

<Título de la propuesta de proyecto> (Propuesta de proyecto final de carrera)

Gabriel Sanhueza Fuentes (Estudiante)
Jimmy Gutierrez Bahamondes (Profesor guía)
Carrera de Ingeniería Civil en Computación
Universidad de Talca

6 de septiembre de 2019

1. Descripción de la propuesta

En el presente capítulo se presentan los conceptos básicos, el problema y su contexto, y la propuesta de solución.

1.1. Conceptos básicos del proyecto

En esta sección se definen algunos conceptos básicos que serán necesarios para la realización del proyecto.

- **Optimización:** La optimización consiste en maximizar o minimizar un conjunto de funciones

$$f_1(x), f_2(x), \dots, f_N(x), x = (x_1, \dots, x_d) | x \in X$$

sujeto a una serie de condiciones

$$h_j(x) = 0, j = 1, 2, \dots, J$$

$$g_k(x) \leq 0, k = 1, 2, \dots, K$$

siendo f_1, \dots, f_N funciones objetivos y h_j junto con g_k una serie de restricciones [1]. En donde de acuerdo a [Optimización multi-objetivo: aplicaciones a problemas del mundo real], X corresponde el espacio de búsqueda, mientras que los valores obtenidos $f_n(x)$ al espacio objetivo. De acuerdo a la cantidad de funciones objetivos que se tenga, se establece que si $N = 1$ la optimización es **mono-objetivo**, mientras que para $N \geq 2$ es **multi-objetivo** [1].

Debido a la definición de las restricciones es posible dividir el espacio de búsqueda en dos regiones [2]:

- Soluciones factibles: Compuesto por los elementos pertenecientes al espacio de búsqueda que satisfacen todas las restricciones.
 - Soluciones no factibles: Integrado por aquellos elementos que no cumplen todas las restricciones.
- **Heurísticas:** Métodos que permiten obtener soluciones de calidad haciendo uso de un bajo nivel de recursos y en un tiempo reducido [2].
 - **Metaheurística:** Heurísticas generalizadas para ser aplicadas en un amplio conjunto de problemas [2].
 - **Algoritmos genéticos:** Algoritmo que busca soluciones óptimas dentro del espacio de soluciones. Para realizar esto, el algoritmo parte desde una solución candidata y lleva a cabo un proceso de reproducción, generando nuevas soluciones. A medida que se agregan nuevas soluciones en la población, soluciones de generaciones anteriores mueren [3].
 - **NSGA-II:** Algoritmo que utiliza la cruce, mutación y reproducción para encontrar un conjunto de soluciones óptimas a problemas que cuentan con más de un objetivo [?].
 - **Epanet:** Software que permite simular el comportamiento hidráulico y la calidad del agua en redes de distribución de aguas compuesta por tuberías, nodos, bombas, válvulas y tanques de almacenamiento [4].
 - **Red de distribución de agua potable:** Conjunto de elementos enlazados de tal manera que permite suministrar cierta cantidad de agua a una presión establecida [5].

1.2. Contexto del proyecto

La escasez de agua potable es sin duda una problemática a nivel mundial. Dentro de este contexto, la optimización de los sistemas de distribución de agua potable es un problema sin resolver hasta la fecha.

A lo mencionado anteriormente, también se suma la escasez energética. Puesto que, es requerida para el tratamiento y distribución del agua. Es por esto que, es importante hacer un uso eficiente de la energía durante la operación del sistema.

La optimización de estos sistemas, a la vez, involucra la participación de múltiples criterios que deben ser tomados en cuenta a la hora de decidir. Sin embargo, la

incorporación de estos criterios, involucra la generación de modelos cada vez más complejos.

Los algoritmos metaheurísticos han demostrado ser un mecanismo eficiente ante problemas de este tipo. Ya que, estos reducen el tiempo necesario que toma el evaluar todas las configuraciones posibles. Dado que, juzgan un conjunto menor de valores logrando como resultado una aproximación a la solución óptima.

1.3. Definición del problema

Los encargados de implementar sistemas de distribución de agua potable, no cuentan con suficientes herramientas y tiempo para su correcta gestión. Por lo tanto, no es posible utilizar los recursos asociados de forma eficiente. Además, las herramientas existentes no satisfacen las necesidades de usabilidad y costo, debido a que son poco intuitivas y de pago.

El escoger las especificaciones de una red de agua potable ya es de por sí difícil debido a que hay que evaluar el rendimiento general del sistema alternando entre distintas configuraciones en busca de una solución que sea eficaz. Debido a esto, el uso de herramientas automatizadas que evalúen el rendimiento de las diversas combinaciones posibles viene a ser necesario.

A lo anterior se suma el hecho de que los interesados en esta área no manejan herramientas informáticas.

1.4. Trabajo relacionado

En este apartado se mostraran algunos trabajos parecidos al que tiene como resultado este proyecto.

