

Díálogo

16 de agosto de 2020

Buenas tardes profesores de la comisión e invitados. Mi nombre es Gabriel Sanhueza y procederé a presentar la defensa para mi proyecto de titulación.

1. Introducción

1.1. Contexto

Como bien se sabe la escasez de agua potable es sin duda una problemática a nivel mundial. Es por esto que optimizar los sistemas que permiten la distribución de agua es importante. Sin embargo, esto no es simple puesto que los sistemas de distribución de agua deben estar activos las 24 horas del día asegurando los niveles de servicio mínimos. Adicionalmente, la optimización de estos sistemas involucra la participación de múltiples criterios lo que dificulta la toma de decisiones respecto a éstos.

Dentro del área de las redes de agua potable nos encontramos con dos clases de problemas. Los problemas de diseño y los de operación.

1.2. Problema

Los problemas identificados con este proyecto son los siguientes:

Primero, no se cuenta con las suficientes herramientas y tiempo para la correcta gestión de las redes de distribución de agua potable (RDA). Por lo tanto, no es posible utilizar los recursos asociados de forma eficiente en estos sistemas.

Junto a lo anteriormente mencionado, escoger las especificaciones de una RDA es una tarea difícil debido a que hay que evaluar el rendimiento general del sistema en función de un conjunto de variables que se mueven en un rango muy elevado de posibilidades.

1.3. Propuesta de solución

La solución que se propone para abordar el problema consiste en el desarrollo de una aplicación de escritorio escalable que permita optimizar el proceso de diseño y operación en redes de agua potable.

Los problemas que se propone implementar son dos:

El problema de diseño basado en el costo de tuberías abordado utilizando el Algoritmo Genético.

El problema problema de operación basado en el régimen de bombeo abordado a través del algoritmo Non-Dominated Sorting Genetic Algorithm II.

La elección de estos problemas se debe a los intereses de los integrantes del equipo del proyecto “Optimization of real-world water distribution systems and hydraulic elements using computational fluid dynamics (cf) and evolutionary algorithms”.

1.4. Objetivo general

El objetivo general de este trabajo es el:

Diseñar y desarrollar una aplicación extensible de escritorio para optimizar el diseño y operación de una red de distribución de agua potable.

1.5. Objetivos específicos

En cuanto a los objetivos específicos se tienen los siguientes:

- Diseñar software orientado a la optimización de RDA basado en la arquitectura lógica del framework multiobjetivo Jmetal.
- Implementar un algoritmo metaheurístico de optimización monoobjetivo para aplicar al problema de diseño de RDA.
- Implementar un algoritmo metaheurístico de optimización multiobjetivo para aplicar al problema de Régimen de bombeo en RDA.
- Diseñar e implementar la interfaz gráfica del sistema de optimización de redes de agua potable desarrollado durante este proyecto.

2. Metodología de desarrollo

La metodología escogida para llevar a cabo el desarrollo de este proyecto es la “Iterativa e Incremental”. Esta metodología divide el proceso de construcción del programa en iteraciones que se componen de 4 fases las que son Análisis, Diseño, Implementación y Pruebas.

Las adaptaciones que son realizadas sobre la metodología consisten en disminuir la cantidad de documentación generada, permitir la realización de más de una fase a la vez y finalmente una única persona desempeña todos los roles establecidos para la metodología.

3. Desarrollo

Ahora se procede a hablar acerca del desarrollo del proyecto.

3.1. Concepción del proyecto

Este proyecto se origina como una propuesta por parte del profesor Daniel Mora Melia del departamento de Ingeniería en Obras Civiles. El junto a un grupo de colaboradores forman parte del proyecto “Optimization of real-world water distribution systems and hydraulic elements using computational fluid dynamics (cfd) and evolutionary algorithms” el cual busca aplicar algoritmos metaheurísticos en la resolución de problemas asociados a las RDA. De este proyecto, surgieron varios artículos, entre ellos, se encuentra uno que consiste en la integración de dos herramientas independientes JMetal y Epanet para llevar a cabo optimizaciones sobre RDA. De dicho artículo surge la idea de implementar una herramienta gráfica con el fin de facilitar la optimización de redes de agua potable.

