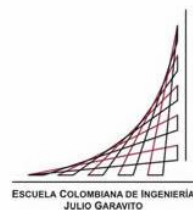


Maestría en Ingeniería Civil

**EciSewer - Programa de uso libre para el diseño de redes de
alcantarillado en perfil**

Miguel Alejandro Jiménez Barajas

Bogotá, D.C., 26 de enero de 2022



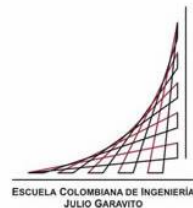
**EciSewer - Programa de uso libre para el diseño de redes de
alcantarillado en perfil**

**Tesis para optar al título de magíster en Ingeniería Civil, con
énfasis en recursos hidráulicos y medio ambiente**

Ing. Germán Ricardo Santos Granados

Director

Bogotá, D.C., 26 de enero de 2022



La tesis de maestría titulada “EciSewer - Programa de uso libre para el diseño de redes de alcantarillado en perfil”, presentada por Miguel Alejandro Jiménez Barajas, cumple con los requisitos establecidos para optar al título de Magíster en Ingeniería Civil con énfasis en recursos hidráulicos y medio ambiente.

Director de la tesis

Ing. Germán Ricardo Santos Granados

Jurado

Ing. Héctor Matamoros Rodríguez

Jurado

Ing. Andrés Humberto Otálora Carmona

Bogotá, D.C., 26 de enero de 2022

Agradecimientos

A mi bella familia, que siempre llevo en mi mente y mi corazón

A mis colegas candidatos de maestría, quienes me brindaron agradables momentos

A German Santos, por permitirme el privilegio de ser mi director de tesis

A todas las personas quienes me brindaron su apoyo

¡A todos, gracias!

Resumen

Las infraestructuras de alcantarillado permiten coleccionar y transportar las aguas de las zonas pobladas a una disposición del fluido. Ya sea por la inexistencia o deficiencia de la infraestructura, se requieren de herramientas para implementar o diagnosticar el servicio de alcantarillado, entre ellas programas para el diseño y modelación de redes.

Las herramientas de uso común en el diseño de redes de alcantarillado traen consigo uno o varios problemas relacionados con su uso y propósito, entre ellos, los costos del software comercial, la necesidad de una capacitación a la medida del software utilizado, y en general su poca flexibilidad y nula posibilidad de ajustes.

Una alternativa de solución para los problemas asociados a los costos y la flexibilidad, es la generación de software libre, para el diseño de redes de alcantarillado, que permitirá al usuario la oportunidad de ajustar los aspectos necesarios para mejorar el diseño o potenciar su rendimiento.

A partir de lo anterior, la hipótesis de este trabajo, es que, mediante la implementación de un modelo numérico en un software de uso libre, permita al usuario, diseñar redes de alcantarillado en perfil de forma eficiente, sencilla y de bajo costo, facilitando la toma de decisiones de inversión o mantenimiento en infraestructura.

El trabajo de grado busca contribuir en el aprovechamiento y la gestión de los recursos hidráulicos de la línea de investigación Modelación física y numérica de los Recursos Hidráulicos del Grupo de investigación centro de estudios hidráulicos de la Escuela Colombiana de Ingeniería Julio Garavito.

Palabras clave: Sistema de alcantarillado, diseño hidráulico alcantarillados, superficie libre, canales abiertos.

Índice General

1	Capítulo I – Justificación, objetivos y estado del arte.....	14
1.1	Justificación	14
1.2	Objetivo general	14
1.3	Objetivos específicos.....	14
1.4	Algunos softwares de análisis y diseño de redes de alcantarillado.....	15
1.4.1	Storm Water Management Model (SWMM):	15
1.4.2	Software de hidráulica e hidrología BENTLEY	15
1.4.3	APIs:.....	16
1.4.3.1	SWMM5 para python:	16
1.5	Normativa Colombiana	16
1.6	Documentos académicos - Tesis.....	17
1.6.1	Metodología para el diseño optimizado de redes de alcantarillado(Duque Villarreal, 2015).....	17
1.6.2	Métodos de diseño en redes de alcantarillado(Torres Dueñas, 2013).....	17
1.7	Criterios de diseño.....	18
1.7.1	Características geométricas de la tubería circular	18
1.7.1.1	Ángulo entre la superficie libre y el eje de la sección de la tubería	19
1.7.1.2	Área mojada transversal	19
1.7.1.3	Perímetro mojado	19
1.7.1.4	Radio Hidráulico	20
1.7.1.5	Ancho de la sección en la superficie libre	20
1.7.2	Dimensionamiento Hidráulico - Ecuación de Manning.....	20
1.7.3	Régimen de flujo.....	21
1.7.4	Alcantarillado de aguas residuales	21
1.7.4.1	Período de diseño.....	21

1.7.4.2	Caudal sanitario de diseño	22
1.7.4.2.1	Caudal de aguas residuales domésticas (QD)	22
1.7.4.2.2	Caudal medio diario de aguas residuales.....	22
1.7.4.2.3	Factor de mayoración (F)	22
1.7.4.2.4	Caudal máximo horario (QMH).....	23
1.7.4.2.5	Caudal de diseño	23
1.7.5	Velocidades Mínimas.....	23
1.7.6	Alcantarillado de aguas lluvias	24
1.7.6.1	Período de diseño.....	24
1.7.6.2	Caudal de diseño.....	24
1.7.6.3	Curvas de Intensidad-Duración- Frecuencia.....	24
1.7.6.4	Coeficiente de escorrentía	25
1.7.6.5	Tiempo de concentración.....	25
1.7.7	Pérdidas de energía en colectores	26
1.7.8	Alcantarillado de aguas combinadas	27
1.8	Criterios de diseño en la norma colombiana	27
1.9	Ecuación de Autolimpieza	29
1.10	Consideraciones para flujo uniforme en sistemas de alcantarillado.....	29
1.11	Tipos de tuberías y accesorios	29
1.12	Consideraciones de estabilidad de flujo uniforme.....	30
1.13	Restricciones de uso del aplicativo	30
2	Capítulo II – Metodología.....	31
2.1	Desarrollo Computacional.....	31
2.1.1	Diagrama de flujo del algoritmo	32
2.1.2	Archivos de entrada requeridos para la ejecución del aplicativo.....	34
2.1.2.1	Archivo red.csv	34

2.1.2.2	Archivo tuberías.csv.....	36
2.1.2.3	Archivo setup.csv.....	37
2.1.3	Archivos de salida del aplicativo.....	39
2.1.4	Variables de interés.....	39
2.1.5	Descripción de funciones y rutinas.....	39
2.1.5.1	Rutinas para la carga/ejecución de proyectos:.....	39
2.1.5.2	Rutinas para la visualización de datos en QGIS.....	40
2.1.5.3	Rutinas para la selección de la tubería – proceso iterativo.....	41
2.1.5.4	Rutinas para el control de acciones en el formulario del aplicativo.....	42
2.1.5.5	Rutinas y funciones auxiliares.....	43
2.3	Cálculo de costos.....	44
2.3.1	Análisis de precios unitarios.....	44
2.3.2	Función de costos CRA.....	48
2.3.3	Función de costos propia.....	49
2.4	Prerrequisitos para el uso del aplicativo.....	50
2.4.1	Instalación de QGIS.....	50
2.4.2	Instalación de SWMM.....	50
2.4.3	Instalación de Anaconda.....	50
2.4.4	Instalación de SewerGems.....	51
2.4.5	Instalación de SWMM5 para Python.....	51
2.4.5.1	En QGIS 3 para Windows.....	51
2.4.5.2	En Anaconda para Windows.....	51
2.4.6	Instalación del aplicativo en QGIS.....	52
2.4.7	Ejecución del aplicativo en QGIS.....	53
2.5	Presentación de los perfiles:.....	53
3	Capítulo III - Resultados y contribución.....	56

3.1	Casos de estudio	56
3.1.1	Caso de estudio 1 - Ejemplo02.....	57
3.1.2	Caso de estudio 2 - Ejemplo03.....	59
3.1.3	Caso de estudio 3 - Ejemplo01.....	60
3.1.4	Caso de estudio 4 - Ejemplo04.....	61
3.1.5	Caso de estudio 5 - Ejemplo05.....	62
3.2	Costo computacional en pruebas de ejecución.....	63
3.3	Comparación del costo mediante análisis de precios unitarios respecto a las funciones de costos.....	64
4	Conclusiones y recomendaciones	67
5	Referencias	70
6	Anexos.....	72

Índice de Tablas

Tabla 1. Ejemplo Coeficientes IDF	25
Tabla 2. Pérdida de energía en colectores	26
Tabla 3. Resumen de los criterios de diseño en la norma colombiana	27
Tabla 4. Estructura archivo “ <i>red.csv</i> ”	34
Tabla 5. Características de las tuberías seleccionadas	36
Tabla 6. Estructura archivo “ <i>setup.csv</i> ”	38
Tabla 7. Precios unitarios de referencia	45
Tabla 8. Precios unitarios seleccionados para tuberías	47
Tabla 9. Resumen de funciones (costos en pesos de diciembre de 2007)	48
Tabla 10. Descripción de casos de estudio	57
Tabla 11. Caso de estudio 1 - Resumen de información de interés de la red diseñada. ..	58
Tabla 12. Caso de estudio 2 - Resumen de información de interés de la red diseñada. ..	59
Tabla 13. Caso de estudio 3 - Resumen de información de interés de la red diseñada. ..	61
Tabla 14. Caso de estudio 4 - Resumen de información de interés de la red diseñada. ..	62
Tabla 15. Caso de estudio 5 - Resumen de información de interés de la red diseñada. ..	63
Tabla 16. Costo computacional de casos de estudio	63
Tabla 17. Comparación de funciones objetivo respecto al cálculo de costos mediante precios unitarios.	64

Índice de Figuras

Figura 1. Características geométricas de la tubería circular.	18
Figura 2. Formulario del aplicativo.....	31
Figura 3. Algoritmo de entrada, proceso y salida.....	31
Figura 4. Diagrama de flujo del algoritmo	33
Figura 5. Imagen ilustrativa para la construcción del archivo “red.csv”.....	35
Figura 6. Proyección de la sección del área de excavación.	44
Figura 7. Resultados ajuste función de costos propia.	49
Figura 8. Instalación de SWMM5 para Python en QGIS – paso 1.....	51
Figura 9. Instalación de SWMM5 para Python en QGIS – paso 2.....	51
Figura 10. Instalación de SWMM5 para Anaconda.....	51
Figura 11. Instalación del aplicativo en QGIS – ruta de usuario.	52
Figura 12. Instalación del aplicativo en QGIS – Menú de plugins de QGIS.	52
Figura 13. Instalación del aplicativo en QGIS – Ícono del aplicativo.....	52
Figura 14. Instalación del aplicativo en QGIS – Formulario del aplicativo.	53
Figura 15. Ejemplo de vista con zoom de escala grande en QGIS.....	54
Figura 16. Ejemplo de vista con zoom de escala pequeña en QGIS.....	54
Figura 17. Vista del perfil de la red “Ejemplo02” en SWMM:	55
Figura 18. Vista del perfil de la red “Ejemplo02” en SewerGems:	55
Figura 19. Esquema en planta de los casos de estudio	56
Figura 20. Caso de estudio 1 - Perfil diseñado entre los pozos 64_5 y AS239.....	57
Figura 21. Caso de estudio 1 - Planta de la red entre los pozos 64_14 y 64_21	58
Figura 22. Caso de estudio 2 - Perfil diseñado entre los pozos AL_211y AL_215.....	59
Figura 23. Caso de estudio 3 - Perfil diseñado entre los pozos AS_115_1 y AS_116.....	60
Figura 24. Caso de estudio 4 - Perfil diseñado entre los pozos AL_28 y AL_32.....	61
Figura 25. Caso de estudio 4 - Perfil diseñado entre los pozos AL_19 y AL_24.....	62
Figura 26. Error relativo porcentual. Costo calculado vs Función de costos CRA (1).....	65
Figura 27. Error relativo porcentual. Costo calculado vs Función de costos propia (1).....	65
Figura 28. Error relativo porcentual. Costo calculado vs Función de costos CRA (2).....	65
Figura 29. Error relativo porcentual. Costo calculado vs Función de costos propia (2).....	66

Índice de Ecuaciones

Ecuación 1. Ángulo entre la superficie libre y el eje de la tubería.....	19
Ecuación 2. Área mojada transversal de la tubería	19
Ecuación 3. Perímetro mojado de la tubería.....	19
Ecuación 4. Radio hidráulico de la tubería	20
Ecuación 5. Ancho de la sección en la superficie libre de la tubería.....	20
Ecuación 6. Ecuación de Manning	20
Ecuación 7. Ecuación Número de Froude	21
Ecuación 8. Ecuación Profundidad Hidráulica	21
Ecuación 9. Ecuación Caudal de aguas residuales domésticas	22
Ecuación 10. Ecuación Caudal medio de aguas residuales	22
Ecuación 11. Ecuación Caudal máximo horario	23
Ecuación 12. Ecuación Caudal de diseño	23
Ecuación 13. Ecuación Esfuerzo cortante en la pared	23
Ecuación 14. Ecuación Método racional.....	24
Ecuación 15. Ecuación Curvas de intensidad-duración-frecuencia	24
Ecuación 16. Tiempo de concentración.....	25
Ecuación 17. Ecuación de Autolimpieza.....	29
Ecuación 18. Volumen del área de excavación	44
Ecuación 19. Ecuación potencial.....	48
Ecuación 20. Ecuación Costos de las tuberías CRA	48
Ecuación 21. Ecuación Costos de excavación CRA.....	48
Ecuación 22. Ecuación Costos de las tuberías por metro lineal	49
Ecuación 23. Ecuación Costos de excavación por metro lineal.....	49

Índice de Anexos

Anexo 1. Código para la generación de la función de costos propia	72
Anexo 2. Código del aplicativo.....	74

Introducción

El diseño de las redes de alcantarillado se compone de dos componentes principales: la topología de la red y el diseño hidráulico: La topología de la red en planta define el recorrido del fluido de acuerdo a las necesidades de descarga de una población. El diseño hidráulico determina la topología en perfil de cada tubería, cuyas restricciones se definen de acuerdo a la normatividad vigente y a los criterios de diseño.

El propósito principal es diseñar un software de uso libre para el diseño de redes de alcantarillado en perfil. El software genera una solución hidráulica tubo a tubo de redes de alcantarillado, minimizando el costo de suministro e instalación de la tubería, analizando cada tramo individualmente. Estos costos se calculan mediante precios unitarios actualizados a la fecha de publicación de este documento, y mediante dos funciones de costos, una propia y otra propuesta por la comisión regulación de agua potable y saneamiento básico – CRA.

El documento está organizado de la siguiente manera

Capítulo I – Justificación, objetivos y estado del arte

- Justificación y objetivos
- Marco teórico
- Normativa colombiana
- Definición de criterios de diseño y restricciones

Capítulo II – Metodología: Descripción de las variables, restricciones y condiciones a usar en el software propuesto

- Detalles del desarrollo computacional
- Descripción del algoritmo y su proceso iterativo, para el diseño hidráulico de redes de alcantarillado en perfil.

Capítulo III - Resultados

Capítulo IV – Conclusiones

Capítulo I – Justificación, objetivos y estado del arte

1.1 Justificación

La modelación con fines de diseño de infraestructuras de alcantarillado, toma importancia como herramienta para la toma de decisiones en la gestión de inversiones de salud pública para zonas urbanas.

El software propuesto resuelve con bases conceptuales soportadas en desarrollos matemáticos consistentes con la normatividad vigente, la física de elementos de las redes de alcantarillado en perfil, mediante un programa de fácil uso. Para esto se valoran alcances y elementos del modelo físico a implementar como:

- Selección y clasificación de criterios de diseño, datos de entrada y restricciones normativas e hidráulicas.
- Selección de tuberías mediante el uso de funciones de costos y análisis de precios unitarios.
- Marco normativo y valoración del “Estado del arte” de software de uso común en el diseño hidráulico de las redes de alcantarillado, e integración del mismo en el aplicativo desarrollado.

1.2 Objetivo general

- Diseñar un software de uso libre para el diseño de redes de alcantarillado en perfil en flujo uniforme.

1.3 Objetivos específicos

- De acuerdo a la revisión bibliográfica y del estado del arte en el diseño de redes de alcantarillado en perfil, identificar las variables más importantes, alcances y metodologías requeridas para la generación del aplicativo.
- Definir metodologías de solución hidráulica que cumplan con los criterios de diseño y restricciones.
- Generar en QGIS una interfaz gráfica de usuario, proporcionando un entorno visual para el control de ejecución, visualización y exportación de datos.

- Determinar las ventajas y desventajas del algoritmo propuesto, mediante el análisis de resultados, pruebas de ejecución y de alcances deseables para software de diseño de alcantarillado en flujo uniforme.

La hipótesis de este trabajo, es que, mediante la implementación de un modelo numérico en un software de uso libre, se permite al usuario, diseñar redes de alcantarillado en perfil, de forma eficiente, sencilla y de bajo costo, facilitando la toma de decisiones de inversión o mantenimiento en infraestructura.

1.4 Algunos softwares de análisis y diseño de redes de alcantarillado

1.4.1 Storm Water Management Model (SWMM):

Storm Water Management Model o modelo de gestión de aguas pluviales - SWMM (United States Environmental Protection Agency - EPA, 2020), es un software de uso libre que permite la modelación dinámica de precipitaciones especialmente en alcantarillados urbanos. Permite realizar la modelación y diseño de redes de drenaje y eventos de precipitación. Cuenta con tres módulos: Hidráulico, hidrológico y de calidad.

Respecto al Modelado Hidráulico: SWMM tiene herramientas de modelación de características hidráulicas, las cuales se integran al software propuesto, y son de utilidad cuando se requieran análisis con flujo gradualmente variado y flujo permanente y no permanente. SWMM cuenta con integración con lenguajes de libre uso como Python, lo cual permite la posibilidad la integración con el software propuesto.

1.4.2 Software de hidráulica e hidrología BENTLEY

Conjunto de aplicativos de uso comercial (Bentley, n.d.), en plataformas GIS o CAD, permite la solución de modelos de alcantarillado, analizando los elementos de la red. Entre sus productos SewerGEMS soporta la importación de archivos con la misma estructura de datos soportada por SWMM. Permite la modelación de redes de alcantarillado cuando se requieran análisis con flujo gradualmente variado, permanente y no permanente.

1.4.3 APIs:

Herramientas de uso libre, que permiten la integración de códigos mediante APIs (Interfaz de programación de aplicaciones), generalmente en visual Basic o Python, permitiendo usar e integrar la potencialidad de programas de análisis hidráulico e hidrológico como SWMM, HEC-RAS y EPANET. Algunos de ellos son Pipedream, QEsg, PyRAS, SWMM5, PySWMM y GISWATER.

1.4.3.1 SWMM5 para python:

SWMM5 permite su integración en lenguaje Python, para la creación y manipulación de redes complejas. Ofrece posibilidades para la manipulación de controles y obtención de los resultados del posprocesamiento del modelado de redes, y otras posibilidades de integración que ofrece la programación en Python.

1.5 Normativa Colombiana

Como principal referencia de normatividad técnica colombiana, se usa la Resolución 0330 de 2017 expedida por el Ministerio de Vivienda, Ciudad y Territorio (Reglamento Técnico Del Sector de Agua Potable y Saneamiento Básico - RAS | Minvivienda, 2017); la cual reglamenta los requisitos técnicos que se deben cumplir en las etapas de diseño construcción, puesta en marcha, operación, mantenimiento y rehabilitación de la infraestructura relacionada con los servicios públicos de acueducto, alcantarillado y aseo, cuyas variables a usar en el desarrollo del software son consistentes con el marco teórico y los criterios de diseño de este documento.

Como referencias normativas secundarias, se usa la Guía para el diseño hidráulico de Redes de Alcantarillado de EPM y la norma técnica del servicio de diseño de sistemas de alcantarillado de la empresa de acueducto y alcantarillado de Bogotá.

1.6 Documentos académicos - Tesis

A continuación, se mencionan trabajos de tesis desarrollados que aportan al conocimiento del desarrollo del presente documento, especialmente en la definición de variables y restricciones del modelo a implementar en el software, así como de buenas prácticas a tener en cuenta en los alcances del aplicativo:

1.6.1 Metodología para el diseño optimizado de redes de alcantarillado(Duque Villarreal, 2015)

Metodología para el diseño optimizado de redes de alcantarillado (Duque Villarreal, 2015) propone *“Desarrollar una metodología para la selección del trazado y el diseño hidráulico costo-óptimo de redes de alcantarillado, teniendo en cuenta los aspectos hidráulicos que aseguran el funcionamiento adecuado del sistema de drenaje urbano, siguiendo las normas colombianas: Reglamento Técnico del Sector de Agua Potable y Saneamiento Básico”*.

El documento es referencia para el uso de funciones objetivo de costos, valoración de restricciones y alcances generales del algoritmo.

1.6.2 Métodos de diseño en redes de alcantarillado(Torres Dueñas, 2013)

Métodos de diseño en redes de alcantarillado (Torres Dueñas, 2013) busca conocer las prácticas más usadas en el diseño actual de redes de alcantarillado en Colombia, para realizar una comparación entre métodos de diseño usados por diseñadores de sistemas de alcantarillado en Colombia.

La tesis presenta varias metodologías para resolver el problema del diseño hidráulico optimizado de alcantarillados, y una selección de preguntas realizadas en la tesis a diseñadores de sistemas de alcantarillado en Colombia y es referencia básica para:

- Aspectos relevantes para la selección del trazado.
- Selección del colector principal como la ruta de mayor pendiente.
- Métodos de optimización usados en el diseño hidráulico en redes de alcantarillado.
- Variables para minimizar los costos de la red.
- Software de uso común en el diseño en redes de alcantarillado.
- Criterios de diseño.

Presenta conclusiones y recomendaciones de acuerdo con los análisis de entrevistas, valora críticas a metodologías de diseño de alcantarillado comunes en consultoría, hace recomendaciones generales para la mejora en los resultados obtenidos, sobre el uso de criterios de diseño y sobre el uso de funciones de costos en sistemas de alcantarillado.

1.7 Criterios de diseño

1.7.1 Características geométricas de la tubería circular

Las propiedades geométricas de la sección transversal de la tubería circular parcialmente llena se describen teniendo en cuenta el diámetro interno de la tubería, el ángulo entre el centro de la sección transversal y los puntos de contacto entre la superficie libre y la tubería de acuerdo a la siguiente figura:

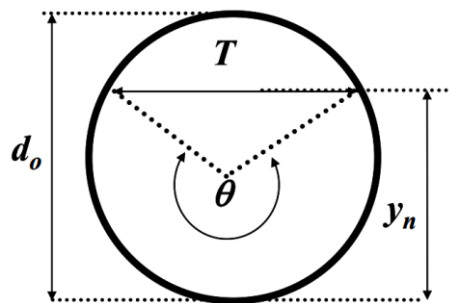


Figura 1. Características geométricas de la tubería circular.

Fuente: (Empresas Públicas de Medellín, 2013).

1.7.1.1 Ángulo entre la superficie libre y el eje de la sección de la tubería

El ángulo desde el centro de la sección transversal y los puntos de contacto entre el fluido y la tubería está dado por la Ecuación:

$$\theta = \pi + 2\arcsen\left(\frac{y_n - d_0 / 2}{d_0 / 2}\right)$$

Ecuación 1. Ángulo entre la superficie libre y el eje de la tubería

Donde,

θ = Ángulo (rad)

y_n = Profundidad del agua (m)

d_0 = Diámetro interno real de la tubería (m)

1.7.1.2 Área mojada transversal

De acuerdo a el ángulo y el diámetro de la tubería, el área mojada de la sección transversal se calcula de acuerdo a:

$$A = \frac{1}{8}(\theta - \text{sen}\theta)d_0^2$$

Ecuación 2. Área mojada transversal de la tubería

Donde A = Área mojada transversal (m²)

1.7.1.3 Perímetro mojado

El perímetro mojado correspondiente al perímetro del área mojada y se calcula de acuerdo a:

$$P = \frac{1}{2}\theta d_0$$

Ecuación 3. Perímetro mojado de la tubería

Donde P = Perímetro mojado (m)

1.7.1.4 Radio Hidráulico

Corresponde al cociente del área mojada y el perímetro mojado y se calcula de acuerdo a:

$$R = \frac{d_0}{4} \left(1 - \frac{\text{sen}\theta}{\theta} \right)$$

Ecuación 4. Radio hidráulico de la tubería

Donde R = Radio hidráulico (m)

1.7.1.5 Ancho de la sección en la superficie libre

Se calcula de acuerdo a:

$$T = d_0 \cos \left(\arcsen \frac{y_n - d_0 / 2}{d_0 / 2} \right)$$

Ecuación 5. Ancho de la sección en la superficie libre de la tubería

Donde T = Ancho de la sección en la superficie libre (m)

1.7.2 Dimensionamiento Hidráulico - Ecuación de Manning

Para el dimensionamiento de la sección transversal de la tubería con flujo hidráulicamente rugoso, se calcula de acuerdo con la Ecuación de Manning:

$$V = \frac{1}{n} \cdot R^{2/3} \cdot S^{1/2} \quad \text{o en términos de caudal: } Q = \frac{1}{n} \cdot AR^{2/3} \cdot S^{1/2}$$

Ecuación 6. Ecuación de Manning

Donde,

v = Velocidad media del flujo (m/s)

n = Coeficiente de rugosidad de Manning (s/m^{1/3})

Q = Caudal del flujo (m³/s)

R = Radio hidráulico (m)

A = Área mojada transversal (m²)

S = Pendiente longitudinal de la tubería (mm)

1.7.3 Régimen de flujo

El régimen de flujo, puede ser crítico, subcrítico o supercrítico de acuerdo con:

$$Fr = 1.0$$

$$Fr < 1.0$$

$$Fr > 1.0$$

El número de Froude está descrito mediante la ecuación:

$$Fr = \frac{v}{\sqrt{gD}}$$

Ecuación 7. Ecuación Número de Froude

Donde,

Fr = Número de Froude (-)

g = Aceleración de la gravedad (m/s²)

D = Profundidad hidráulica (m)

v = Velocidad de flujo (m/s)

La profundidad hidráulica está descrita mediante la ecuación:

$$D = \frac{d}{8} \left(\frac{\theta - \text{sen}\theta}{\text{sen}(\theta / 2)} \right)$$

Ecuación 8. Ecuación Profundidad Hidráulica

Donde,

θ = Ángulo (rad)

d = Diámetro interno de la tubería (m)

1.7.4 Alcantarillado de aguas residuales

1.7.4.1 Período de diseño

Requerido para el cálculo de las demandas de caudal de la población a futuro, y para el dimensionamiento de las tuberías de alcantarillado sanitario

El usuario ingresara la información requerida para el periodo de diseño deseado, como proyección de clientes, densidad población futura y dotación neta futura.

