

Trabajo Fin de Grado

Optimización de rutas de recogida de fangos en
las depuradoras

Sludge Collection Routes Optimization in Treatment Plants

Autor

Guillermo Aldunate Ozcariz

Director

Luis Mariano Esteban

Escuela Universitaria Politécnica La Almunia

Junio 2024



**Escuela Universitaria
Politécnica** - La Almunia
Centro adscrito
Universidad Zaragoza

**ESCUELA UNIVERSITARIA POLITÉCNICA
DE LA ALMUNIA DE DOÑA GODINA (ZARAGOZA)**

MEMORIA

Optimización de rutas de recogida de fangos en las depuradoras

Sludge Collection Routes Optimization in Treatment Plants

608.23.38

Autor: Guillermo Aldunate Ozcariz

Director: Luis Mariano Esteban

Fecha: 29/02/2024



INDICE DE CONTENIDO BREVE

1. RESUMEN	-1-
2. ABSTRACT	-2-
3. INTRODUCCIÓN	-3-
4. MARCO TEÓRICO	-4-
5. DESARROLLO	-10-
6. RESULTADOS	-40-
7. CONCLUSIONES	-43-
8. OBJETIVOS DE DESARROLLO SOSTENIBLE	-45-
9. BIBLIOGRAFÍA	-46-

INDICE DE CONTENIDO

1. RESUMEN	- 1 -
1.1. PALABRAS CLAVE	- 1 -
2. ABSTRACT	- 2 -
2.1. KEY WORDS	- 2 -
3. INTRODUCCIÓN	- 3 -
3.1. OBJETIVOS	- 4 -
3.2. DEFINICIÓN DEL PROBLEMA	- 4 -
3.3. METODOLOGÍA PARA LOGRAR LOS OBJETIVOS	- 5 -
3.4. PARTES EN LAS QUE SE ESTRUCTURA EL TFG	- 5 -
4. MARCO TEÓRICO	- 4 -
4.1. ESTACIÓN DEPURADORA DE AGUAS RESIDUALES (EDAR)	- 4 -
4.1.1. <i>Etapas del agua en una EDAR</i>	- 5 -
4.1.2. <i>Residuos generados por una EDAR</i>	- 8 -
5. DESARROLLO	- 10 -
5.1. GESTIÓN Y EXTRACCIÓN DE FANGOS DE LAS EDAR	- 10 -
5.1.1. <i>Gestión de los fangos</i>	- 10 -
5.1.2. <i>Necesidad de la gestión de fangos</i>	- 11 -
5.1.3. <i>Recogida de Fangos de las EDAR</i>	- 12 -
5.1.4. <i>Estudio de producción de fangos de las EDAR</i>	- 12 -
5.1.5. <i>Estudio de mercado</i>	- 14 -
5.1.6. <i>Recopilación de datos</i>	- 15 -
5.1.7. <i>Comparación de sistemas</i>	- 18 -
5.1.7.1. Sistema actual de recogida de fangos	- 18 -
5.1.7.2. Recogida de fangos semanal	- 21 -
5.2. PROBLEMA DEL VIAJANTE Y ALGORITMOS GENÉTICOS	- 23 -
5.2.1. <i>Definición teórica</i>	- 23 -
5.2.2. <i>Tipos de TSPs</i>	- 24 -
5.2.3. <i>Algoritmos más utilizados</i>	- 25 -
5.2.4. <i>Aplicaciones</i>	- 25 -
5.2.5. <i>Algoritmos genéticos</i>	- 26 -
5.2.5.1. Procedimiento de un algoritmo genético	- 26 -
5.3. OBTENCIÓN DE LA SOLUCIÓN.	- 27 -
5.3.1. <i>Aplicación Shiny/R</i>	- 27 -
5.3.2. <i>Archivos del Programa.R</i>	- 29 -
5.4. INTERFAZ DE LA APLICACIÓN "RUTA DE RECOGIDA DE FANGOS"	- 37 -
6. RESULTADOS	- 40 -
6.1. RESULTADOS DE LA RUTA ACTUAL	- 40 -
7. CONCLUSIONES	- 43 -
8. OBJETIVOS DE DESARROLLO SOSTENIBLE	- 45 -
9. BIBLIOGRAFÍA	- 46 -

INDICE DE ILUSTRACIONES

Ilustración 1: Estación Depuradora de Aguas Residuales.	4 -
Ilustración 2: Aspecto de los Fangos generados.	4 -
Ilustración 3: Ciclo completo del Agua.	5 -
Ilustración 4: Entrada o Pretratamiento.	5 -
Ilustración 5: Decantador primario.	6 -
Ilustración 6: Tratamiento biológico.	6 -
Ilustración 7: Decantador Secundario.	7 -
Ilustración 8: Salida de Fangos y de agua del decantador secundario.	7 -
Ilustración 9: Separación de los lodos mediante decantación.	8 -
Ilustración 10: Almacenamiento de los fangos.	9 -
Ilustración 11: Mapa de Navarra con las EDAR.	11 -
Ilustración 12: Camiones de extracción de fangos.	13 -
Ilustración 13: Medidas de los almacenes.	15 -
Ilustración 14: Decantación de fangos.	15 -
Ilustración 15: Etapas de fangos en el Almacén.	16 -
Ilustración 16: Fangos solidificados.	17 -
Ilustración 17: Zona decantación de fangos en el Almacén.	18 -
Ilustración 18: Zonas críticas de atascos.	19 -
Ilustración 19: Solidificación de fangos.	19 -
Ilustración 20: Funcionamiento de un ratón de desatascos.	19 -
Ilustración 21: Extracción de tubería atascada.	20 -
Ilustración 22: Atasco de fangos.	20 -
Ilustración 23: Librería Shiny.	27 -
Ilustración 24: Librería TSP.	28 -
Ilustración 25: Librería RSconnect.	28 -
Ilustración 26: Librería xlsx.	28 -
Ilustración 27: Librería igraph.	28 -
Ilustración 28: Directorio del proyecto.	29 -
Ilustración 29: Coordenadas GPS de las EDAR.	29 -
Ilustración 30: Lectura de Coordenadas en R.	29 -
Ilustración 31: lectura del Excel de coordenadas.	30 -
Ilustración 32: Obtención de Matriz de distancias.	30 -
Ilustración 33: Matriz de distancias.	30 -
Ilustración 34: Código para la visualización de la aplicación.	31 -
Ilustración 35: width.	31 -
Ilustración 36: Código de salida de app.	31 -
Ilustración 37: Código de Solve_TSP.	32 -
Ilustración 38: Código de representación de rutas.	33 -
Ilustración 39: Solución de ruta con algoritmo genético.	33 -
Ilustración 40: Código de representación del mapa en la app.	34 -
Ilustración 41: Código para la representación de la ruta en el mapa.	35 -
Ilustración 42: Código de representación del algoritmo genético.	35 -
Ilustración 43: Código de representación de ruta.	36 -
Ilustración 44: Representación de valores en la aplicación.	36 -
Ilustración 45: Interfaz de la aplicación.	37 -
Ilustración 46: Panel de selección de municipios de la app.	38 -
Ilustración 47: Algoritmos para el cálculo de rutas.	38 -
Ilustración 48: Longitud de las rutas.	38 -
Ilustración 49: Ruta de reparto en la App.	39 -
Ilustración 50: Mapa del trayecto a realizar.	39 -



INDICE DE TABLAS

Tabla 1: Litros de fango a la semana en cada EDAR.....	12 -
Tabla 2: Frecuencia de extracción de fangos anuales.	16 -

1. RESUMEN

La optimización de rutas se erige como un proceso esencial que debe llevarse a cabo de manera sistemática en cualquier entidad o empresa que gestione un servicio de distribución o recolección. La eficiente optimización de estas rutas no solo conlleva una reducción considerable de costos, sino también una marcada disminución del tiempo destinado a las actividades de distribución o recolección.

En este proyecto, se ha analizado las rutas de recolección de fangos, así como el estudio de los costes y la organización que se requería para realizar esta tarea, además de realizar la optimización de las rutas de recolección de fangos provenientes de las depuradoras en diversas localidades de la zona media de Navarra. Este enfoque se orienta a resolver un problema real relacionado con la gestión de residuos derivados de estaciones de tratamiento de aguas residuales.

Los fangos, que en este caso abarcan todas las localidades navarras que cuentan con depuradoras, han sido objeto de estudio para su óptima recolección. El análisis de las rutas de recolección se fundamentó en la información proporcionada por los directores de las depuradoras de estas localidades de Navarra, que incluyó la ubicación de los puntos de recolección, el calendario de recolección de los fangos y las distancias y tiempos asociados a esta tarea.

Posteriormente, se detallaron los algoritmos seleccionados para abordar el problema de TSP (Travelling Salesman Problem), que serán los algoritmos del vecino más cercano, más lejano y el algoritmo genético, así como los programas y paquetes necesarios para llevar a cabo esta optimización. El propósito era desarrollar una aplicación capaz de ofrecer una ruta optimizada, permitiendo la incorporación eficaz de nuevos puntos de recolección para lograr una nueva ruta óptima en el contexto de las localidades navarras.

Finalmente, se presenta una explicación detallada de los pasos seguidos para obtener la solución al problema, mostrando los resultados obtenidos y comparándolos con los datos iniciales. Esto posibilita extraer conclusiones propias sobre la eficacia de la optimización de rutas aplicada a la gestión de fangos de depuradoras en diversas localidades de Navarra

1.1. PALABRAS CLAVE

Optimización, Rutas, TSP, Recogida, Fangos.

2. ABSTRACT

La optimization of routes emerges as an essential process that must be carried out systematically in any entity or company that manages a distribution or collection service. The efficient optimization of these routes not only entails a considerable reduction in costs but also a marked decrease in the time allocated to distribution or collection activities.

In this project, the collection routes of sludge have been analyzed, as well as the study of the costs and organization required to perform this task, in addition to optimizing the collection routes of sludge from sewage treatment plants in various locations in the central area of Navarra. This approach is aimed at solving a real problem related to the management of waste derived from wastewater treatment plants.

The sludge, which in this case includes all the Navarre towns with treatment plants, has been studied for its optimal collection. The analysis of the collection routes was based on information provided by the directors of the treatment plants in these Navarre towns, which included the location of the collection points, the collection schedule for the sludge, and the distances and times associated with this task.

Subsequently, the selected algorithms to address the TSP (Travelling Salesman Problem) were detailed, which will be the nearest neighbor, farthest neighbor, and genetic algorithm, as well as the programs and packages necessary to carry out this optimization. The purpose was to develop an application capable of offering an optimized route, allowing the effective incorporation of new collection points to achieve a new optimal route in the context of Navarre towns.

Finally, a detailed explanation of the steps followed to obtain the solution to the problem is presented, showing the results obtained and comparing them with the initial data. This enables drawing conclusions about the effectiveness of route optimization applied to the management of sludge from treatment plants in various locations in Navarra.

2.1. KEY WORDS

Optimization, Routes, TSP, Collection, Sludge.

3. INTRODUCCIÓN

La elección de este Trabajo de Fin de Grado (TFG) se basa en mi creciente interés a lo largo de la carrera en los temas de optimización, específicamente en la optimización de rutas.

En el ámbito del Grado de Ingeniería de Organización Industrial, se concede gran importancia a la mejora económica de las empresas mediante una gestión adecuada, y en este contexto, la optimización, tanto de rutas como en otros aspectos empresariales, desempeña un papel crucial.

La optimización de rutas resulta vital para la reducción de costos en una empresa, y este concepto se ha abordado en diversas asignaturas a lo largo de este grado, como Investigación Operativa, Ampliación de Investigación Operativa o Logística. Sin embargo, fue tras cursar la asignatura de Investigación Operativa cuando despertó mi interés particular en los temas de optimización.

Además, otra de las razones y motivaciones por las que realizo este trabajo es por el hecho de que durante el verano he realizado labores de mantenimiento en las depuradoras de la zona media de Navarra, teniendo la sede en Olite, próxima a mi localidad de residencia que es Tafalla. Por ello, planear una optimización de rutas para la recogida de fangos es algo que puede beneficiar a las localidades próximas, evitando así pérdidas tanto temporales como económicas.