3.2. Tecnologías a utilizar

Las tecnologías utilizadas fueron Java, como lenguaje de programación, JavaFX, para implementar la interfaz y la librería EpanetToolkit para realizar las simulaciones hidráulicas y evaluar las soluciones generadas por los algoritmos metaheurísticos.

La elección de Java como lenguaje de programación se debe a que éste fue el solicitado por el cliente, puesto que permite la incorporación directa de algunas herramientas ya desarrolladas.

3.3. Requisitos

Durante la fase de requisitos, se capturaron y priorizaron los requisitos generados en cada iteración agregando estos finalmente a un documento de especificación formal de requisitos.

En total se obtuvieron 32 requisitos de usuarios y se generó como salida de esta fase un documento en que se especifican estos requisitos siguiendo el formato mostrado en la ilustración.

3.4. Diseño

Durante la fase de diseño se realizó la conceptualización del software generando como resultado un documento de diseño en que se especifica la arquitectura física y arquitectura lógica, así como una serie de diagramas de clase y secuencia junto con los diseños de las interfaces.

La arquitectura física utilizada consiste en la monolítica, puesto que la aplicación no tiene interacción con otros sistemas computacionales y está diseñada para operar localmente en el equipo del usuario que la utilice.

En cuanto a la arquitectura lógica, ésta consiste en la arquitectura Modelo-Vista-Controlador la cual fue escogida porque nos permite separar los componentes de la aplicación en módulos reduciendo la dependencia entre el código de la interfaz de usuario y la lógica de la aplicación. (Y la librería JavaFX trabaja en base a esta arquitectura)

3.5. Implementación

Durante la fase de implementación se realizó la codificación de las funcionalidades de la aplicación y el manual de usuario de ésta.

A continuación se mostrara como resultado de esta fase las interfaces gráficas implementadas para la aplicación.

La primera interfaz corresponde a la principal desde esta interfaz se pueden visualizar el esquema de la red, ver los elementos que la conforman y su configuración, realizar la simulación utilizando los valores definidos en el archivo de configuración de red y seleccionar el problema a optimizar.

En la ventana de configuración del problema se puede ver una descripción de éste y establecer las configuraciones utilizadas para realizar la optimización.

Para la aplicación se tienen dos ventanas de estado de ejecución. La primera correspondiente a la imagen de la izquierda es la utilizada para los problemas monoobjetivos, mientras que la derecha se utiliza para los multiobjetivos.

También, desde la ventana de estado se puede visualizar el gráfico de la ejecución, si este está activo. En este caso se muestra el gráfico de la ejecución monoobjetivo, en donde cada color corresponde a una ejecución independiente del algoritmo.

En la ventana de resultados como su nombre lo dice se visualizan los resultados de la optimización. En esta ilustración, se presentan los resultados para cada ejecución independiente de un algoritmo monoobjetivo, marcando con el color verde la mejor de las soluciones encontradas.

En la ventana de simulación se muestran los resultados de la simulación utilizando la configuración del archivo de configuración de red.

3.5.1. Java Reflection y Java Annotation

Una de las capacidades más importantes de la aplicación que es la extensibilidad, se logra haciendo uso de dos funcionalidades de Java, las cuales son la reflexión y las anotaciones.

Las anotaciones fueron utilizadas para agregar metadatos a los operadores y a la clase que hereden de la interfaz *Registrable*. Estas anotaciones son utilizados para identificar los parámetros requeridos para cumplir con su función.

Los Experimentos son contenedores que poseen varias instancias de un mismo algoritmo para resolver un problema, esto le da a la aplicación la capacidad de realizar simulaciones independientes.

Para agregar un nuevo algoritmo, operador o problema se debe implementar la interfaz correspondiente, presentada en el diagrama y posteriormente crear una clase que herede de *Registrable*. Ésta clase es utilizada como una plantilla para la creación de los Experimentos. Posteriormente, la clase debe ser registrada en una estructura de datos presente dentro de la aplicación.

Estas dos funcionalidades permiten que el usuario que quiera agregar nuevos operadores o problemas no necesite modificar el código de la interfaz de usuario para que los nuevos elementos sean visualizados en la aplicación.