1.7.4.2 Caudal sanitario de diseño

1.7.4.2.1 Caudal de aguas residuales domésticas (Q_D)

Mediante la proyección de la población y su demanda, el caudal de diseño se debe calcular de acuerdo con:

$$Q_D = \frac{C_R \times P \times D_{NETA}}{86400}$$

Ecuación 9. Ecuación Caudal de aguas residuales domésticas

Donde,

Q_D = Caudal de aguas residuales domésticas (m³/s – l/s)

C_R = Coeficiente de retorno (adimensional)

P = Número de habitantes proyectados al período de diseño (hab)

D_{NETA} = Demanda neta (m³/hab/día - l/hab/día)

1.7.4.2.2 Caudal medio diario de aguas residuales

El caudal medio diario de aguas residuales (Q_{MD}) para una tubería con un área tributaria dada, corresponde a la suma de los caudales de aguas residuales domésticas, industriales, comerciales, oficiales y especiales, de acuerdo con:

$$Q_{MD} = Q_D + Q_I + Q_C$$

Ecuación 10. Ecuación Caudal medio de aguas residuales

Donde,

Q_{MD} = Caudal medio diario de aguas residuales (m³/s)

Q_I = Caudal de aguas residuales industriales (m³/s)

Q_C = Caudal de aguas residuales comerciales (m³/s)

1.7.4.2.3 Factor de mayoración (F)

Usado para calcular el caudal máximo horario con base en el caudal medio diario de aguas residuales domésticas. El factor de mayoración, puede establecerse utilizando datos de campo o aproximaciones indicadas por la norma vigente.

1.7.4.2.4 Caudal máximo horario (Q_{MH})

El caudal máximo horario de aguas residuales (Q_{MH}), establece el caudal de diseño de aguas residuales. Se calcula a partir del caudal medio diario, utilizando un factor de mayoración F, de acuerdo con:

$$Q_{MH} = F Q_D + Q_I + Q_C$$

Ecuación 11. Ecuación Caudal máximo horario

1.7.4.2.5 Caudal de diseño

Suma del caudal máximo horario (Q_{MH}), los aportes de caudal de infiltración y de aguas lluvias por conexiones erradas, de acuerdo con:

$$Q = Q_{MH} + Q_{INF} + Q_{CE}$$

Ecuación 12. Ecuación Caudal de diseño

Donde,

Q = Caudal de diseño para cada tubería (m^3/s)

Q_{MH} = Caudal máximo horario (m^3/s)

Q_{INF} = Caudal por infiltraciones (m^3/s)

Q_{CE} = Caudal por conexiones erradas (m^3/s)

1.7.5 Velocidades Mínimas

El diseño debe verificar el comportamiento autolimpiante del flujo, donde se debe utilizar el criterio de esfuerzo cortante en la pared:

$$\tau = \gamma \cdot R \cdot S$$

Ecuación 13. Ecuación Esfuerzo cortante en la pared

Donde,

τ = Esfuerzo cortante en la pared (Pa - N/m^2)

$\gamma = \rho \cdot g$ = Peso específico del agua residual (N/m^3)

R = Radio hidráulico (m)

S = Pendiente (m/m)

1.7.6 Alcantarillado de aguas lluvias

1.7.6.1 Período de diseño

Se tendrán en cuenta tanto el período de diseño como el período de retorno de la lluvia de diseño.

1.7.6.2 Caudal de diseño

Se usará el Método Racional:

$$Q = C i A \text{ (mm/h)} \text{ ó } Q = 2.78 C i A \text{ ((l/Ha/s))}$$

Ecuación 14. Ecuación Método racional

Donde,

Q =Caudal pico de aguas lluvias (L/s)

C = Coeficiente de escorrentía (adimensional)

i =Intensidad de precipitación (mm/h o L/s/ha)

A =Área tributaria (ha)

1.7.6.3 Curvas de Intensidad-Duración- Frecuencia

Las curvas de intensidad-duración–frecuencia (IDF) son base para el cálculo de los caudales de diseño de las redes de alcantarillado de aguas lluvias. Determinan y establecen la intensidad media de lluvia para diferentes duraciones de eventos de precipitación con períodos de retorno específicos. Se muestra la estructura de las ecuaciones usadas por el instituto de hidrología, meteorología y estudios ambientales IDEAM:

$$i = C / (D + X0)^{C2}$$

Ecuación 15. Ecuación Curvas de intensidad-duración-frecuencia

Donde,

i = Intensidad de precipitación (mm/h)

D = Duración de lluvia (min)

C,D,X0= Coeficientes de cada estación (adimensional)

La información requerida por el software son las constantes C1, X0 y C2 de las tablas IDF suministradas en la estación seleccionada:

Tabla 1. Ejemplo Coeficientes IDF

TR(años)	C1	X0	C2
2	6343.361	33.417	1.176
3	7077.232	33.889	1.175
5	7913.103	34.348	1.174
10	8984.033	34.840	1.173
25	10361.209	35.358	1.173
50	11395.816	35.682	1.173
100	12431.185	35.964	1.173

Fuente: Estación Obs Met Nacional - Estación Código: 2120523

Donde TR: Periodo de retorno (años)

1.7.6.4 Coeficiente de escorrentía

El coeficiente de escorrentía, C, es función del tipo del suelo de la cuenca, del grado de permeabilidad de la zona, de la pendiente del terreno y de los factores que determina qué parte de la precipitación se convierte en escorrentía superficial. Suministrado por el usuario en el software.

1.7.6.5 Tiempo de concentración

Tiempo que se demora el agua en recorrer el fluido en una cuenca, tubería o red de tuberías desde el punto más alejado, de acuerdo a:

$$T_C = T_e + T_t$$

Ecuación 16. Tiempo de concentración

Donde,

T_C = Tiempo de concentración (min)

T_e = Tiempo de entrada (min)

T_t = Tiempo de recorrido (min)

1.7.7 Pérdidas de energía en colectores

La pérdida de energía localizada h_L en el pozo, ya sea por el cambio en la pendiente del fondo del pozo y/o cambio de dirección en la cañuela, puede ser calculada de acuerdo a:

Tabla 2. Pérdida de energía en colectores

Régimen del Flujo	r_c/ϕ	Pérdida de Energía k_L
Subcrítico	1.0 a 1.5	$0.40 V^2/2g$
Subcrítico	1.5 a 3.0	$0.20 V^2/2g$
Subcrítico	Mayor de 3	$0.05 V^2/2g$
Supercrítico	6 a 8	$0.40 V^2/2g$
Supercrítico	8 a 10	$0.20 V^2/2g$
Supercrítico	Mayor de 10	$0.05 V^2/2g$

Fuente: Norma técnica de servicio – EAAB

Donde,

ϕ = diámetro del colector (m)

r_c = radio de curvatura del colector (-)

g = aceleración de la gravedad (m/s²)

V = velocidad del colector para flujo uniforme (m/s)

Se tiene en cuenta que:

- Para la transición de flujo de supercrítico a subcrítico: Cuando el régimen del flujo cambia de flujo de supercrítico a subcrítico entre los colectores de entrada y de salida, la elevación de la cota de batea del colector de salida se calculará empatando los niveles del agua entre la entrada y la salida.
- Para la transición de flujo de subcrítico a supercrítico: Cuando el régimen del flujo cambia de flujo de subcrítico a supercrítico entre los colectores de entrada y de salida, la elevación de la cota de batea del colector de salida se calculará empatando el nivel del Hw con el nivel del flujo de entrada.
- Cuando se presenten pérdidas de energía negativas o iguales a cero, se debe dejar una caída mínima de 2 cm entre cotas claves de las tuberías de acuerdo al artículo 137 del Reglamento Técnico para el Sector de Agua Potable y Saneamiento Básico.

1.7.8 Alcantarillado de aguas combinadas

Los parámetros de diseño para las redes de alcantarillados combinados corresponden a los criterios establecidos para los sistemas individuales de alcantarillado de aguas residuales y de alcantarillado de aguas lluvias. En general, los valores máximos y mínimos que gobiernan el diseño de los sistemas combinados corresponden a las redes de alcantarillado de aguas lluvias. Los caudales se asumen como pluviales a menos que el cálculo del caudal sanitario supere el 5% del caudal pluvial. El software no realiza cálculos asociados al diseño de alivios.

1.8 Criterios de diseño en la norma colombiana

De la normatividad de referencia se extrae:

Tabla 3. Resumen de los criterios de diseño en la norma colombiana

Parámetro	RAS	EPM	EAAB	EciSewer
Dotación neta máxima	120-140 l/hab-día	Proy población/demanda	Estudios de población y demanda	120 l/hab-día
Coeficiente AASS	0.85	0.85	0.85	0.85
F Caudal máximo horario	1.4 - 3.8	mayor a 1.4	2.1 - 3.0	3
Caudal de conexiones erradas	0.2l/s.ha máx.	0.2l/s.ha máx.	0.2l/s.ha máx.	0.2l/s.ha
Caudal de infiltración	entre 0.1 y 0.3 l/s.ha	entre 0.1 y 0.3 l/s.ha	entre 0.1 y 0.2	0.2l/s.ha
Caudal aguas resid industriales		1.5 L/s.ha ind		0
Caudal aguas resid comerciales		0.5 L/s.ha		0
Q AASS de uso oficial o especial		0.5 L/s.ha		0
Caudal de diseño sanitario	1.5 l/s mínimo	1.5 l/s mínimo	1.5 l/s mínimo	1.5 l/s mínimo
Tiempo de concentración	3-10 minutos			5
Diámetro mínimo	AASS: 170MM AALL: 260MM	AASS secundaria 170/180mm AASS colectores: 284mm AALL: 215mm	AASS: 200MM AALL: 300MM	AASS: 200mm AALL: 300mm
Factor y/D - Capacidad	AALL: 85% AALL: 93%	D<0.5:70% 0.5<=D<1:80% D>1:85%	AASS: D<0.5:70% 0.5<=D<1:80%	D<0.5:70% 0.5<=D<1:80% D>1:85%

Parámetro	RAS	EPM	EAAB	EciSewer
			D>1:85% AALL: 93%	
Velocidad mínima	AASS: TAO MIN 1.0 Pa AALL: TAO MIN 2.0 Pa	AASS: V>0.45m/s ó TAO 1Pa AASS Industrial: TAO 3Pa AALL: V=0.75m/s ó TAO 3Pa	AASS: TAO MIN 1.5-2 Pa AALL: TAO MIN 3.0 Pa	AASS: TAO MIN 1.5 Pa <450mm TAO MIN 2.0 Pa >450mm AALL: TAO MIN 3.0 Pa
Velocidad máxima	Referencia:5.0 m/s Especial S>30%:10m/s	Concreto, GRP, acero: 5m/s Polietileno y PVC: 10m/s	Concreto: 6m/s - PVC/PEAD: 9m/s Fibra de vidrio/GRP: 4m/s	6 m/s
Manning (n)		Conc-Hierro fundido 0.013 Gres-Acero 0.012 PEAD-GRP-PVC liso 0.010	PVC/PEAD/GRP: 0.010 GRES/CONC PREFAB: 0.013 Concreto/Metálica corrugada: 0.015	0.010 PVC (D<750mm) 0.013 CONC (D>750mm)
Recubrimiento mínimo a clave	V.Peat/Z.Verde: 0.75 Vía vehiculas: 1.20m	1.2m	Vías peatonales o zonas verdes: 0.75 Vías vehiculares: 1.20m	1.2m
Perdida min de energía en pozo	0.02 (m)			0.02 (m)
Diámetros cámaras	200<=D<=500:Dp=1.20 500<D<=750:Dp=1.50 750<D<=900:Dp=1.80 d>900: Diseño estruc	D<595mm 1.2 595<=D<1200 1.5 D>=1200 2 Otros: prefabricadas	1.2	1.2
Cámaras de caída (m)	0.75	0.7	0.75	0.75
Distancia máxima entre pozos	120m			120m
Froude				0.7<F>1.5

Fuente: Elaboración propia

1.9 Ecuación de Autolimpieza

Conociendo el esfuerzo cortante mínimo requerido en la pared de la tubería y combinándolo con la ecuación de Manning se obtiene:

$$Q = \frac{1}{n} \cdot AR^{2/3} \cdot S^{1/2}$$
$$A = \frac{1}{8} \cdot (\theta - \text{sen}\theta)d^2$$
$$R = \frac{d}{4} \cdot \left(1 - \frac{\text{sen}\theta}{\theta}\right)$$
$$\tau = \gamma \cdot R \cdot S$$
$$S = \frac{\tau}{\gamma \cdot \left[\frac{d}{4} \cdot \left(1 - \frac{\text{sen}\theta}{\theta}\right)\right]}$$
$$Q = \frac{1}{n} \cdot \left[\frac{1}{8} \cdot (\theta - \text{sen}\theta)d^2\right] \left[\frac{d}{4} \cdot \left(1 - \frac{\text{sen}\theta}{\theta}\right)\right]^{2/3} \cdot \left[\frac{\tau}{\gamma \cdot \left[\frac{d}{4} \cdot \left(1 - \frac{\text{sen}\theta}{\theta}\right)\right]}\right]^{1/2}$$

Ecuación 17. Ecuación de Autolimpieza

1.10 Consideraciones para flujo uniforme en sistemas de alcantarillado

Si alguna tubería presenta un diámetro nominal igual o superior a 1300 mm, el diseño debe incluir el análisis de flujo no permanente, el software permitirá al usuario exportar la red diseñada para la modelación en otros softwares.

En el caso de requerir un análisis más detallado, el usuario deberá preparar la información requerida por el programa seleccionado (ej: SWMM, SewerGems), como características de la cuenca, datos de infiltración, hidrogramas, series de tiempo de variables físicas, etc.

1.11 Tipos de tuberías y accesorios

Las tuberías a considerar en los diseños, serán de uso comercial en Colombia (Hormigón Reforzado y PVC).

1.12 Consideraciones de estabilidad de flujo uniforme

Cuando el número de Froude se encuentra entre 0.8 y 1.3 aproximadamente, presenta inestabilidad y variabilidad de la profundidad de flujo alrededor de la profundidad crítica de flujo. El software propuesto evita tuberías con números de Froude en este intervalo.

1.13 Restricciones de uso del aplicativo

- El software y los criterios mencionados aplican para flujo uniforme.
- No se tendrán en cuenta estaciones de Bombeo
- No se tendrán en cuenta diseños eléctricos ni de instrumentación
- El usuario define el trazado en planta.
- No se tendrán en cuenta diseños estructurales ni de anclajes por pendientes o cambio de dirección
- No se desarrollarán modelos hidrológicos.
- No se desarrollarán modelos poblacionales. El usuario estima la información poblacional proyectada para los tiempos de retorno requeridos
- Se tendrán en cuenta de forma básica interferencias ingresadas por el usuario

Capítulo II – Metodología

2.1 Desarrollo Computacional

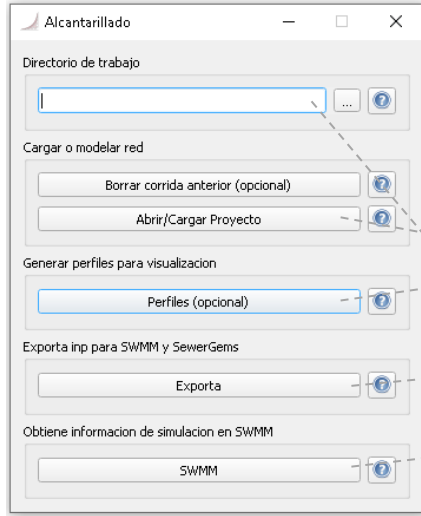


Figura 2. Formulario del aplicativo.

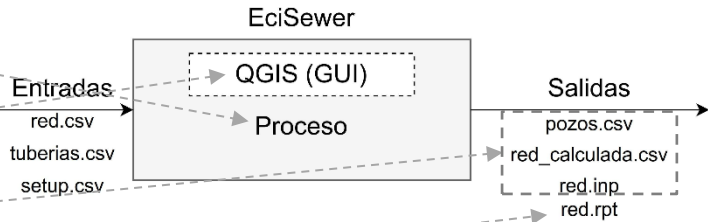


Figura 3. Algoritmo de entrada, proceso y salida.

El programa a partir de la información conocida de la topología en planta de una red de alcantarillado y sus cotas rasantes en los pozos (archivo red.csv), información de tuberías comerciales (tuberías.csv) y valores iniciales y por defecto de variables y restricciones (setup.csv), genera una solución hidráulica tubo a tubo de acuerdo a los criterios de diseño seleccionados en el subcapítulo 1.8 y en la solución de la *Ecuación 17. Ecuación de Autolimpieza*, aplicando el uso de la biblioteca de Python *scipy.optimize*. Esta solución hidráulica se genera para las tuberías comerciales disponibles, para luego seleccionar la de menor costo, mediante dos funciones de costos y un análisis de precios unitarios descritos en el subcapítulo 2.3 *Cálculo de costos*. Las rutinas para el proceso iterativo y de selección de tuberías de menor costo, se describen en 2.1.5.3 *Rutinas para la selección de la tubería – proceso iterativo*.

A partir del diseño en perfil, con el uso del botón “exporta”, se generan los archivos “pozos.csv”, “red_calculada.csv” y “red.inp”. Los dos primeros archivos contienen la topología en planta y perfil, costos calculados y otras variables de interés. El tercer archivo contiene la topología de la red en planta y perfil y los datos necesarios para su importación

en SWMM y en SewerGems. El archivo “red.inp”, no está configurado para la ejecución de la red en flujo gradualmente variado. Se requiere que el usuario ajuste la variación de caudales en función del tiempo al usarse en SWMM o en SewerGems.

El aplicativo permite la posibilidad del uso de la biblioteca SWMM5 para Python. Cuando el usuario usa el botón “SWMM”, se generan los archivos “red.rpt” y “red.dat”. El archivo “red.rpt” contiene el reporte de la simulación de la red con SWMM5.

El formulario del aplicativo (Figura *Formulario del aplicativo*), controla rutinas que esperan la acción del usuario en él.

El botón “Abrir/Cargar proyecto” genera en la interfaz gráfica de usuario de QGIS (GUI), las capas temporales “Pozos” y “Tubos”, donde se muestra la topología en planta de la red.

El botón “Perfiles” genera en la interfaz gráfica de usuario de QGIS, las capas temporales de perfiles de rasante (tuberías y cámaras) del proyecto seleccionado, con una deformación horizontal/vertical de 1 a 5. Todas las capas vectoriales creadas en el QGIS son temporales. QGIS se usa solo como interfaz gráfica y no se requiere guardar ninguna información.

Posteriormente en este capítulo, se describe la información requerida de entrada, variables de interés, archivos de salida, descripción de rutinas del software, detalle del proceso iterativo para la selección de la tubería de menor costo y los prerequisites para la instalación y ejecución del software.

2.1.1 Diagrama de flujo del algoritmo

El diagrama de flujo del algoritmo, muestra la interacción de las principales rutinas del programa. Está organizado de tal forma, que el proceso iterativo de selección de tuberías se muestre en la parte derecha de la figura, cuyas rutinas se describen en *2.1.5.3 Rutinas para la selección de la tubería – proceso iterativo*.

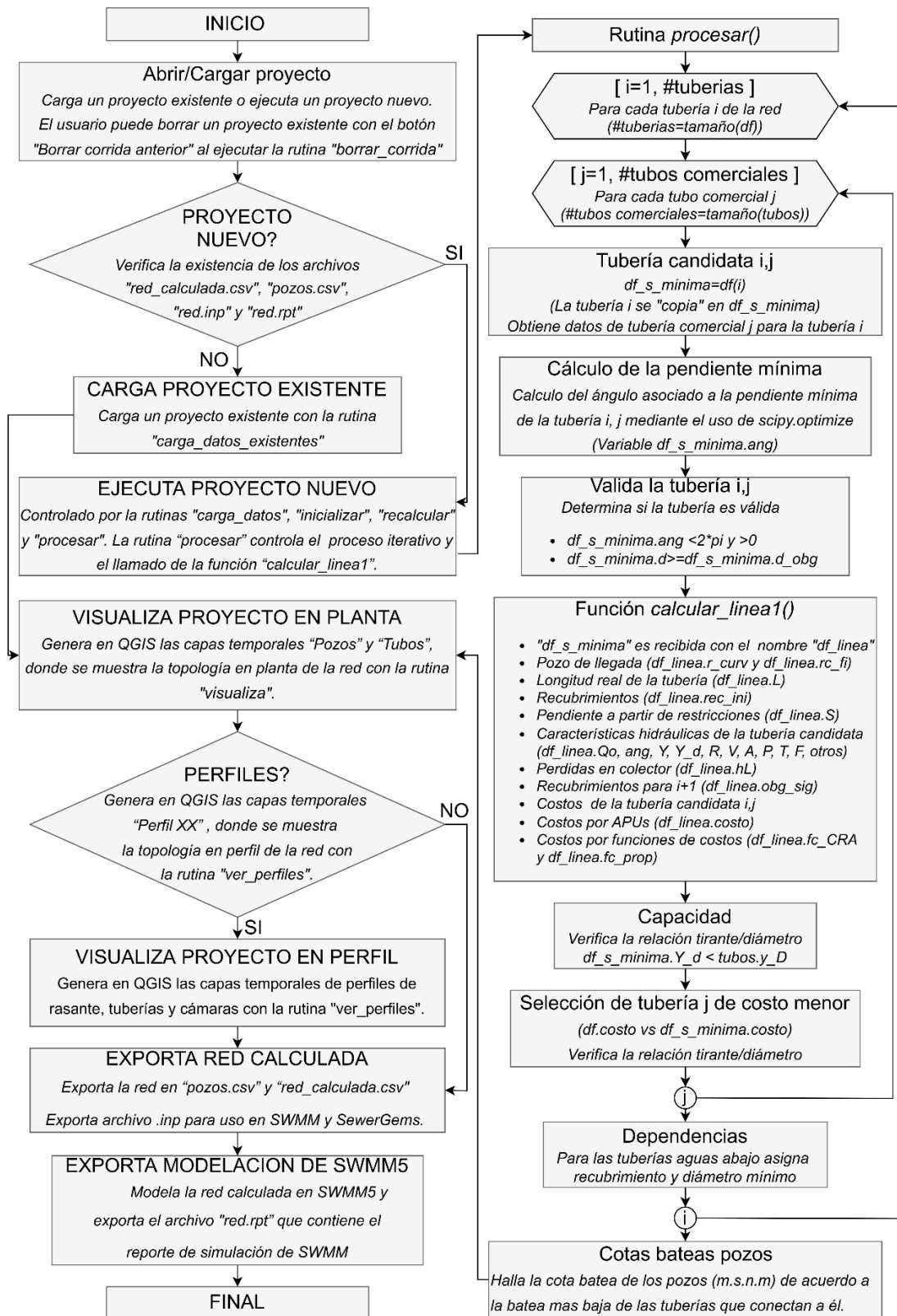


Figura 4. Diagrama de flujo del algoritmo

2.1.2 Archivos de entrada requeridos para la ejecución del aplicativo

Para la ejecución de cada proyecto requiere que el usuario genere tres archivos descritos a continuación:

2.1.2.1 Archivo red.csv

Contiene la topología de la red en planta, información para el cálculo del caudal (áreas) y recubrimientos a clave de tubería requeridos por el usuario cuando son diferentes al recubrimiento mínimo. Las filas de la red están ordenadas de aguas arriba hacia aguas abajo, donde las tuberías afluentes de otras estarán en las primeras filas.