La empresa encargada de la gestión de las depuradoras en este caso se trata de una subcontrata, que actúa como responsable principal del correcto funcionamiento de las depuradoras. Esta a su vez contrata a otras compañías para que realicen la extracción de los residuos tales como fangos.

El sistema actual de recogida a primera vista puede aparentar ser el más económico, debido a que se realizan menos extracciones de fangos, sin embargo, a causa de las diferencias de precios entre compañías y los numerosos atascos que se producen de manera reiterada, son un claro indicador de grandes pérdidas de tiempo y de dinero.

Es por ello, por lo que el TFG tiene como principal finalidad la de lograr una optimización de las rutas de recogida de fangos en las diferentes depuradoras. Con la idea de comparar los resultados existentes hoy en día y poder ofrecer una solución al problema.

3.1. OBJETIVOS

Como principal objetivo a desarrollar en este trabajo, se encuentra la idea de optimizar la ruta y el sistema organizativo de recogida de fangos de las diferentes depuradoras.

Con la intención de reducir las distancias recorridas por las compañías de extracción de fangos, minimizando así los costes económicos y de tiempo empleado en esta tarea.

Esto genera dos objetivos principales a cumplir:

- Búsqueda de la optimización de rutas realizadas por la empresa de recogida de fangos.
- Realizar un estudio previo del sistema actual de recogida de fangos y de posibles reformas en el sistema organizativo que permitan reducir costes.

3.2. DEFINICIÓN DEL PROBLEMA

Para definir el problema a resolver en este TFG, se deberá comenzar comentando que se trata de buscar una optimización de las rutas de recogida de fangos en algunas depuradoras de la zona media de Navarra, por lo que se trata de un problema y una situación real.

Actualmente, no existe ningún documento que planifique o indique con más o menos previsión de la frecuencia con la que se deben extraer fangos de cada depuradora.

Por ello, se estudiará los litros diarios de fango que se producen en cada depuradora. Buscando tener, una previsión fiable que indique la frecuencia optima con la que se deben extraer fangos de cada depuradora.

Una vez conocida esta frecuencia de extracción de fangos, se procederá obteniendo las coordenadas GPS específicas de cada uno de los puntos de recogida (Depuradoras). Para, posteriormente, convertir estas coordenadas en una matriz de distancias entre cada uno de los puntos de recogida en las que se relacionen todas ellas directamente con sus distancias exactas.

Tras obtener esta matriz, se procederá con la creación del código necesario en el software R, mediante el uso de librerías como TSP y Gor en la que se conseguirá obtener la solución a este problema, que se basa en la idea de encontrar la ruta más corta posible pasando una vez por cada punto de recogida y pasando al punto de partida.

3.3. METODOLOGÍA PARA LOGRAR LOS OBJETIVOS

Para lograr cumplir con los objetivos se van a tomar como referencia los siguientes pasos:

- Estudio del funcionamiento de las depuradoras y sus ciclos de fangos.
- Estudiar cual es el precio de cada compañía en función de las distancias y las cantidades de fangos a transportar.
- Obtención de las coordenadas de las depuradoras en las que se deberá realizar el control de nivel de fangos.
- Creación de la matriz de distancias entre puntos de extracción de fangos, eligiendo cual es el punto de partida y cuál es el punto final.
- Desarrollo del código y ejecución del algoritmo mediante el software R que realizará el cálculo de la distancia más corta.
- Implementación de la aplicación web con los datos obtenidos de las rutas optimas.
- Conclusiones y posibles propuestas de mejora.

3.4. PARTES EN LAS QUE SE ESTRUCTURA EL TFG

Para el desarrollo del trabajo se seguirá una estructura clara que trata los siguientes puntos:

1. Funcionamiento de Estación Depuradora de Aguas Residuales (EDAR).
2. Gestión y extracción de fangos de las depuradoras.
3. Estudio del mercado de las compañías de transporte de fangos.
4. Desarrollar el problema del viajante y el algoritmo TSP.
5. Herramientas necesarias para dar con el cálculo de la ruta óptima.
6. Desarrollo y análisis de la solución.
7. Conclusiones y propuestas de mejora.

4. MARCO TEÓRICO

Antes de comenzar con el desarrollo del sistema de optimización de rutas, se realizará una breve explicación de los aspectos más importantes que se deben conocer acerca de las depuradoras.

Se explicará de forma breve su funcionamiento, con la idea de poner en contexto el tema del que trata el trabajo, así como la importancia que tienen para el medio ambiente y para el ecosistema.

4.1. ESTACIÓN DEPURADORA DE AGUAS RESIDUALES (EDAR)

Una estación depuradora de aguas residuales (EDAR) puede definirse como un conjunto de instalaciones próximas a los ríos o canales, que han sido diseñadas para reducir la contaminación de las aguas residuales hasta límites que sean óptimos y aceptables para los canales en los que se va a depositar el agua.



Ilustración 1: Estación Depuradora de Aguas Residuales.

Además de ello, las EDAR cuentan con otro objetivo tan importante como el anterior. Ya que buscan tratar los fangos generados durante la depuración de estas aguas residuales para conseguir que sean aptos para su destino (vertedero, uso agrícola, compost, etc.).



Ilustración 2: Aspecto de los Fangos generados.

El esquema general básico de una planta de tratamiento de aguas residuales consta de dos circuitos principales: El de agua y el de fangos.[1]

4.1.1. Etapas del agua en una EDAR

Dentro de las EDAR como se ha comentado anteriormente, disponemos de dos circuitos principales, que son el del agua y el de fangos. En este apartado se tratará principalmente el del agua, la cual sigue un recorrido que está claramente explicado en la siguiente imagen.[2]

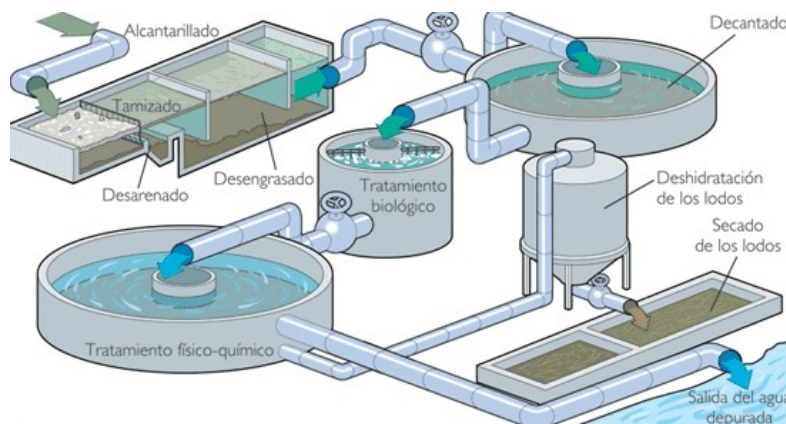


Ilustración 3: Ciclo completo del Agua.

Para realizar la explicación de este ciclo se va a dividir en cuatro etapas principales:[3]

- 1. Entrada o pretratamiento:** En esta etapa se busca eliminar materias gruesas, que se encuentran flotando o en suspensión en las aguas, y que por su tamaño u origen pueden causar problemas a lo largo del ciclo.



Ilustración 4: Entrada o Pretratamiento.

Se tamiza para quitar las materias más grandes, posteriormente se realiza un desarenado, cuya función principal consiste en soplar aire para así eliminar las partículas más pesadas.

Y por último se realiza un desengrasado, para eliminar las grasas, aceites, espumas y materiales más ligeros, que se encuentran flotando.

2. Decantación primaria: La decantación primaria trata de separar gran parte de los sólidos y material flotante que no lograron ser eliminados por los procesos anteriores.

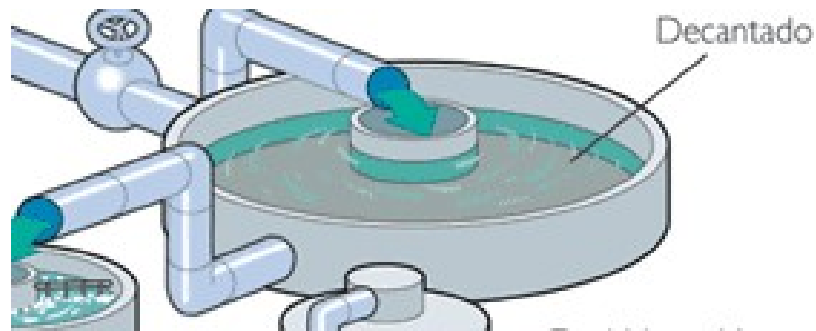


Ilustración 5: Decantador primario.

Los sólidos se depositan en la zona inferior del decantador, y mediante unas rasquetas giratorias se retiran. El agua decantada es enviada al tratamiento biológico.

3. Tratamiento biológico: En esta tercera fase, el agua es llevada a unos reactores biológicos, donde la materia orgánica será degradada por acción de una serie de microorganismos.



Ilustración 6: Tratamiento biológico.

En esta etapa se consigue eliminar la contaminación del agua por un proceso llamado biocenosis, que generalmente son microorganismos que se encuentran en un medio adecuado y con un ambiente controlado. Estos microorganismos, también se conocen como fango biológico.

4. Decantación secundaria: En este proceso o etapa el agua entra al decantador secundario, donde será separado el fango biológico.



Ilustración 7: Decantador Secundario.

El fango se deposita en el fondo mediante decantación, que mediante unas rasquetas, serán dirigidas al centro del decantador, para su posterior extracción mediante unas bombas.

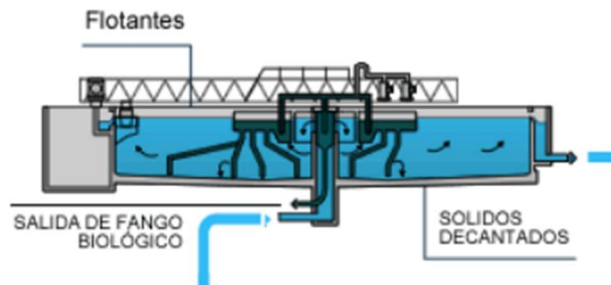


Ilustración 8: Salida de Fangos y de agua del decantador secundario.

En este caso el agua ya ha superado la última etapa, es decir el ciclo de tratamiento del agua ha finalizado, una vez superada esta etapa, el agua es vertida al río o mar con la garantía de calidad exigida.

4.1.2. Residuos generados por una EDAR

Según el Instituto Nacional de Estadística (INE), los procesos de tratamiento de aguas generan anualmente alrededor de 1.200.000 toneladas de materia seca al año.

Estos lodos destacan por ser un residuo extremadamente líquido y es que el 95% de su composición es agua.

A pesar de ello, su composición varía según una serie de factores, como puede ser la ubicación, la carga de contaminación del agua residual inicial, así como los tratamientos llevados a cabo en las EDAR.[4]

¿Que son los fangos?

Los lodos o fangos son una mezcla de sólidos separados del agua residual con un porcentaje de agua muy elevado, que surge del proceso de purificación de aguas residuales.

Dependiendo del tratamiento puede ser un residuo o un recurso. Debido a que su principal inconveniente es la posibilidad de que contenga una serie de compuestos tóxicos, que dependiendo de su concentración puede ser un aspecto a tener en cuenta a la hora de aplicarlo en agricultura, para evitar posibles intoxicaciones.

¿Dónde se generan?

Como bien se ha comentado anteriormente, los lodos surgen como residuos durante el proceso de purificación del agua.

En concreto se obtienen o se separan del agua, mediante la decantación que se produce tanto en el decantador primario como en el secundario, como se puede observar en la siguiente imagen.