- **Magmoredes:** En [6] se describe la existencia de un software de diseño basado en micro-algoritmos genéticos multiobjetivos, que de acuerdo al autor, tiene un mejor rendimiento y es mas eficiente debido a que requiere una menor cantidad de memoria y tiene un tiempo de computo que el algoritmo NSGA-II. Este programa puede cargar cualquier red y realiza los cálculos utilizando librerías de java. Sin embargo, este programa tiene como objetivo optimizar el costo y la confiabilidad final de la red.
- **Hidra software:** Es un Software de paga que permite el diseño y la realización de distintos tipos de cálculos, como la capacidad de estanque, los recursos requeridos para la construcción de la red, demanda, diámetros, etc.
- **WaterGEMS:** De acuerdo a [7], este software permite la construcción de modelos geoespaciales; optimización de diseño, ciclos de bombeo y calibración automática del modelo; y la gestión de activos.

1.5. Propuesta de solución

(Esta sección debe incluir el planteamiento y justificación de la solución y/o idea, incluyendo aspectos novedosos.) Esta sección responde a la pregunta ¿Cómo voy a resolver el problema planteado?

Para resolver el problema, se llevara a cabo la implementación de una aplicación de escritorio. Esta aplicación, permitirá buscar posibles soluciones a los siguientes problemas:

- Problema de diseño: E
- Problema del régimen de bombeo: Este problema se abarcara desde el enfoque multiobjetivo a través del algoritmo conocido como NSGA-II. Este algoritmo podrá ser ejecutado con distintos operadores los cuales estarán disponibles en el sistema. Como solución otorgara los ...

2. Objetivos

A continuación se darán a conocer el objetivo general y los objetivos específicos que se desean lograr con el desarrollo del proyecto.

Objetivo general

- Diseñar y desarrollar una aplicación de apoyo a la toma de decisiones, integrando algoritmos de optimización aplicados al problema de diseño y operación en redes de distribución de agua potable.

Objetivos específicos *(Una lista de puntos que detallan el objetivo general.)*

- Generar soluciones frente al problema monoobjetivo de diseño de redes de distribución de agua potable a través de la implementación de algoritmos genéticos.
- Generar soluciones frente al problema multiobjetivo de operación de redes de distribución de agua potable a través de la implementación de NSGA-II.
- Diseñar he implementar interfaz grafica del sistema.
- Evaluar el desempeño de los algoritmos, contrastando los resultados obtenidos en redes de benchmarking con óptimos conocidos.

3. Alcances

- Este trabajo no contempla la creación de la red, por lo que estas serán ingresadas como entradas.
- Este trabajo solo contempla la utilización e implementación del algoritmo genético y NSGA-II.
- El producto obtenido a través de la realización de este trabajo solo será compatible con el sistema operativo Windows. Esto se debe a que harán llamadas a librerías nativas para realizar los cálculos de las redes.

4. Metodología

En esta sección se describirá y justificará la metodología de desarrollo y la metodología de evaluación que se usarán a lo largo del desarrollo del proyecto..

4.1. Metodología de desarrollo

Esta sección describe y justifica la metodología de desarrollo que se utilizará.

4.1.1. Iterativo incremental

Esta metodología lleva a cabo el desarrollo de un proyecto de software dividiéndolo en iteraciones que generan un incremento, el cual contribuye en el desarrollo del producto final. Cada iteración se compone de las fases de análisis, diseño, implementación y testing. La fase de análisis lleva a cabo la obtención y definición de los requerimientos del software. La etapa de diseño se encarga de la conceptualización del software basado en los requerimientos definidos anteriormente. Durante la implementación se codifican las funcionalidades siguiendo las directivas establecidas durante el diseño, con el fin de satisfacer los requerimientos. Y finalmente, durante la fase de testing, se valida y verifica la correctitud de las funcionalidades implementadas, así como el cumplimiento de los requisitos. El hecho de llevar a cabo un desarrollo iterativo permite la obtención de retroalimentación del producto que se está desarrollando tempranamente y de esta manera poder refinar el trabajo en etapas posteriores del desarrollo. [8, 9, 10, 11].

Debido a que la metodología está creada para ser llevada a cabo por un equipo de trabajo se adaptará la metodología para poder ser aplicada en el desarrollo llevado a cabo por una sola persona. Esta adaptación consiste en la disminución de la cantidad de la documentación generada, permitir llevar a cabo más de una fase de la iteración al mismo tiempo, el rol de analista, diseñador, implementador y tester

sera realizado por una sola persona. Los documentos a realizar serán el de requisitos y el de diseño. El documento de requisitos contara con los requerimientos de usuario y de sistema. En el documento de diseño se documentara el como la aplicación será implementada utilizando los diagramas de casos de uso, arquitectura lógica y diagrama de clases.

Debido a que la metodología esta creada para ser llevada a cabo por un equipo de trabajo se adaptara la metodología para poder ser aplicada en el desarrollo llevado a cabo por una sola persona. Esta adaptación consiste en lo siguiente:

- Disminución de la cantidad de la documentación generada.
- Se podra llevar a cabo mas de una base de la iteración a la vez.
- Las actividades de los roles de analista, diseñador, implementador y tester sera realizado por una sola persona.
- Los documentos a realizar serán el de requisitos y el de diseño. El documento de requisitos contara con los requerimientos de usuario y de sistema. En el documento de diseño se documentara el como la aplicación será implementada utilizando los diagramas de casos de uso, arquitectura lógica y diagrama de clases.