Se muestra el archivo “red.csv” para el caso de estudio “Ejemplo02”, que tiene una estructura como se muestra a continuación:

Tabla 4. Estructura archivo “red.csv”.

num	idd	red	Strahler	tipo	pz_ini	x_ini	y_ini	z_ini	pz_fin	x_fin	y_fin	z_fin	int_ini	int_fin	a_tramo
0	64_14-64_15	1	3	1	64_14			20.1	64_15			19.54	0	0	0.37
1	64_15-64_21	1	3	0	64_15			19.54	64_21			19.3	0	0	0.36
2	64_2-64_3	2	2	1	64_2			32.18	64_3			25.89	0	0	1.51
3	64_3-64_4	2	2	0	64_3			25.89	64_4			22.37	0	0	1.34
4	64_4-64_10	2	2	0	64_4			22.37	64_10			21.16	0	0	0.8
5	64_16-64_17	3	2	1	64_16			31.16	64_17			27.29	0	0	0.17
6	64_17-64_18	3	2	0	64_17			27.29	64_18			21.7	0	0	0.18
7	64_18-64_19	3	2	0	64_18			21.7	64_19			19.6	0	0	0.17
8	64_19-64_21	3	2	0	64_19			19.6	64_21			19.3	0	0	0.17
9	64_21-64_A23	3	2	0	64_21			19.3	64_A23			18.67	0	0	0.35
10	64_5-64_6	4	1	1	64_5			37.31	64_6			33.01	0	0	0.54
11	64_6-64_7	4	1	0	64_6			33.01	64_7			32.33	0	0	0.2
12	64_7-64_8	4	1	0	64_7			32.33	64_8			31.15	0	0	0.2
13	64_8-64_9	4	1	0	64_8			31.15	64_9			30.96	0	0	0.21
14	64_9-64_10	4	1	0	64_9			30.96	64_10			21.16	0	0	0.18
15	64_10-64_11	4	1	0	64_10			21.16	64_11			20.47	0	0	0.79
16	64_11-64_12	4	1	0	64_11			20.47	64_12			19.81	0	0	0.37
17	64_12-64_13	4	1	0	64_12			19.81	64_13			19.36	0	0	0.38
18	64_13-64_22	4	1	0	64_13			19.36	64_22			18.8	0	0	0.47
19	64_22-64_A23	4	1	0	64_22			18.8	64_A23			18.67	0	0	0.68
20	64_A23-AS239	4	1	2	64_A23			18.67	AS_239			18	0	0	0.68

Fuente: Elaboración propia.

Donde,

- num: Numero entero consecutivo (-)
- idd: Texto identificador del tramo (-)
- red: Consecutivo de la red (-)
- Strahler: Orden de la red (-)
- tipo: se refiere a si el pozo es inicial [1], final [2] o intermedio [0] (-)
- pz_ini, pz_fin: Texto identificador del pozo inicial y final(-)
- int_ini, int_fin: Recubrimiento a clave inicial y final de la tubería (m)
- a_tramo: Área de servicio del tramo (ha)

Las redes están ordenadas de tal forma que, las redes ubicadas aguas arriba, están en las primeras filas. La columna *Strahler* corresponde de forma inversa con el número de Strahler-Horton, donde la red principal será de orden 1:

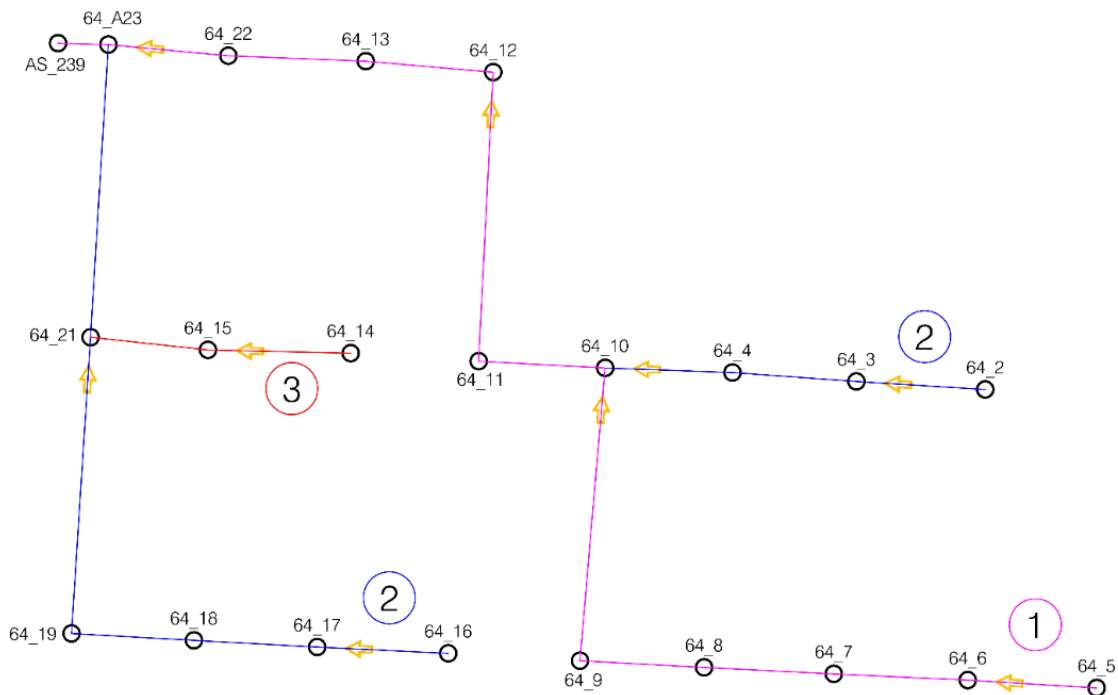


Figura 5. Imagen ilustrativa para la construcción del archivo "red.csv".

Fuente: Elaboración propia.

Las columnas int_ini e int_fin, indican el valor relativo a la rasante de la profundidad mínima a clave de la tubería. Por ejemplo, si el recubrimiento mínimo es de 1.2 metros y la columna de interferencia indica 1.5, la tubería tendrá como mínimo una profundidad a clave de 1.5 metros.

2.1.2.2 Archivo tuberias.csv

Contiene información de las tuberías comerciales seleccionadas como diámetro, material, costo por metro lineal de tubería e instalación, anchos de zanja y restricciones de uso de acuerdo a la norma, de acuerdo a la siguiente tabla. **Error! No se encuentra el origen de la referencia.:**

Tabla 5. Características de las tuberías seleccionadas

d_interno	d externo	n_mann	tao	v_max	y_D	valor_mi	material	d_pozo	valor_inst a	a_zanja
0.145	0.16	0.01	1.5	5	0.7	30635	PVC	1.2	7316	0.4
0.182	0.2	0.01	1.5	5	0.7	43793	PVC	1.2	7316	0.4
0.227	0.25	0.01	1.5	5	0.7	63962	PVC	1.2	7316	0.5
0.284	0.315	0.01	1.5	5	0.7	92037	PVC	1.2	7316	0.5
0.327	0.355	0.01	1.5	5	0.7	137923	PVC	1.2	18183	0.7
0.362	0.4	0.01	1.5	5	0.7	168703	PVC	1.2	18183	0.7
0.407	0.45	0.01	2	5	0.7	217880	PVC	1.2	18183	0.7
0.452	0.5	0.01	2	5	0.7	284051	PVC	1.5	41189	0.7
0.6	0.744	0.013	2	5	0.8	409486	CR	1.5	41428	0.85
0.7	0.866	0.013	2	5	0.8	514044	CR	1.8	51705	0.85
0.8	0.982	0.013	2	5	0.8	641793	CR	1.8	51705	0.85
0.9	1.1	0.013	2	5	0.85	805078	CR	1.8	60946	0.85
1	1.226	0.013	2	5	0.85	1007397	CR	2.2	60946	0.85
1.1	1.352	0.013	2	5	0.85	1128199	CR	2.2	75670	0.85
1.2	1.472	0.013	2	5	0.85	1322371	CR	2.4	75670	0.85
1.3	1.58	0.013	2	5	0.85	1467313	CR	2.4	83134	1
1.4	1.68	0.013	2	5	0.85	1627149	CR	2.6	83134	1
1.5	1.8	0.013	2	5	0.85	1816873	CR	2.6	100102	1
1.6	1.92	0.013	2	5	0.85	2065174	CR	2.8	100102	1
1.7	2.04	0.013	2	5	0.85	2334216	CR	3	126389	1
1.8	2.16	0.013	2	5	0.85	2579118	CR	3	126389	1

d_interno	d_externo	n_mann	tao	v_max	y_D	valor_ml	material	d_pozo	valor_inst a	a_zanja
2	2.4	0.013	2	5	0.85	3243752	CR	3.2	147991	1
2.15	2.57	0.013	2	5	0.85	3704817	CR	3.4	198622	1
2.3	2.74	0.013	2	5	0.85	4169382	CR	3.6	250504	1
2.45	2.946	0.013	2	5	0.85	5306577	CR	3.8	343742	1
2.75	3.34	0.013	2	5	0.85	6545981	CR	4.2	437351	1

Fuente: Elaboración propia

Donde,

d_interno: Diámetro interno de la tubería (m)

d_externo: Diámetro externo de la tubería (m)

n_mann: Coeficiente de rugosidad de Manning de la tubería (-)

tao: Esfuerzo cortante en la pared (Pa - N/m²).

v_max: Velocidad máxima permitida en la tubería (m/s)

y_D: Factor tirante/diámetro - Capacidad de la tubería (%)

valor_ml: Valor por metro lineal de la tubería (\$Col)

material: Material de la tubería (-)

d_pozo: Diámetro del pozo (m)

valor_insta: Valor de instalación de la tubería (\$Col)

a_zanja: Ancho de zanja para la instalación de la tubería (m)

2.1.2.3 Archivo setup.csv

Contiene valores iniciales y por defecto de algunas variables y restricciones del aplicativo, como tipo de red, información para el cálculo de caudales de acuerdo al tipo de red, costos de excavación, y otros valores de variables y restricciones.

Tiene una estructura como se muestra a continuación:

Tabla 6. Estructura archivo "setup.csv".

tipo_red	AASS	Tipo de red
g	9.81	Gravedad (m/s ²)
rho	1000	Densidad del agua (kg/m ³)
fr_sc	0.8	Froude inferior (-)
fr_cr	1.3	Froude superior (-)
delta_S	0.001	Variable auxiliar pendiente (m/m)
d_pz_min	1.2	Diámetro mínimo de pozo
v_ini_asumida	1.5	Velocidad asumida pluvial (m/s)
rec_min	1.2	Recubrimiento mínimo a clave (m)
Q_minimo	1.5	Caudal mínimo (lps)
hL_minimo	0.02	Perdidas mínimas en el colector (m)
mat_cama	0.15	Altura base granular de cama (m)
exc_0_2	3917	Excavación en zanja a máquina en material común, para rangos de alturas entre 0-2m, 2-5m, 5-7m y mayores a 7m(m ³)
exc_2_5	4116	
exc_5_7	4752	
exc_7	5808	
ent_0_3	21412	Entibado continuo en cajones de acero, para rangos de alturas entre 0-3m, 3-4.8m, 4.8-7.2m y mayores a 7.2m (m ²)
ent_3_48	23739	
ent_48_72	25063	
ent_72	30828	
mat_rec	70006	Suministro, transporte, instalación de materiales de relleno: recebo, material de excavación y granular (m ³).
mat_sel	28652	
mat_are	101850	Retiro de material de excavación (m ³)
mat_retiro	2112	Acarreo al sitio de disposición (km)
d_disposicion	6	Coeficientes Información Hidrológica (idf)
C1	2077	
X0	33.89	
C2	1.175	Diámetro mínimo tubería pluvial (m)
d_minimo_AALL	0.284	Tiempo concentración entrada (min)
tc_entrada	5	Coeficiente escorrentía (-)
c_escorrentia	0.6	Diámetro mínimo sanitario (m)
d_minimo_AASS	0.182	Densidad Población Futura (hab/ha)
pob	200	Dotación neta futura (l/hab/día)
dot	120	Factor de Retorno
fR	0.85	Factor Mayoración Caudal
fmQ	1	Tasa Conexiones Erradas (lps/ha)
Qce	0.15	Tasa Infiltración (lps/ha)
Qinf	0.15	Otros caudales de aguas resid (lps/ha)
Qotros	0	Factor de mayoración QMH
F	3	

Fuente: Elaboración propia.

2.1.3 Archivos de salida del aplicativo

- Cuando se genera la ejecución de un proyecto nuevo, y al usar el botón “Exporta”, se generan los archivos “pozos.csv”, “red_calculada.csv” y “red.inp”. Los dos primeros archivos contienen la topología en planta y perfil, costos calculados y otras variables de interés. El tercer archivo contiene la topología de la red en planta y perfil y los datos necesarios para su importación en SWMM y en SewerGems.
- Cuando se genera la ejecución de un proyecto nuevo y el usuario utiliza el botón “SWMM”, se generan los archivos “red.rpt” y “red.dat”. “red.rpt” contiene el reporte que se puede obtener de la simulación de la red de forma independiente en SWMM. Requiere previamente la generación del archivo “red.inp”.

Si el directorio seleccionado en el aplicativo contiene estos archivos de salida, no se resuelve nuevamente la topología en perfil, sino que se carga la información de la ejecución guardada en estos archivos.

2.1.4 Variables de interés

- “df” contiene la información asociada a la tubería y los cálculos realizados en la solución de la red seleccionada. Al ejecutar el aplicativo para un proyecto nuevo, obtiene información de base del archivo “red.csv”.
- “tubos” contiene la información asociada a las tuberías comerciales de referencia. Al iniciar el aplicativo en un proyecto nuevo, obtiene información base del archivo “tuberias.csv” como diámetros, rugosidad, ancho de zanja y costos.
- “setup” contiene la información del archivo “setup.csv”.
- “files_path” guarda la ruta del directorio de la red seleccionada por el usuario.

2.1.5 Descripción de funciones y rutinas

2.1.5.1 Rutinas para la carga/ejecución de proyectos:

- “carga_datos_existentes” lee los archivos “red_calculada.csv”, “tuberias.csv”, “pozos.csv” y “setup.csv” si el directorio seleccionado contiene estos archivos.
- “borrar_corrida”: borra los archivos “pozos.csv”, “red.inp”, “red_calculada.csv”, “pozos_SWMM.csv”, “red.dat”, “red.rpt”, y “red_calculada_SWMM.csv” en caso de ser hallados en el directorio seleccionado por el usuario.
- “carga_datos” lee los archivos “red.csv”, “tuberias.csv” y “setup.csv”, cuando el aplicativo no encuentra archivos de una ejecución previa en el directorio seleccionado. En caso contrario carga los archivos “red_calculada.csv” y “pozos.csv”.
- “inicializar” inicializa las variables con los valores por defecto o de acuerdo a restricciones para cada variable para cada una de las tuberías.

2.1.5.2 Rutinas para la visualización de datos en QGIS

- “visualiza” genera en QGIS las capas temporales “Pozos” y “Tubos”, donde se muestra la topología en planta de la red. El usuario accediendo a la tabla de atributos de las capas, puede visualizar información de algunas variables calculadas en la ejecución del proyecto seleccionado.
- “ver_perfiles” genera en QGIS las capas temporales de perfiles de rasante, tuberías y cámaras del proyecto seleccionado, con una deformación horizontal/vertical de 1 a 5.

2.1.5.3 Rutinas para la selección de la tubería – proceso iterativo

- “procesar” controla el proceso iterativo y el llamado de la función “calcular_linea1” y verifica la validez de la tubería que esta función retorna. Luego se selecciona la tubería válida de menor costo. Adicionalmente genera restricciones para las tuberías aguas abajo que dependan de la tubería calculada.
- “calcular_linea1” para una tubería con información de pendiente, diámetro y caudal, halla las variables hidráulicas descritas en el subcapítulo 1.7 - Criterios de diseño y los costos de tubería e instalación, mediante las dos funciones de costos y el análisis de precios unitarios mencionados en el subcapítulo 2.1.
- “exportar” genera un archivo de extensión .inp, para la importación de la red en los programas SWMM y SewerGems. Adicionalmente exporta el resultado de la modelación en los archivos de texto separados por comas “pozos.csv” y “red_calculada.csv”
- “simula_SWMM” a partir del archivo “red.inp”, ejecuta una simulación básica en flujo permanente, mediante el complemento para Python SWMM5 y exporta los archivos “red.rpt” y “red.dat”. “red.rpt” contiene el reporte de procesamiento simulación de la red de SWMM5.

“procesar” y “calcular_linea1” son las rutinas encargadas de solucionar la topología en perfil de la red y del cálculo de las variables. En estas rutinas se selecciona la tubería comercial válida, con el menor costo calculado y se consolida la selección en la matriz de variables “*df*” de acuerdo a la siguiente secuencia:

- Itera tuberías de la red (i)
 - Itera diámetros disponibles (j)

- Cálculo de la pendiente mínima de la tubería que cumple los requerimientos mínimos de los criterios de diseño y restricciones mediante `scipy.optimize`
 - Verifica la validez de la tubería - Valida si no existe tubería válida para la combinación de caudal y diámetro de acuerdo a la ejecución de la rutina “`calcular_linea1`”, donde se valoran las funciones objetivo y se calcula el costo mediante análisis de precios unitarios
 - Evita flujos con Froude cercano al crítico definidos en las restricciones ingresadas en el archivo “`setup.csv`”.
 - Verifica la relación tirante/diámetro
 - Verifica si el costo calculado para el diámetro(n) es menor que el costo calculado para el diámetro(n-1), y selecciona la tubería con menor costo.
- Obliga un recubrimiento mínimo para las tuberías dependientes
- Terminadas las iteraciones anteriores, halla la cota batea final de los pozos

2.1.5.4 Rutinas para el control de acciones en el formulario del aplicativo

- “`__init__`” define el tamaño y la posición del formulario, y vincula el llamado de rutinas de acuerdo al botón seleccionado por el usuario.
- “`cargar`” ejecuta un proyecto nuevo o carga un proyecto existente si en el directorio seleccionado encuentra los archivos generados en una corrida previa.
- “`borrar`”, “`ver_perfil`”, “`exporta`”, “`corre_SWMM`”, “`ayuda1`”, “`ayuda2`”, “`ayuda3`”, “`ayuda4`”, “`ayuda5`” y “`ayuda6`” controlan la ejecución de rutinas de acuerdo a la selección del usuario.
- “`initGui`”, “`unload`” y “`run`” controlan la visualización o cierre del formulario.

2.1.5.5 Rutinas y funciones auxiliares

- *“func” ejecuta el cálculo del ángulo desde el centro de la sección transversal y los puntos de contacto entre el fluido y la tubería, que cumple con los requerimientos mínimos de autolimpieza de acuerdo a la Ecuación 17.*
- *“func2” ejecuta el cálculo del ángulo desde el centro de la sección transversal y los puntos de contacto entre el fluido y la tubería, cuando la pendiente es mayor a la calculada a la función anterior para cumplir con el recubrimiento mínimo a clave.*
- *“func3” ejecuta el cálculo del ángulo desde el centro de la sección transversal y los puntos de contacto entre el fluido y la tubería en régimen crítico.*
- *“crear_barra_progreso” y “actualiza_barra_progreso” actualizan la barra de progreso cuando se ejecuta un proyecto nuevo.*
- *“deflexion” halla el ángulo de deflexión entre dos tuberías. Usado para hallar una aproximación de las pérdidas de energía del flujo producidas en los pozos.*
- *“recalcular” y “tiempo_concentracion” se ejecutan al recalcular el tiempo de concentración en tuberías de aguas lluvias, ya que se asume una velocidad inicial en la tubería y para los casos ocasionales donde la longitud de la tubería cambia por cambios en el diámetro del pozo de llegada de la tubería. Solo se usan estas rutinas en redes de aguas lluvias.*

2.3 Cálculo de costos

La ecuación para el cálculo del volumen de excavaciones de la instalación de una tubería, donde en perfil se forma un trapecio escaleno es:

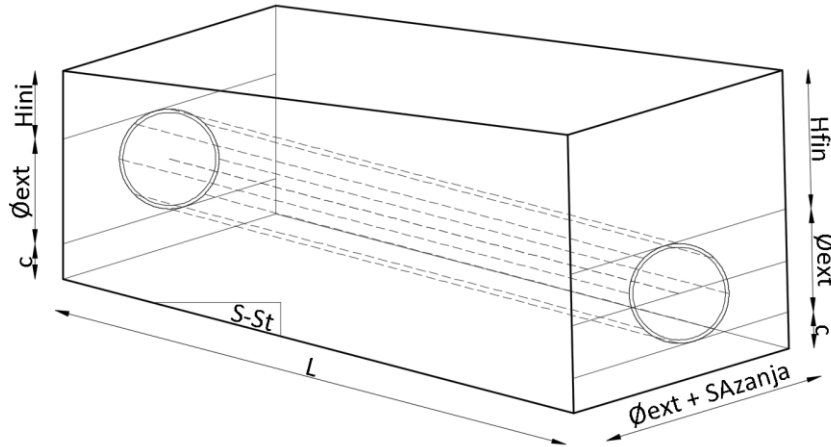


Figura 6. Proyección de la sección del área de excavación.

Fuente: Elaboración propia

$$V = \left(\frac{[Hini + Hfin]}{2} + \text{Øext} + c \right) * (\text{SAzanja} + \text{Øext}) * L$$

Ecuación 18. Volumen del área de excavación

Fuente: Elaboración propia

Donde,

V: Volumen excavado (m³).

Hini: Profundidad de excavación hasta la cota clave aguas arriba de la tubería (m).

Hfin: Profundidad de excavación hasta la cota clave aguas abajo de la tubería (m).

Øext: Diámetro externo de la tubería (m)

c: Relleno que debe disponerse bajo la tubería, comúnmente llamado "material granular de cama" (m)

SAzanja: Sobreecho de excavación. Depende del diámetro de la tubería (m).

L: Longitud de la tubería en el plano horizontal (m).

2.3.1 Análisis de precios unitarios

Se seleccionaron precios unitarios actualizados (03 de noviembre de 2021) de los costos de suministro e instalación de tubería de la prestadora de servicios públicos de acueducto y alcantarillado de la ciudad de Bogotá (EAAB):

Tabla 7. Precios unitarios de referencia

ITEM	DESCRIPCIÓN	U	VU (\$Col)
104001	Excavaciones		
104.001.002	Excavaciones en material común a máquina		
104.001.002.001	Excavación En Zanja a máquina en material común h <2 m	M3	\$ 3,917
104.001.002.002	Excavación En Zanja a máquina en material común 2 <= h	M3	\$ 4,116
104.001.002.003	Excavación En Zanja a máquina en material común 5 < h	M3	\$ 4,752
104.001.002.004	Excavación En Zanja a máquina en material común h > 7	M3	\$ 5,808
105	ENTIBADOS TABLAESTACA Y CORTINAS PILOTES		
105.001	Entibado		
105.001.009	Entibado continuo en cajones de aluminio H<=2,00 M	M2	\$ 18,494
105.001.010	Entibado continuo en cajones de acero H<=3,00 M	M2	\$ 21,412
105.001.011	Entibado deslizante 3,00<H<=4,80 M	M2	\$ 23,739
105.001.012	Entibado deslizante 4,80<H<=7,20 M	M2	\$ 25,063
105.001.013	Entibado deslizante H>7,20 M	M2	\$ 30,828
106	RELLENOS		
106.001	Suministro, transporte e instalación Recebo	M3	\$ 70,006
106.002	Suministro, transporte e instalación material seleccionado	M3	\$ 28,652
106.006.001	Suministro, transporte e instalación de Arena de Peña	M3	\$ 101,850
107	RETIRO Y DISPOSIC. MATERIALES SOBRANTES		
107.001	Retiro y disposic. materiales sobrantes excav y demol	m3k	\$ 2,112
109	INSTALACIÓN TUBERÍAS		
109.001	Instalación Tuberías alcantarillado		
109.001.001	Instalación tub flexibles alcantarillado		
109.001.001.001	Inst tub flexibles alcant Dn6,8,10,12"	M	\$ 7,316
109.001.001.002	Inst tub flexibles alcant Dn14,16,18"	M	\$ 18,183
109.001.001.003	Inst tub flexibles alcant Dn20 y 24"	M	\$ 41,189
109.001.001.004	Inst tub flexibles alcant Dn27 y 30"	M	\$ 49,545
109.001.001.005	Inst tub flexibles alcant, Dn36"-1.00m	M	\$ 64,746
109.001.001.006	Inst tub flexibles alcant Dn1.10 y 1.20m	M	\$ 75,400
109.001.001.007	Inst tub flexibles alcant Dn1.30 y 1.40m	M	\$ 78,791
109.001.001.008	Inst tub flexibles alcant Dn1.50 y 1.60m	M	\$ 92,463
109.001.001.009	Inst tub flexibles alcant Dn1.70 y 1.80m	M	\$ 101,852
109.001.001.010	Inst tub flexibles alcant Dn 2.00m	M	\$ 109,854
109.001.001.011	Inst tub flexibles alcant Dn 2.15m	M	\$ 131,976
109.001.001.012	Inst tub flexibles alcant Dn 2.30m	M	\$ 146,362
109.001.001.013	Inst tub flexibles alcant Dn 2.45m	M	\$ 158,253
109.001.001.014	Inst tub flexibles alcant Dn 2.75m	M	\$ 165,874
109.001.001.015	Inst tub flexibles alcant Dn1.90m	M	\$ 105,408
109.001.001.016	Inst tub flexibles alcant Dn2.20m	M	\$ 139,859
109.001.001.017	Inst tub flexibles alcant Dn2.50m	M	\$ 163,722
109.001.001.018	Inst tub flexibles alcant Dn2.60m	M	\$ 167,775
109.001.002	Instalación tub. Rígidas alcantarillado		
109.001.002.001	Inst tub rígidas alcant Dn6, 8, 10 y 12"	M	\$ 27,644
109.001.002.002	Inst tub rígidas alcant Dn14, 16 y 18"	M	\$ 34,011
109.001.002.003	Inst tub rígidas alcant Dn20 y 24"	M	\$ 41,428
109.001.002.004	Inst tub rígidas alcant Dn27 y 30"	M	\$ 51,705
109.001.002.005	Inst tub rígidas alcant Dn 36" - 1.00 m	M	\$ 60,946
109.001.002.006	Inst tub rígidas alcant Dn1.10 y 1.20m	M	\$ 75,670
109.001.002.007	Inst tub rígidas alcant Dn1.30 y 1.40m	M	\$ 83,134
109.001.002.008	Inst tub rígidas alcant Dn1.50 y 1.60m	M	\$ 100,102
109.001.002.009	Inst tub rígidas alcant Dn1.70 y 1.80m	M	\$ 126,389
109.001.002.010	Inst tub rígidas alcant Dn 2.00m	M	\$ 147,991