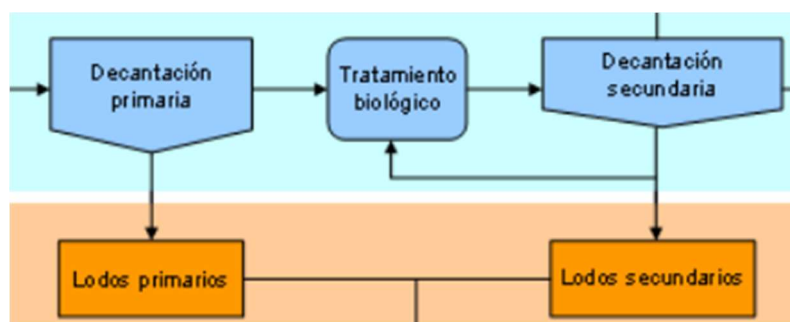


Ilustración 9: Separación de los lodos mediante decantación.

En estos decantadores, mediante unas rasquetas mueven hacia el centro los lodos depositados en el fondo, para que posteriormente puedan ser extraídos mediante unas bombas.

¿Qué etapas siguen los fangos?

Una vez conocido que son los fangos y de donde se obtienen se debe conocer cuál es su proceso o ciclo. Una vez extraído el fango de los decantadores, este debe dirigirse al espesador, donde se almacena temporalmente el fango extraído.[5]

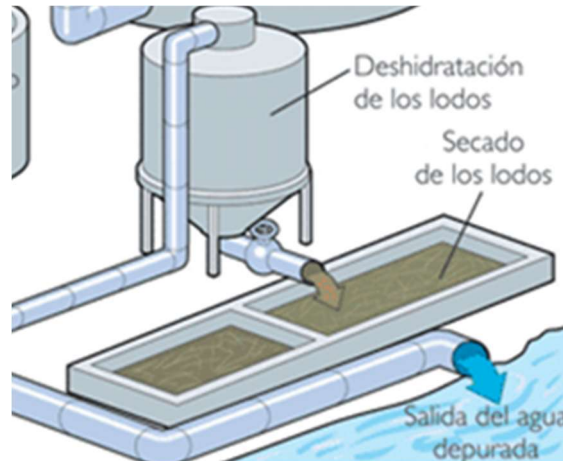


Ilustración 10: Almacenamiento de los fangos.

En los espesadores, conseguimos separar gran parte del agua que conforma el fango, que como bien hemos comentado anteriormente un 95% está formado por agua. Sin embargo, en los espesadores conseguimos separar gran parte de esta agua, que por reposo los lodos se dirigen al fondo y el agua se queda en la zona superior.

Esta agua se redirigirá a los decantadores, para que pueda ser tratada de nuevo, mientras que los fangos se enviarán al almacén donde posteriormente serán extraídos por las compañías de recogida de fangos.

5. DESARROLLO

5.1. GESTIÓN Y EXTRACCIÓN DE FANGOS DE LAS EDAR

5.1.1. *Gestión de los fangos*

Las plantas de tratamiento de aguas residuales (EDAR) son generadoras de lodos, por lo que los generadores de residuos deben velar por su correcta gestión, la cual pueden realizar directamente o encomendar a gestores autorizados, todo ello en cumplimiento de lo dispuesto en la Ley 7/2022, de 8 de abril, de residuos y suelos contaminados.

La dirección de su gestión debe respetar los principios de la política de residuos relacionados con la protección del medio ambiente y la salud humana, y aplicar una jerarquía en los programas de gestión que prioricen la prevención sobre el reciclaje, otros tipos de reciclaje, incluida la recuperación de energía. Los sedimentos de los vertederos son los últimos. Actualmente, sus directrices de gestión están incorporadas al Plan Nacional Integrado de Residuos (PNIR).

Se considera como gestión de fangos al conjunto de operaciones que se realizan con ellos desde que se generan en las EDAR, hasta la última fase en su tratamiento. Se pueden diferenciar cuatro etapas:

- Almacenamiento.
- Recogida.
- Transporte.
- Tratamiento.

Todas las EDAR que se van a tratar en este proyecto, cuentan con un sistema de almacenamiento. Cuya función principal del sistema de almacenamiento es la de extraer la mayor cantidad de agua posible de los fangos, para reducir el volumen de estos fangos, y así reducir costes de envíos.

La parte de recogida, transporte y tratamiento pasará a formar parte de la empresa contratada para el tratamiento de fangos o lodos, a los que se les encargará para realizar una recogida periódica, de estos fangos, entorno a una vez por semana.

5.1.2. Necesidad de la gestión de fangos

La extracción de fangos de las EDAR, son un proceso necesario en el ciclo de tratamiento de agua de las depuradoras.

Los servicios de retirada se realizarán en las EDAR que vamos a tomar como referencia para este proyecto: Barasoain, Pueyo, San Martín, Ujue, Olite, Pitillas, Santacara, Carcastillo, Figarol, Rada.



Ilustración 11: Mapa de Navarra con las EDAR.

Como bien se ha comentado anteriormente todas las operaciones con los fangos ya sean la carga, transporte y descarga en vertedero será por cuenta del contratista. Además, por contrato los camiones, salvo por ocasiones excepcionales, deberán efectuar la carga o extracción dentro de los horarios de presencia de personal.

5.1.3. Recogida de Fangos de las EDAR

Cada servicio comprenderá las siguientes operaciones y trabajos:

- Carga o extracción de los fangos dependiendo del dispositivo de almacenamiento que exista en cada depuradora.
- Transporte de estos mismos fangos, respetando las condiciones de higiene, seguridad, llevando un estricto cumplimiento de la normativa para transportes por carretera.
- Descarga en vertederos autorizados u otros lugares apropiados para almacenar este tipo de residuos.
- Gestión de tipo administrativo.

5.1.4. Estudio de producción de fangos de las EDAR

Antes de comenzar a estudiar el mercado, deberemos conocer la cantidad aproximada de fango que se produce en cada depuradora a la semana.

EDAR	Litros de fango a la semana
Barasoain	3500 litros
Pueyo	700 litros
San Martin	350 litros
Ujue	250 litros
Olite	6500 litros
Pitillas	3000 litros
Santacara	2500 litros
Carcastillo	2000 litros
Figarol	550 litros
Rada	650 litros
TOTAL	20000 litros

Tabla 1: Litros de fango a la semana en cada EDAR.

Cabe destacar que la anterior tabla, muestra unos valores medios recogidos de un cálculo aproximado a partir del número de días que tardan en llenarse los almacenes y teniendo en cuenta el número de litros que entran en el almacén, de esta forma conseguimos estos resultados.

Cabe destacar que son valores aproximados, ya que los fangos que produce una planta se ven afectados según el clima y la época del año.

Cabe destacar que los vehículos utilizados para este tipo de tareas son camiones cisterna con una capacidad de unos 26000 litros.



Ilustración 12: Camiones de extracción de fangos.

Y según las previsiones de producción de fangos, obtenemos una media semanal de 20000 litros, lo que nos indica que claramente la extracción de estos fangos deberá realizarse semanalmente, para evitar posibles imprevistos.

Actualmente no existe una periodicidad en la extracción de fangos, es decir cuando los operarios observan un nivel elevado de estos fangos, llaman con urgencia a la compañía que esté disponible en ese día para realizar la extracción de las depuradoras. Lo que hace que siempre se contrate a la compañía con precios más elevados ya que es la que mayor disponibilidad tiene.

De esta forma no se puede asegurar tener un control de este proceso de extracción de fangos de las EDAR, lo que supone que en ocasiones la cantidad de fangos de las EDAR, superen la cantidad de litros que pueden transportar en estos camiones.

Esto supone un gasto enorme, ya que la empresa contratada para la extracción deberá volver a extraer los fangos restantes que no han podido sacar en el primer viaje.

A esto se le debe añadir el hecho de que, con este sistema de extracción, hay depuradoras que almacenan los fangos durante mas de dos años, lo que hace que se taponen los conductos de extracción, generando averías costosas y muy aparatosas.

5.1.5. Estudio de mercado

Para este tipo de labores, podemos encontrar dos compañías principales en el área de Navarra. Ambas compañías trabajan de forma similar, en cuanto a profesionalidad a la hora de realizar las recogidas.

- **Compañía A:** es una compañía ubicada en Pamplona, cuyo presupuesto es el siguiente.

170€ por cada 1000 litros a transportar

Lo que supone un total anual:

$20.000\text{l/sem} * 4 \text{ semanas al mes} * 12 \text{ meses} = 960.000\text{l/año}$

Por lo que supondría un coste anual de:

$960 \text{ mil litros} * 170 \text{ €} = 163.000 \text{ €/anuales}$

Mayor rapidez de reacción a la hora de llamarles para una extracción.

- **Compañía B:** Es una compañía ubicada en Mutilva, cuyo presupuesto es el siguiente.

150€ por cada 1000 litros a transportar

Lo que supone un total anual:

$20.000\text{l/sem} * 4 \text{ semanas al mes} * 12 \text{ meses} = 960.000\text{l/año}$

Por lo que supondría un coste anual de:

$960 \text{ mil litros} * 150 \text{ €} = 144.000 \text{ €/anuales}$

Necesitan más tiempo de reacción desde que les llamas hasta que realizan la extracción. Es decir, necesitan tener una mayor planificación.

Además de este coste actualmente se debe añadir el coste por inutilizar a los operarios encargados de mantener las plantas en condiciones óptimas. Ya que cada vez que se contrata a la compañía de extracción de fangos, ellos deben acompañarlos para abrir la puerta de las EDAR.

Sin embargo, si anualmente se contrata una compañía que será la encargada de realizar todas las extracciones anuales que se necesiten, podremos facilitar de forma independiente el acceso a las EDAR.

Debido a esta anterioridad con la que se pactará por contrato que compañía se encargará de realizar las extracciones, se llegará a un

acuerdo para realizar las extracciones con la compañía B, que en este caso se trata de la más barata de las dos.

Ya que ambas cuentan con los mismos servicios y lo único que se diferencian entre ellas es la planificación, que en este caso no supone ningún inconveniente ya que se planificara las extracciones en el contrato.

5.1.6. Recopilación de datos

- **Medidas de los depósitos:** los depósitos de todas las EDAR, en este caso cuentan con las mismas medidas

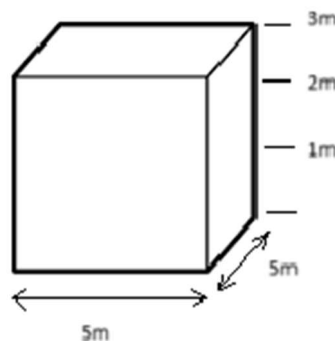


Ilustración 13: Medidas de los almacenes.

Un aspecto a tener en cuenta con estos almacenes, es el hecho de que cuando entra la mezcla de agua con fangos, los fangos se depositan en el fondo por decantación, y el agua se mantiene arriba de estos fangos.

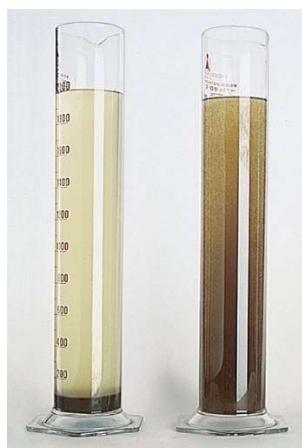


Ilustración 14: Decantación de fangos.

Para evitar que estos fangos se solidifiquen en el interior y produzcan atascos, siempre se deja un metro de altura en su interior.

- **Cuando se recogen los fangos:** Actualmente los fangos se recogen cuando el nivel de los almacenes supera o alcanza los dos metros de altura. Es entonces cuando se llama al camión para que extraiga 25.000 litros que equivalen a un metro de altura.

El camión extrae 1 metro de altura del fondo, es decir extrae los fangos y deja 1 metro de agua en el almacén.