3.5.2. Problema de diseño de RDA basado en el costo de las tuberías

Ahora hablare un poco de los problemas y algoritmos implementados por defecto.

El primero de estos problemas consiste en el problema de diseño abordado desde el enfoque monoobjetivo. Éste, busca optimizar el costo de inversión de las tuberías el cual se calcula como la sumatoria del costo por diámetro por el largo de la tubería.

3.5.3. Problema de operación de RDA basado en el régimen de bombeo

En cuanto al problema operación, es utilizando el enfoque multiobjetivo para optimizar el costos energético, así como el costo de mantención de las bombas.

El costo energético para cada bomba en un tiempo determinado se calcula como precio de la energía por el consumo energético por el estado de la bomba, tomando este último el valor 0 o 1 dependiendo si la bomba esta apagada o encendida respectivamente.

En cuanto al costo de mantención, éste se calcula como “el numero de cambios de estado desde apagado a encendido por cada bomba en cada uno de los periodos de tiempo simulados”

3.5.4. GA

El algoritmo implementado para el problema de diseño mencionado anteriormente es el Algoritmo Genético. Los pasos para realizar este algoritmo son:

1. Crear la población inicial
2. Evaluar ésta población
3. Iterar hasta alcanzar la condición de término establecida aplicando los operadores de selección y reproducción para generar una nueva población.
4. Una vez alcanzada la condición de termino se obtiene el mejor individuo de la población.

3.5.5. NSGAII

En cuanto al algoritmo implementado para el problema de operación se encuentra el algoritmo NSGAII. Éste algoritmo sigue el mismo procedimiento mencionado anteriormente para el Algoritmo Genético, pero en vez de obtener como resultado una única solución se obtiene un conjunto de ellas conocido como la Frontera de Pareto.

Una cosa importante a mencionar para este algoritmo es la función de remplazo utilizada.

Para realizar el remplazo se deben operar tanto la población original asi como la descendiente generada a través del proceso de selección y reproducción.

Estas dos poblaciones se agrupan y posteriormente se ordenan sus individuos en diversos Frentes formado por las soluciones que no se dominan entre sí. Posteriormente, se agregan a la nueva población las soluciones de los frentes hasta alcanzar el tope establecido. Si el último frente agregado no cabe en su totalidad, éste debe ser ordenado en base a su densidad y las mejores soluciones son las conservadas.

3.6. Pruebas

Durante la fase de pruebas se realizaron pruebas manuales y automatizadas, las cuales fueron especificadas en su documento respectivo.

En total se realizaron 39 pruebas automatizadas y 16 pruebas manuales.

En el documento de especificación de los casos de prueba se utiliza el formato presentado en la ilustración que se compone de el ID de la prueba, el título, la característica a evaluar, el objetivo del caso, la configuración de la aplicación al momento de realizar la prueba, los datos utilizados para la prueba, las acciones realizadas para ejecutar el caso y finalmente el resultado esperado. Este formato se usa tanto para las pruebas automatizadas como las manuales.

4. Metodología de evaluación

La metodología de evaluación escogida para aplicar sobre el sistema construido consiste en los estudios de caso. El estudio de caso es una metodología de investigación que permite analizar un proyecto en un contexto real de aplicación con el fin de responder la pregunta de investigación planteada.

El estudio de caso se dividió en las siguientes etapas. Diseño del estudio de caso, recolección de datos, análisis de resultados y presentación de los resultados y conclusiones.

Durante el diseño se definió el caso a estudiar, los objetivos del estudio, los protocolos para llevar la recolección de datos a cabo, las características que se buscan evaluar y los sujetos de prueba para la recolección de la información.

Durante la recolección de datos se aplicó el estudio a los sujetos de prueba con el fin de recabar datos que posteriormente fueron analizados.

Durante el análisis de datos se analizaron los datos recolectados.

Y finalmente durante la presentación de los resultados se dan a conocer las conclusiones del estudio.

5. Evaluación

A continuación se presentan los detalles de la evaluación realizada.

5.1. Diseño del caso

Como estudio de caso se establece la aplicación desarrollada durante este proyecto con el objetivo de la investigación responder a la pregunta “¿Cómo el

sistema desarrollado trabaja en la practica?”.