ITEM	DESCRIPCIÓN	U	VU (\$Col)
109.001.002.011	Inst tub rígidas alcant Dn 2.15m	M	\$ 198,622
109.001.002.012	Inst tub rígidas alcant Dn 2.30m	M	\$ 250,504
109.001.002.013	Inst tub rígidas alcant Dn 2.45m	M	\$ 343,742
109.001.002.014	Inst tub rígidas alcant Dn 2.75m	M	\$ 437,351
903.003	SUMINISTRO TUBERÍA PARA ALCANTARILLADO		
Tubería de concreto reforzado con varillas de acero			
903.003.001.022	Tubería CR,alcant., clase II,D 24"	M	\$ 409,486
903.003.001.023	Tubería CR,alcant., clase II,D 27"	M	\$ 514,044
903.003.001.024	Tubería CR,alcant., clase II,D 30"	M	\$ 641,793
903.003.001.026	Tubería CR,alcant., clase II,D 36"	M	\$ 805,078
903.003.001.027	Tubería CR,alcant., clase II,D 1.00 m	M	\$ 1,007,397
903.003.001.028	Tubería CR,alcant., clase II,D 1.10 m	M	\$ 1,128,199
903.003.001.029	Tubería CR,alcant., clase II,D 1.20 m	M	\$ 1,322,371
903.003.001.030	Tubería CR,alcant., clase II,D 1.30 m	M	\$ 1,467,313
903.003.001.031	Tubería CR,alcant., clase II,D 1.40 m	M	\$ 1,627,149
903.003.001.032	Tubería CR,alcant., clase II,D 1.50 m	M	\$ 1,816,873
903.003.001.033	Tubería CR,alcant., clase II,D 1.60 m	M	\$ 2,065,174
903.003.001.034	Tubería CR,alcant., clase II,D 1.70 m	M	\$ 2,334,216
903.003.001.035	Tubería CR,alcant., clase II,D 1.80 m	M	\$ 2,579,118
903.003.001.036	Tubería CR,alcant., clase II,D 2.00 m	M	\$ 3,243,752
903.003.001.037	Tubería CR,alcant., clase II,D 2.15 m	M	\$ 3,704,817
903.003.001.038	Tubería CR,alcant., clase II,D 2.30 m	M	\$ 4,169,382
903.003.001.039	Tubería CR,alcant., clase II,D 2.45 m	M	\$ 5,306,577
903.003.001.040	Tubería CR,alcant., clase II,D 2.75 m	M	\$ 6,545,981
Tubería de PVC pared interior lisa y exterior corrugada - Rigidez mínima de 57 PSI			
903.003.005.001	Tub PVCcorrú ext,lisa int,PS57psi,D160mm	M	\$ 30,635
903.003.005.002	Tub PVCcorrú ext,lisa int,PS57psi,D200mm	M	\$ 43,793
903.003.005.003	Tub PVCcorrú ext,lisa int,PS57psi,D250mm	M	\$ 63,962
903.003.005.004	Tub PVCcorrú ext,lisa int,PS57psi,D315mm	M	\$ 92,037
903.003.005.008	Tub PVCcorrú ext,lisa int,PS57psi,D355mm	M	\$ 137,923
903.003.005.005	Tub PVCcorrú ext,lisa int,PS57psi,D400mm	M	\$ 168,703
903.003.005.006	Tub PVCcorrú ext,lisa int,PS57psi,D450mm	M	\$ 217,880
903.003.005.007	Tub PVCcorrú ext,lisa int,PS57psi,D500mm	M	\$ 284,051
Tubería de PVC pared interior lisa y exterior corrugada - Rigidez mínima de 28 PSI			
903.003.012.002	Tub PVCcorrú ext,lisa int,PS28psi,D27"	M	\$ 520,605
903.003.012.003	Tub PVCcorrú ext,lisa int,PS28psi,D30"	M	\$ 669,672
903.003.012.004	Tub PVCcorrú ext,lisa int,PS28psi,D33"	M	\$ 891,062
903.003.012.005	Tub PVCcorrú ext,lisa int,PS28psi,D36"	M	\$ 1,242,997
903.003.012.006	Tub PVCcorrú ext,lisa int,PS28psi,D39"	M	\$ 1,663,077
903.003.012.007	Tub PVCcorrú ext,lisa int,PS28psi,D42"	M	\$ 1,949,290
Tubería de PVC de perfil abierto			
903.003.006.022	Tub PVC Perfil Abierto,PS46,D600 mm	M	\$ 566,206
903.003.006.023	Tub PVC Perfil Abierto,PS46,D630 mm	M	\$ 627,974
903.003.006.024	Tub PVC Perfil Abierto,PS46,D675 mm	M	\$ 679,933
903.003.006.025	Tub PVC Perfil Abierto,PS46,D700 mm	M	\$ 755,683
903.003.006.026	Tub PVC Perfil Abierto,PS46,D725 mm	M	\$ 800,899
903.003.006.027	Tub PVC Perfil Abierto,PS46,D800 mm	M	\$ 976,069
903.003.006.028	Tub PVC Perfil Abierto,PS46,D900 mm	M	\$ 1,454,529
903.003.006.029	Tub PVC Perfil Abierto,PS46,D1000 mm	M	\$ 1,879,750
903.003.006.030	Tub PVC Perfil Abierto,PS46,D1200 mm	M	\$ 3,031,084
903.003.006.031	Tub PVC Perfil Abierto,PS46,D1250 mm	M	\$ 3,135,107
903.003.006.032	Tub PVC Perfil Abierto,PS46,D1400 mm	M	\$ 3,515,105
903.003.006.033	Tub PVC Perfil Abierto,PS46,D1500 mm	M	\$ 3,956,298

ITEM	DESCRIPCIÓN	U	VU (\$Col)
Tubería de PVC de perfil cerrado			
903.003.007.014	Tub PVC Perfil Cerrado PS28, D24"	M	\$ 448,179
903.003.007.015	Tub PVC Perfil Cerrado PS28, D27"	M	\$ 520,606
903.003.007.016	Tub PVC Perfil Cerrado PS28, D30"	M	\$ 669,672
903.003.007.017	Tub PVC Perfil Cerrado PS28, D33"	M	\$ 891,062
903.003.007.018	Tub PVC Perfil Cerrado PS28, D36"	M	\$ 1,242,997
903.003.007.019	Tub PVC Perfil Cerrado PS28, D39"	M	\$ 1,663,077
903.003.007.020	Tub PVC Perfil Cerrado PS28, D42"	M	\$ 1,949,290
903.003.007.021	Tuberia PVC Perfil Cerrado PS28, D45"	M	\$ 1,924,739
903.003.007.022	Tuberia PVC Perfil Cerrado PS28, D48"	M	\$ 2,521,536
903.003.007.023	Tuberia PVC Perfil Cerrado PS28, D51"	M	\$ 2,799,972
903.003.007.024	Tuberia PVC Perfil Cerrado PS28, D54"	M	\$ 2,947,636
903.003.007.025	Tuberia PVC Perfil Cerrado PS28, D57"	M	\$ 3,078,057
903.003.007.026	Tuberia PVC Perfil Cerrado PS28, D60"	M	\$ 3,486,562

Fuente: Empresa de acueducto y alcantarillado de Bogotá

Al tener varias posibilidades de tipos de tuberías para el mismo diámetro, se seleccionan los más económicos de acuerdo a la siguiente tabla

Tabla 8. Precios unitarios seleccionados para tuberías

Material	Nominal	Interno	Externo	espesor	Valor
PVC	160mm	0.145	0.160	0.0075	\$ 30,635
PVC	200mm	0.182	0.200	0.0090	\$ 43,793
PVC	250mm	0.227	0.250	0.0115	\$ 63,962
PVC	315mm	0.284	0.315	0.0155	\$ 92,037
PVC	355mm	0.327	0.355	0.0140	\$ 137,923
PVC	400mm	0.362	0.400	0.0190	\$ 168,703
PVC	450mm	0.407	0.450	0.0215	\$ 217,880
PVC	500mm	0.452	0.500	0.0240	\$ 284,051
CR	24"	0.600	0.744	0.0720	\$ 409,486
CR	27"	0.700	0.866	0.0830	\$ 514,044
CR	10"	0.800	0.982	0.0910	\$ 641,793
CR	12"	0.900	1.100	0.1000	\$ 805,078
CR	14"	1.000	1.226	0.1130	\$ 1,007,397
CR	16"	1.100	1.352	0.1260	\$ 1,128,199
CR	18"	1.200	1.472	0.1360	\$ 1,322,371
CR	20"	1.300	1.580	0.1400	\$ 1,467,313
CR	24"	1.400	1.680	0.1400	\$ 1,627,149
CR	27"	1.500	1.800	0.1500	\$ 1,816,873
CR	30"	1.600	1.920	0.1600	\$ 2,065,174
CR	36"	1.700	2.040	0.1700	\$ 2,334,216
CR	40"	1.800	2.160	0.1800	\$ 2,579,118
CR	44"	2.000	2.400	0.2000	\$ 3,243,752
CR	48"	2.150	2.570	0.2100	\$ 3,704,817
CR	52"	2.300	2.740	0.2200	\$ 4,169,382
CR	56"	2.450	2.946	0.2480	\$ 5,306,577
CR	60"	2.750	3.340	0.2950	\$ 6,545,981

Fuente: Elaboración propia

2.3.2 Función de costos CRA

A partir de una base de datos de 224 proyectos de alcantarillado, el documento propone funciones de tipo potencial de la forma $C = Ax^b$, donde, por ejemplo, presenta los coeficientes asociados al costo de la tubería respecto a su diámetro de la siguiente forma:

Tabla 9. Resumen de funciones (costos en pesos de diciembre de 2007)

Componente	Variable	Un	Un costo	Coef cen	Exp cen
Recolección (10%)	Diámetro	m	Col\$/m	9579.31	0.5737
Recolección (25%)	Diámetro	m	Col\$/m	9579.31	0.5737
Recolección (50%)	Diámetro	m	Col\$/m	9579.31	0.5737

Fuente: Comisión de Regulación de Agua potable y saneamiento básico – CRA. Revista No.15 agosto 2009.

$$C = Ax^b$$

Ecuación 19. Ecuación potencial

Donde,

C: Costo del componente (Costo total o costo unitario)

A: Coeficiente de la función potencial

b: Exponente de la función potencial

x: Variable física (caudal, diámetro, volumen, etc.)

Costos de las tuberías:

$$C = 1.5302 * 9579.31 * d^{0.5737}$$

Ecuación 20. Ecuación Costos de las tuberías CRA

$$C = 1.5302 * 1163.77 * V^{1.31}$$

Ecuación 21. Ecuación Costos de excavación CRA

Donde,

C: Costo por metro lineal de tubería a mayo de 2009(\$Cop)

d: Diámetro de la tubería en milímetros (mm)

V: Volumen de excavación por tubería (m3)

1.532: Factor de conversión de pesos de diciembre de 2007 a noviembre de 2021 (-)

2.3.3 Función de costos propia

A partir de los casos de estudio descritos en el capítulo 3 “Resultados y contribución”, se proponen funciones de tipo potencial de la forma $C = Ax^b$, donde los coeficientes asociados al costo de la tubería respecto a su diámetro y volumen de excavación son:

$$C = 1004861.4272 * d^{1.7928}$$

Ecuación 22. Ecuación Costos de las tuberías por metro lineal

Fuente: Elaboración propia

$$C = 43644.8004 * V^{0.9583}$$

Ecuación 23. Ecuación Costos de excavación por metro lineal

Fuente: Elaboración propia

Donde,

C: Costo por metro lineal de tubería [COP/m].

d: Diámetro de la tubería en milímetros [m].

V: Volumen de excavación por tubería [m3].

La función de costos propuesta, se obtuvo mediante la generación de un código denominado “AplicativoFuncionCostos”. Se presenta la visualización de los resultados obtenidos en el ajuste de la curva potencial:

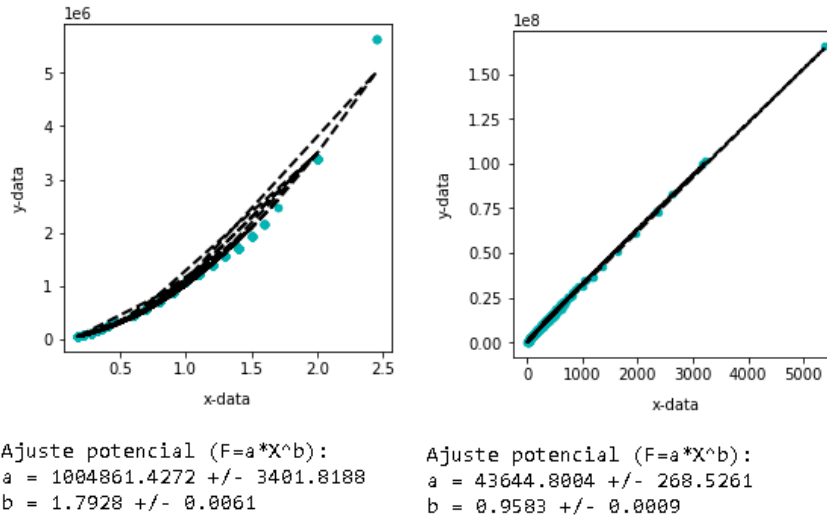


Figura 7. Resultados ajuste función de costos propia.

Fuente: Elaboración propia

2.4 Prerrequisitos para el uso del aplicativo

2.4.1 Instalación de QGIS

Se recomienda descargar el instalador autónomo de la versión 3.16 con soporte a largo plazo en el link: <https://qgis.org/es/site/forusers/download.html>



Otras versiones de QGIS con versiones de Python superiores o iguales a 3.0 deberían funcionar sin problemas. El software se probó en la versión 3.20.

2.4.2 Instalación de SWMM

Se recomienda descargar el instalador de la versión 5.1.015 en el link: <https://www.epa.gov/water-research/storm-water-management-model-swmm>



Versiones 5.XX de SWMM publicadas hasta 2021, funcionan sin problemas de compatibilidad.

2.4.3 Instalación de Anaconda

Se incluye una versión compacta del aplicativo para ser ejecutado en Anaconda-Jupyter, sin las funcionalidades de QGIS, en el archivo “Aplicativo para jupyter.ipynb”. La instalación del software Anaconda es opcional.

Se recomienda descargar el instalador de la versión 3.9 o superior en el link: <https://www.anaconda.com/products/individual>

Anaconda Individual Edition

Download 

For Windows

Python 3.9 • 64-Bit Graphical Installer • 510 MB

Versiones 3.XX deberían funcionar sin problemas de uso.

2.4.4 Instalación de SewerGems

Si se cuenta con una licencia de este programa se puede importar la red obtenida del programa propuesto de manera similar a como se indica para ser importado en SWMM. La instalación del software es opcional.

2.4.5 Instalación de SWMM5 para Python

2.4.5.1 En QGIS 3 para Windows

Abra como administrador el shell de OSGeo4W en el menú de inicio:

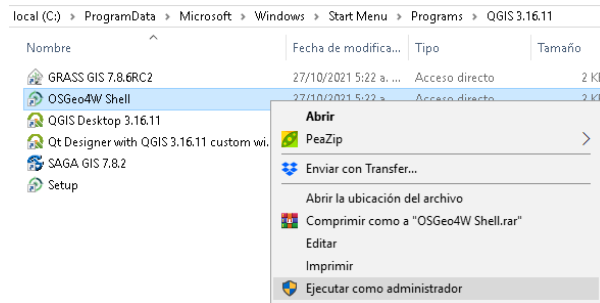


Figura 8. Instalación de SWMM5 para Python en QGIS – paso 1.

Ejecute el comando: `python -m pip install SWMM5`:

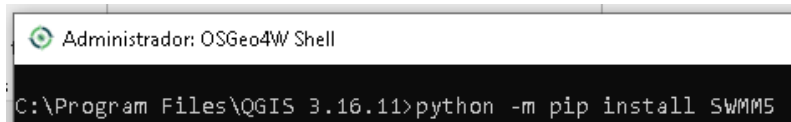


Figura 9. Instalación de SWMM5 para Python en QGIS – paso 2.

2.4.5.2 En Anaconda para Windows

Abra el Prompt de Anaconda en el menú de inicio como administrador

Ejecute el comando: `pip install SWMM5`

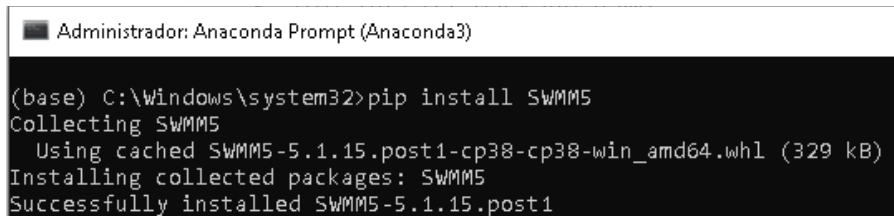


Figura 10. Instalación de SWMM5 para Anaconda.

2.4.6 Instalación del aplicativo en QGIS

Ejecute QGIS. El menú “Settings -> User Profiles -> Open Active Profile folder”, ubica la carpeta de perfil de usuario de QGIS (es diferente para cada PC, de acuerdo a la versión de Windows y el usuario que instala el programa):

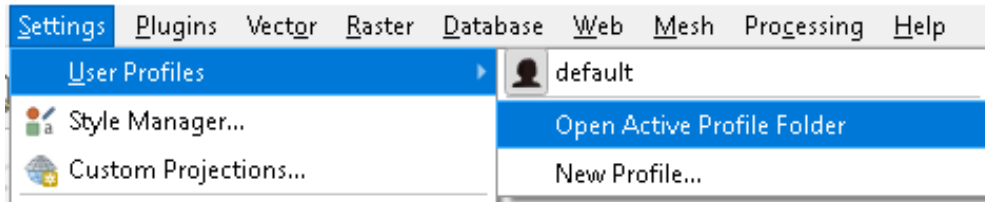


Figura 11. Instalación del aplicativo en QGIS – ruta de usuario.

Se accede a la carpeta Python, luego a la carpeta plugins y se pega la carpeta “sewer_plugin” anexa y se reinicia QGIS

El software estará instalado, se puede verificar en el menú “plugins -> Installed Plugins”

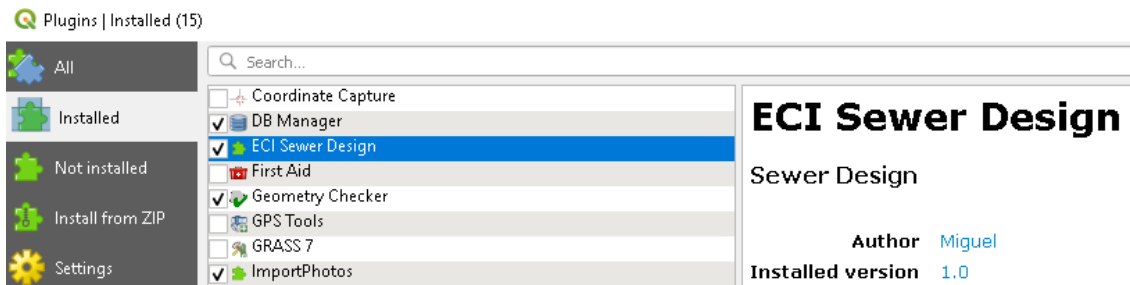


Figura 12. Instalación del aplicativo en QGIS – Menú de plugins de QGIS.

Aparecerá un icono personalizado en la barra de iconos de acceso rápido del programa:

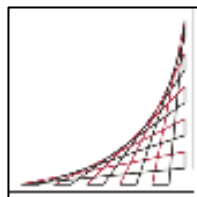
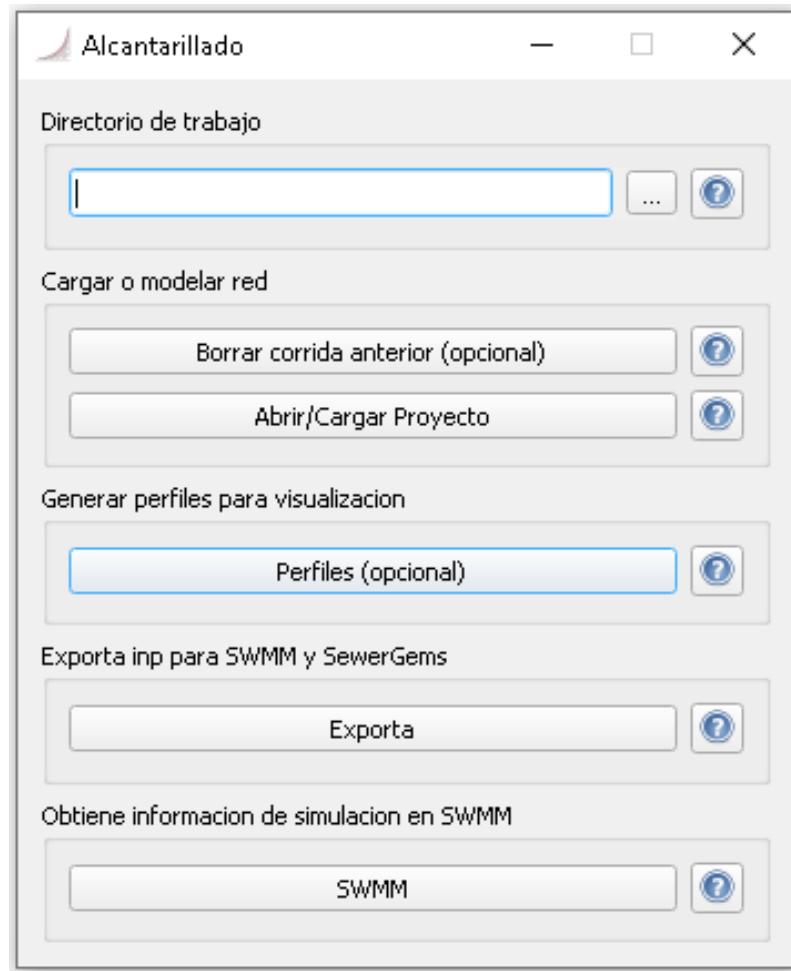


Figura 13. Instalación del aplicativo en QGIS – Ícono del aplicativo.

2.4.7 Ejecución del aplicativo en QGIS

El aplicativo se inicia picando el icono en QGIS. Se desplegará el siguiente formulario:



Alcantarillado

Directorio de trabajo

Cargar o modelar red

Generar perfiles para visualizacion

Exporta inp para SWMM y SewerGems

Obtiene informacion de simulacion en SWMM

Figura 14. Instalación del aplicativo en QGIS – Formulario del aplicativo.

2.5 Presentación de los perfiles:

La visualización de los perfiles tiene diferentes aspectos de uso respecto al programa usado.

La generación de perfiles en QGIS por el aplicativo es muy rápida, dibuja todos los perfiles y solo depende del archivo “red_calculada.csv” de cada red. Presenta como desventajas que no tiene un formato con grilla acotada en ninguno de los dos ejes y que depende de la escala seleccionada de la interfaz gráfica, para la correcta visualización de las etiquetas de las cadenas de texto identificadores de los pozos.

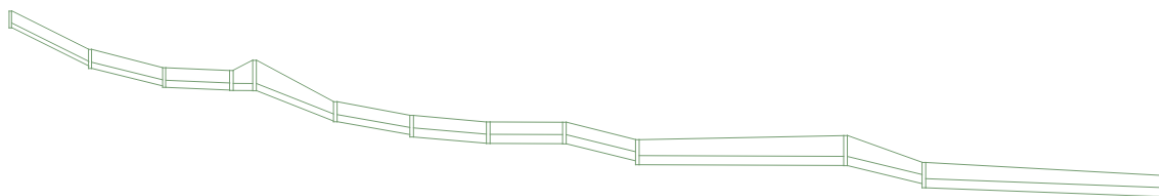


Figura 15. Ejemplo de vista con zoom de escala grande en QGIS

Fuente: Elaboración propia.