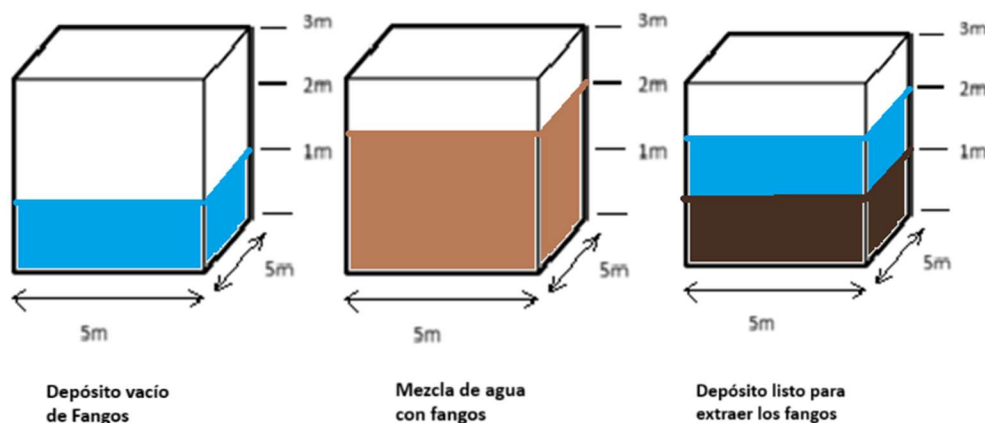


Ilustración 15: Etapas de fangos en el Almacén.

Además de ello cada vez que se realiza una extracción, se limpian los canales o tuberías por las que pasan los fangos, para evitar atascos, ya que cuando los fangos se secan, obstruye todos los circuitos.

- **Frecuencia de extracción de fangos:** Tras recopilar información con el coordinador de las EDAR, se han obtenido los siguientes datos

		Cuando el almacen supera los 2 m llaman al camión de 25000 litros	Semanas al año
		25000	48
	Litros semana	Cuantas semanas pasan hasta que el camion va a extraer los fangos en cada ciudad	Cuantas veces pasa el camion al año por cada EDAR
Olite	6500	3,846153846	12,48
Barasoain	3500	7,142857143	6,72
Pitillas	3000	8,333333333	5,76
Santacara	2500	10	4,8
Carcastillo	2000	12,5	3,84
Pueyo	700	35,71428571	1,344
Rada	650	38,46153846	1,248
Figarol	550	45,45454545	1,056
San martin	350	71,42857143	0,672
Ujue	250	100	0,48
TOTAL	20000		38,4

Tabla 2: Frecuencia de extracción de fangos anuales.

Como se puede observar, actualmente se realizan 38,4 recogidas anuales, en las EDAR que se va a tratar en este proyecto.

Tras recopilar estos datos, se puede observar que el número actual de recogidas anuales es de 38.4, mientras que la propuesta de este proyecto sería realizar 48 recogidas anuales.

A primera vista puede parecer un gasto extra innecesario, sin embargo, por parte del coordinador de estas depuradoras ha observado que el sistema actual necesita un cambio.

Ya que su principal problema es que en depuradoras como la de San Martín y Ujue, se extrae fango casi cada dos años, lo que provoca enormes problemas de taponamientos en el circuito de transporte de fangos ya que, estos fangos se espesan y se solidifican provocando numerosas averías.



Ilustración 16: Fangos solidificados.

Por ello, se ha llegado a la conclusión de que el sistema más apropiado sería el simple hecho de realizar semanalmente una recogida y una limpieza a todas las EDAR. Es decir, un día a la semana recoger los fangos producidos en todas las EDAR esa semana.

Que, aunque pueda parecer a primera vista que supone un mayor coste, a la larga produce menos averías y menos costes.

5.1.7. Comparación de sistemas

A modo de breve resumen, a continuación, se va a realizar un pequeño estudio que permitirá entender de manera más clara cuales son los inconvenientes del sistema o formato de extracción de fangos que se puede encontrar actualmente.

5.1.7.1. Sistema actual de recogida de fangos

Actualmente en estas depuradoras se emplea un sistema de recogida por nivel.

Es decir, cuando el almacén de fangos supera los 2 metros de altura, se llama de urgencia a las dos únicas compañías aptas para realizar este trabajo.

Cuando se extrae los fangos del almacén, se extraen unos 25000 litros, que supone un metro de altura del almacén, dejando siempre otro metro en el interior de este para que no se produzcan atascos en la zona recuadrada en rojo.



Ilustración 17: Zona decantación de fangos en el Almacén.

En ese momento, la depuradora se encuentra a merced de las compañías de extracción de fango, que serán las que decidan si pueden acudir en ese momento a la recogida o no. Ambas compañías tienen una diferencia de precio notable de hasta 20€ por cada 1000 litros.

El inconveniente de este sistema se centra, en el aspecto de depender de estas compañías, además de que siempre se termina contratando a la compañía con precios más elevados, porque suele tener mayor disponibilidad.

Otro de los principales problemas, que se puede encontrar con esta metodología es el simple hecho de que algunas depuradoras tardan más de dos años en que se alcancen los dos metros de altura. Que a primera vista puede parecer una ventaja, ya que significa menos veces que hay que contratar a la compañía de extracción.

Sin embargo, es todo lo contrario. Ya que al realizar una extracción cada dos años o más, habitualmente se generan atascos en la zona crítica de extracción.



Ilustración 18: Zonas críticas de atascos.

Debido a las propiedades de estos fangos, cuando se extraen de los decantadores, cuentan con un elevado porcentaje de agua. Pero cuando se secan comienza su proceso de solidificación y compactación de estos fangos.



Ilustración 19: Solidificación de fangos.

Generalmente cuando se van a extraer fangos de las depuradoras más pequeñas, simplemente con pasar una manguera con un "ratón" por las tuberías con fangos, se consigue reblandecer. Como se puede observar en las siguientes imágenes.



Ilustración 20: Funcionamiento de un ratón de desatascos.

Como bien se ha comentado anteriormente, aunque pueda parecer a primera vista más rentable económicamente hablando el hecho de recoger los fangos cada vez que se superan los dos metros.

A estos costes hay que añadirle el hecho de que, en las EDAR de Rada, Figarol, Sanmartín y Ujue. Entorno a una vez cada 3 años se produce un atasco que no se puede solucionar mediante el uso del ratón.

Lo que supone una avería de gran calibre, ya que se necesitan de 3 o 4 operarios durante una semana para conseguir abrir el cemento hasta llegar a la tubería.



Ilustración 21: Extracción de tubería atascada.

Para posteriormente extraerla y sustituirla por otra nueva, con los gastos de materiales, maquinaria y personal que eso supone.



Ilustración 22: Atasco de fangos.

5.1.7.2. Recogida de fangos semanal

Este nuevo sistema que se pretende implantar es muy sencillo. Se centra en la idea de realizar semanalmente una recogida del fango producido esa semana por todas las plantas.

Consiguiendo así reducir considerablemente las averías producidas por la obstrucción de fangos en las tuberías, con los gastos que conllevan.

Como bien se ha comentado en apartados anteriores. Actualmente se realizan 38,4 recogidas anuales. Con este sistema se realizarían 48 al año.

Sin embargo, como se realizarán semanalmente, podremos llegar a un acuerdo con la compañía más barata para que un determinado día a la semana reserve un camión para estas depuradoras. Pactando de antemano un precio.

Esta compañía tendrá acceso a todas las EDAR, sin necesidad de llamar a ningún operario de las plantas.

Diferencia de precios en recogida de fangos:

- Sistema actual: Se contrata a una compañía de recogida de fangos que cobra

170€ por cada 1000 litros.

Cada recogida son 25000 litros

$25 \text{ mil litros} \times 170\text{€} \times 38,4 \text{ semanas} = 163200\text{€ al año}$

- Sistema semanal: Se contrataría a la compañía de recogida más barata.

150€ por cada 1000 litros

Cada recogida semanal será de 20000 litros

$20 \text{ mil litros} \times 150\text{€} \times 48 \text{ semanas} = 144000\text{€ al año}$

Como se puede observar el ahorro anual por las recogidas supone una suma de dinero muy considerable.

A este ahorro se le deberán añadir otros costes que suponen las averías como pueden ser:

- Frecuencia de averías anuales. Debido al corto periodo de tiempo que el coordinador de las EDAR lleva en su puesto, No dispone de un registro exacto y detallado del número de averías que se han producido por atascamiento durante los últimos años.

Sin embargo, por parte de sus superiores remarcan que suelen tener este tipo de averías una o dos veces al año.

- Metodología: Una vez se detecta la obstrucción del conducto, se intenta reblandecer el fango solidificado, introduciendo un "ratón", cuya finalidad es expulsar agua a una elevada presión, haciendo que esta pieza se mueva en el interior del conducto.

Generalmente, con esta metodología se suelen solucionar. Pero en caso de que no se consiga solventar, se comienza con el proceso de extracción del conducto.

Para ello se necesita una excavadora que disponga de martillo percutor para agrietar el cemento del suelo y así acceder a la tubería.

Una vez retirado todo el material que se encuentra sobre la tubería, se corta la sección de tubería obstruida, para acoplar otra nueva.

Posteriormente se cubre con tierra y se cubre la parte superior con cemento.

- Materiales: Alquiler de una excavadora con martillo percutor, la nueva sección de tubería, con sus empalmes y un camión de cemento, ya que en ocasiones la superficie extraída puede llegar a ser de más de cinco metros.
- El costo de mano de obra es un aspecto a tener en cuenta ya que se necesita de tres o cuatro personas trabajando constantemente durante una semana, para llegar a solucionar el problema.
- Duración de estas averías: Tratar este tipo de averías, requiere de al menos una semana completa de trabajo, dejando inutilizada la planta y a los operarios de mantenimiento.

A modo de conclusión de este punto, se debe reconocer que mediante la recogida de fangos semanal. Se ahorran 19200€ al año únicamente en transporte y a ello hay que añadirle el costo por materiales y por la mano de obra.

5.2. PROBLEMA DEL VIAJANTE Y ALGORITMOS GENETICOS

El problema del viajante (TSP) es un desafío que consiste en hallar la ruta más breve que conecte distintos puntos, visitándolos una sola vez y regresando al punto de partida. Es considerado uno de los problemas más notables en la disciplina de optimización combinatoria computacional.

En el contexto de este proyecto, la meta es determinar la ruta más eficiente entre los diversos puntos de recolección de fangos en Navarra, asegurándose de pasar por cada punto exactamente una vez y regresando al punto de inicio.

5.2.1. Definición teórica

Se expondrá a continuación una clásica formulación del TSP, teniendo en cuenta para ello una serie de conceptos como los siguientes:

- Un Grafo $G = (N, A)$ se trata de un conjunto de N elementos también llamados vértices o nodos y un conjunto A cuyos elementos se conocen como arcos o aristas. Cada uno de estos arcos o aristas vienen a representar una conexión directa entre dos nodos o elementos de N .
- Un grafo dirigido u orientado, se conoce a aquel cuyos arcos son pares ordenados; el arco (i, j) empieza en el nodo i y termina en el nodo j .
- Un grafo no dirigido o no orientado, es un grafo donde (i, j) y (j, i) vienen a representar el mismo arco.
- Un grafo completo es aquel grafo (orientado o no orientado) en el que están presentes todas las aristas posibles.
- Un camino está formado por una secuencia de aristas en las que todos los vértices son distintos.
- Un circuito, es una secuencia de aristas donde el primer vértice y el último son el mismo, además aparte de estos dos vértices no hay más vértices coincidentes.
- Un camino hamiltoniano, es un camino donde se recorren una vez todos los nodos del grafo.
- Un circuito hamiltoniano, es un circuito que recorre una vez todos los nodos del grafo.

Una vez se conocen estas definiciones, se puede describir de la siguiente manera el problema del TSP según la teoría de grafos.

Sea $G = (N, A)$ un grafo completo, en donde $N = 1, \dots, n$ es el conjunto de vértices o nodos y A es el conjunto de arcos. Los nodos $i = 2, \dots, n$ se corresponden con los clientes a visitar mientras que el nodo 1 se corresponde con la ciudad de origen y destino.

A cada arco (i, j) es asociado un valor no negativo d_{ij} , que viene a representar la distancia que existe del vértice i al j .

