Software para la optimización de redes de distribución de agua potable

JHawanet Framework

Gabriel Sanhueza

Defensa de Título
Universidad de Talca

Agosto, 2020

Contexto

- Escasez de agua
- Sistemas que deben estar las 24 horas del día activo
- Múltiples criterios para optimizar los sistemas de distribución
- Las redes de agua (RDA) involucran dos tipos de problemas:
  - Problema de diseño
  - Problema de operación

Problema

- No se cuenta con las suficientes herramientas y tiempo para la correcta gestión de las redes de distribución de agua potable.
- El escoger las especificaciones es una tarea difícil debido a que hay que evaluar el rendimiento general del sistema.

Propuesta de solución

Aplicación de escritorio extensible que permita optimizar procesos de diseño y operación en RDA.
Problemas implementados:

- Problema de diseño de RDA basado en el costo de tuberías.
- Problema de operación basado en el Régimen de bombeo.

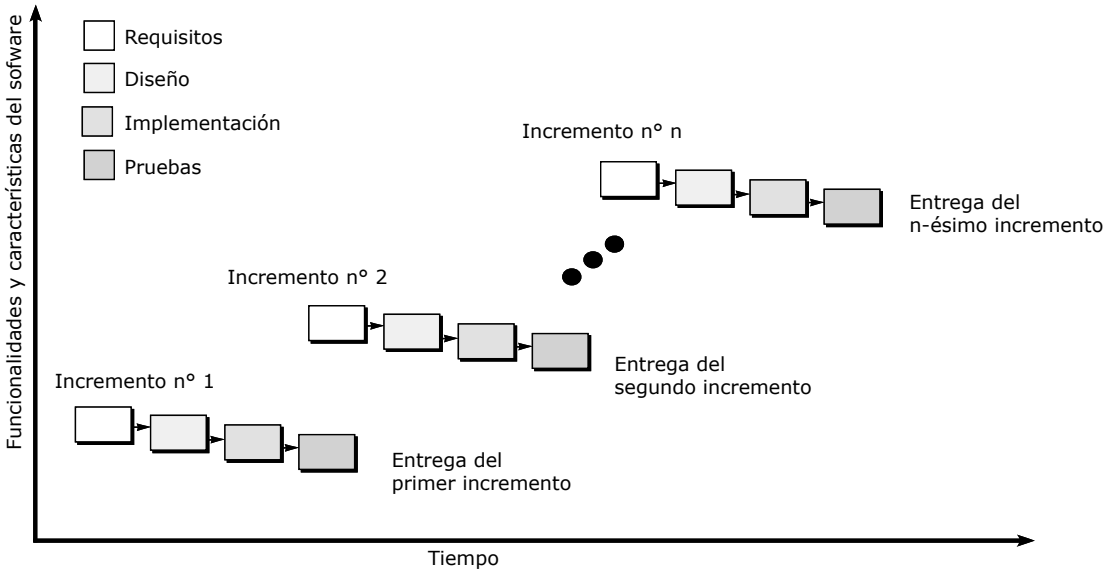
Objetivos

- Objetivo general
  - Diseñar y desarrollar una aplicación extensible de escritorio para optimizar el diseño y operación de una red de distribución de agua.
- Objetivos específicos
  - 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.

Iterativa e incremental

Fases:

- Análisis
- Diseño
- Implementación
- Pruebas



Iterativa e incremental

Adaptaciones realizadas a la metodología

- Disminuir la cantidad de documentación de cada fase
- Llevar a cabo más de una fase en una iteración al mismo tiempo
- Los roles de analista, diseñador, implementador y tester son llevado a cabo por una sola persona.

Concepción del proyecto

- Propuesta por parte del profesor Daniel Mora
- Proyecto Optimization of real-world water distribution systems and hydraulic elements using computational fluid dynamics (cfd) and evolutionary algorithms
- Jmetal + EpanetToolkit

Planificación

- **Iteración 1:** Requisitos y Arquitectura
- **Iteración 2:** Problema monoobjetivo y Algoritmo Genético (GA)
- **Iteración 3:** Interfaz gráfica (GUI)
- **Iteración 4:** Problema multiobjetivo y Algoritmo Non-Dominated Sorting Genetic Algorithm II (NSGAII)
- **Iteración 5:** Experimentos y simulación hidraúlica.
- **Iteración 6:** Afinación de detalles.