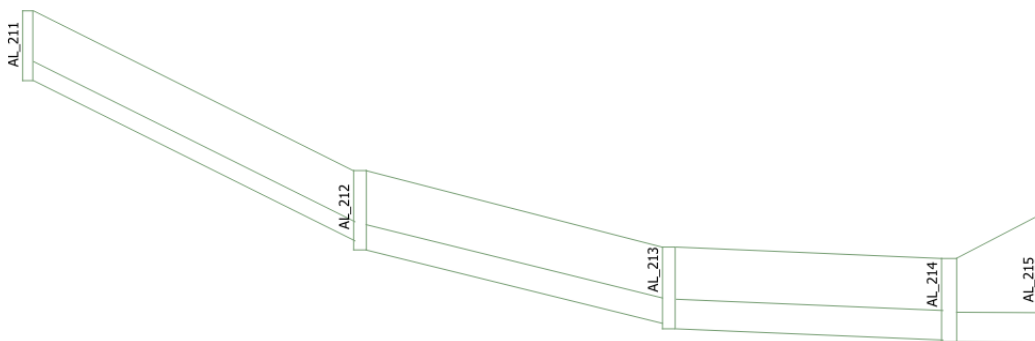


Figura 16. Ejemplo de vista con zoom de escala pequeña en QGIS

Fuente: Elaboración propia.

La generación de perfiles en SWMM y en SewerGems, depende únicamente de la importación del archivo “red.inp”. Requiere la ejecución de la modelación de la red importada y de la generación de los perfiles. En SewerGems la generación de los perfiles es casi automatizada, mientras que en SWMM se requiere indicar el pozo inicial y final de la red a dibujar en perfil. Presenta como ventajas: un formato con grilla donde se muestra información relevante de la red (cota, abscisa), y la posibilidad de ocultar o mostrar información. Las etiquetas de texto del perfil se ven correctamente en escalas pequeñas.

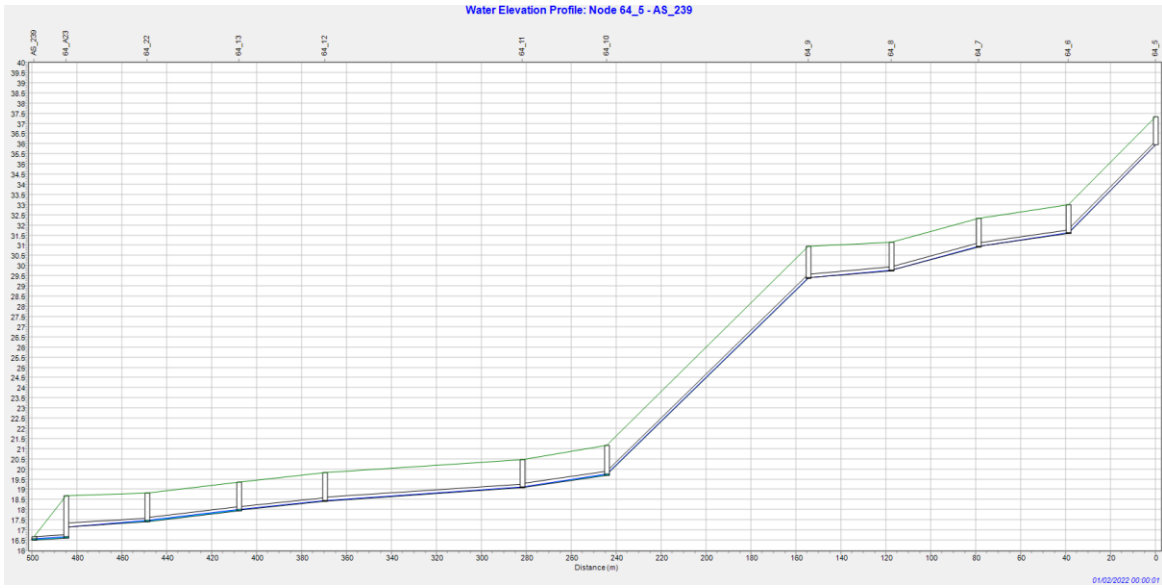


Figura 17. Vista del perfil de la red “Ejemplo02” en SWMM:

Fuente: Elaboración propia.

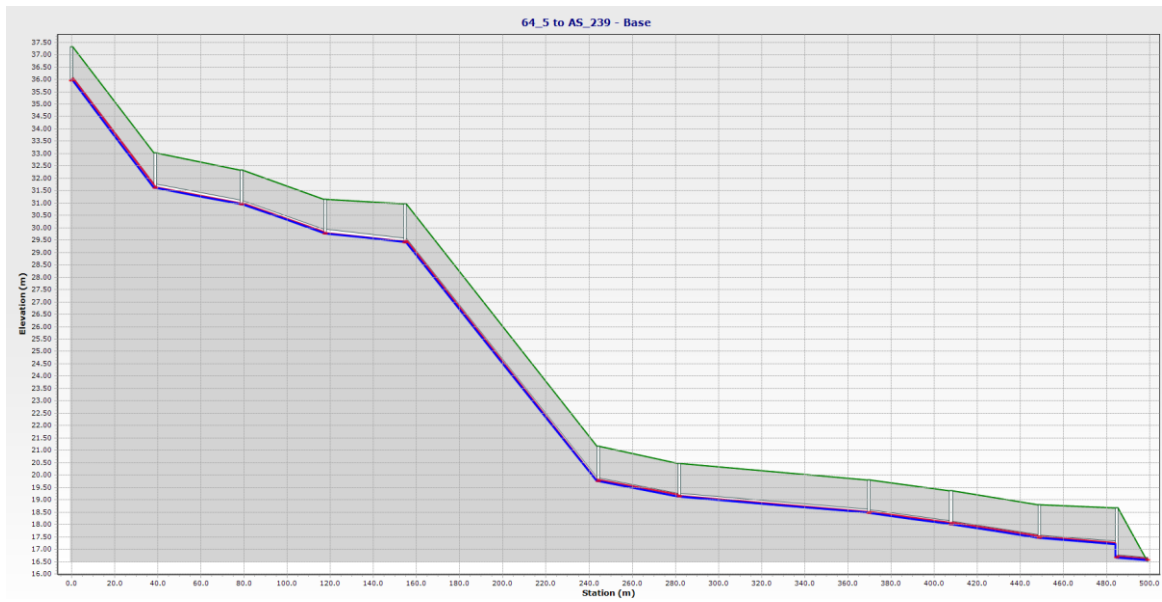


Figura 18. Vista del perfil de la red “Ejemplo02” en SewerGems:

Fuente: Elaboración propia.

Capítulo III - Resultados y contribución

3.1 Casos de estudio

Como se menciona en el subcapítulo 2.3.3, se cuentan con 5 topologías en planta que incluyen la cota rasante de los pozos. La construcción de la obra civil de estas redes de alcantarillado, fueron ejecutadas en la zona de Flor de Bastión de la Ciudad de Guayaquil, entre los años 2013 y 2015.

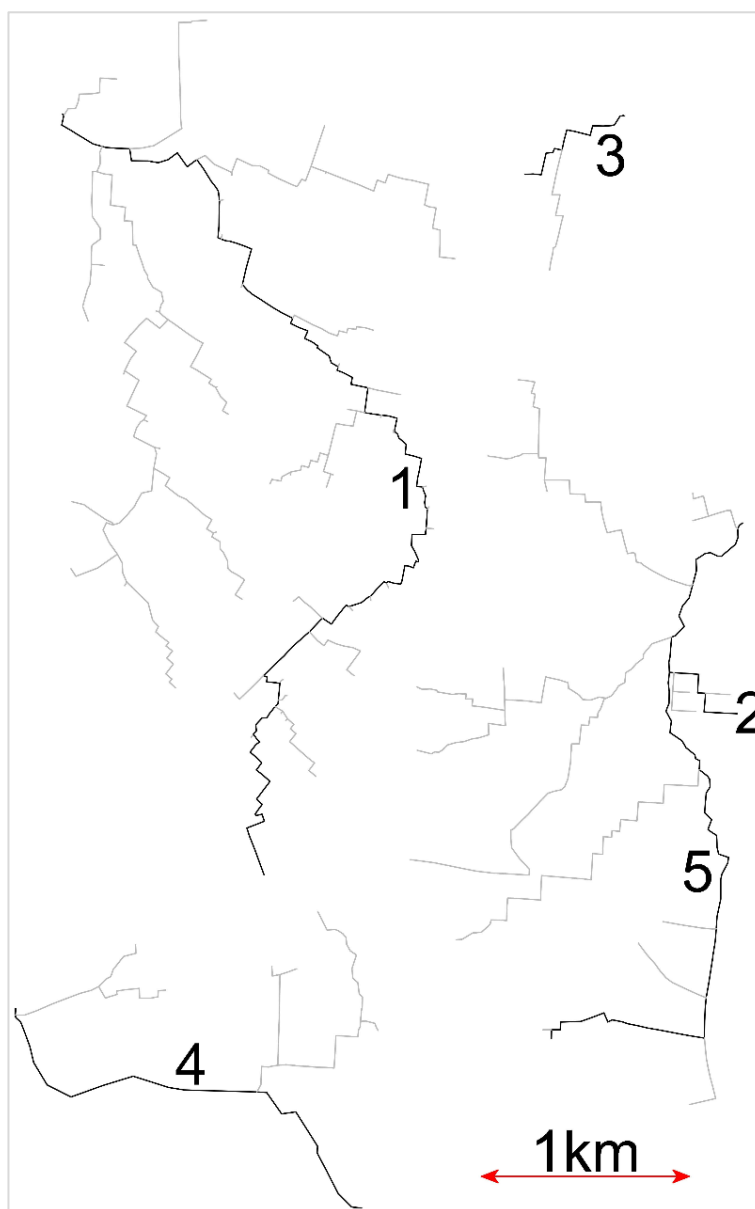


Figura 19. Esquema en planta de los casos de estudio

Fuente: Elaboración propia.

A continuación, se presenta un resumen de las características de las redes:

Tabla 10. Descripción de casos de estudio

Red	Ejemplo01	Ejemplo02	Ejemplo03	Ejemplo04	Ejemplo05
Pozos	474	22	32	105	323
Tramos	473	21	31	104	322
Longitud	19447.53	1006.15	1384.67	5663.94	13446.95
Tipo	AASS	AASS	AALL	AALL	AALL
Área (ha)	637.24	10.12	51.91	541.08	638.04

Fuente: Elaboración propia.

3.1.1 Caso de estudio 1 - Ejemplo02

Pequeña red sanitaria con 1 kilómetro de tuberías. Se muestra la planta de la red entre los pozos 64_14 y 64_21 graficada desde SewerGems y el perfil diseñado de la red entre los pozos 64_5 y AS_239, graficado desde SWMM:

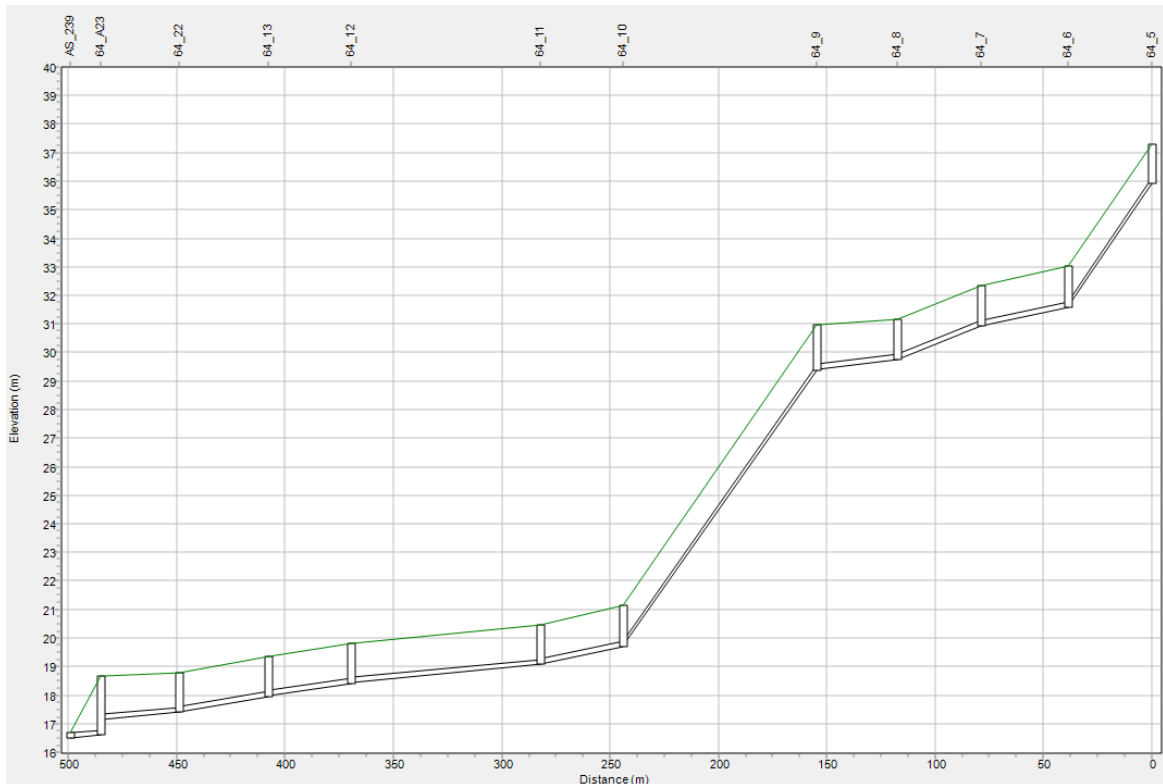


Figura 20. Caso de estudio 1 - Perfil diseñado entre los pozos 64_5 y AS239

Fuente: Elaboración propia.

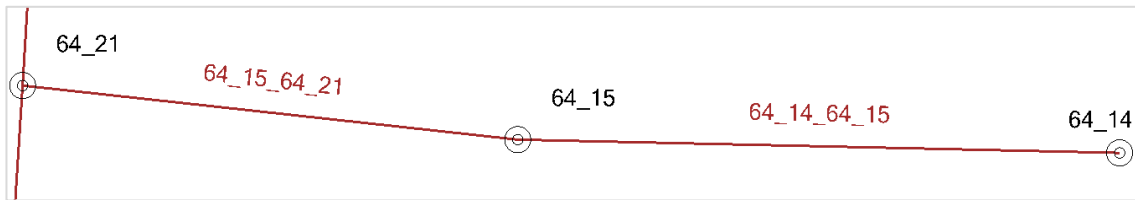


Figura 21. Caso de estudio 1 - Planta de la red entre los pozos 64_14 y 64_21

Fuente: Elaboración propia.

A continuación, se presenta un resumen de información de interés de la red diseñada:

Tabla 11. Caso de estudio 1 - Resumen de información de interés de la red diseñada.

Caso de estudio	1
Directorio	Ejemplo02
Tipo de red	Sanitaria
Área (ha)	10.12
Caudal máximo (lps)	10.20
Pendiente máxima de tubería (m/m)	16.3%
Pendiente mínima de tubería (m/m)	0.7%
Factor Y/d máximo (%)	41.58%
Factor Y/d mínimo (%)	7.50%
Velocidad máxima del fluido (m/s)	1.72
Velocidad mínima del fluido (m/s)	0.08
Tirante máximo (cm)	7.57
Diámetro máximo (pg)	8
Diámetro mínimo (pg)	8
Altura promedio de excavación (m)	1.60
Volumen de excavación (m3)	942
Costo (APU's)	\$ 89,463,971

Fuente: Elaboración propia.

3.1.2 Caso de estudio 2 - Ejemplo03

Pequeña red pluvial con 1.4 kilómetros de tuberías. Se muestra el perfil diseñado de la red entre los pozos 64_5 y AS_239, graficado desde QGIS por el aplicativo:

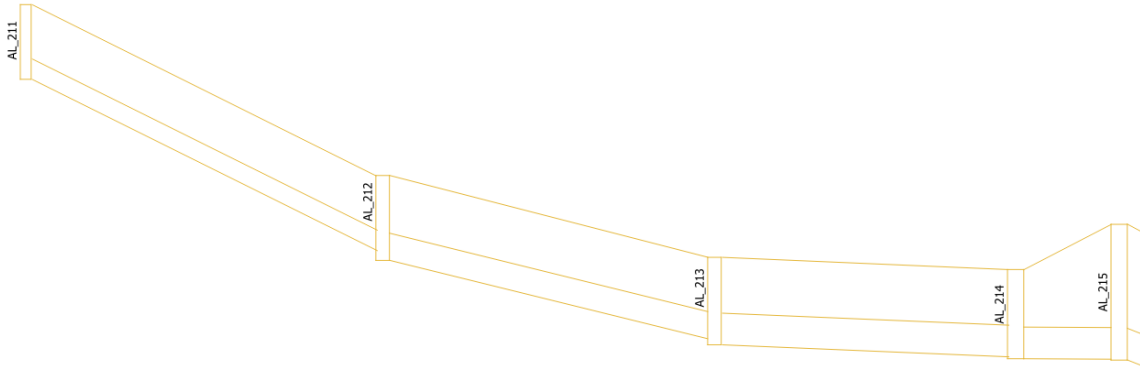


Figura 22. Caso de estudio 2 - Perfil diseñado entre los pozos AL_211y AL_215

Fuente: Elaboración propia.

A continuación, se presenta un resumen de información de interés de la red diseñada:

Tabla 12. Caso de estudio 2 - Resumen de información de interés de la red diseñada.

Caso de estudio	2
Directorio	Ejemplo03
Tipo de red	Pluvial
Área (ha)	51.91
Caudal máximo (lps)	1651.21
Pendiente máxima de tubería (m/m)	9.9%
Pendiente mínima de tubería (m/m)	0.0%
Factor Y/d máximo (%)	87.56%
Factor Y/d mínimo (%)	23.61%
Velocidad máxima del fluido (m/s)	5.39
Velocidad mínima del fluido (m/s)	1.31
Tirante máximo (cm)	131.34
Diámetro máximo (pg)	78
Diámetro mínimo (pg)	20
Altura promedio de excavación (m)	2.79
Volumen de excavación (m3)	8016
Costo (APU's)	\$ 1,793,077,593

Fuente: Elaboración propia.

3.1.3 Caso de estudio 3 - Ejemplo01

Red sanitaria con 19.4 kilómetros de tuberías. Se muestra el perfil diseñado de la red entre los pozos AS_115_1 y AS_116, graficado desde SewerGems:

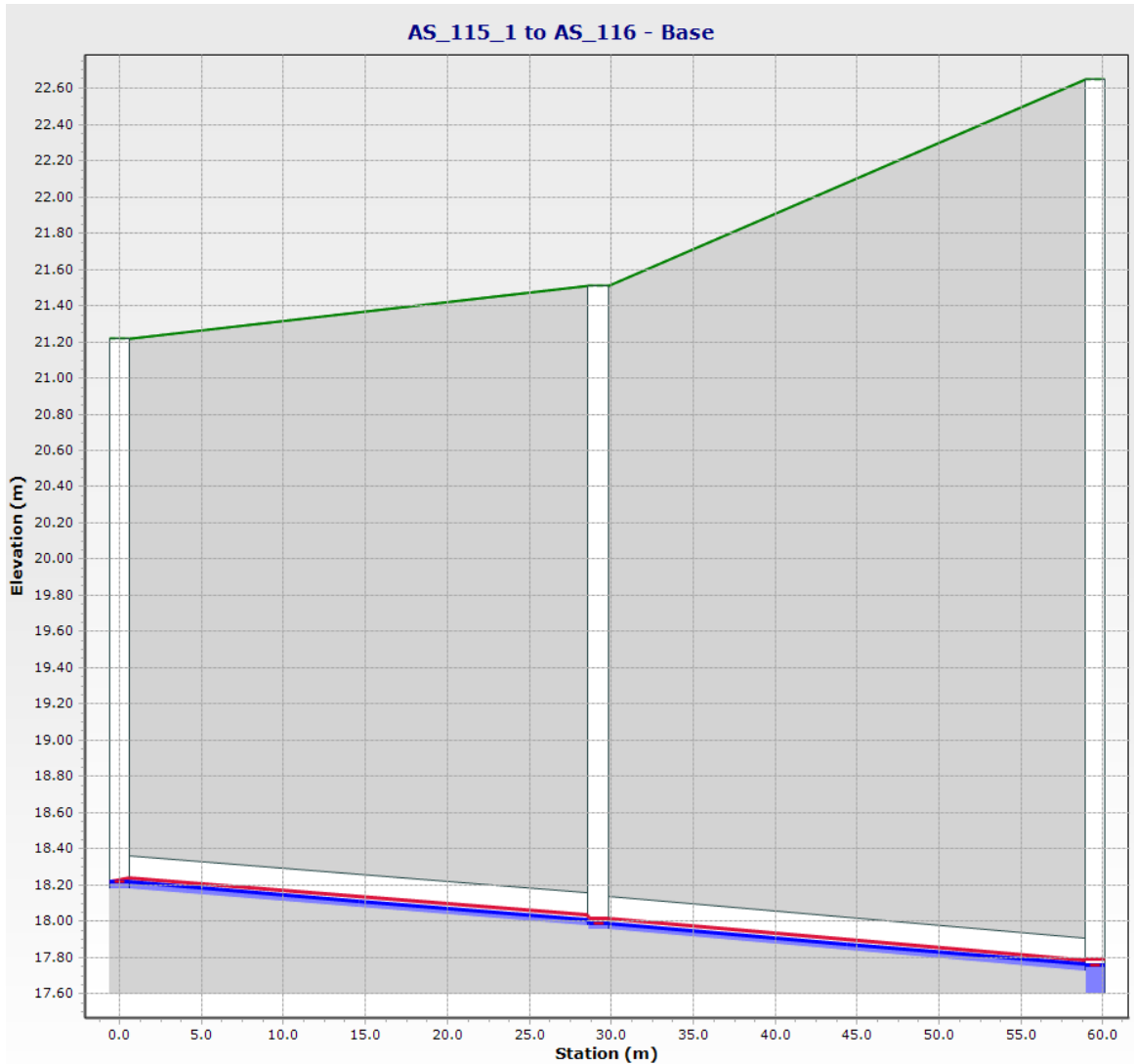


Figura 23. Caso de estudio 3 - Perfil diseñado entre los pozos AS_115_1 y AS_116

Fuente: Elaboración propia.

A continuación, se presenta un resumen de información de interés de la red diseñada:

Tabla 13. Caso de estudio 3 - Resumen de información de interés de la red diseñada.

Caso de estudio	3
Directorio	Ejemplo01
Tipo de red	Sanitaria
Área (ha)	637.24
Caudal máximo (lps)	642.55
Pendiente máxima de tubería (m/m)	42.6%
Pendiente mínima de tubería (m/m)	0.1%
Factor Y/d máximo (%)	81.06%
Factor Y/d mínimo (%)	6.36%
Velocidad máxima del fluido (m/s)	4.24
Velocidad mínima del fluido (m/s)	0.81
Tirante máximo (cm)	81.06
Diámetro máximo (pg)	56
Diámetro mínimo (pg)	8
Altura promedio de excavación (m)	2.65
Volumen de excavación (m3)	55150
Costo (APU's)	\$ 6,396,377,337

Fuente: Elaboración propia.

3.1.4 Caso de estudio 4 - Ejemplo04

Red pluvial con 5.6 kilómetros de tuberías. Se muestra el perfil diseñado de la red entre los pozos AL_28 y AL_32, graficado desde QGIS por el aplicativo:



Figura 24. Caso de estudio 4 - Perfil diseñado entre los pozos AL_28 y AL_32

Fuente: Elaboración propia.

A continuación, se presenta un resumen de información de interés de la red diseñada:

Tabla 14. Caso de estudio 4 - Resumen de información de interés de la red diseñada.

Caso de estudio	4
Directorio	Ejemplo04
Tipo de red	Pluvial
Área (ha)	541.08
Caudal máximo (lps)	5388.54
Pendiente máxima de tubería (m/m)	8.7%
Pendiente mínima de tubería (m/m)	0.0%
Factor Y/d máximo (%)	92.94%
Factor Y/d mínimo (%)	16.93%
Velocidad máxima del fluido (m/s)	5.85
Velocidad mínima del fluido (m/s)	2.19
Tirante máximo (cm)	219.35
Díámetro máximo (pg)	134
Díámetro mínimo (pg)	36
Altura promedio de excavación (m)	3.08
Volumen de excavación (m3)	47699
Costo (APU's)	\$ 12,759,316,695

Fuente: Elaboración propia.

3.1.5 Caso de estudio 5 - Ejemplo05

Red pluvial con 13.4 kilómetros de tuberías. Se muestra el perfil diseñado de la red entre los pozos AL_19 y AL_24, graficado desde QGIS por el aplicativo:

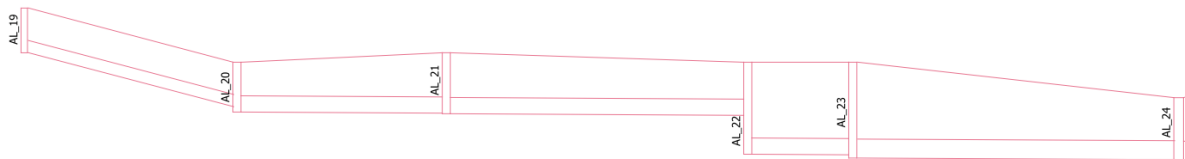


Figura 25. Caso de estudio 4 - Perfil diseñado entre los pozos AL_19 y AL_24

Fuente: Elaboración propia.

A continuación, se presenta un resumen de información de interés de la red diseñada:

Tabla 15. Caso de estudio 5 - Resumen de información de interés de la red diseñada.