En este caso la utilización de los arcos (i, i) no está permitido, por lo que se impone que $d_{ii} = 1 \forall i \in N^2$ [6]

5.2.2. Tipos de TSPs

A la hora de estudiar el TSP, se pueden diferenciar cuatro tipos principales de TSPs:

- Simétrico: este tipo se caracteriza por tener siempre la misma distancia entre un par de puntos, independientemente del sentido. En este tipo de TSP, cuenta con una matriz de distancias simétricas.
- Asimétrico: este tipo de TSP se caracteriza por no tener que ser siempre las mismas distancias entre dos puntos, dependiendo del sentido que se tome, por ello la matriz de distancias en este caso no es simétrica. Este es el tipo de TSP que se ha utilizado para realizar este Proyecto.
- Euclidiano: este tipo de TSP se caracteriza por tener distancias euclidianas entre dos puntos.

La distancia euclidiana sirve para indicar la separación que hay entre dos puntos en un espacio donde se cumplen los teoremas de Euclides. La distancia entre dos puntos de un espacio euclidiano es la longitud del vector de esos dos puntos que pertenece a la única recta que pasa por esos dos puntos.

- Desigualdad triangular: este tipo de TSP se diferencia por destacar que la distancia desde A hasta B no sea mayor que la distancia de A a B pasando por el punto C.

5.2.3. Algoritmos más utilizados

Dentro de los algoritmos, podemos diferenciar claramente dos que son los más utilizados para el problema del viajante.

- Algoritmo del vecino más cercano: comenzando desde cualquier punto, seleccione el punto más cercano entre los puntos que no han sido visitados hasta que se atraviesen todos los puntos existentes y regrese al punto de partida.
- Algoritmos de inserción: estos algoritmos consisten en construir ciclos que primero visitan ciertos puntos y luego extienden estos bucles insertando puntos fuera del bucle. En cada paso se incorpora un nuevo punto al bucle hasta obtener el bucle óptimo.

Existen cuatro criterios destacables dentro de estos algoritmos.[7]

- Inserción más cercana: Seleccionar el vértice j más cercano al ciclo.
- Inserción más lejana: Seleccionar el vértice j más lejano al ciclo.
- Inserción más barata: Seleccionar el vértice j con el menor incremento del coste.
- Inserción aleatoria: Seleccionar el vértice j al azar.

5.2.4. Aplicaciones

El problema del viajante tiene muchas aplicaciones diferentes en el mundo en el que vivimos. La principal aplicación es la planificación de rutas logísticas, ya que este problema genera soluciones óptimas de rutas, en lo que se basa la primera aplicación mencionada.

Además, se utiliza en otros ámbitos muy diferentes, como la fabricación de placas de circuitos, en este caso el foco está en conseguir la mejor secuencia de perforación de agujeros en la placa y en conseguir las mejores vías de comunicación a través del cableado entre chips.

Finalmente encontramos una aplicación en un servidor de red, porque al final del día Internet es una plataforma de distribución y queremos aprovecharla al máximo.

Estas son las aplicaciones más importantes y destacables de este problema, pero hay muchas otras, como la secuenciación de ADN o el análisis de datos estadísticos.

5.2.5. Algoritmos genéticos

Los algoritmos genéticos se incluyen en las siguientes categorías de algoritmos evolutivos en el campo de las metaheurística. Este tipo los algoritmos son el resultado de simular las características evolutivas de los organismos.

Específicamente, estos algoritmos pretenden imitar la evolución de las especies, es decir basarse en el fenómeno de la selección natural mediante la herencia genética.

En numerosas ocasiones se han tomado como referencia la naturaleza para la resolución de problemas. El simple hecho de tener que estar constantemente adaptándose hacen que las especies se garanticen la supervivencia buscando constantemente la mejora.

Estos algoritmos, se modelan de manera semejante al del proceso de reproducción de los individuos, teniendo en cuenta sobre todo las características heredadas por parte de la progenie. Además de ello se debe tener en cuenta las variaciones que existen debido al medio donde se encuentran los miembros.

5.2.5.1. Procedimiento de un algoritmo genético

Para comenzar, se genera una población inicial de soluciones, haciendo alusión a los individuos. Posteriormente, se continúa realizando una evaluación de la población generada y se le asigna a cada solución su correspondiente valor de adecuación.

Tras realizar la evaluación, se realiza un proceso de selección según el valor de adecuación de cada uno de los individuos, donde a más valor, más veces se copiará.

La selección, asegura que los individuos que cuenten con mayor adecuación tengan más participación en el proceso.

Posteriormente comienza la fase de cruce, donde los individuos se emparejan de manera aleatoria, donde las parejas tienen probabilidad de reproducirse.

Al crearse nuevas soluciones, estas tienen la posibilidad de la estructura de las soluciones pueden alterarse incorporando nuevas características de población

Por ultimo se realiza una evaluación para comprobar si cumple con los objetivos del problema descrito, y si las condiciones no resultan satisfactorias, se repite el procedimiento hasta que la condición se cumpla.

5.3. OBTENCIÓN DE LA SOLUCIÓN.

En este apartado se pretende realizar el desarrollo y explicación de forma detallada del trabajo realizado con sus respectivos pasos para llegar a la solución del problema descrito y tratado anteriormente en este proyecto.[8]

5.3.1. Aplicación Shiny/R

Para comenzar con la explicación de las partes que conforman este programa, se realizara una breve introducción sobre la aplicación Shiny del software de R.

Esta aplicación, será la que utilizemos para desarrollar las rutas optimas de cada recogida de fangos.

Mediante el uso del paquete Shiny perteneciente al software de R se generará el código necesario para crear una aplicación web en la que aparezca representada la solución al problema definido.

Esta aplicación Shiny, está constituida por dos tipos de archivos:

- Server. R en este apartado, se genera el contenido que viene influenciado por las interacciones que realiza el usuario en la pantalla. Además, se define la lógica del servidor.
- Ui. R alberga la parte donde se encuentran definidos los elementos que componen la interfaz, así como la disposición de estos mismos en la propia pantalla.

Además de ello, se encuentra el código necesario para mostrar la información en la pantalla, así como los parámetros que la hacen interactiva.

Con estos dos archivos, resultará más que suficiente para el desarrollo de la aplicación en la que se nos mostrará la solución del problema.

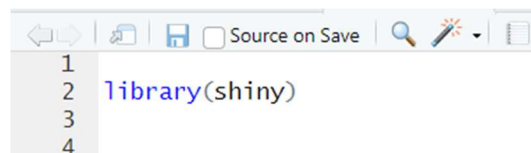
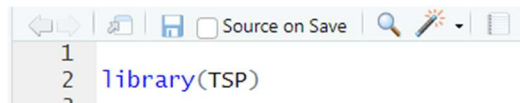


Ilustración 23: Librería Shiny.

Librería TSP

Mediante el uso de esta librería, se van a aportar todo lo necesario en cuanto a infraestructuras y algoritmos para el problema del viajante, es decir.

Con esta librería vamos a conseguir obtener la solución de las rutas optimas de este problema.

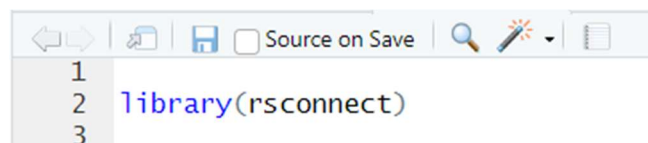


```
1  
2 library(TSP)  
3
```

Ilustración 24: Librería TSP.

Librería RShiny

Instalando la librería RShiny, se obtiene un paquete que nos permite desplegar las aplicaciones de Shiny.

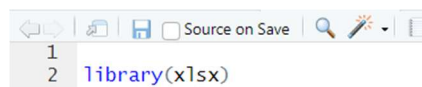


```
1  
2 library(rsconnect)  
3
```

Ilustración 25: Librería RShiny.

Librería xlsx

Este paquete permite importar archivos en formato Excel. Con esta librería podemos leer y escribir este formato de archivos.

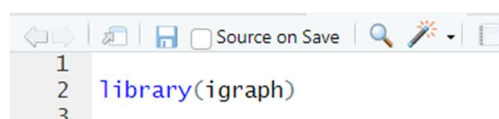


```
1  
2 library(xlsx)  
3
```

Ilustración 26: Librería xlsx.

Librería igraph

El paquete igraph, es un tipo de librería que permite proporcionar rutinas y funciones que permiten generar y manipular grafos fácilmente.



```
1  
2 library(igraph)  
3
```

Ilustración 27: Librería igraph.

5.3.2. Archivos del Programa.R

En este apartado se va a explicar cómo se ha construido y desarrollado el archivo del programa completo, así como de la aplicación.

Para realizar este programa, se debe definir el directorio en el que se van a encontrar tanto los documentos Excel como los archivos del programa. En este caso será el siguiente.

```
setwd("C:/Guillermo Aldunate/Estudios/2. Universidad/5º Mecatrónica e IOI/  
Semestre 2/TFG/TFG/Programa TFG/Programa Guillermo")
```

Ilustración 28: Directorio del proyecto.

Primeramente, se debe realizar un estudio con las coordenadas de cada punto de recogida de fangos. Con ello, conseguiremos conocer cuales son los puntos o ubicaciones exactas de cada EDAR. Incluyendo la ubicación de Iraola, que será desde donde comenzará la ruta el camión.

IRAOLA	42°47'16"N	1°37'33"W
Barasoain	42° 36'03"N	1°38'58"W
pueyo	42°33'49"N	1°33'39"W
San Martin	42°31'24"N	1°34'07"W
Ujue	42°30'12"N	1°30'02"W
Olite	42°28'57"N	1°38'28"W
Pitillas	42°24'52"N	1°37'46"W
Santacara	42°21'57"N	1°33'03"W
Carcastillo	42°22'57"N	1°27'26"W
Rada	42°19'25"N	1°34'50"W
Figarol	42°20'54"N	1°22'33"W

Ilustración 29: Coordenadas GPS de las EDAR.

Deposito	LATITUD	LONGITUD
1 IRAOLA	42.78778	-1.625833
2 BARASOAIN	42.60083	-1.649444
3 PUEYO	42.56361	-1.560833
4 SAN MARTIN DE UNX	42.52333	-1.568611
5 UJUE	42.50333	-1.500556
6 OLITE	42.48250	-1.641111
7 PITILLAS	42.41444	-1.629444
8 SANTACARA	42.36583	-1.550833
9 CARCASTILLO	42.38250	-1.457222
10 RADA	42.32361	-1.580556
11 FIGAROL	42.34833	-1.375833

Ilustración 30: Lectura de Coordenadas en R.

Estas coordenadas, se encontrarán escritas en un Excel. Que conociendo la ruta del directorio de ese archivo podremos leer su interior mediante el siguiente comando.

```
Coordenadas<-read_excel("C:/Guillermo Aldunate/Estudios/2. Universidad/5º Mecatrónica e IOI/  
Semestre 2/TFG/TFG/Programa TFG/Programa Guillermo  
/CoordenadasGuillermo.xlsx",col_names = TRUE,sheet=1)
```

Ilustración 31: lectura del Excel de coordenadas.

Una vez tenemos las coordenadas en formato xlsx, y conocemos su directorio y la forma de leerlos mediante el software de R.

El siguiente paso consiste en realizar la Matriz de distancias. Para ello utilizaremos el siguiente código.

```
MatrizDistancias<-geodist(Coordenadas)  
Coordenadas<-as.data.frame(Coordenadas)  
row.names(Coordenadas)<-Coordenadas[,1]  
Coordenadas<-Coordenadas[,c(2,3)]  
names(Coordenadas)<-c("LATITUD", "LONGITUD")  
Distancias<-as.matrix(MatrizDistancias)  
rownames(Distancias)<-row.names(Coordenadas)  
colnames(Distancias)<-rownames(Distancias)  
Coordenadas<-as.matrix(Coordenadas)  
coordenadas<-Coordenadas  
ID1=rownames(Coordenadas)<-rownames(Distancias)
```

Ilustración 32: Obtención de Matriz de distancias.