Tecnología utilizada

Requisitos

- Captura, priorización y especificación formal de los requisitos
- 32 Requisitos de usuario
- Documento de especificación formal de requisitos.

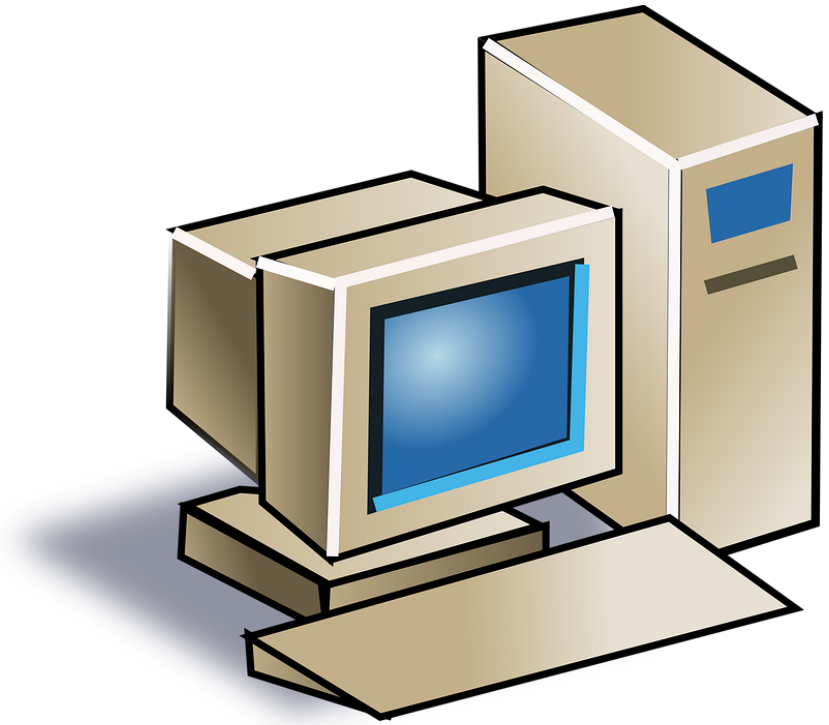
RU004 – Visualizar red en una interfaz grafica.	
Descripción:	Se debe mostrar en la interfaz gráfica una representación de la red (Un dibujo, etc) generada a partir de la información contenida en el archivo inp.
Fuente:	Jimmy Gutiérrez
Prioridad:	Moderada
Estabilidad:	Intransable
Fecha de actualización:	09/09/2019
Estado:	Cumple
Incremento:	3
Tipo:	Funcional

Diseño

- Diseñar los módulos de la aplicación y la interacción entre ellos.
- Documento de diseño
  - Arquitectura Física
  - Arquitectura Lógica
  - Diagrama de clases
  - Diagrama de secuencia
  - Diseño de interfaces