Caso de estudio	5
Directorio	Ejemplo05
Tipo de red	Pluvial
Área (ha)	638.04
Caudal máximo (lps)	1788.48
Pendiente máxima de tubería (m/m)	11.9%
Pendiente mínima de tubería (m/m)	0.0%
Factor Y/d máximo (%)	92.50%
Factor Y/d mínimo (%)	17.85%
Velocidad máxima del fluido (m/s)	6.22
Velocidad mínima del fluido (m/s)	1.38
Tirante máximo (cm)	137.93
Diámetro máximo (pg)	78
Diámetro mínimo (pg)	44
Altura promedio de excavación (m)	2.75
Volumen de excavación (m3)	74933
Costo (APU's)	\$ 15,961,310,129

Fuente: Elaboración propia.

3.2 Costo computacional en pruebas de ejecución

A continuación, se presentan las características de las redes de referencia, incluyendo el costo computacional del diseño de la red:

Tabla 16. Costo computacional de casos de estudio.

	Ejemplo01	Ejemplo02	Ejemplo03	Ejemplo04	Ejemplo05
Pozos	474	22	32	105	323
Tramos	473	21	31	104	322
Longitud	19447.53	1006.15	1384.67	5663.94	13446.95
Tipo	AASS	AASS	AALL	AALL	AALL
Tiempo procesamiento (s)	185	10.94	15.69	50.37	312.65

Fuente: Elaboración propia.

Se observa una demanda de tiempo adicional en las tuberías pluviales, debido a los reprocesos en el cálculo de la velocidad del fluido y en el tiempo de concentración de la red.

3.3 Comparación del costo mediante análisis de precios unitarios respecto a las funciones de costos.

Mediante un aplicativo para Python “*AplicativoFuncionCostos.ipynb*”, se genera una función de costos propia, a partir del cálculo de los costos mediante análisis de precios unitarios del costo de suministro e instalación de la tubería, como se indica en 2.3.3. Los costos obtenidos por la función de costos indicada en 2.3.2, se comparan con los costos calculados mediante análisis de precios unitarios descritos en Análisis de precios unitarios 2.3.1. Se hallan errores relativos porcentuales agrupados por diámetro y por volumen de excavación por metro lineal, obteniendo un estimativo de la calidad de la aproximación de las funciones de costos en términos de un error relativo porcentual.

Tabla 17. Comparación de funciones objetivo respecto al cálculo de costos mediante precios unitarios.

D (m)	% Error	% Error	Vol/ml	% Error	% Error
0.182	290.48%	7.95%	1	314.59%	6.81%
0.227	252.25%	4.20%	2	213.75%	6.61%
0.284	214.17%	0.65%	3	109.45%	11.05%
0.327	137.70%	10.59%	4	52.61%	7.74%
0.362	119.31%	10.43%	5	23.58%	5.03%
0.407	92.04%	11.52%	6	14.33%	3.45%
0.452	63.36%	19.93%	7	17.99%	3.83%
0.6	39.52%	8.54%	8	21.94%	3.98%
0.7	29.26%	5.38%	9	22.59%	5.02%
0.8	18.50%	2.64%	10	26.92%	5.73%
0.9	5.59%	3.36%	11	33.44%	7.35%
1	7.61%	4.80%	12	28.76%	6.42%
1.1	11.76%	0.57%	13	50.26%	3.73%
1.2	19.04%	0.21%	14	50.29%	3.27%
1.3	23.12%	3.60%	15	49.51%	3.11%
1.4	26.77%	6.88%	16	48.87%	3.01%
1.5	31.23%	7.70%	18	50.65%	3.20%
1.6	35.85%	7.49%	19	45.27%	2.33%
1.7	40.99%	5.81%	20	43.55%	2.07%
2	49.38%	3.09%	21	43.86%	2.08%
2.45	63.06%	9.25%	22	64.22%	9.15%
			23	66.46%	8.88%
			24	63.96%	9.11%
			25	61.96%	9.39%
			26	60.87%	9.48%

Fuente: Elaboración propia.

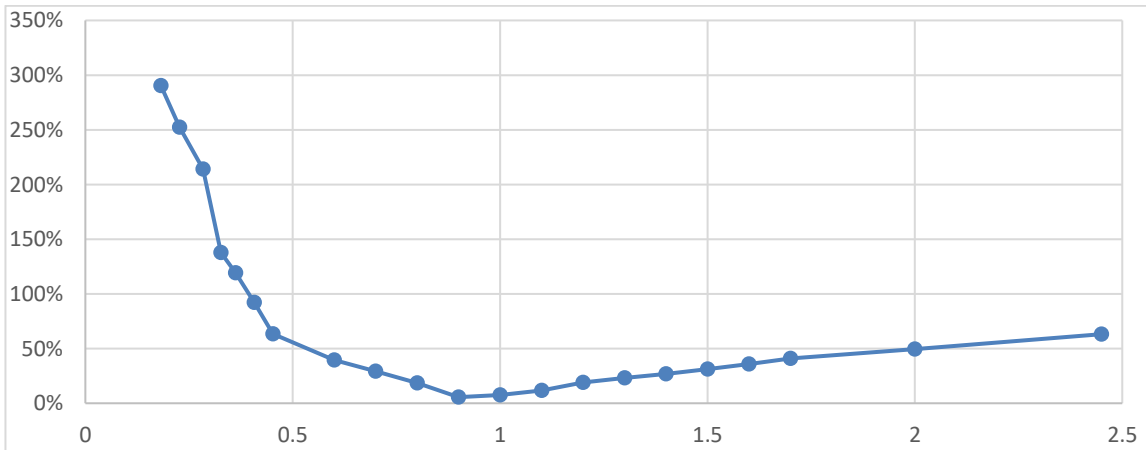


Figura 26. Error relativo porcentual. Costo calculado vs Función de costos CRA (1)
(Agrupados por diámetro)

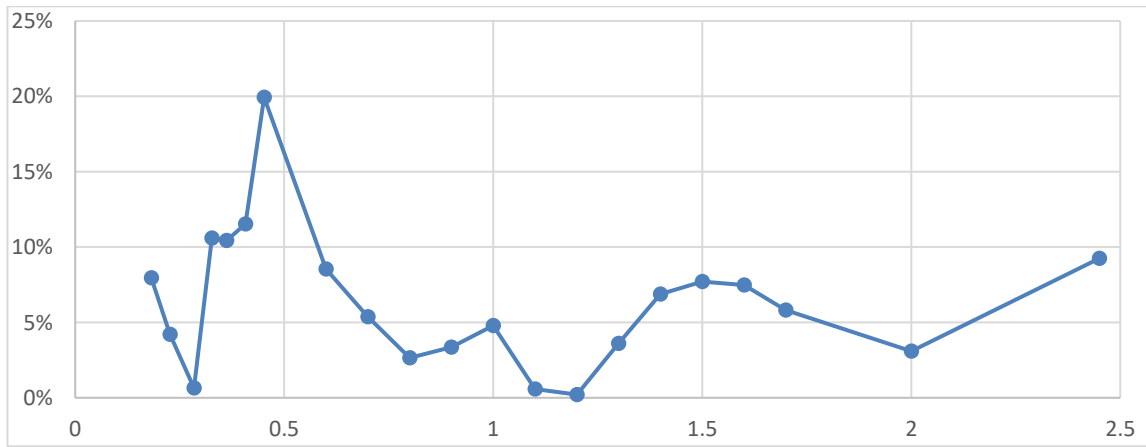


Figura 27. Error relativo porcentual. Costo calculado vs Función de costos propia (1)
(Agrupados por diámetro)

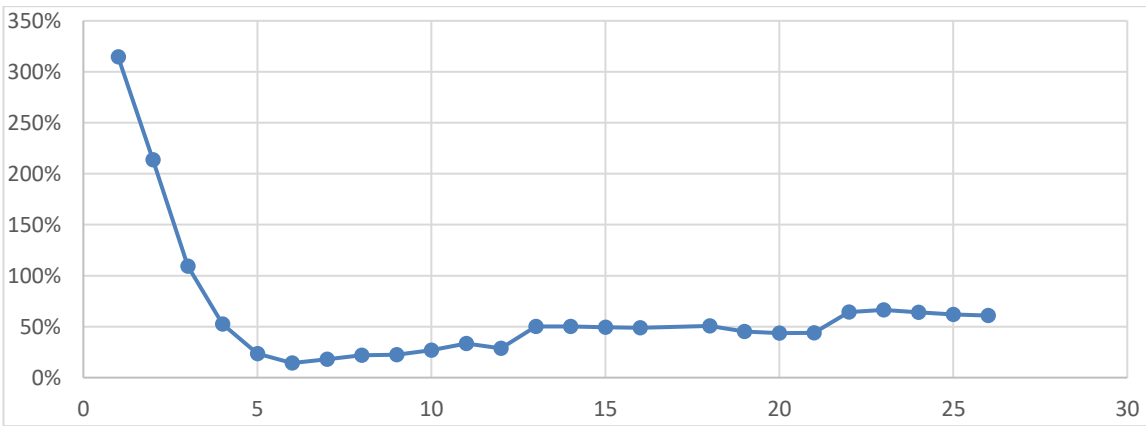


Figura 28. Error relativo porcentual. Costo calculado vs Función de costos CRA (2)
(Agrupados por volumen de excavación por metro)

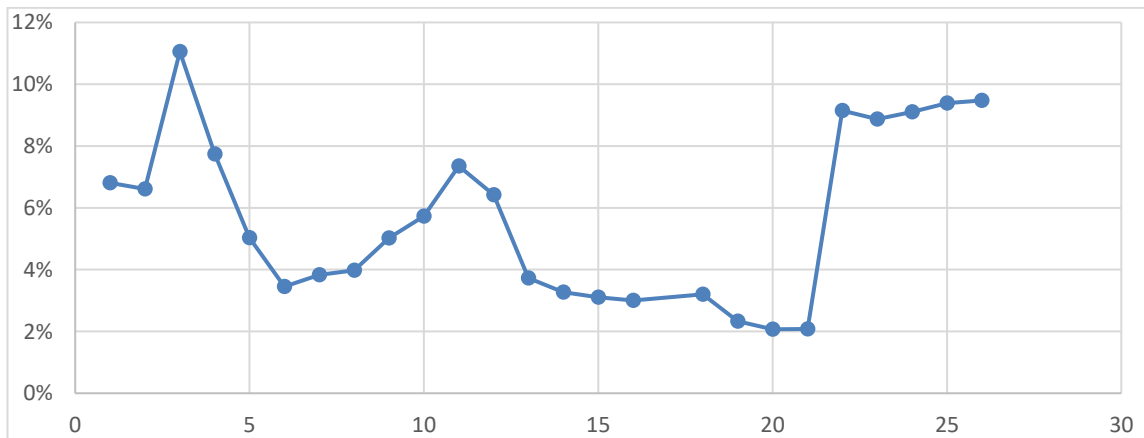


Figura 29. Error relativo porcentual. Costo calculado vs Función de costos propia (2)
(Agrupados por volumen de excavación por metro)

Se aprecian diferencias importantes entre el cálculo mediante precios unitarios respecto a la función de costos propuesta con la Comisión de Regulación de Agua potable y saneamiento básico. Son en promedio del 70%, y se concentran los diámetros y volúmenes de excavación más pequeños.

Las diferencias presentadas entre el cálculo mediante precios unitarios, y la función de costos propia, son en promedio son del 7%, y no presenta una concentración o sesgo del error respecto al diámetro o al volumen de excavación, sin embargo, se recomienda la selección de las tuberías solamente por el análisis por precios unitarios.

Conclusiones y recomendaciones

- El software cumple con el objetivo de diseñar la topología en perfil de la red seleccionada, calculada con el menor costo de acuerdo a las restricciones hidráulicas indicadas por las normas de referencia.
- El programa a partir de la información conocida de la topología en planta de una red de alcantarillado, las cotas rasantes de los pozos, genera una solución hidráulica de menor costo utilizando flujo uniforme, mediante funciones de costos y análisis de precios unitarios. La topología de la red en planta y perfil puede ser usada en programas con el potencial de modelar flujo gradualmente variado como SWMM y SewerGems.
- El software fue generado con el lenguaje de programación Python y se ejecuta en el entorno de desarrollo (IDE) de QGIS.
- Las herramientas de uso común para la modelación de redes de alcantarillado como SWMM y SewerGems permiten la modelación en flujo gradualmente variado y permiten el diseño de las redes con un bajo nivel de automatización. EciSewer a pesar de diseñar la red en solo perfil y en flujo uniforme, lo hace con un nivel completo de automatización, y puede usarse la topología de la red que genera para la modelación en los programas mencionados.
- La interfaz gráfica de usuario en QGIS permite la visualización de la información de la topología de la red en planta y perfil. QGIS permite la manipulación y visualización gráfica de cualquier información principalmente de forma vectorial, y mediante ajustes menores permite la flexibilidad de mostrar la información de acuerdo a las necesidades del usuario, donde no es obligatoria la programación en Python. Permite adicionalmente generar interfaces de visualización para la generación de planos.
- El software EciSewer es de libre uso, y su código fuente, documentación y futuras actualizaciones son de dominio público estarán disponible en el portal de github en el link: <https://github.com/jimenezbarajasmiguel/EciSewer>.
- Los valores obtenidos en las ejecuciones de la red exportada por el software en SWMM y en SewerGems son consistentes y sus valores son muy cercanos a los calculados por el aplicativo.

- De los análisis hechos, no se recomienda el uso de las funciones objetivo de costos, puesto que no representa una ventaja respecto al cálculo del costo asociado al suministro e instalación de la tubería, mediante el análisis de precios unitarios.

Trabajo futuro

El uso del portal de github, permite que cualquier usuario reporte fallos menores o recomendaciones de nuevo alcance, así como la posibilidad del uso total o parcial del código, en otros proyectos.

Para la descripción del trabajo futuro se parte del potencial deseable en un software para el diseño de alcantarillado. Un aplicativo completo para el diseño de redes de alcantarillado, debe solucionar, a partir de la ubicación de los pozos en planta, la topografía del área de interés y los caudales asociados a los pozos, la solución en planta y en perfil de la topología de la red. Además, no debe limitarse al cálculo de la red con la restricción de flujo uniforme. Los alcances deseables deben incluir algoritmos para:

1. Trazado de la red en planta a partir de la solución del trazado de la tubería entre nodos. Este problema se puede resolver mediante la solución de la conexión entre nodos, usando programación no lineal u otras metodologías de programación
2. Optimización de grupos de tuberías mediante técnicas de programación y métodos de procesamiento más complejos, ya sean métodos metaheurísticos como el recocido simulado o métodos evolutivos, donde se pueden optimizar grupos de tuberías como una entidad, evitando la solución tubo a tubo de la red.
3. Permitir solucionar la topología en perfil a partir del punto de descarga hacia aguas arriba, para los problemas donde se tiene una restricción en la cota de llegada.
4. El cálculo de costos debe incluir la opción de ingresar la rasante y calcular los costos asociados a los pozos.

El archivo "red.inp".no está configurado para la ejecución de la red en flujo no permanente. Se requiere permitir que el usuario ingrese información de la variación de caudales en función de series de tiempo para usarse en SWMM o en SewerGems. De forma similar, esta información puede usarse con el complemento de SWMM5 para

generar modelaciones de redes de alcantarillado en flujo gradualmente variado y no permanente.

Referencias

- Bentley. (10 de abril de 2021). *SewerGEMS. Software de modelado para alcantarillados sanitarios y combinados.* Obtenido de <https://www.bentley.com/es/products/product-line/hydraulics-and-hydrology-software/sewergems>
- Duque, N. (2015). *Metodología para el diseño optimizado de redes de alcantarillado (tesis de maestría).* Universidad de los Andes. Obtenido de <https://repositorio.uniandes.edu.co/handle/1992/13046>
- Empresa de Acueducto y Alcantarillado de Bogotá. (2020). *Norma técnica del servicio NS-085. Criterios de diseño de sistemas de alcantarillado.* Recuperado el 22 de Febrero de 2021, de <https://www.acueducto.com.co/webdomino/sistec/consultas.nsf>
- Empresas Públicas de Medellín. (2013). *Guía para el diseño Hidráulico de Redes de Alcantarillado.* Recuperado el 25 de Marzo de 2021, de https://www.epm.com.co/site/Portals/0/centro_de_documentos/GuiaDisenoHidraulicoRedesAlcantarillado.pdf
- Marín, C., Revollo, D., Del Valle, J., & Irreño, N. (2007). Estudio de estructuración y análisis de información de inversiones de los prestadores de acueducto y alcantarillado. *Regulación de agua potable y saneamiento básico*(15), 104-133. Obtenido de <https://www.cra.gov.co/revista-ndeg-15>
- Ministerio de Vivienda, Ciudad y Territorio. (2017). *Resolución 330. Reglamento Técnico del Sector de Agua Potable y Saneamiento Básico - RAS.* Recuperado el 12 de Marzo de 2021, de <https://www.minvivienda.gov.co/viceministerio-de-agua-y-saneamiento-basico/reglamento-tecnico-sector/reglamento-tecnico-del-sector-de-agua-potable-y-saneamiento-basico-ras>
- Quantum GIS. (5 de Agosto de 2021). *QGIS Sistema de Información Geográfica libre y de Código Abierto.* Obtenido de <https://qgis.org/es/site/>

Torres, L. (2013). *Métodos de diseño en redes de alcantarillado (tesis de maestría)*. Universidad de los Andes. Obtenido de <https://repositorio.uniandes.edu.co/handle/1992/19576>

United States Environmental Protection Agency. (9 de marzo de 2021). *Storm Water Management Model (SWMM) Water Research US EPA (5.1)*. Obtenido de <https://www.epa.gov/water-research/storm-water-management-model-swmm>

Anexos

Anexo 1. Código para la generación de la función de costos propia

Archivo “*AplicativoFuncionCostos.ipynb*”

```
## Bibliotecas requeridas en la ejecucion del codigo
import matplotlib as mpl
import matplotlib.pyplot as plt
import numpy as np
from scipy.optimize import curve_fit
import pandas as pd
import os
from IPython.display import HTML

## Funcion para calcular la funcion potencial con constantes a y b
def power_law(x, a, b):
    return a*np.power(x, b)

## Lee el archivo con el consolidado de las redes calculadas de referencia y muestra la
informacion
ruta='D:\Datos\Maestria\Tesis\Redes\FuncionCostos'
df=pd.read_csv(os.path.join(ruta, 'redes_calculadas.csv'))
display(HTML(df.to_html(na_rep='-',formatters={'S_ras': '{:,.2%}'.format, 'Y_d':
'{:,.1%}'.format, 'S': '{:,.2%}'.format, 'S_min': '{:,.2%}'.format, 'n': '{:,.3f}'.format,
'costo': '${:0,.0f}'.format, 'fc_CRA': '${:0,.0f}'.format, 'fc_prop':
'${:0,.0f}'.format, 'costo_mat': '${:0,.0f}'.format, 'costo_tub':
'${:0,.0f}'.format, 'valor_insta': '${:0,.0f}'.format, 'valor_ml':
'${:0,.0f}'.format, 'vol_exc': '{:,.1f}'.format})))

## Ajusta la curva potencial a los costos asociados por metro lineal a la tuberia
x_dummy=df["d"]
y_dummy=df["costo_tub"]/df["L"]

# Ajusta a la curva potencial
pars, cov = curve_fit(f=power_law, xdata=x_dummy, ydata=y_dummy, p0=[0, 0],
bounds=(-np.inf, np.inf))
power_law(df["d"][20], pars[0], pars[1])

# Grafica del ajuste
fig = plt.figure(figsize=(3, 3))
ax = fig.add_axes([0, 0, 1, 1])
ax.scatter(x_dummy, y_dummy, s=20, color='#00b3b3', label='Data')
ax.plot(x_dummy, power_law(x_dummy, *pars), linewidth=2, linestyle='--', color='black')
ax.set_xlabel('x-data', labelpad=10)
ax.set_ylabel('y-data', labelpad=10)
plt.show()
```



```

# Muestra los resultados del ajuste y el rango de confianza para una desviacion estandar
print("Ajuste potencial (F=a*X^b): ")
print("a =", round(pars[0],4), "+/-", round(cov[0,0]**0.5,4))
print("b =", round(pars[1],4), "+/-", round(cov[1,1]**0.5,4))

## Ajusta la curva potencial a los costos asociados a la excavacion
x_dummy=df["vol_exc"]
y_dummy=df["costo_mat"]

# Ajusta a la curva potencial
pars2, cov2 = curve_fit(f=power_law, xdata=x_dummy, ydata=y_dummy, p0=[0, 0],
bounds=(-np.inf, np.inf))
power_law(df["h_barra"][20], pars2[0], pars2[1])

# Grafica del ajuste
fig = plt.figure(figsize=(3, 3))
ax = fig.add_axes([0, 0, 1, 1])
ax.scatter(x_dummy, y_dummy, s=20, color='#00b3b3', label='Data')
ax.plot(x_dummy, power_law(x_dummy, *pars2), linewidth=2, linestyle='--', color='black')
ax.set_xlabel('x-data', labelpad=10)
ax.set_ylabel('y-data', labelpad=10)
plt.show()

# Muestra los resultados del ajuste
# Muestra los resultados del ajuste y el rango de confianza para una desviacion estandar
print("Ajuste potencial (F=a*X^b): ")
print("a =", round(pars2[0],4), "+/-", round(cov2[0,0]**0.5,4))
print("b =", round(pars2[1],4), "+/-", round(cov2[1,1]**0.5,4))

```

Anexo 2. Código del aplicativo

Archivo "sewer_plugin.py"

```
## Bibliotecas requeridas en la ejecucion del codigo
#from .utilitarios import *
from qgis.core import *#QgsVectorLayer, QgsProject, Qgis, QgsPointXY,
QgsGeometryUtils, QgsCoordinateReferenceSystem, QgsGeometry, QgsPoint,
QgsDistanceArea, QgsFeature, QgsField, QgsWkbTypes, QgsMapLayerProxyModel,
QgsPalLayerSettings
from qgis.utils import iface
from qgis.gui import QgsMapCanvas, QgsMapLayerComboBox
from qgis.PyQt.QtGui import *#QIcon, QFont, QColor
from PyQt5.QtWidgets import QProgressBar
from PyQt5.QtCore import Qt, QVariant
from PyQt5.QtWidgets import QAction
from PyQt5 import uic
import wx
import os
import math
import time
import numpy as np
import pandas as pd
import math
from scipy.optimize import root

plugin_path = os.path.dirname(__file__)
files_path = ""
DialogBase, DialogType = uic.loadUiType(os.path.join(plugin_path, 'formulario.ui'))
escala=5.0

## Funciones comunes para la manipulacion de QGIS
def crear_barra_progreso(final):
    progressMessageBar = iface.messageBar().createMessage("Ejecución en progreso:")
    progress = QProgressBar()
    progress.setMaximum(final)
    progress.setAlignment(Qt.AlignLeft|Qt.AlignVCenter)
    progressMessageBar.layout().addWidget(progress)
    iface.messageBar().pushWidget(progressMessageBar, Qgis.Info)
    progress.setVisible(True)
    return progress

def actualiza_barra_progreso(progress, valor, final):
    progress.setValue(valor)
    iface.mapCanvas().refresh()
    iface.mapCanvas().waitWhileRendering()
    if valor >= final:
        progress.setVisible(False)
```

```

        iface.messageBar().pushMessage("Informacion:", " Proceso completado", Qgis.Info,
5) #Info Warning Critical Success

```

```

def crear_canvas(layer):
    canvas = QgsMapCanvas()
    canvas.setExtent(layer.extent())
    canvas.setLayers([layer])
    canvas.show()
    return canvas

```

```

## Funciones comunes para el calculo de redes tuberias

```

```

# Calculo del angulo de deflexion

```

```

def deflexion(x1,y1,x2,y2,x3,y3):
    return 180-math.degrees(math.pi
        -abs(math.atan(((y2-y1)/(x2-x1)-(y3-y1)/(x3-x1)))/(1+((y2-y1)/(x2-x1)*(y3-y1)/(x3-
x1))))))
        -abs(math.atan(((y2-y3)/(x2-x3)-(y1-y3)/(x1-x3)))/(1+((y2-y3)/(x2-x3)*(y1-y3)/(x1-
x3))))))

```

```

# Calculo del angulo de la tuberia que cumple los requerimientos minimos

```

```

def func(x):
    return Q-(1/n)*((x-np.sin(x))*d**2/8)*(d*(1-np.sin(x)/x)/4)**(2/3)*(tao/(g*rho*(d*(1-
np.sin(x)/x)/4)))**(1/2)

```

```

# Calculo del angulo de la tuberia

```

```

def func2(x):
    return Q-(1/n)*((x-np.sin(x))*d**2/8)*(d*(1-np.sin(x)/x)/4)**(2/3)*S**(1/2)

```

```

# Calculo del angulo critico de la tuberia

```

```

def func3(x):
    return Q/g**0.5-2**0.5/32*(x-np.sin(x))**1.5*d**2.5/(np.sin(x/2))**0.5

```

```

## Carga de datos - Carga la informacion de corridas previas

```

```

def carga_datos_existentes():
    global df, tubos, setup, pozos
    df=pd.read_csv(os.path.join(files_path, 'red_calculada.csv'))
    tubos=pd.read_csv(os.path.join(files_path, 'tuberias.csv'))
    setup=pd.read_csv(os.path.join(files_path, 'setup.csv'))
    pozos=pd.read_csv(os.path.join(files_path, 'pozos.csv'))

```

```

## Carga de datos - Carga la informacion de topologia de la red, características de las
tuberias y costos asociados

```

```

def carga_datos():

```

```

# La instrucción global habilita el acceso de la variable dentro de la funciones.