Como bien se ha comentado anteriormente, necesitaremos de la matriz de distancias, que nos indica las distancias que existen entre todos los puntos respectivamente.

```
> MatrizDistancias  
      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]      [,8]      [,9]     [,10]     [,11]  
[1,]  0.00 20865.412 25474.239 29759.486 33233.149 33949.466 41490.693 47292.173 47107.30 51717.208 52951.62  
[2,] 20865.41 0.000 8350.532 10861.320 16308.457 13168.410 20778.552 27335.411 28919.88 31320.410 35904.31  
[3,] 25474.24 8350.532 0.000 4521.248 8320.093 11155.373 17503.008 21994.869 21841.59 26720.727 28315.00  
[4,] 29759.49 10861.320 4521.248 0.000 5998.223 7471.118 13085.792 17563.837 18113.80 22217.244 25045.67  
[5,] 33233.15 16308.457 8320.093 5998.223 0.000 11736.895 14453.801 15825.381 13889.21 21019.342 20024.18  
[6,] 33949.47 13168.410 11155.373 7471.118 11736.895 0.000 7623.283 14923.889 18711.44 18340.444 26342.35  
[7,] 41490.69 20778.552 17503.008 13085.792 14453.801 7623.283 0.000 8402.275 14538.68 10858.998 22023.00  
[8,] 47292.17 27335.411 21994.869 17563.837 15825.381 14923.889 8402.275 0.000 7883.93 5285.609 14457.40  
[9,] 47107.30 28919.884 21841.593 18113.797 13889.212 18711.439 14538.684 7883.930 0.00 12031.956 7668.73  
[10,] 51717.21 31320.410 26720.727 22217.244 21019.342 18340.444 10858.998 5285.609 12031.96 0.000 16982.85  
[11,] 52951.62 35904.311 28314.997 25045.665 20024.182 26342.351 22022.997 14457.395 7668.73 16982.846 0.00
```

Ilustración 33: Matriz de distancias.

A continuación, el siguiente paso a realizar consiste en desarrollar la parte de visualización de la aplicación. Es decir, se deben insertar los nombres de los pueblos donde se encuentran las EDAR que se van a tener en cuenta para la extracción de fangos de este proyecto.


```
ui<-fluidPage(  #definir todo lo que aparece en la web
  titlePanel("RUTA DE RECOGIDA DE FANGOS EN LAS EDAR DE NAVARRA"),
  sidebarPanel(  #entradas
    tabsetPanel(
      tabPanel(title="Sector zona Media Navarra",
        checkboxGroupInput("variable1","Seleccione las EDAR de recogida", #definir los puntos
          choices=c("IRAOLA"="IRAOLA","BARASOAIN"="BARASOAIN","PUEYO"="PUEYO",
            "SAN MARTIN DE UNX"="SAN MARTIN DE UNX","UJUE"="UJUE","OLITE"="OLITE",
            "PITILLAS"="PITILLAS","SANTACARA"="SANTACARA","CARCASTILLO"="CARCASTILLO",
            "RADA"="RADA","FIGAROL"="FIGAROL"),
          selected=c("IRAOLA"="IRAOLA","BARASOAIN"="BARASOAIN","PUEYO"="PUEYO",
            "SAN MARTIN DE UNX"="SAN MARTIN DE UNX","UJUE"="UJUE","OLITE"="OLITE",
            "PITILLAS"="PITILLAS","SANTACARA"="SANTACARA","CARCASTILLO"="CARCASTILLO",
            "RADA"="RADA","FIGAROL"="FIGAROL"),
          inline=TRUE)),
```

Ilustración 34: Código para la visualización de la aplicación.

A continuación, se establecen los parámetros del algoritmo genético, seleccionando aspectos tales como la población inicial, el numero de generaciones, el parámetro local y el parámetro Elite.

Utilizando el tabpanel, conseguiremos generar diferentes pestañas dentro de la aplicación.

El código de los elementos de entrada se cierra con width que determina la anchura de entrada. El valor es 45 que se ajusta al ancho de una ventana HTML.

```
tabPanel(title="Parámetros algoritmo genético",
  numericInput("Npop","Seleccione la población inicial",value=6, min=2,max=100, step=1),
  numericInput("Ngen","Seleccione el número de generaciones",value=2, min=1,max=100, step=1),
  numericInput("Local","Seleccione el parámetro Local",value=0.05, min=0,max=1, step=0.05),
  numericInput("Elite","Seleccione el parámetro Elite",value=2, min=1,max=6, step=1))
),
width=45
```

Ilustración 35: width.

En esta parte de código, se establece el botón que nos permite calcular la ruta.

Además de que nos muestre en la app las distancias entre el algoritmo del vecino más cercano, el algoritmo del vecino más lejano y el algoritmo genético.

Y por ultimo que nos muestre el mapa.

```
submitButton("Calcular ruta"),
mainPanel(textOutput("Distancia"), #salida de app
  # verbatimTextOutput("values"),
  tableOutput("values"),#comparacion algoritmo tsp y genetico
  leafletOutput("mapa"))##coloca el mapa
)
```

Ilustración 36: Código de salida de app.

Para realizar el cálculo de la ruta más optima, se debe conocer primeramente el trabajo de la función `Solve_TSP` incluida en la librería TSP que nos permite ejecutar el problema del viajante.

`Solve_TSP (MatrizTSP, method, control)`

- `MatrizTSP`: representa la matriz de distancias entre los municipios a tener en cuenta y que se incluyen dentro de la ruta dada.
- `Method`: Es una variable que define el algoritmo que se va a emplear en la función para calcular la ruta.
- `Control`: permite tener un control de la forma en que se calculan las soluciones, además de utilizarse para imponer el nodo que debe aparecer como inicio de la ruta.

La solución que se va a obtener será un ciclo de carácter hamiltoniano donde se incluyan todos los nodos o pueblos de la ruta por los que pasará una única vez. La matriz de municipios que se incluye debe ser de clase TSP.

Los algoritmos impuestos en el presente proyecto son: "nearest_insertion", "farthest_insertion".

La elección de estos algoritmos se debe a que son los únicos algoritmos que permiten respetar la función de control. Ya que es importante en este caso, el hecho de respetar el punto de inicio de la ruta.

Finalmente, se obtiene la solución para ambos métodos (solución 1 y solución 2) que nos permitirá facilitarnos el escoger la ruta más optima comparando las longitudes de las rutas.

```
server<-function(input,output){
  slidervalues<-reactive({

    MatrizTrabajo<-Distancias[c("IRAOLA",c(input$variable1)),
                               c("IRAOLA",c(input$variable1))]

    if(length(c("IRAOLA",c(input$variable1)))<10){
      MatrizTrabajo[lower.tri(MatrizTrabajo)]=t(MatrizTrabajo)[lower.tri(MatrizTrabajo)]
      isSymmetric(MatrizTrabajo)
      MatrizTSP<-TSP(as.matrix(MatrizTrabajo), labels = NULL)
      initialtour<-as.integer(which(labels(MatrizTrabajo)[[1]]=="IRAOLA"))
      solucion1<-solve_TSP(MatrizTSP,method="farthest_insertion",control=list(start=initialtour))
      solucion2<-solve_TSP(MatrizTSP,method="nearest_insertion",control=list(start=initialtour))

      Longitud1<-tour_length(solucion1)
      Longitud2<-tour_length(solucion2)
    }
  })
}
```

Ilustración 37: Código de `Solve_TSP`.

Seguidamente se utilizan los siguientes comandos, que servirán a modo de indicador de la longitud, de ambas soluciones, además de mostrar el resultado de las distancias con el orden de la ruta a seguir.

```
Ruta<-list(c(paste("Longitud (km):",round(Longitud1,digits=2)),labels(solucion1)),
          c(paste("Longitud (km):",round(Longitud2,digits=2)),labels(solucion1)))
Longitudes<-c(Longitud1,Longitud2)
Seleccion<-which.min(Longitudes)
Ruta}
```

Ilustración 38: Código de representación de rutas.

En caso de que se seleccionen más de 10 municipios sobre los que realizar la ruta de recogida de fangos, se ejecutará esta parte del programa.

El único aspecto para tener en cuenta es que en este caso únicamente se disponen de 10 EDAR además de la ubicación inicial. Pero este proyecto se ha diseñado con visiones a futuro, en donde se podrían añadir nuevas depuradoras.

En este caso además de tener las soluciones de la ruta más larga y mas corta, se añade el algoritmo genético.

```
else {
  MatrizTrabajo[lower.tri(MatrizTrabajo)]=t(MatrizTrabajo)[lower.tri(MatrizTrabajo)]
  isSymmetric(MatrizTrabajo)
  MatrizAG<-MatrizTrabajo
  colnames(MatrizAG)<-NULL
  MatrizTSP<-TSP(as.matrix(MatrizTrabajo), labels = NULL)
  initialtour<-as.integer(which(labels(MatrizTrabajo)[[1]]=="IRAOLA"))
  solucion1<-solve_TSP((MatrizTSP),method="farthest_insertion",control=list(start=initialtour))
  solucion2<-solve_TSP((MatrizTSP),method="nearest_insertion",control=list(start=initialtour))
  solucion3<-search_tour_genetic(MatrizAG,dim(MatrizAG)[1],input$Npop,input$Ngen,input$Local,input$Elite)
  Longitud1<-tour_length(solucion1)
  Longitud2<-tour_length(solucion2)
  Longitud3<-solucion3$distance

  Ruta<-list(c(paste("Longitud (km):",round(Longitud1,digits=2)),labels(solucion1)),
            c(paste("Longitud (km):",round(Longitud2,digits=2)),labels(solucion1)),
            c(paste("Longitud (km):",round(Longitud3,digits=2)),colnames(MatrizTrabajo)[solucion3$tour]))
  Longitudes<-c(Longitud1,Longitud2,Longitud3)
  Seleccion<-which.min(Longitudes)
  Ruta
}
```

Ilustración 39: Solución de ruta con algoritmo genético.

Por último, en cuanto al código necesario para incluir el apartado de representación del mapa dentro de la aplicación, se puede observar que es similar al anterior apartado del código de MatrizTrabajo.

Se puede observar que en este apartado se cuenta también con dos soluciones, relacionadas con la ruta más larga y la ruta más corta en caso de disponer de menos de 10 localidades.

```
mapaselect<-reactive({
  if(length(c("IRAOLA",c(input$variable1)))<10){
    coordenadas<-coordenadas[c("IRAOLA",c(input$variable1))]
    coordenadas<-as.data.frame(coordenadas)
    MatrizTrabajo<-Distancias[c("IRAOLA",c(input$variable1)),
                               c("IRAOLA",c(input$variable1))]
    MatrizTrabajo[lower.tri(MatrizTrabajo)]=t(MatrizTrabajo)[lower.tri(MatrizTrabajo)]
    isSymmetric(MatrizTrabajo)

    MatrizTSP<-TSP(as.matrix(MatrizTrabajo), labels = NULL)
    initialtour<-as.integer(which(labels(MatrizTrabajo)[[1]]=="IRAOLA"))
    solucion1<-solve_TSP((MatrizTSP),method="farthest_insertion",control=list(start=initialtour))
    solucion2<-solve_TSP((MatrizTSP),method="nearest_insertion",control=list(start=initialtour))
    Longitud1<-tour_length(solucion1)
    Longitud2<-tour_length(solucion2)
  }
```

Ilustración 40: Código de representación del mapa en la app.

En esta parte del programa se encuentran los comandos, que se utilizan para la representación del trayecto en el mapa, en donde aparecen representados las ubicaciones de los municipios que deben visitarse.

Como se puede observar en las siguientes líneas de código, es necesario registrarse en Openrouteservice. De este modo, al registrarse te proporcionan un token que se incluye en la orden `ors_api_key`.

```
ors_api_key('5b3ce3597851110001cf62485fce866458364a43b3aa39e829eef59c')
```

En la creación y diseño del trayecto, se utiliza la función `ors_directions` que cuenta con varios parámetros. Primero se indican las coordenadas mediante la lista de longitudes y latitudes en orden. Seguido, se ha indicado el formato, que por defecto es "geojson", y se deja indicado el radio máximo de 10000 kilómetros entre municipios.

Antes de poder representar el trayecto en el mapa, se crea un nuevo indicador. En este caso, se pretende relacionar el nombre de los municipios según el orden que se desean visitar, según lo obtenido en la ruta optima.

```
ID1=rownames(coordenadasok)
```

Para finalizar, utilizando el paquete Leaflet se obtiene la representación de forma gráfica que se muestra tras el código. Haciendo uso de dos nuevos parámetros correspondientes a la función Leaflet:

-addGeoJSON (): acepta datos GeoJSON en formato analizado (listas anidadas) o en forma de cadena (vector de caracteres de elemento único).