La razón por la que se utilizara esta metodología sobre otras es porque el producto resultante de este proyecto esta pensado para servir como base para futuros trabajos. Debido a esto es necesario documentar correctamente para que otros programadores puedan continuar con su desarrollo en el futuro. Aunque existen otras metodologías como cascada u otras tradicionales, estas son difíciles de llevar a cabo por la cantidad de documentación que se requiere, mientras que metodologías de desarrollo ágil carecen en cuanto a la documentación que se necesita para el sistema a desarrollar. Ademas, esta metodología nos permite obtener una retroalimentacion al final de la iteración, obtener nuevos requisitos que no hayan quedado definidos en etapas anteriores o refinar los requisitos y el diseño ya existente, permitiendo asi mejorar la calidad del producto final.

La implementación de esta metodología para el desarrollo del proyecto se llevara a cabo repartiendo las tareas necesarias para el cumplimiento de los objetivos en iteraciones. De este modo al final de cada iteración se contará con un prototipo funcional de la aplicación sobre el que se agregaran las nuevas funcionalidades en las iteraciones siguientes.

4.2. Metodología de evaluación del proyecto

(En esta subsección se deben describir y justificar los metodos de evaluación/validación que se aplicarán a lo largo del desarrollo del proyecto.)

4.2.1. Pruebas unitarias

4.2.2. Pruebas de integración

4.2.3. Caso de estudio

Por ejemplo:

- Metodología de evaluación basado en experimentos
- Metodología de evaluación basado en casos de estudio
- Metodología de evaluación basado action research

Parte I

5. Plan de trabajo

(En esta sección se debe definir como organizar y planificar, en términos de etapas y tiempo, las actividades a desarrollar así como los resultados a obtener. Cada actividad debe generar un entregable. Se entiende como entregable a la generación de artefactos que sean útiles al desarrollo del proyecto, por ejemplo: documentos bibliográficos con resumen, documentos de desarrollo de software, código fuente, planificaciones, diagramas, esquemas, algoritmos, etc.)

(Esta sección debe incluir además una Carta Gantt, la cual define fechas de inicio y término. La longitud máxima de esta sección es de 1 página, sin considerar la Carta Gantt.)

Etapas 1: Desarrollar el objetivo 1 (inicio-término)

- Analizar ... (inicio-término)
- Redactar ... (inicio-término)

Etapas 2: Desarrollar los objetivos 2 y 3 (inicio-término)

- Diseñar ... (inicio-término)
- Programar ... (inicio-término)
- Redactar ... (inicio-término)

Referencias

- [1] Xin She Yang. Metaheuristic Optimization What is Metaheuristic Optimiza-
tion. 6(2011):1–17, 2015.
- [2] Javier López. *Optimización multi-objetivo: aplicaciones a problemas del
mundo real*. D - Editorial de la Universidad Nacional de La Plata, La Pla-
ta, ARGENTINA, 2013.
- [3] Dorothea Heiss-Czedik. An Introduction to Genetic Algorithms. *Artificial
Life*, 3(1):63–65, 1997.
- [4] Lewis Rossman. *EPANET 2.0 en Español. Analisis Hidraulico y de Calidad
del Agua en Redes de Distribución de Agua. Manual del Usuario*. 2017.
- [5] Tesis Doctoral. DISEÑO DE REDES DE DISTRIBUCIÓN DE AGUA ME-
DIANTE ALGORITMOS EVOLUTIVOS. ANÁLISIS DE EFICIENCIA.
2012.
- [6] Pino V. Edwin, Valle C. Angely, Condori P. Franz, Mejia M. Jesus, Chava-
rri V. Eduardo, and Alfaro R. Luis. Diseño Óptimo de Redes de Distribución
de Agua Usando Un Software Basado En Microalgoritmos Genéticos Multi-
objetivos. *Ribagua*, 4(1):6–23, 2017.
- [7] Bentley. WaterGEMS® CONNECT Edition. 2017.
- [8] R Victor. Iterative and Incremental Development: A Brief History. 2003.
- [9] Susan M. Mitchell and Carolyn B. Seaman. A comparison of software cost,
duration, and quality for waterfall vs. iterative and incremental development:
A systematic review. *2009 3rd International Symposium on Empirical Soft-
ware Engineering and Measurement, ESEM 2009*, (February 2008):511–515,
2009.
- [10] Robert C Martin. Iterative and Incremental Development (IID). *Design*,
(Iid), 1999.
- [11] Adel Alshamrani and Abdullah Bahattab. A Comparison Between Three
SDLC Models Waterfall Model, Spiral Model, and Incremental/Iterative Mo-
del. *IJCSI International Journal of Computer Science Issues*, 12(1):106–111,
2015.