```

```

# Requerido para el uso de algunas funciones como por ejemplo scipy.optimize en func2
(Calculo del angulo de la tuberia)

```

```

global df, tubos, setup, c_escorrentia, C1, C2, d_disposicion, d_minimo,
d_minimo_AALL, d_minimo_AASS
global d_pz_min, delta_S, dot, ent_0_3, ent_3_48, ent_48_72, ent_72, exc_0_2,
exc_2_5, exc_5_7, exc_7
global F, fmQ, fR, fr_cr, fr_sc, g, hL_minimo, mat_are, mat_cama, mat_rec, mat_retiro,
mat_sel, pob
global Q_minimo, Qce, Qinf, Qotros, rec_min, rho, sobrependiente, tc_entrada,
tipo_red, v_ini_asumida, X0,files_path
## Carga informacion en dataframes
# Obtiene la topologia de la red y su informacion geometrica del trazado en planta
df=pd.read_csv(os.path.join(files_path, 'red.csv'))
# Obtiene la informacion asociada a la tuberia de acuerdo a su diametro como
# diamtro interno y externo, rugosidad, ancho de zanja, costos y otros.
tubos=pd.read_csv(os.path.join(files_path, 'tuberias.csv'))
# Obtiene informacion asociada al tipo de red (Pluvial o sanitaria), restricciones y
condiciones de frontera,
# costos de materiales, e informacion general de restricciones
setup=pd.read_csv(os.path.join(files_path, 'setup.csv'))

## Carga de informacion comundiametro interno, externo
tipo_red=setup['tipo_red'][0]
g=float(setup['g'])
rho=float(setup['rho'])
fr_sc=float(setup['fr_sc'])
fr_cr=float(setup['fr_cr'])
delta_S=float(setup['delta_S'])
d_pz_min=float(setup['d_pz_min'])
v_ini_asumida=float(setup['v_ini_asumida'])
rec_min=float(setup['rec_min'])
Q_minimo=float(setup['Q_minimo'])
# Perdidas minimas en el colector: RAS artículo 137 - 0.02m
hL_minimo=float(setup['hL_minimo'])

## Informacion de costos de materiales, entibados y excavaciones
# Altura base granular de cama
mat_cama=float(setup['mat_cama'])
# Excavación En Zanja a máquina en material común h <2 m (M3)
exc_0_2=float(setup['exc_0_2'])
# Excavación En Zanja a máquina en material común 2<= h<=5 m (M3)
exc_2_5=float(setup['exc_2_5'])
# Excavación En Zanja a máquina en material común 5<h<=7 m (M3)
exc_5_7=float(setup['exc_5_7'])
# Excavación En Zanja a máquina en material común h > 7 m (M3)
exc_7=float(setup['exc_7'])
# Entibado continuo en cajones de acero H<=3,00 M (M2)
ent_0_3=float(setup['ent_0_3'])
# Entibado deslizante 3,00<H<=4,80 M (M2)
ent_3_48=float(setup['ent_3_48'])
# Entibado deslizante 4,80<H<=7,20 M (M2)

```

```

ent_48_72=float(setup['ent_48_72'])
# Entibado deslizante H>7,20 M (M2)
ent_72=float(setup['ent_72'])
# Suministro, transporte e instalación Recebo (M3)
mat_rec=float(setup['mat_rec'])
# Suministro, transporte e instalación material excavación (M3)
mat_sel=float(setup['mat_sel'])
# Suministro, transporte e instalación de Arena de Peña (M3)
mat_are=float(setup['mat_are'])
# Retiro y disposic. materiales sobrantes excav y demol (M3.km)
mat_retiro=float(setup['mat_retiro'])
# Distancia al sitio de disposic de materiales sobrantes y de excavacion (km)
d_disposicion=float(setup['d_disposicion'])

## Informacion para diseño de tuberia pluvial
# Informacion hidrológica (idf)
C1=float(setup['C1'])
X0=float(setup['X0'])
C2=float(setup['C2'])
# Diametro minimo pluvial
d_minimo_AALL=float(setup['d_minimo_AALL'])
# Tiempo concentracion entrada
tc_entrada=float(setup['tc_entrada'])
# Coeficiente escorrentia
c_escorrentia=float(setup['c_escorrentia'])

## Informacion para diseño de tuberia sanitaria
# Diametro minimo sanitario
d_minimo_AASS=float(setup['d_minimo_AASS'])
# Densidad Población Futura (hab/ha)
pob=float(setup['pob'])
# Dotación neta futura (l/hab/día)
dot=float(setup['dot'])
# Factor de Retorno
fR=float(setup['fR'])
# Factor Mayoración Caudal
fmQ=float(setup['fmQ'])
# Tasa Conexiones Erradas (lps/ha)
Qce=float(setup['Qce'])
# Tasa Infiltración (lps/ha)
Qinf=float(setup['Qinf'])
# Caudal de aguas residuales comerciales y/o industriales (lps/ha)
Qotros=float(setup['Qotros'])
# Factor de mayoración que para calcular el caudal máximo horario con base en el caudal
medio diario
F=float(setup['F'])

## Inicializa variables
d_minimo=d_minimo_AASS

```

sobrependiente=0.0
df['a_aflu']=0.0
df['a_tot']=df['a_tramo']
df['x_sig']=0.0
df['y_sig']=0.0
df['S_ras']=0.0
df['L_XY']=0.0
df['L']=0.0
df['absc']=0.0
df['deflex']=0.0
df['Tc_tramo']=0.0
df['Tc_aflu']=0.0
df['Tc_tot']=tc_entrada
df['I']=0.0
df['C']=c_escorrentia
df['Q_AALL']=0.0
df['poblac']=0.0
df['Qd']=0.0
df['QMH']=0.0
df['QCE']=0.0
df['QINF']=0.0
df['Qotros']=0.0
df['Q_AASS']=0.0
df['Q_propio']=0.0
df['Q']=0.0
df['S']=0.0
df['S_min']=0.0
df['d']=d_minimo_AASS
df['n']=0.010
df['Qo']=0.0
df['Y_d']=0.0
df['ang']=0.0
df['A']=0.0
df['P']=0.0
df['T']=0.0
df['V']=v_ini_asumida
df['Y']=0.0
df['Dh']=0.0
df['R']=0.0
df['tao']=0.0
df['F']=0.0
df['Hv']=0.0
df['EE']=0.0
df['ang_cr']=0.0
df['Yc']=0.0
df['Vc']=0.0
df['EEc']=0.0
df['d_pz_ini']=d_pz_min
df['d_pz_fin']=d_pz_min

```

df['pz_ini_bat'] = 1000000.0
df['pz_fin_bat'] = 1000000.0
df['r_curv']=0.0
df['rc_fi']=0.0
df['c_caida']=0.0
df['k']=0.05
df['hL']=0.0
df['cla_ini']=0.0
df['cla_fin']=0.0
df['bat_ini']=0.0
df['bat_fin']=0.0
df['obg_ini']=rec_min
df['obg_fin']=rec_min
df['obg_sig']=rec_min
df['rec_ini']=rec_min
df['rec_fin']=rec_min
df['d_obg']=d_minimo_AASS
df['d_ext']=0.0
df['h_barra']=0.0
df['L_XY_tubo']=0.0
df['sa_zanja']=0.0
df['vol_exc']=0.0
df['valor_ml']=0.0
df['valor_insta']=0.0
df['costo_tub']=0.0
df['costo_mat']=0.0
df['costo']=1000000000000
df['fc_CRA']=0.0
df['fc_prop']=0.0
df=df.replace(np.nan, 0.0)

## Calculos preliminares - Abscisas, deflexiones, caudales iniciales
def inicializar():
## Calculos requeridos para la iteracion cero
    df['idd']=(df['pz_ini'].map(str)+'_'+df['pz_fin'].map(str))
# Longitud en el plano horizontal entre pozos
    df['L_XY']=((df['x_fin']-df['x_ini'])**2+(df['y_fin']-df['y_ini'])**2)**0.5
# Pendiente del terreno
    df['S_ras']=(df['z_ini']-df['z_fin'])/df['L_XY']
# Valor inicial para el calculo del abscisado
    df['absc']=df['L_XY']

## Coordenadas siguiente pozo - Requerido para el calculo del angulo de deflexion
for i, row in df.iterrows():
    try:
        df.at[i,'x_sig']=df.loc[np.where((df['pz_ini']==df.at[i,'pz_fin']))]['x_fin'].max()
        df.at[i,'y_sig']=df.loc[np.where((df['pz_ini']==df.at[i,'pz_fin']))]['y_fin'].max()
    except:
        pass

```

```

## Abscisa punto final
try:
    if df.at[i,'red']==df.at[i-1,'red']:
        df.at[i,'absc']=df.at[i-1,'absc']+df.at[i,'L_XY']
except:
    pass

## Calculo del angulo de deflexion
try:
    df.at[i,'deflex']=deflexion(df.at[i,'x_ini'],df.at[i,'y_ini'],df.at[i,'x_fin'],
                                df.at[i,'y_fin'],df.at[i,'x_sig'],df.at[i,'y_sig'])
except:
    pass

## Area aferente predecesores y Area aferente acumulada
df.at[i,'a_afllu']=df.loc[np.where((df['pz_fin']==df.at[i,'pz_ini'])
    & (df['red']!=df.at[i,'red']))]['a_tot'].sum()
try:
    if df.at[i,'red']==df.at[i-1,'red']:
        df.at[i,'a_tot']=df.at[i-1,'a_tot']+df.at[i,'a_tramo']+df.at[i,'a_afllu']
    else:
        df.at[i,'a_tot']=df.at[i,'a_tramo']
except: pass

## Calculos para el caudal de diseño para tuberías sanitarias
if tipo_red=='AASS':
    df['poblac']=np.round(pob*df['a_tot'],0)
    df['Qd']=df['poblac']*dot*fR*fmq/86400
    df['QMH']=df['Qd']*F
    df['QCE']=df['a_tot']*Qce
    df['QINF']=df['a_tot']*Qinf
    df['Qotros']=df['a_tot']*Qotros
# Se requiere un caudal minimo de 1.5lps
df['Q_AASS']=Q_minimo

df['Q_AASS']=np.maximum(df['Q_AASS'],(df['QMH']+df['QCE']+df['QINF']+df['Qotros']))
df['Q']=df['Q_AASS']
d_minimo=d_minimo_AASS
# Inicializa el diametro minimo
df['d']=d_minimo

## Calculos para el caudal de diseño para tuberías pluviales
else:
# Cambia el factor tirante/diametro maximo
tubos['y_D']=0.93
# Calculos iniciales hidrológicos
df['I']=2.78*C1/(df['Tc_tot']+X0)**C2
df['Q_AALL']=df['C']*df['I']*df['a_tot']

```



```

    df['Q']=df['Q_AALL']
# Tiempo de concentracion
    tiempo_concentracion()
    d_minimo=d_minimo_AALL
# Inicializa el diametro minimo
    df['d']=d_minimo

## Calculo del caudal de acuerdo al Tiempo de concentracion - Usado solamente para
tuberias pluviales
# Recalcula el tiempo de concentracion por cambio de diametro del tubo o del pozo
# El tc depende de la logitud de la tuberia y su pendiente
def recalcular():
    if tipo_red=='AALL':
        df['l']=2.78*C1/(df['Tc_tot']+X0)**C2
        df['Q_AALL']=df['C']*df['l']*df['a_tot']
        df['Q']=df['Q_AALL']
        tiempo_concentracion()

## Calculo del Tiempo de concentracion - Usado solamente para tuberias pluviales
def tiempo_concentracion():
# La primera entrada a esta funcion asume una velocidad del fluido de 1.5m/s
    if tipo_red=='AALL':
        df['Tc_tramo']=df['L_XY']/df['V']/60
        df['Tc_tot']=df['Tc_tramo']+df['Tc_tot']
# Tiempo_concentracion acumulado
    for i, row in df.iterrows():
        try:
            if df.at[i,'red']==df.at[i-1,'red']:
                df.at[i,'Tc_tot']=df.at[i-1,'Tc_tot']+df.at[i,'Tc_tramo']
            if df.at[i,'Tc_tot']<df.at[i,'Tc_aflu']:
                df.at[i,'Tc_tot']=df.at[i,'Tc_aflu']+df.at[i,'Tc_tramo']
        except:
            pass
    df.at[i,'Tc_aflu']=df.loc[np.where((df['pz_fin']==df.at[i,'pz_ini'])
        & (df['red']!=df.at[i,'red']))]['Tc_tot'].max()
    df['Tc_aflu'] = df['Tc_aflu'].fillna(0)

## Calculos requeridos por cada tuberia de la red
def calcular_linea1(i,df_linea):
    global S
    global sobrependiente

# Caracteristicas del pozo - Radio de curvatura
    df_linea.at[i,'r_curv']=(df_linea.at[i,'d_pz_fin']/2)/(np.tan(df_linea.at[i,'deflex']/180)/2)
    df_linea.at[i,'rc_fi']=df_linea.at[i,'r_curv']/df_linea.at[i,'d']

```

```

# Longitud real de la tuberia - Se requiere recalculer por cambio de diametro de pozo
df_linea.at[i,'L']=((df_linea.at[i,'L_XY']-df_linea.at[i,'d_pz_ini']/2-
df_linea.at[i,'d_pz_fin']/2)**2+
(df_linea.at[i,'z_fin']-df_linea.at[i,'z_ini'])**2)**0.5 #

# Cotas obligadas - Garantiza que la profundidad del recubrimiento tenga en cuenta
tuberias afluentes
df_linea.at[i,'rec_ini']=max(df_linea.at[i,'int_ini'],df_linea.at[i,'obg_ini'])

# Las perdidas en colector Obliga un recubrimiento del tramo siguiente
df_linea.at[i,'rec_fin']=max(df_linea.at[i,'int_fin'],df_linea.at[i,'obg_fin'])

# Pendiente del tramo a partir de condiciones de frontera (scipy.optimize y topografia)
# La variable sobrependiente se incluye en caso de requerir cambiar el numero de Froude
del tramo evaluado
df_linea.at[i,'S']=max(df_linea.at[i,'S_min'],sobrependiente+((df_linea.at[i,'z_ini']
-df_linea.at[i,'rec_ini'])-(df_linea.at[i,'z_fin']-df_linea.at[i,'rec_fin']))
/(df_linea.at[i,'L_XY']-df_linea.at[i,'d_pz_ini']/2-df_linea.at[i,'d_pz_ini']/2))
S=df_linea.at[i,'S']

# Caracteristicas hidraulicas del tubo
df_linea.at[i,'Qo']=(1/df_linea.at[i,'n'])*((math.pi*df_linea.at[i,'d']**2/4)/
(math.pi*df_linea.at[i,'d'])**(2/3)*(math.pi*df_linea.at[i,'d']**2/4)*(df_linea.at[i,'S']/1)**0.5*1
000
df_linea.at[i,'ang']=float(root(func2,math.pi).x)
df_linea.at[i,'Y']=(1-np.cos(df_linea.at[i,'ang']/2))*df_linea.at[i,'d']/2
df_linea.at[i,'Y_d']=df_linea.at[i,'Y']/df_linea.at[i,'d']
df_linea.at[i,'R']=df_linea.at[i,'d']/4*(1-np.sin(df_linea.at[i,'ang'])/df_linea.at[i,'ang'])
df_linea.at[i,'tao']=rho*g*df_linea.at[i,'S']*df_linea.at[i,'R']
df_linea.at[i,'V']=df_linea.at[i,'R']**(2/3)*df_linea.at[i,'S']**0.5/df_linea.at[i,'n']
df_linea.at[i,'A']=df_linea.at[i,'d']**2/8*(df_linea.at[i,'ang']-np.sin(df_linea.at[i,'ang']))
df_linea.at[i,'P']=df_linea.at[i,'ang']*df_linea.at[i,'d']/2
df_linea.at[i,'T']=df_linea.at[i,'d']*np.cos(np.arcsin((df_linea.at[i,'Y']-df_linea.at[i,'d']/2)/
(df_linea.at[i,'d']/2)))
df_linea.at[i,'Dh']=df_linea.at[i,'A']/df_linea.at[i,'T']
df_linea.at[i,'F']=df_linea.at[i,'V']/(g*df_linea.at[i,'Dh'])**0.5
df_linea.at[i,'Hv']=df_linea.at[i,'V']**2/(2*g)
df_linea.at[i,'EE']=df_linea.at[i,'Y']+df_linea.at[i,'Hv']

# Regimen critico
df_linea.at[i,'ang_cr']=float(root(func3,math.pi).x)
df_linea.at[i,'Yc']=(1-np.cos(df_linea.at[i,'ang_cr']/2))*df_linea.at[i,'d']/2
df_linea.at[i,'Vc']=(df_linea.at[i,'Q']/1000)/df_linea.at[i,'d']**2/8*(df_linea.at[i,'ang_cr']-
np.sin(df_linea.at[i,'ang_cr']))
df_linea.at[i,'EEc']=df_linea.at[i,'Yc']+df_linea.at[i,'Vc']**2/(2*g)

# Perdidas en el colector
if df_linea.at[i,'F']<1:

```

```

if df_linea.at[i,'rc_fi']>3:
    df_linea.at[i,'k']=0.05
elif df_linea.at[i,'rc_fi']>1.5 and df_linea.at[i,'rc_fi']<=3:
    df_linea.at[i,'k']=0.20
else:
    df_linea.at[i,'k']=0.40
else:
    if df_linea.at[i,'rc_fi']>10:
        df_linea.at[i,'k']=0.05
    elif df_linea.at[i,'rc_fi']>8 and df_linea.at[i,'rc_fi']<=10:
        df_linea.at[i,'k']=0.20
    else:
        df_linea.at[i,'k']=0.40
df_linea.at[i,'hL']=max(df_linea.at[i,'k']*df_linea.at[i,'Hv'], hL_minimo)

# Perfil de alcantarillado (m.s.n.m)
df_linea.at[i,'cla_ini']=df_linea.at[i,'z_ini']-df_linea.at[i,'rec_ini']
df_linea.at[i,'cla_fin']=df_linea.at[i,'cla_ini']-df_linea.at[i,'L']*df_linea.at[i,'S']
df_linea.at[i,'bat_ini']=df_linea.at[i,'cla_ini']-df_linea.at[i,'d']
df_linea.at[i,'bat_fin']=df_linea.at[i,'cla_fin']-df_linea.at[i,'d']

# Cota obligada siguiente clave
df_linea.at[i,'obg_sig']=df_linea.at[i,'z_fin']-df_linea.at[i,'cla_fin']+df_linea.at[i,'hL']

## Costos

# Volumen de la excavacion. Seccion formada en perfil por un trapecio escaleno
df_linea.at[i,'h_barra']=(df_linea.at[i,'rec_ini']+df_linea.at[i,'d_ext']+
    mat_cama+df_linea.at[i,'rec_fin']+df_linea.at[i,'d_ext']+mat_cama)/2
df_linea.at[i,'L_XY_tubo']=df_linea.at[i,'L_XY']-df_linea.at[i,'d_pz_fin']/2-
df_linea.at[i,'d_pz_ini']/2

df_linea.at[i,'vol_exc']=df_linea.at[i,'L_XY_tubo']*df_linea.at[i,'h_barra']*(df_linea.at[i,'d_ext']+
    df_linea.at[i,'sa_zanja'])

# Costos asociados con la tuberia
df_linea.at[i,'costo_tub']=df_linea.at[i,'costo']=df_linea.at[i,'L']*(df_linea.at[i,'valor_ml']+
    df_linea.at[i,'valor_insta'])

# Costos asociados con los materiales y con la excavacion (calculado por proporcion a la
altura)
# Material de cama
df_linea.at[i,'costo']+=mat_are*(mat_cama/df_linea.at[i,'h_barra'])*
df_linea.at[i,'vol_exc']
# Material de reemplazo
df_linea.at[i,'costo']+=mat_sel*(df_linea.at[i,'vol_exc']*
    (df_linea.at[i,'h_barra']-mat_cama)/df_linea.at[i,'h_barra'])

```

```

        -df_linea.at[i,'d_ext']**2*math.pi/4*df_linea.at[i,'L'])
# Costos de disposicion del material sobrante de excavacion
    df_linea.at[i,'costo']+=mat_retiro*d_disposicion*((mat_cama/df_linea.at[i,'h_barra'])*
df_linea.at[i,'vol_exc']+
        df_linea.at[i,'L']*df_linea.at[i,'d_ext']**2*np.pi/4)

# Costos excavacion por rango de profundidad
if df_linea.at[i,'h_barra']<2:
    df_linea.at[i,'costo']+=exc_0_2*df_linea.at[i,'vol_exc']
elif df_linea.at[i,'h_barra']>=2 and df_linea.at[i,'h_barra']<=5:
    df_linea.at[i,'costo']+=exc_0_2*df_linea.at[i,'vol_exc']*(2/df_linea.at[i,'h_barra'])
    df_linea.at[i,'costo']+=exc_2_5*df_linea.at[i,'vol_exc']*((df_linea.at[i,'h_barra']-
2)/df_linea.at[i,'h_barra'])
elif df_linea.at[i,'h_barra']>5 and df_linea.at[i,'h_barra']<=7:
    df_linea.at[i,'costo']+=exc_0_2*df_linea.at[i,'vol_exc']*(2/df_linea.at[i,'h_barra'])
    df_linea.at[i,'costo']+=exc_2_5*df_linea.at[i,'vol_exc']*(3/df_linea.at[i,'h_barra'])
    df_linea.at[i,'costo']+=exc_5_7*df_linea.at[i,'vol_exc']*((df_linea.at[i,'h_barra']-
5)/df_linea.at[i,'h_barra'])
elif df_linea.at[i,'h_barra']>7:
    df_linea.at[i,'costo']+=exc_0_2*df_linea.at[i,'vol_exc']*(2/df_linea.at[i,'h_barra'])
    df_linea.at[i,'costo']+=exc_2_5*df_linea.at[i,'vol_exc']*(3/df_linea.at[i,'h_barra'])
    df_linea.at[i,'costo']+=exc_5_7*df_linea.at[i,'vol_exc']*(2/df_linea.at[i,'h_barra'])
    df_linea.at[i,'costo']+=exc_7*df_linea.at[i,'vol_exc']*((df_linea.at[i,'h_barra']-
7)/df_linea.at[i,'h_barra'])

# Costos entibados por rango de profundidad
if df_linea.at[i,'h_barra']<=3:
    df_linea.at[i,'costo']+=ent_0_3*2*df_linea.at[i,'h_barra']
elif df_linea.at[i,'h_barra']>3 and df_linea.at[i,'h_barra']<=4.8:
    df_linea.at[i,'costo']+=ent_3_48*2*df_linea.at[i,'h_barra']
elif df_linea.at[i,'h_barra']>5 and df_linea.at[i,'h_barra']<=7.2:
    df_linea.at[i,'costo']+=ent_48_72*2*df_linea.at[i,'h_barra']
elif df_linea.at[i,'h_barra']>7.2:
    df_linea.at[i,'costo']+=ent_72*2*df_linea.at[i,'h_barra']
df_linea.at[i,'costo_mat']=df_linea.at[i,'costo']-df_linea.at[i,'costo_tub']

# Funcion de costos por la comision de regulaci3n de agua potable y saneamiento b3sico
CRA
df_linea.at[i,'fc_CRA']=1.5302*(9579.31*(df_linea.at[i,'d']*1000)**0.5737*df_linea.at[i,'L']+
1163.77*df_linea.at[i,'vol_exc']**1.31)

# Funcion de costos propia obtenida mediante el ajuste a una funcion pontencial del tipo
F=A.X^B
df_linea.at[i,'fc_prop']=(df_linea.at[i,'L']*1004861.4272*df_linea.at[i,'d']**1.7928+
43644.8004*df_linea.at[i,'vol_exc']**0.9583)
return df_linea

```

```

## Funcion iterativa de tuberias
def procesar():
# Barra de progreso
    final = len(df)*len(tubos)
    barra_progreso = crear_barra_progreso(final)
    ij=0
# Itera tuberias de la red
    for i, row in df.iterrows():
# Itera diametros disponibles
        for j, row in tubos.iterrows():
            ij+=1
            actualiza_barra_progreso(barra_progreso, ij, final)
            global n
            global d
            global Q
            global tao
            global d_minimo
            global sobrependiente
            global d_sobrependiente
            valido=1
            df_s_minima=df.iloc[[i]]
            n=df_s_minima.at[i,'n']=tubos.at[j,'n_mann']
            d=df_s_minima.at[i,'d']=tubos.at[j,'d_interno']
            df_s_minima.at[i,'d_pz_fin']=tubos.at[j,'d_pozo']
            tao=df_s_minima.at[i,'tao']=tubos.at[j,'tao']
            df_s_minima.at[i,'d_pz_fin']=tubos.at[j,'d_pozo']
            df_s_minima.at[i,'d_ext']=tubos.at[j,'d_externo']
            df_s_minima.at[i,'sa_zanja']=tubos.at[j,'a_zanja']
            df_s_minima.at[i,'valor_ml']=tubos.at[j,'valor_ml']
            df_s_minima.at[i,'valor_insta']=tubos.at[j,'valor_insta']
            Q=df_s_minima.at[i,'Q']/1000

# Calculo de la pendiente minima de la tuberia que cumple los requerimientos minimos
mediante scipy.optimize
    df_s_minima.at[i,'ang']=float(root(func,math.pi).x)

# Verifica la validez de la tuberia - Valida si no existe pendiente para la combinacion de
caudal y diametro
    if (df_s_minima.at[i,'ang']>2*np.pi or
        df_s_minima.at[i,'ang']<0.0 or
        df_s_minima.at[i,'d']<d_minimo or
        df_s_minima.at[i,'d']<df_s_minima.at[i,'d_obg']):
        valido=0
    else:
        df_s_minima.at[i,'S_min']=tao/(rho*g*(d*(1-math.sin(df_s_minima.at[i,'ang'])
            /df_s_minima.at[i,'ang'])/4))
        try:
            df_s_minima=calcular_linea1(i,df_s_minima)
        except:

```

```

        print("No hay solucion para la combinacion de Q,d,S..... supero los
diametros disponibles")
# Nota: Este mensaje no deberia visualizarse, de suceder requiere diametros mas grandes
adicionales.
# Al validar la combinacion D,d,S. De acuerdo a la norma requeriria calculo de flujo no
permanente con otro software.

# Evita flujos con Froude cercano al critico en el rango (fr_sc, fr_cr) definidos en setup.csv
# cambiando la pendiente y recalculando hasta que el numero de Froude sea superior a
fr_cr
    while df_s_minima.at[i,'F']>fr_sc and df_s_minima.at[i,'F']<fr_cr:
        sobrependiente+=delta_S
        df_s_minima=calcular_linea1(i,df_s_minima)
        sobrependiente=0.0

# Verifica la relacion tirante/diametro
    if df_s_minima.at[i,'Y_d']>tubos.at[j,'y_D']:
        valido=0
# Verifica si el costo calculado para el diametro(n) es meor que el costo calculado para el
diametro(n-1)
    if float(df.iloc[[i]]['costo'])>df_s_minima.at[i,'costo'] and valido==1:
        df.iloc[[i]]=df_s_minima
# Obliga un recubrimiento minimo para las tuberias dependientes
    for k in range(i+1, len(df), 1):
        if df.at[k,'pz_ini']==df.at[i,'pz_fin']:
            df.at[k,'d_obg']=max(df.at[i,'d'],df.at[k,'d_obg'])
            df.at[k,'obg_ini']=max(df.at[i,'obg_sig'],df.at[k,'obg_ini'])
            df.at[k,'d_pz_ini']=max(df.at[i,'d_pz_fin'],df.at[k,'d_pz_ini'])
# En el caso de tuberia pluvial requiere recalculer
    recalculer()

# Caudal propio
    df['Q_propio']=df['Q']
    for i, row in df.iterrows():
        for k in range(i+1, len(df), 1):
            if df.at[k,'pz_ini']==df.at[i,'pz_fin']:
                df.at[k,'Q_propio']=df.at[k,'Q_propio']-df.at[i,'Q']

## Halla la cota batea de los pozos (m.s.n.m) de acuerdo a la batea mas baja de las
tuberias que conectan a el pozo
    global pozos
    pozos=df.loc[:, "pz_ini"]
    pozos=pozos.append(df.loc[:, "pz_fin"])
    pozos=pd.DataFrame(pozos).drop_duplicates().rename(columns={0:
"pozo"}).reset_index(drop=True)
    pozos['batea'] = 1000000.0
    pozos['Q_propio'] = 0.0
    pozos['Coord_X'] = 0.0
    pozos['Coord_Y'] = 0.0

```

```

pozos['Coord_Z'] = 0.0
for i, row in pozos.iterrows():
    for j, row in df.iterrows():
        if pozos.at[i,'pozo']==df.at[j,'pz_ini']:
            pozos.at[i,'batea']=min(df.at[j,'bat_ini'],pozos.at[i,'batea'])
        if pozos.at[i,'pozo']==df.at[j,'pz_fin']:
            pozos.at[i,'batea']=min(df.at[j,'bat_fin'],pozos.at[i,'batea'])

for i, row in pozos.iterrows():
    for j, row in df.iterrows():
        if pozos.at[i,'pozo']==df.at[j,'pz_fin']:
            df.at[j,'pz_fin_bat']=min(df.at[j,'pz_fin_bat'],pozos.at[i,'batea'])
        if pozos.at[i,'pozo']==df.at[j,'pz_ini']:
            df.at[j,'pz_ini_bat']=min(df.at[j,'pz_ini_bat'],pozos.at[i,'batea'])
# Para exportar a SWMM y SewerGems
    pozos.at[i,'Q_propio']=df.at[j,'Q_propio']
    pozos.at[i,'Coord_X'] = df.at[j,'x_ini']
    pozos.at[i,'Coord_Y'] = df.at[j,'y_ini']
    pozos.at[i,'Coord_Z'] = df.at[j,'z_ini']
    pozos.at[i,'Coord_X'] = df.at[j,'x_fin']
    pozos.at[i,'Coord_Y'] = df.at[j,'y_fin']
    pozos.at[i,'Coord_Z'] = df.at[j,'z_fin']
    print("Proyecto ejecutado")

def exportar():
    global files_path
    ## Exporta la red calculada
    # csv con la informacion de los pozos
    pozos.to_csv(os.path.join(files_path, 'pozos.csv'))
    # csv con la red calculada
    df.to_csv(os.path.join(files_path, 'red_calculada.csv'))

    ## Exporta inp para el uso en SewerGems y SWMM
    f=open(os.path.join(files_path, 'red.inp'),'w')
    # Titulo
    f.write("[TITLE]\n")
    f.write("Modelacion\n")
    f.write("\n")

    # Opciones
    f.write("[OPTIONS]\n")
    f.write(";;Option Value\n")
    f.write("FLOW_UNITS LPS\n")
    f.write("INFILTRATION HORTON\n")
    f.write("FLOW_ROUTING DYNWAVE\n")
    f.write("LINK_OFFSETS DEPTH\n")
    f.write("MIN_SLOPE 0\n")

```

```

f.write("ALLOW_PONDING NO\n")
f.write("SKIP_STEADY_STATE NO\n")
f.write("START_DATE 01/01/2022\n")
f.write("START_TIME 23:00:02\n")
f.write("REPORT_START_DATE 01/01/2022\n")
f.write("REPORT_START_TIME 23:00:02\n")
f.write("END_DATE 01/02/2022\n")
f.write("END_TIME 00:00:02\n")
f.write("SWEEP_START 01/01\n")
f.write("SWEEP_END 12/31\n")
f.write("DRY_DAYS 0\n")
f.write("REPORT_STEP 1:00:00\n")
f.write("WET_STEP 0:05:00\n")
f.write("DRY_STEP 1:00:00\n")
f.write("ROUTING_STEP 0:00:05\n")
f.write("RULE_STEP 00:00:00\n")
f.write("INERTIAL_DAMPING PARTIAL\n")
f.write("NORMAL_FLOW_LIMITED BOTH\n")
f.write("FORCE_MAIN_EQUATION D-W\n")
f.write("VARIABLE_STEP 0.75\n")
f.write("LENGTHENING_STEP 0\n")
f.write("MIN_SURFAREA 0\n")
f.write("MAX_TRIALS 0\n")
f.write("HEAD_TOLERANCE 0\n")
f.write("SYS_FLOW_TOL 5\n")
f.write("LAT_FLOW_TOL 5\n")
f.write("MINIMUM_STEP 0.5\n")
f.write("THREADS 1\n")
f.write("\n")

# Evaporacion
f.write("[EVAPORATION]\n")
f.write(";;DataSource Parameters\n")
f.write("CONSTANT 0\n")
f.write("DRY_ONLY NO\n")
f.write("\n")

# Nodos
f.write("[JUNCTIONS]\n")
f.write(";;Name Elevation MaxDepth InitDepth SurDepth Aponded\n")
for i, row in pozos.iterrows():
    if i<len(pozos.index)-1:
        f.write(str(pozos.at[i,'pozo'])+' '+
            "{:.4f}".format(pozos.at[i,'batea'])+' '+
            "{:.4f}".format(pozos.at[i,'Coord_Z']-pozos.at[i,'batea'])+' '
            '0 0 0'+ "\n")
    f.write("\n")

# Descargas

```



```

f.write("[OUTFALLS]\n")
f.write(";;Name Elevation Type StageData Gated RouteTo\n")
f.write(str(pozos.at[i,'pozo'])+' '+
        "{:.4f}".format(pozos.at[i,'batea'])+' '+
        'FREE NO'+"\n")
f.write("\n")

# Conexion de Tuberias
f.write("[CONDUITS]\n")
f.write(";;Name FromNode ToNode Length Roughness InOffset OutOffset InitFlow
MaxFlow\n")
for j, row in df.iterrows():
    f.write(str(df.at[j,'idd'])+' '+
            str(df.at[j,'pz_ini'])+' '+
            str(df.at[j,'pz_fin'])+' '+
            "{:.4f}".format(df.at[j,'L'])+' '+
            "{:.4f}".format(df.at[j,'n'])+' '+
            "{:.4f}".format(df.at[j,'bat_ini']-df.at[j,'pz_ini_bat'])+' '+
            "{:.4f}".format(df.at[j,'bat_fin']-df.at[j,'pz_fin_bat'])+' '+
            '0 0'+"\n")
f.write("\n")

# Seccion de Tuberias
f.write("[XSECTIONS]\n")
f.write(";;Link Shape Geom1 Geom2 Geom3 Geom4 Barrels Culvert\n")
for j, row in df.iterrows():
    f.write(str(df.at[j,'idd'])+' CIRCULAR '+
            "{:.4f}".format(df.at[j,'d'])+' '+
            '0 0 0 1'+"\n")
f.write("\n")

# Caudales
f.write("[DWF]\n")
f.write(";;Node Constituent Baseline Patterns\n")
for i, row in pozos.iterrows():
    f.write(str(pozos.at[i,'pozo'])+' FLOW '+
            "{:.4f}".format(pozos.at[i,'Q_propio'])+' '+
            "" "" "" ""+"\n")
f.write("\n")

# Reportes
f.write("[REPORT]\n")
f.write(";;Reporting Options\n")
f.write("SUBCATCHMENTS ALL\n")
f.write("NODES ALL\n")
f.write("LINKS ALL\n")
f.write("\n")

# Tags

```

```

f.write("[TAGS]\n")
f.write("\n")

# Mapa
f.write("[MAP]\n")
f.write("Units None\n")
f.write("\n")

# Coordenadas de nodos
f.write("[COORDINATES]\n")
f.write(";;Nodo X Y\n")
for i, row in pozos.iterrows():
    f.write(str(pozos.at[i,'pozo'])+' '+
            "{:.4f}".format(pozos.at[i,'Coord_X'])+' '+
            "{:.4f}".format(pozos.at[i,'Coord_Y'])+"\n")
f.write("\n")

# Vertices
f.write("[VERTICES]\n")
f.write(";;Link X Y\n")
f.write("\n")
f.close()

print("Archivo red.inp exportado para el uso en SWMM y SewerGems")

def visualiza():
    global df
    global pozos
    QgsProject.instance().clear()
    layer = QgsVectorLayer("linestring?crs=epsg:3116&field=num", "Tubos", "memory")
    QgsProject.instance().addMapLayer(layer)
    pr = layer.dataProvider()
    for column in df:
        pr.addAttributes([QgsField(column, QVariant.String)])
    layer.updateFields()
    for j, row in df.iterrows():
        fet = QgsFeature(layer.fields())
        for column in df:
            fet.setAttribute(column, str(df.at[j,column]))
        fet.setGeometry(QgsGeometry.fromPolyline([QgsPoint(int(df.at[j,'x_ini']),
int(df.at[j,'y_ini']), QgsPoint(int(df.at[j,'x_fin']), int(df.at[j,'y_fin'])))]))
        (res, outFeats) = layer.dataProvider().addFeatures([fet])
    layer.updateExtents()

    layer = QgsVectorLayer("point?crs=epsg:3116&field=pozo", "Pozos", "memory")
    QgsProject.instance().addMapLayer(layer)
    pr = layer.dataProvider()
    for column in pozos:
        pr.addAttributes([QgsField(column, QVariant.String)])

```

```

layer.updateFields()
for j, row in pozos.iterrows():
    fet = QgsFeature(layer.fields())
    for column in pozos:
        fet.setAttribute(column, str(pezos.at[j,column]))

fet.setGeometry(QgsGeometry.fromPointXY(QgsPointXY(int(pezos.at[j,'Coord_X']),int(pezos.at[j,'Coord_Y']))))
(res, outFeats) = layer.dataProvider().addFeatures([fet])
layer.updateExtents()

layer_settings = QgsPalLayerSettings()
text_format = QgsTextFormat()

text_format.setFont(QFont("Arial", 10))
text_format.setSize(12)

buffer_settings = QgsTextBufferSettings()
buffer_settings.setEnabled(True)
buffer_settings.setSize(1)
buffer_settings.setColor(QColor("white"))

text_format.setBuffer(buffer_settings)
layer_settings.setFormat(text_format)

layer_settings.fieldName = "pozo"
layer_settings.placement = 2

layer_settings.enabled = True

layer_settings = QgsVectorLayerSimpleLabeling(layer_settings)
layer.setLabelsEnabled(True)
layer.setLabeling(layer_settings)
layer.triggerRepaint()

def ver_perfiles():
    global df
    global escala
    k=0
    for i in range(0, len(df['red'].unique()), 1):
        df2=df.query('red == ' + str(i+1))
        layer = QgsVectorLayer("linestring?crs=epsg:3116&field=num", "Perfil" + str(i+1),
"memory")
        QgsProject.instance().addMapLayer(layer)
        pr = layer.dataProvider()
        for j, row in df2.iterrows():

            fet = QgsFeature(layer.fields())

```

```

        fet.setGeometry(QgsGeometry.fromPolyline([QgsPoint(float(df2.at[j,'absc']-
df2.at[j,'L_XY_tubo']-df2.at[j,'d_pz_ini']-df2.at[j,'d_pz_fin']/2), escala*float(df2.at[j,'z_ini'])),
        QgsPoint(float(df2.at[j,'absc']-df2.at[j,'L_XY_tubo']-
df2.at[j,'d_pz_fin']/2), escala*float(df2.at[j,'z_ini']))))))
        (res, outFeats) = layer.dataProvider().addFeatures([fet])
        fet = QgsFeature(layer.fields())
        fet.setGeometry(QgsGeometry.fromPolyline([QgsPoint(float(df2.at[j,'absc']-
df2.at[j,'L_XY_tubo']-df2.at[j,'d_pz_ini']-df2.at[j,'d_pz_fin']/2),
escala*float(df2.at[j,'pz_ini_bat'])),
        QgsPoint(float(df2.at[j,'absc']-df2.at[j,'L_XY_tubo']-
df2.at[j,'d_pz_fin']/2), escala*float(df2.at[j,'pz_ini_bat']))))))
        (res, outFeats) = layer.dataProvider().addFeatures([fet])
        fet = QgsFeature(layer.fields())
        fet.setAttribute("num", df2.at[j,'pz_ini'])
        fet.setGeometry(QgsGeometry.fromPolyline([QgsPoint(float(df2.at[j,'absc']-
df2.at[j,'L_XY_tubo']-df2.at[j,'d_pz_ini']-df2.at[j,'d_pz_fin']/2), escala*float(df2.at[j,'z_ini'])),
        QgsPoint(float(df2.at[j,'absc']-df2.at[j,'L_XY_tubo']-
df2.at[j,'d_pz_ini']-df2.at[j,'d_pz_fin']/2), escala*float(df2.at[j,'pz_ini_bat']))))))
        (res, outFeats) = layer.dataProvider().addFeatures([fet])
        fet = QgsFeature(layer.fields())
        fet.setGeometry(QgsGeometry.fromPolyline([QgsPoint(float(df2.at[j,'absc']-
df2.at[j,'L_XY_tubo']-df2.at[j,'d_pz_fin']/2), escala*float(df2.at[j,'z_ini'])),
        QgsPoint(float(df2.at[j,'absc']-df2.at[j,'L_XY_tubo']-
df2.at[j,'d_pz_fin']/2), escala*float(df2.at[j,'pz_ini_bat']))))))
        (res, outFeats) = layer.dataProvider().addFeatures([fet])

        fet.setGeometry(QgsGeometry.fromPolyline([QgsPoint(float(df2.at[j,'absc']-
df2.at[j,'L_XY_tubo']-df2.at[j,'d_pz_ini']/2), escala*float(df2.at[j,'bat_ini'])),
        QgsPoint(float(df2.at[j,'absc']-df2.at[j,'d_pz_ini']/2),
escala*float(df2.at[j,'bat_fin']))))))
        (res, outFeats) = layer.dataProvider().addFeatures([fet])
        fet.setGeometry(QgsGeometry.fromPolyline([QgsPoint(float(df2.at[j,'absc']-
df2.at[j,'L_XY_tubo']-df2.at[j,'d_pz_ini']/2), escala*float(df2.at[j,'cla_ini'])),
        QgsPoint(float(df2.at[j,'absc']-df2.at[j,'d_pz_ini']/2),
escala*float(df2.at[j,'cla_fin']))))))
        (res, outFeats) = layer.dataProvider().addFeatures([fet])
        fet.setGeometry(QgsGeometry.fromPolyline([QgsPoint(float(df2.at[j,'absc']-
df2.at[j,'L_XY_tubo']-df2.at[j,'d_pz_ini']/2), escala*float(df2.at[j,'z_ini'])),
        QgsPoint(float(df2.at[j,'absc']-df2.at[j,'d_pz_fin']/2),
escala*float(df2.at[j,'z_fin']))))))
        (res, outFeats) = layer.dataProvider().addFeatures([fet])

layer.updateExtents()

layer_settings = QgsPalLayerSettings()
layer_settings.fieldName = "num"
layer_settings.placement = 2
layer_settings.enabled = True
layer_settings = QgsVectorLayerSimpleLabeling(layer_settings)

```

```

        layer.setLabelsEnabled(True)
        layer.setLabeling(layer_settings)
        layer.triggerRepaint()

print("Perfiles generados")

def borrar_corrida():
    if os.path.isfile(os.path.join(files_path, 'red_calculada.csv')):
        os.remove(os.path.join(files_path, 'red_calculada.csv'))
    if os.path.isfile(os.path.join(files_path, 'pozos.csv')):
        os.remove(os.path.join(files_path, 'pozos.csv'))
    if os.path.isfile(os.path.join(files_path, 'red_calculada_SWMM.csv')):
        os.remove(os.path.join(files_path, 'red_calculada_SWMM.csv'))
    if os.path.isfile(os.path.join(files_path, 'pozos_SWMM.csv')):
        os.remove(os.path.join(files_path, 'pozos_SWMM.csv'))
    if os.path.isfile(os.path.join(files_path, 'red.inp')):
        os.remove(os.path.join(files_path, 'red.inp'))
    if os.path.isfile(os.path.join(files_path, 'red.dat')):
        os.remove(os.path.join(files_path, 'red.dat'))
    if os.path.isfile(os.path.join(files_path, 'red.rpt')):
        os.remove(os.path.join(files_path, 'red.rpt'))
    print("Corrida previa eliminada")

def simula_SWMM():
    global df
    global pozos
    global files_path
    df_SWMM=df

    pozos_SWMM=pozos
    archivoINP=os.path.join(files_path, 'red.inp')
    archivoRPT=os.path.join(files_path, 'red.rpt')
    archivoDAT=os.path.join(files_path, 'red.dat')

    from swmm5.swmm5tools import SWMM5Simulation
    from swmm5 import swmm5 as sw
    st=SWMM5Simulation(archivoINP)

#Resultados
    pozos_SWMM['SWMM_WS']=0.0
    pozos_SWMM['SWMM_H']=0.0
    pozos_SWMM['SWMM_Q']=0.0
    pozos_SWMM['SWMM_Qt']=0.0

    df_SWMM['SWMM_Q']=0.0
    df_SWMM['SWMM_Y']=0.0
    df_SWMM['SWMM_V']=0.0

    for i, row in pozos_SWMM.iterrows():

```

```

    for j in range(0, int(st.SWMM_Nnodes), 1):
        if pozos_SWMM.at[i,'pozo']==st.Node()[j]:
#0 Depth of water above invert (ft or m) - Variable: SWMM_WS
        pozos_SWMM.at[i,'SWMM_WS']=list(st.Results('NODE',st.Node()[j], 0))[0]
#1 Hydraulic head (ft or m) - Variable: SWMM_H
        pozos_SWMM.at[i,'SWMM_H']=list(st.Results('NODE',st.Node()[j], 1))[0]
#3 Lateral inflow (flow units) - Variable: SWMM_Q
        pozos_SWMM.at[i,'SWMM_Q']=list(st.Results('NODE',st.Node()[j], 3))[0]
#4 Total inflow (lateral + upstream) (flow units) - Variable: SWMM_Qt
        pozos_SWMM.at[i,'SWMM_Qt']=list(st.Results('NODE',st.Node()[j], 4))[0]

    for i, row in df_SWMM.iterrows():
        for j in range(0, int(st.SWMM_Nlinks), 1):
            if df_SWMM.at[i,'idd']==st.Link()[j]:
#0 Flow rate (flow units) - Variable: SWMM_Q
            df_SWMM.at[i,'SWMM_Q']=list(st.Results('LINK',st.Link()[j], 0))[0]
#1 Flow depth (ft or m) - Variable: SWMM_Y
            df_SWMM.at[i,'SWMM_Y']=list(st.Results('LINK',st.Link()[j], 1))[0]
#2 Flow velocity (ft/s or m/s) - Variable: SWMM_V
            df_SWMM.at[i,'SWMM_V']=list(st.Results('LINK',st.Link()[j], 2))[0]

#Genera archivos .dat y .rpt
    ret=sw.RunSwmmDll(archivoINP,archivoRPT,archivoDAT)

## Exporta la red calculada con informacion de modelacion con SWMM
# csv con la informacion de los pozos
    pozos_SWMM.to_csv(os.path.join(files_path, 'pozos_SWMM.csv'))
# csv con la red calculada
    df_SWMM.to_csv(os.path.join(files_path, 'red_calculada_SWMM.csv'))
    print("Simulacion en SWMM ejecutada")

class IniciarWidget(DialogType, DialogBase):
    def __init__(self, parent=None):
        super().__init__(parent)
        self.setupUi(self)
        self.setWindowFlags(Qt.WindowStaysOnTopHint)
        self.setWindowIcon(QIcon(os.path.join(plugin_path, 'data\\img', 'icon2.png')))
        self.mQgsFileWidget.setDefaultRoot(os.path.join(plugin_path, 'data\\ejemplos'))
        app=None
        app = wx.App(False)
        self.setGeometry(wx.GetDisplaySize()[0]-350-20, int(wx.GetDisplaySize()[1]/2-200),
350, 400) # left, top, width, height

        self.toolButton_1.setIcon(QIcon(os.path.join(plugin_path, 'data\\img', 'help.png')))
        self.toolButton_2.setIcon(QIcon(os.path.join(plugin_path, 'data\\img', 'help.png')))
        self.toolButton_3.setIcon(QIcon(os.path.join(plugin_path, 'data\\img', 'help.png')))
        self.toolButton_4.setIcon(QIcon(os.path.join(plugin_path, 'data\\img', 'help.png')))
        self.toolButton_5.setIcon(QIcon(os.path.join(plugin_path, 'data\\img', 'help.png')))
        self.toolButton_6.setIcon(QIcon(os.path.join(plugin_path, 'data\\img', 'help.png')))

```

```

# Signals
self.pushButton_abrir_proyecto.clicked.connect(self.cargar)
self.pushButton_borrar_corrida.clicked.connect(self.borrar)
self.pushButton_perfil.clicked.connect(self.ver_perfil)
self.pushButton_exporta.clicked.connect(self.exporta)
self.pushButton_SWMM.clicked.connect(self.corre_SWMM)

def cargar(self):
    global df
    global escala
    global files_path
    global plugin_path
    files_path= self.mQgsFileWidget.filePath()
    print("\nProyecto cargado")

    if (os.path.isfile(os.path.join(files_path, 'red_calculada.csv')) and
        os.path.isfile(os.path.join(files_path, 'pozos.csv')) and
        os.path.isfile(os.path.join(files_path, 'red.inp')) and
        os.path.isfile(os.path.join(files_path, 'red.dat')) and
        os.path.isfile(os.path.join(files_path, 'red.rpt'))):
        carga_datos_existentes()
    else:
        carga_datos()
        inicializar()
        recalcular()
        procesar()
        visualiza()

def borrar(self):
    borrar_corrida()

def ver_perfil(self):
    ver_perfiles()

def exporta(self):
    exportar()

def corre_SWMM(self):
    try:
        simula_SWMM()
    except:
        print("Paquete SWMM5 No instalado")

class Formulario:
    def __init__(self, iface):
        self.iface = iface

```

```
def initGui(self):
    self.action = QAction(u'Iniciar/Cargar diseño de red de alcantarillado',
self.iface.mainWindow())
    self.action.triggered.connect(self.run)
    self.action.setIcon(QIcon(os.path.join(plugin_path, 'data\\img', 'icon2.png')))
    self.iface.addToolBarIcon(self.action)

def unload(self):
    print("Plugin recargado")
    self.iface.removeToolBarIcon(self.action)
    del self.action

def run(self):
    self.dlg = IniciarWidget()
    self.dlg.show()
    print("Inicializando")
```