-fitBBox (): es una función auxiliar que permite marcar los límites de la representación del mapa.

```
Ruta<-list(labels(solucion1),labels(solucion2))
Longitudes<-c(Longitud1,Longitud2)
Seleccion<-which.min(Longitudes)
Rutafinal<-Ruta[[Seleccion]]
coordenadas<-coordenadas[order(Rutafinal),]
coordenadas<-as.data.frame(coordenadas)
ruta<-data.frame(coordenadas$LONGITUD,coordenadas$LATITUD)
names(ruta)<-c("LONGITUD","LATITUD")
ID1=rownames(coordenadas)
ors_api_key('5b3ce3597851110001cf6248c461a41c55e84e7fac8efbda9e23fc4e')
route <- ors_directions(asplit(ruta,1), format="geojson",radiuses=1000000)
mapa <- leaflet(data = coordenadas) %>%
  addTiles()
  setView(lng = coordenadas$LONGITUD[1], lat = coordenadas$LATITUD[1] , zoom = 1)
  addMarkers(~LONGITUD, ~LATITUD, popup = ~as.character(ID1), label = ~as.character(ID1))
  addGeoJSON(route, fill=FALSE) %>%
  fitBBox(route$bbox)}
```

Ilustración 41: Código para la representación de la ruta en el mapa.

Como en el caso anterior de la MatrizTrabajo, en este apartado de código, también se añade una parte específica para los casos de seleccionar más de 10 municipios sobre los que realizar la ruta de recogida de fangos, se ejecutará esta parte del programa.

El único aspecto para tener en cuenta es que en este caso únicamente se disponen de 10 EDAR además de la ubicación inicial. Pero este proyecto se ha diseñado con visiones a futuro, en donde se podrían añadir nuevas depuradoras.

En este caso además de tener las soluciones de la ruta más larga y corta, se añade el algoritmo genético.

```
else{
  coordenadas<-coordenadas[c("IRAOLA",c(input$variable1)),]
  coordenadas<-as.data.frame(coordenadas)

  MatrizTrabajo<-Distancias[c("IRAOLA",c(input$variable1)),
                             c("IRAOLA",c(input$variable1))]
  MatrizTrabajo[lower.tri(MatrizTrabajo)]=t(MatrizTrabajo)[lower.tri(MatrizTrabajo)]
  isSymmetric(MatrizTrabajo)
  MatrizAG<-MatrizTrabajo
  colnames(MatrizAG)<-NULL
  MatrizTSP<-TSP(as.matrix(MatrizTrabajo), labels = NULL)
  initialtour<-as.integer(which(labels(MatrizTrabajo)[1]=="IRAOLA"))
  solucion1<-solve_TSP((MatrizTSP),method="farthest_insertion",control=list(start=initialtour))
  solucion2<-solve_TSP((MatrizTSP),method="nearest_insertion",control=list(start=initialtour))
  solucion3<-search_tour_genetic(MatrizAG,dim(MatrizAG)[1],input$Npop,input$Ngen,input$Local,input$Elite)
  Longitud1<-tour_length(solucion1)
  Longitud2<-tour_length(solucion2)
  Longitud3<-solucion3$distance}
```

Ilustración 42: Código de representación del algoritmo genético.

La siguiente sección de programa, es muy similar a la de la ilustración 41.

En esta parte del programa se encuentran los comandos, que se utilizan para la representación del trayecto en el mapa, en donde aparecen representados las ubicaciones de los municipios que deben visitarse.

```
Ruta<-list(labels(solucion1),labels(solucion2),colnames(MatrizTrabajo)[solucion3$tour])
Longitudes<-c(Longitud1,Longitud2,Longitud3)
Seleccion<-which.min(Longitudes)
Rutafinal<-Ruta[[Seleccion]]
coordenadas<-coordenadas[order(Rutafinal),]
coordenadas<-as.data.frame(coordnadas)
ruta<-data.frame(coordnadas$LONGITUD,coordnadas$LATITUD)
names(ruta)<-c("LONGITUD","LATITUD")
ID1=rownames(coordnadas)
ors_api_key('5b3ce3597851110001cf6248c461a41c55e84e7fac8efbda9e23fc4e')
route <- ors_directions(asplit(ruta,1), format="geojson",radiuses=1000000)
mapa <- leaflet(data = coordenadas) %>% #crea el mapa
  addTiles() %>% # Add default OpenStreetMap map tiles
  setView(lng = coordenadas$LONGITUD[1], lat = coordenadas$LATITUD[1] , zoom = 1)
  addMarkers(~LONGITUD, ~LATITUD, popup = ~as.character(ID1), label = ~as.character(ID1))
  addGeoJSON(route, fill=FALSE) %>%
  fitBoundingBox(route$bbox)
}}
```

Ilustración 43: Código de representación de ruta.

Para finalizar con la explicación de la programación, se hace huso de los siguientes comandos.

Con estos comandos, se consigue mostrar por pantalla la representación de la tabla donde se muestran las distancias y rutas separadas en tres columnas, donde se separan por el algoritmo utilizado. Es decir, una columna mostrará la distancia y ruta tanto del algoritmo del vecino más cercano, del más lejano y del algoritmo genético.

```
output$values<-renderTable({
  Tabla<-data.frame(slidervalues())
  if(length(c("IRAOLA",c(input$variable1)))<10){names(Tabla)<-c("ALGORITMO DEL VECINO MÁS CERCANO",
    "ALGORITMO DEL VECINO MÁS LEJANO")
    Tabla}
  else{names(Tabla)<-c("ALGORITMO DEL VECINO MÁS CERCANO","ALGORITMO DEL VECINO MÁS LEJANO",
    "ALGORITMO GENÉTICO")
    Tabla}
  })

output$mapa<-renderLeaflet({mapaselect()})
}

shinyApp(ui = ui, server = server)
```

Ilustración 44: Representación de valores en la aplicación.

5.4. INTERFAZ DE LA APLICACIÓN "RUTA DE RECOGIDA DE FANGOS"

En este apartado se pretende explicar detalladamente el aspecto de la aplicación HTML de Shinny, es decir de la parte visible con la que interactúan los operarios.

Con esta explicación, se conseguirá diferenciar las entradas y salidas de la aplicación. Además, esta explicación servirá también como manual de uso de esta.

RUTA DE RECOGIDA DE FANGOS EN LAS EDAR DE NAVARRA

Sector zona Media Navarra

Parámetros algoritmo genético

Seleccione las EDAR de recogida
☐ IRAOLA ☒ BARASOAIN ☒ PUEYO ☒ SAN MARTIN DE UNX ☒ UJUE ☒ OLITE ☒ PITILLAS
☒ SANTACARA ☒ CARCASTILLO ☒ RADA ☒ FIGAROL

Calcular ruta

ALGORITMO DEL VECINO MÁS CERCANO	ALGORITMO DEL VECINO MÁS LEJANO	ALGORITMO GENÉTICO
Longitud (km): 129.81	Longitud (km): 137.83	Longitud (km): 129.37
IRAOLA	IRAOLA	IRAOLA
PUEYO	PUEYO	BARASOAIN
SAN MARTIN DE UNX	SAN MARTIN DE UNX	OLITE
UJUE	UJUE	PITILLAS
CARCASTILLO	CARCASTILLO	RADA
FIGAROL	FIGAROL	SANTACARA
SANTACARA	SANTACARA	CARCASTILLO
RADA	RADA	FIGAROL
PITILLAS	PITILLAS	UJUE
OLITE	OLITE	SAN MARTIN DE UNX

Ilustración 45: Interfaz de la aplicación.

Primeramente, se debe tener en cuenta que la ubicación inicial siempre será la misma "Iraola", siendo este el inicio y el fin de la ruta. Este punto no se marca con un tick debido a que ya se da por hecho que tiene que estar contemplado en la ruta.

Una vez seleccionado el origen de la ruta, se seleccionan los nombres de los pueblos a los que se desea acudir para la recogida semanal de fangos como bien se muestran en un panel de selección con todos los municipios que forman parte de la red de distribución y que están incluidos en la matriz de distancias.

RUTA DE RECOGIDA DE FANGOS EN LAS EDAR DE NAVARRA

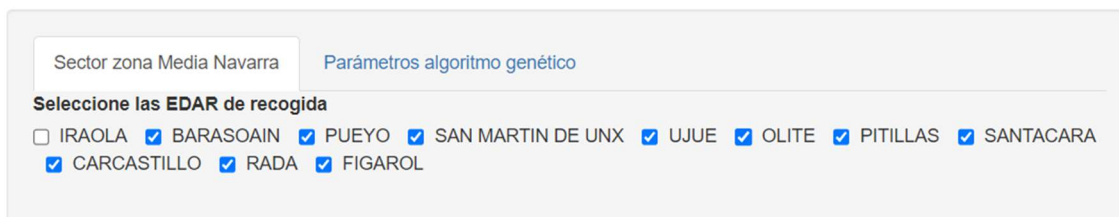


Ilustración 46: Panel de selección de municipios de la app.

Una vez se ha interactuado con las entradas de la aplicación, pasamos a observar las salidas de esta.

La primera vista que se tiene de las salidas son tres columnas que representan cada uno de los algoritmos que se tienen en cuenta para el cálculo de rutas.

**ALGORITMO DEL
VECINO MÁS
CERCANO**

**ALGORITMO DEL
VECINO MÁS LEJANO**

**ALGORITMO
GENÉTICO**

Ilustración 47: Algoritmos para el cálculo de rutas.

Dentro de cada una de las tres columnas diferenciadas en este apartado, se puede observar la información extraída de estos algoritmos, como puede ser la longitud o distancia para recorrer en esta ruta. La longitud viene representada en km.

**ALGORITMO DEL
VECINO MÁS
CERCANO**

**ALGORITMO DEL
VECINO MÁS LEJANO**

**ALGORITMO
GENÉTICO**

Longitud (km): 129.81

Longitud (km): 137.83

Longitud (km):
129.37

Ilustración 48: Longitud de las rutas.

Debajo de estas distancias, se encuentran las listas de ruta de reparto, con el orden de los municipios en el que deben ser visitados.

ALGORITMO DEL VECINO MÁS CERCANO	ALGORITMO DEL VECINO MÁS LEJANO	ALGORITMO GENÉTICO
Longitud (km): 129.81	Longitud (km): 137.83	Longitud (km): 129.37
IRAOLA	IRAOLA	IRAOLA
PUEYO	PUEYO	BARASOAIN
SAN MARTIN DE UNX	SAN MARTIN DE UNX	OLITE
UJUE	UJUE	PITILLAS
CARCASTILLO	CARCASTILLO	RADA
FIGAROL	FIGAROL	SANTACARA
SANTACARA	SANTACARA	CARCASTILLO
RADA	RADA	FIGAROL
PITILLAS	PITILLAS	UJUE
OLITE	OLITE	SAN MARTIN DE UNX
BARASOAIN	BARASOAIN	PUEYO

Ilustración 49: Ruta de reparto en la App.

Para finalizar, se puede observar un mapa con todos los municipios por los que pasa la ruta más óptima obtenida anteriormente.

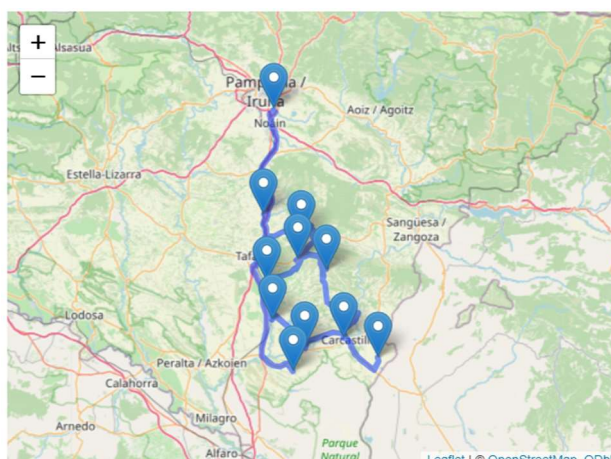


Ilustración 50: Mapa del trayecto a realizar.