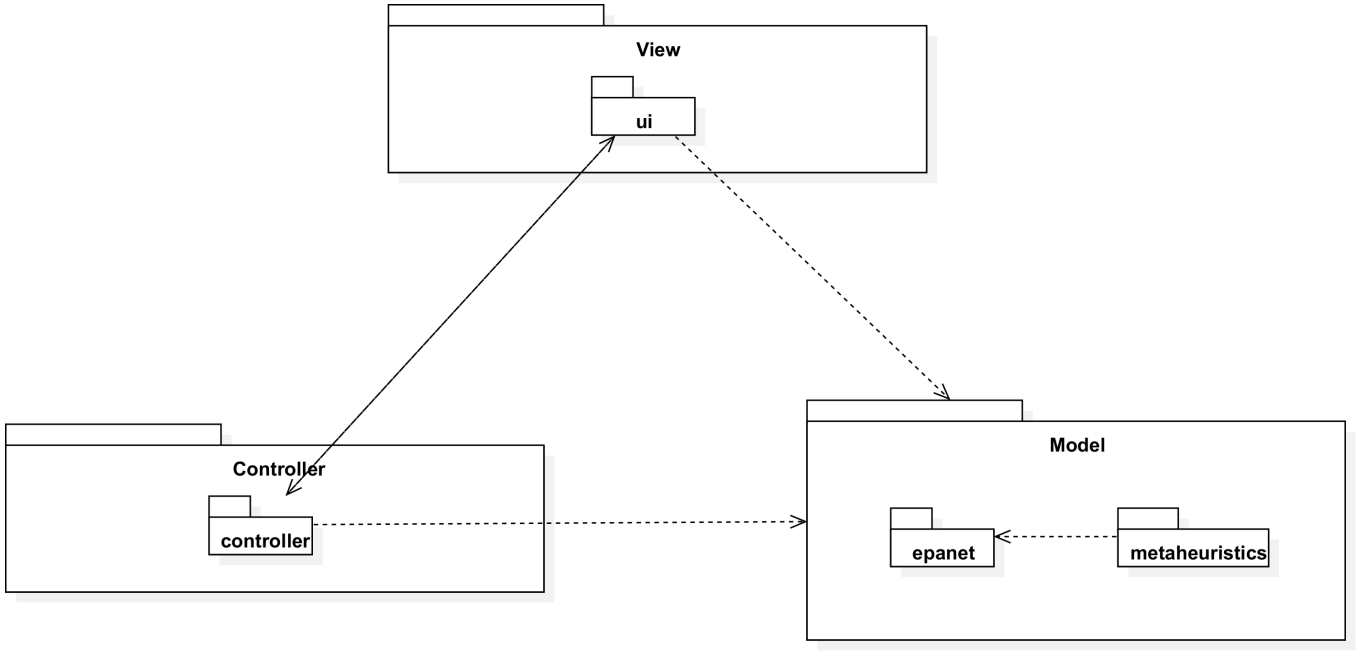
Arquitectura Fisica

Monolítica



Arquitectura Lógica

Modelo-Vista-Controlador

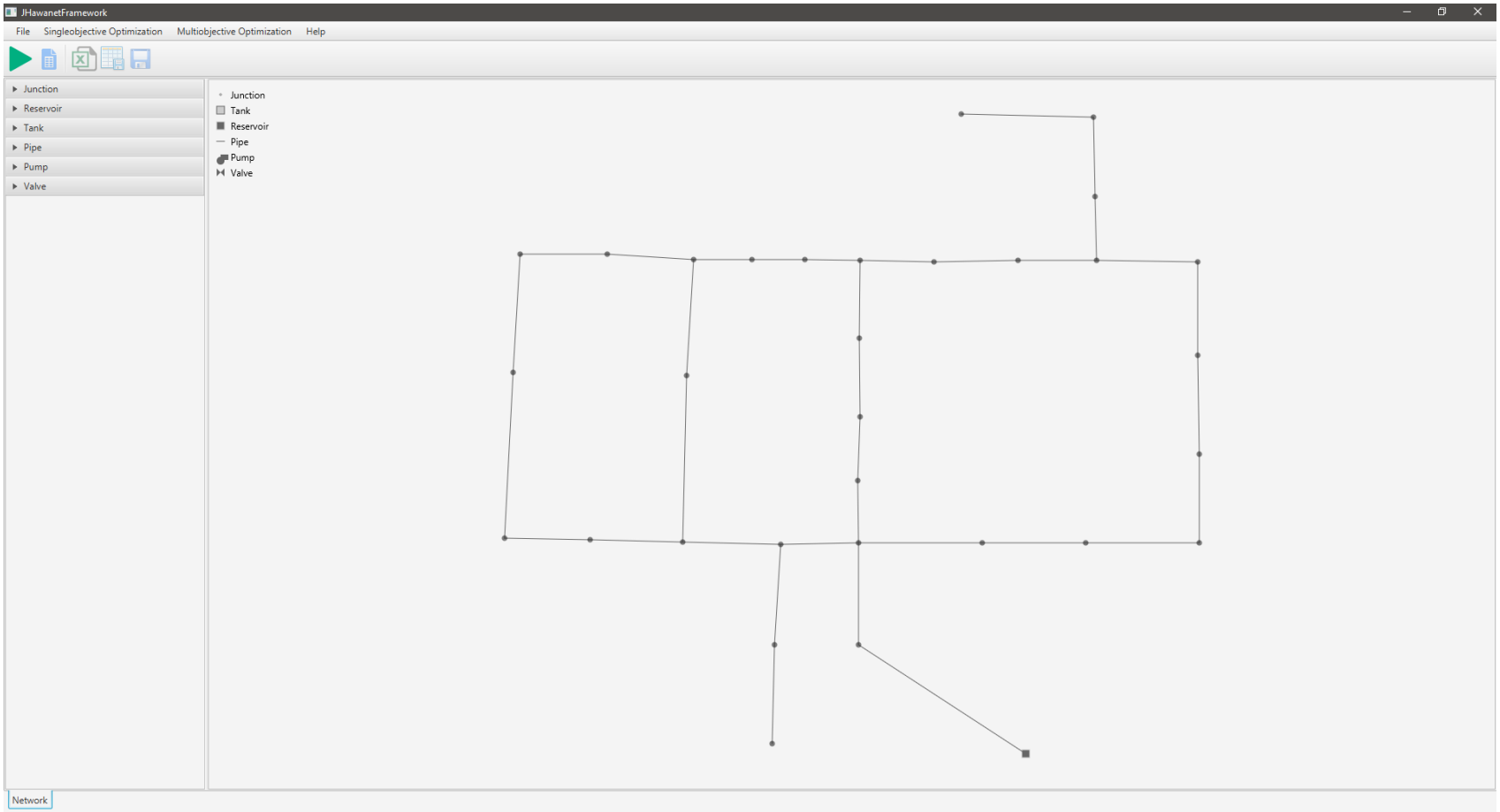


Implementación

Codificación y generación del manual de usuario.