Las características de la aplicación evaluadas fueron las siguientes:

- **Funcionalidad:** Con esta característica se busca evaluar lo que la aplicación puede hacer, es decir, si el sistema cumple con las operaciones solicitadas por el cliente.
- **Usabilidad:** Esta característica busca medir la usabilidad del software, es decir, que tan fácil es de usar la aplicación.
- **Utilidad:** Con este criterio se busca medir el valor de la aplicación. Con valor, nos referimos a si las funcionalidades implementadas por el sistema son de utilidad para quien haga uso de solución.
- **Utilidad del manual de usuario:** Con este criterio se busca evaluar el manual de usuario realizado y si este fue de ayuda para aprender y poder conocer la aplicación.

Como protocolo para realizar el caso se estableció que el caso sería aplicado de manera asíncrona debido a las problemáticas a nivel país y la dificultad de coordinar una reunión. Y que los pasos para realizar el caso consisten en:

- Enviar la aplicación junto con su manual a los participantes del estudio.
- Dar un tiempo para que el participante del estudio se interiorice con la aplicación.
- Enviar la encuesta a los participantes.

Cómo unidad de análisis para llevar a cabo es estudio se escogieron a profesionales con conocimientos en computación, hidráulica y metaheurísticas.

Como consideraciones para aplicar el caso, los encuestados debían tener un equipo con un sistema operativo Windows de 64 bits y la versión de Java 1.8 instalada. Adicionalmente, se establece que éstos deben contar con conocimientos en metaheurísticas e hidráulica para interpretar los resultados y en caso de que quisieran agregar nuevos problemas o algoritmos también deben saber programación, específicamente en Java.

5.2. Recolección de datos

Como participantes del caso de estudio se contó con la participación de la profesora Yamsleydi Salgueiro, el profesor Marco Alsina y Sergio Silva.

Los instrumentos utilizados para la evaluación son encuestas utilizando la escala de Likert. En el caso del test de usabilidad este se realizó utilizando un test de usabilidad bien conocido llamado SUS el cual cuenta con 10 preguntas realizadas en un orden específico y que permiten medir el nivel de usabilidad de la aplicación.

5.3. Análisis de datos

Como resultado de la evaluación de la funcionalidad se obtuvo que el 92.3 % de las respuestas indican que los requisitos establecidos fueron cumplidos.

Como resultado de la evaluación de usabilidad se obtuvo un puntaje promedio de 79.2 % con una desviación estándar de 9.5 % dando como resultado, utilizando la escala de la ilustración una evaluación de “Buena”.

En cuanto a la evaluación de la utilizada aproximadamente el 89 % de las respuestas apuntan a que la aplicación es útil para realizar tareas de optimización en redes de agua potable.

Como parte de la evaluación utilizada para evaluar el manual de usuario, el 44 % de las respuestas indican que este fue útil, mientras que el 22 % indica que no fue útil para comprender la aplicación.

5.4. Conclusión estudio de caso

Para concluir, del caso de estudio realizado se establece que para la aplicación, los criterios de funcionalidad, usabilidad y utilizad fueron cumplidos. Mientras que para el manual de usuario de la aplicación se establece que este no era deficiente debido a la falta de más detalles o ejemplos o a la carencia de algunos temas.

6. Conclusión

Finalmente, como resultado de este proyecto se posee una aplicación escalable, la cual permite extender los algoritmos, problemas u operadores sin necesidad de conocer en detalle el funcionamiento total de la aplicación. Adicionalmente, la aplicación incorpora por defecto 2 problemas y dos algoritmos. Los cuales son el problema de diseño basado en el costo de las tuberías y el problema de operación basado en el régimen de bombeo junto con el Algoritmo Genético y el algoritmo NSGAIL.

Adicionalmente, se establece que los objetivos establecidos al comienzo del desarrollo son logrados exitosamente y que la metodología aplicada era la adecuada para generar la suficiente documentación con el fin de permitir a otros desarrolladores continuar con el desarrollo del proyecto.

Finalmente se establece como resultado de la evaluación del software que la aplicación es valorada positivamente, mientras que su manual debe ser mejorado.

7. Trabajo Futuro