6. RESULTADOS

En este apartado de resultados, se van a mostrar y se van a desarrollar los resultados obtenidos durante el desarrollo de este proyecto que se centraba en conseguir una optimización de las rutas de recogida de fangos.

Como bien se ha comentado con anterioridad a lo largo de la memoria, los resultados obtenidos se encuentran dentro de una página interactiva, que permite a los operarios encargados de realizar la ruta de extracción de fangos conocer cuál es la ruta más corta, además de conocer cuál es la ubicación exacta de estas EDAR.

Los resultados obtenidos por la aplicación en este proyecto han sido resultado de la actual situación de red de recogida de fangos, es decir la ruta ha sido obtenida teniendo en cuenta los municipios cuyas depuradoras corren a cargo de la sede de Olite. Esta ruta cuenta con 10 municipios sin contar el punto inicial, desde donde se encuentra la sede de la compañía de extracción de fangos.

Sin embargo, esta aplicación ha sido diseñada con visión a futuro es decir se ha diseñado de tal forma que sea fácil de editar o añadir nuevos municipios que se quieran introducir en esta ruta de extracción de fangos. Es por ello por lo que, aunque cambien los puntos de recogida, así como el orden de ruta, esta página siempre será actualizada para obtener la nueva ruta optimizada.

6.1. RESULTADOS DE LA RUTA ACTUAL

Los principales resultados a destacar son los relacionados al apartado de optimización de rutas de extracción de fangos en las EDAR de la zona media de Navarra.

Como se ha comentado anteriormente el funcionamiento de la aplicación en el apartado 5.4, primeramente, se seleccionan los municipios a los que se va a acudir para extraer los fangos, dejando sin seleccionar el punto de inicio de la ruta IRAOLA.

Una vez seleccionados estos puntos, nos dirigimos a la zona inferior y pinchamos en el botón azul de calcular ruta.

En este caso se puede observar que nos muestra tres rutas calculadas con diferentes algoritmos.

Pero únicamente nos centramos en la ruta más corta, es decir en la columna de algoritmo genético que como se puede observar muestra que esta opción es la ruta más corta con una distancia por recorrer de tan solo 129.37 km, frente a los 129.81 km del algoritmo más cercano y 137.83 km del algoritmo del vecino más lejano.

En cuanto a la ruta a seguir dentro del algoritmo genético, indica que se debe comenzar por IRAOLA, Barasoain, Olite, Pitillas, Rada, Santacara, Carcastillo, Figarol, Ujue, San Martín y Pueyo.

Se puede observar como la distancia recorrida con la ruta calculada con el algoritmo genético, no supone una disminución enorme en la distancia recorrida por los otros dos algoritmos, lo que a su vez resulta en una reducción de costes poco destacable.

Sin embargo, la reducción de costes más considerables se encuentra en la modificación que se pretende realizar en la organización y planificación del sistema de recogida de fangos.

Como bien se ha comentado en apartados anteriores de este proyecto, hasta la actualidad la recogida de fangos se realizaba de manera individual, es decir cuando se alcanzaba un límite máximo de fangos en el almacén de una EDAR, se contactaba de urgencia con la única compañía que tenía disponibilidad para realizar esta tarea.

Esta compañía, dispone de una flota de camiones muy elevada, lo que le permite estar siempre disponible. El único inconveniente es que su precio por la extracción de 1000 litros supone un costo de 170€.

170€ por cada 1000 litros.

Cada recogida son 25000 litros

$25 \text{ mil litros} \times 170€ \times 38,4 \text{ semanas} = 163200€ \text{ al año}$

Durante el desarrollo de este proyecto, se observó que uno de los grandes inconvenientes de este sistema era el simple hecho de retirar los fangos únicamente cuando los niveles del almacén superan los límites máximos preestablecidos.

Este sistema suponía que en algunas edar se extraigan los fangos una vez cada dos años, lo que supone grandísimos problemas por averías y atascos, causados por la solidificación de los conductos de extracción.

Estas averías se producían 2 o 3 veces anuales, con una inversión por cada avería de en torno a 6000€ sin tener en cuenta los gastos de personal de cuatro operarios durante una semana que supone en torno a 2000€. Es decir, unos 18000€ solo en gastos por averías.

Con el sistema anterior de recogida, se calculan unos gastos de entorno:

$$163200 \text{ €} + 18000 \text{ €} + 2000 \text{ €} = 183200 \text{ € anuales.}$$

Con el sistema que se ha propuesto a partir de este proyecto, se ha descubierto que, si se extraen los fangos semanales producidos en cada depuradora, estas extracciones evitan que se solidifiquen los fangos de los conductos. Por ello, se observó que este sistema nos ahorraría 20000€ anuales en averías.

Además, se llegó a un acuerdo con otra compañía de extracción que era capaz de realizar este tipo de tareas ya que sus precios son mucho más bajos debido a su poca disponibilidad, sin embargo, al acordar con una gran anterioridad el hecho de realizar una extracción semanal en cada depuradora, no les suponía ningún inconveniente ya que tienen un mayor margen de maniobra y de tiempo.

Esta nueva compañía cuenta con unos precios inferiores a la competencia:

150€ por cada 1000 litros

Cada recogida semanal será de 20000 litros

$$20 \text{ mil litros} \times 150\text{€} \times 48 \text{ semanas} = 144000\text{€ al año}$$

Como se puede observar, el ahorro realizado simplemente por cambiar de compañía supone un total de unos 20000 € anuales incluso realizando 10 extracciones más al año, algo que puede parecer contradictorio a primera vista.

Pero estos 20000 € de diferencia se le tienen que añadir los gastos de materiales y de personal por averías que suponen al año un total de 20000 €.

$$183200 \text{ €} - 144000 \text{ €} = 38200 \text{ €}$$

Es decir, con el cambio de compañía y de sistema de extracción conseguiríamos un ahorro anual de 38200 €, lo que supone un ahorro considerable.

7. CONCLUSIONES

Durante el desarrollo de este proyecto, se ha enfatizado los beneficios de afrontar la optimización desde un punto de vista más global, tratando de tener en cuenta la gran mayoría de los puntos que consideremos necesarios, incluso los que puedan parecer poco relevantes en un comienzo.

Inicialmente, el proyecto giraba en torno al diseño de una aplicación que permita a los usuarios optimizar las rutas de recogida de fangos.

Sin embargo, durante la investigación se abrió una puerta que añadía mucha mayor dificultad en la gestión de los tiempos de extracción de los fangos.

Por el simple hecho de observar y destacar la existencia de depuradoras que operan con una frecuencia de extracción de fangos de tan solo una vez cada dos años.

Este aspecto hizo que tuviéramos que posicionarnos en numerosas y variadas situaciones resultando un desafío crítico, ya que la solidificación del fango producida por la inactividad en los conductos de extracción durante largos períodos de tiempo, hacía que resultara en problemas operativos tanto de tiempos como de costos adicionales por averías.

Tras realizar numerosos estudios de análisis de costos y beneficios, se observó que para sorpresa de todos resulta más barato llegar a acuerdo o realizar un contrato anual con la compañía de gestión de residuos con precios más bajos y realizar semanalmente recogidas del fango generado en cada depuradora, en comparación con la estrategia actual de llamar a la compañía de recogida con mayor disponibilidad únicamente cuando los almacenes superen el límite máximo preestablecido.

Aunque inicialmente este descubrimiento pueda parecer contraintuitivo, muestra claramente la importancia que tiene el simple hecho de evaluar todos los aspectos y opciones disponibles, que puedan tener implicaciones a la hora de tomar decisiones a largo plazo.

El simple hecho de tomar una estrategia con mayor iniciativa y más activa en cuanto a realizar recogida de fangos con mayor regularidad no solo supone una reducción considerable de los riesgos tanto por solidificación como por las averías, sino que se traduce directamente en una cantidad considerable de ahorros en términos de costos operativos y de mantenimiento.

A modo de resumen, con este proyecto se pretende mostrar las enormes ventajas que supone el simple hecho de adoptar una visión más global y exhaustiva a la hora de solventar situaciones o proyectos en los que se pretende realizar una optimización como la de este proyecto.

Simplemente, intentando no focalizarse en únicamente la resolución del problema sino tomando una evaluación cuidadosa y una consideración minuciosa de todas las variables que se involucran, con este tipo de metodología se permite desarrollar soluciones más eficaces y que resulten más viables en términos económicos que optimicen los procesos y reduzcan los costos en el largo plazo.

8. OBJETIVOS DE DESARROLLO SOSTENIBLE

Los objetivos de este Trabajo Fin de Grado están alineados con los siguientes Objetivos de Desarrollo Sostenible (ODS) y metas, de la Agenda 2030:

- **Objetivo 6** – Garantizar la disponibilidad de agua y su gestión sostenible y el saneamiento para todos. Agua limpia y saneamiento. (Mediante la reducción significativa del tiempo que se encuentra el agua mezclada con los fangos).

- **Meta 6.1.** Separar el agua potable de las aguas residuales.



- **Objetivo 13.** Adoptar medidas urgentes para combatir el cambio climático y sus efectos. Acción por el clima. (Integración de métodos sostenibles para la gestión de rutas para disminuir la huella de carbono).

- **Meta 13.2.** Incorporar medidas relativas al cambio climático en las políticas, estrategias y planes nacionales.



- **Objetivo 14.** Conservar y utilizar sosteniblemente los océanos, los mares y los recursos marinos para el desarrollo sostenible. Vida submarina (Disminución drástica de vertidos de fangos en hábitats acuáticos).



- **Objetivo 15.** Pretende conservar la vida de ecosistemas terrestres (Reducción en el impacto del suelo causado por las averías y reparaciones del sistema de recolección de fangos)



9. BIBLIOGRAFÍA

- [1] «Estación depuradora de aguas residuales (EDAR)». Accedido: 12 de febrero de 2024. [En línea]. Disponible en: <https://www.miteco.gob.es/es/agua/temas/saneamiento-depuracion/sistemas/edar.html>
- [2] «Contenidos y esquemas CTMA de María Espinosa: Funcionamiento de una EDAR (Estación Depuradora de Aguas Residuales)». Accedido: 12 de febrero de 2024. [En línea]. Disponible en: <https://mariaespinosactma.blogspot.com/2017/05/funcionamiento-de-una-edar-estacion.html>
- [3] «Línea de agua». Accedido: 12 de febrero de 2024. [En línea]. Disponible en: https://www.miteco.gob.es/es/agua/temas/saneamiento-depuracion/sistemas-tratamiento/linea_de_agua.html
- [4] «Lodos de depuración de aguas residuales». Accedido: 12 de febrero de 2024. [En línea]. Disponible en: <https://www.miteco.gob.es/es/calidad-y-evaluacion-ambiental/temas/prevencion-y-gestion-residuos/flujos/lodos-depuradora.html>
- [5] «Línea de fangos». Accedido: 12 de febrero de 2024. [En línea]. Disponible en: https://www.miteco.gob.es/es/agua/temas/saneamiento-depuracion/sistemas-tratamiento/linea_de_fangos_v01.html
- [6] «Problema del viajante - Wikipedia, la enciclopedia libre». Accedido: 12 de febrero de 2024. [En línea]. Disponible en: https://es.wikipedia.org/wiki/Problema_del_viajante
- [7] «TSP: Algoritmos de resolución». Accedido: 12 de febrero de 2024. [En línea]. Disponible en: <https://knuth.uca.es/moodle/mod/page/view.php?id=3417>
- [8] «R-foro: Calculo de rutas en R». Accedido: 12 de febrero de 2024. [En línea]. Disponible en: <https://knuth.uca.es/moodle/mod/forum/discuss.php?d=1376>



Relación de documentos

(X) Memoria 58 páginas

(_) Anexos 17 páginas

La Almunia, a 29 de 02 de 2024

Firmado: Guillermo Aldunate Ozcariz