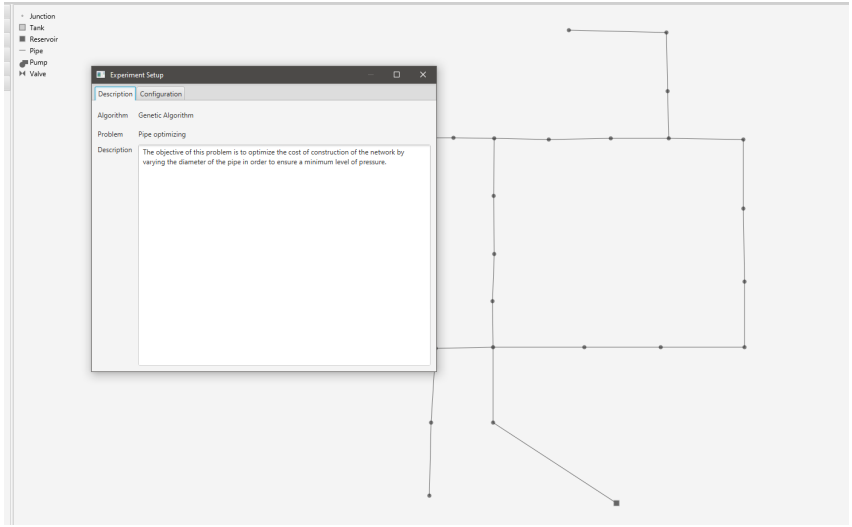
Implementación

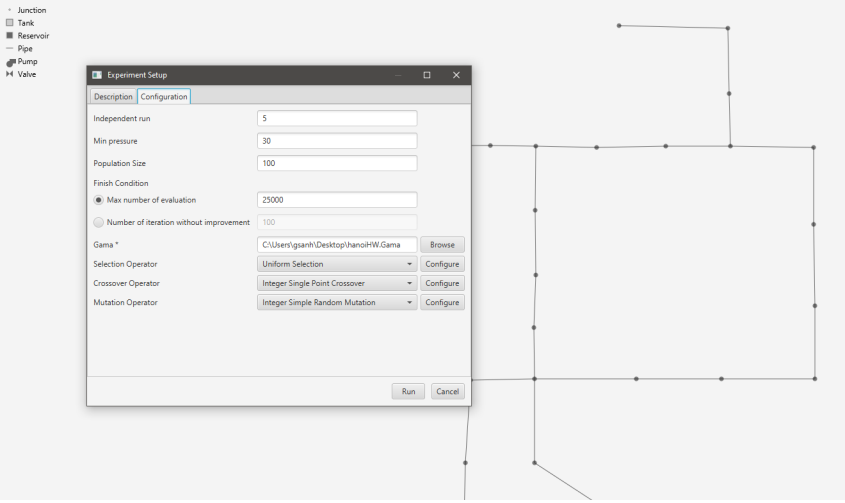
Ventana Principal



Implementación

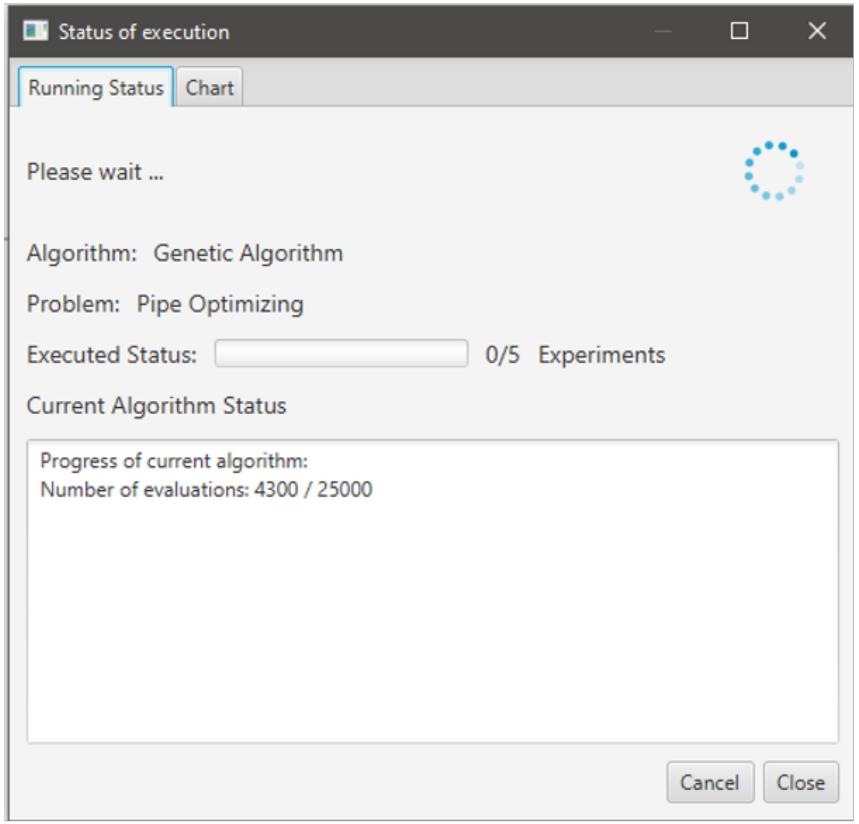
Ventana de Configuración

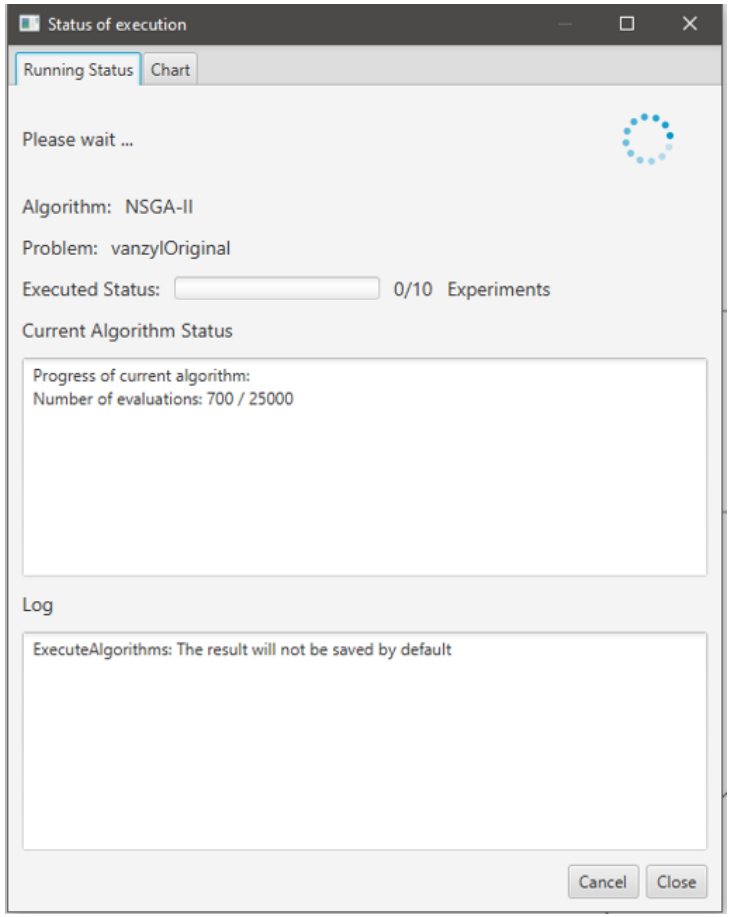




Implementación

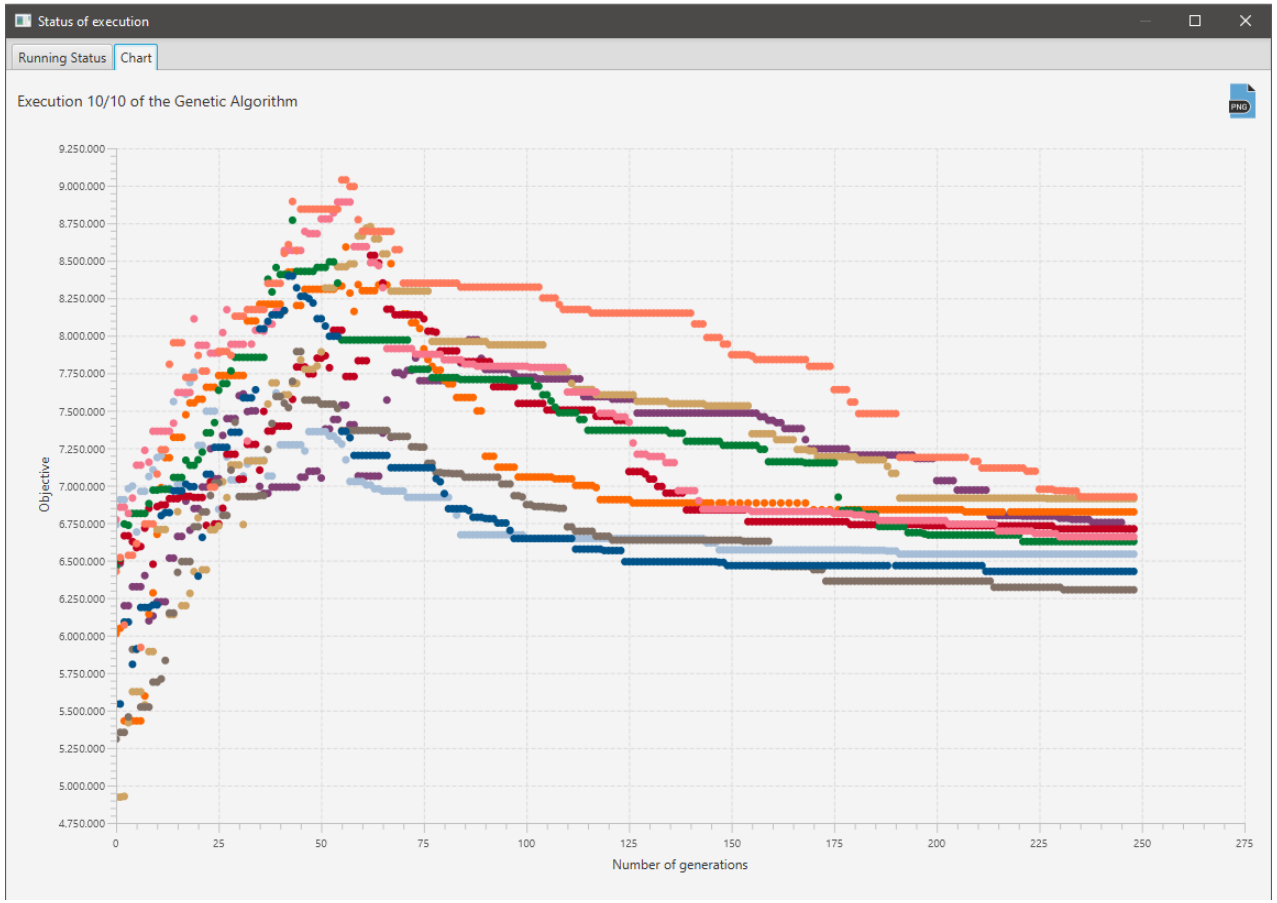
Ventana de ejecución





