1,node_color="black") ###RECIBE
DICcionario NODOXY Y LA LISTA DE NODOS
815.
816.
817. nx.draw_networkx_labels(G, pos=nodosXY, labels=e, font_color="black",font_size=7)
818. nx.draw_networkx_edges(G, nodosXY, arc, edge_color="silver",arrowstyle="-")
819. #print(nodosXY, '<-')
820.

```

```

821. k=0
822. Cont4=0
823. Cont5=0
824. Cont10=0
825. for i in arc_grafo:
826.     if arc_grafo[i]==1:
827.         arc_grafo_list=list(arc_grafo.keys())
828.
829.     nx.draw_networkx_edges(G,nodosXY,[arc_grafo_list[k]],edge_color="green",arrowstyle="-
|>",arrows=False)
830.     Cont4=Cont4+1
831.     if arc_grafo[i]==2:
832.         arc_grafo_list = list(arc_grafo.keys())
833.         nx.draw_networkx_edges(G, nodosXY,[arc_grafo_list[k]],
834.         edge_color="red",arrowstyle="-|>",arrows=False)
835.         Cont5=Cont5+1
836.         if arc_grafo[i]==0:
837.             arc_grafo_list = list(arc_grafo.keys())
838.             nx.draw_networkx_edges(G, nodosXY,[arc_grafo_list[k]],
839.             edge_color="black",arrowstyle="-|>",arrows=False)
840.             Cont10=Cont10+1
841.             k=k+1
842.             #print(arc_grafo)
843.             if DIBUJARED == 1:
844.                 plt.show()
845.
846. ##### IMPRIME RESULTADOS #####
847. arcos_activos2=[]
848. for i,j in arcos_activos1:
849.     if (i<500 ):
850.         if (j<600):
851.             arcos_activos2.append([i,j])
852.
853. Cont=0
854. for i in arcos_activos2:
855.     Cont=C[tuple(i)]+Cont
856.     print('El presupuesto disponible ----->',P,'u.m')
857.     print('Coste total de construcción de la red fue ----->', Cont , 'u.m.')
858.     #print("Coste total de viaje del usuario")
859.
860. Cont1=0
861. for i in arcos_activos2:
862.     Cont1=U[tuple(i)]+Cont1
863.     #print("U" + str(i) + " =" + str(U[tuple(i)]))
864.     print('Coste total de viaje del usuario:----->', Cont1 , 'u.m.')
865.
866. #print('Utilizacion del presupuesto')
867. Cont6=(Cont/P)*100
868.     print('Se ha utilizado----->-----', Cont6, '% del presupuesto')
869.
870. print('La red cuenta con', Cont10 , 'carriles con infraestructura 0')
871. print('La red cuenta con', Cont4 , 'carriles con infraestructura 1')
872. print('La red cuenta con', Cont5 , 'carriles con infraestructura 2')
873.

```