Implementación

Gráfico de resultados



Implementación

Resultados Optimización

JHawanetFramework

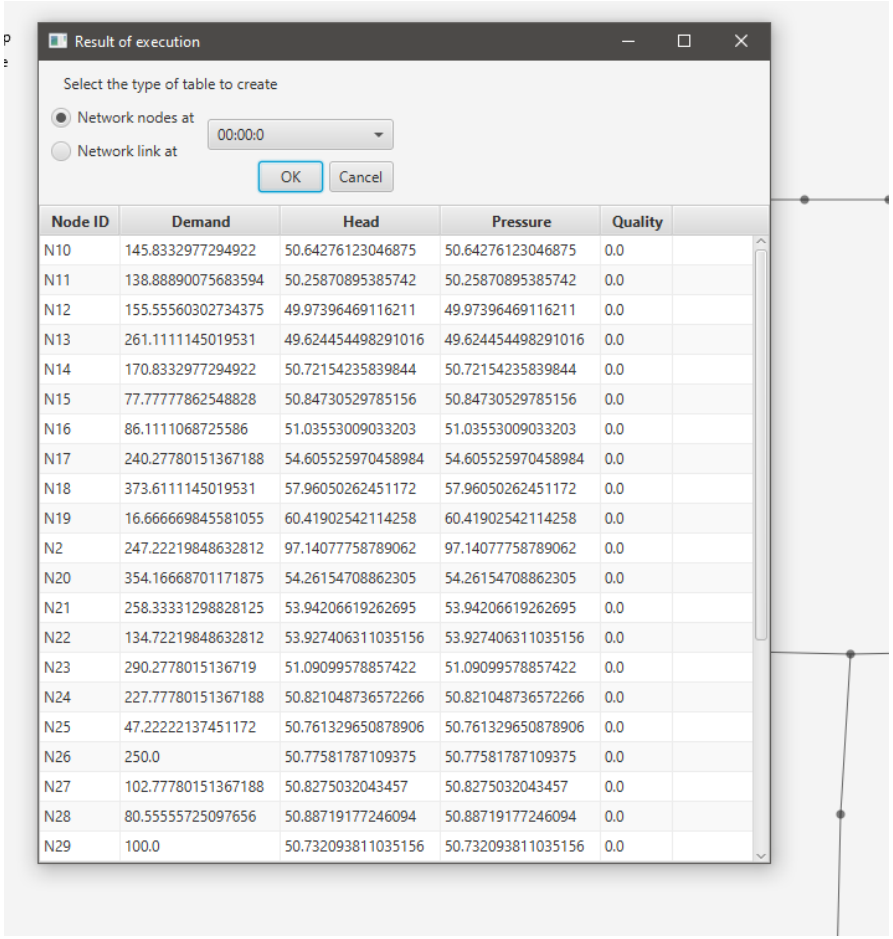
File Singleobjective Optimization Multiobjective Optimization Help

Objective 1	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17	X18	X19	X20	X21	X22	X23	X24	X25	X26	X27	X28	X29	X30	X31	X32	X33	X34	Overall Constrain Violation	Number of Generation	Min Pressure	Population
6928459.299171602	6	5	4	5	4	3	5	5	6	6	6	6	6	3	1	6	2	1	1	2	2	4	6	5	3	2	3	1	6	5	6	5	5	4	0.0	250	30	100
7078116.190952501	6	5	4	5	4	4	4	6	6	6	6	6	6	3	2	6	6	3	1	1	1	4	6	4	2	3	2	1	5	5	5	5	4	2	0.0	250	30	100
6788874.769704701	6	5	5	5	4	5	5	5	6	6	6	6	6	3	1	6	5	3	2	1	3	3	6	2	2	1	1	4	5	5	5	5	4	4	0.0	250	30	100
6637209.140444202	6	5	5	4	5	5	4	5	5	6	5	6	6	3	1	6	5	5	2	1	1	2	6	2	1	1	1	3	6	6	5	5	3	5	0.0	250	30	100
6429161.079557701	6	4	5	5	2	2	3	2	3	4	3	6	6	3	2	6	5	5	3	2	2	2	6	2	1	2	1	5	6	6	6	6	5	6	0.0	250	30	100

Network 0 - Pipe Optimizing-hanoi-Frankenstein origin-2020-06-23

Implementación

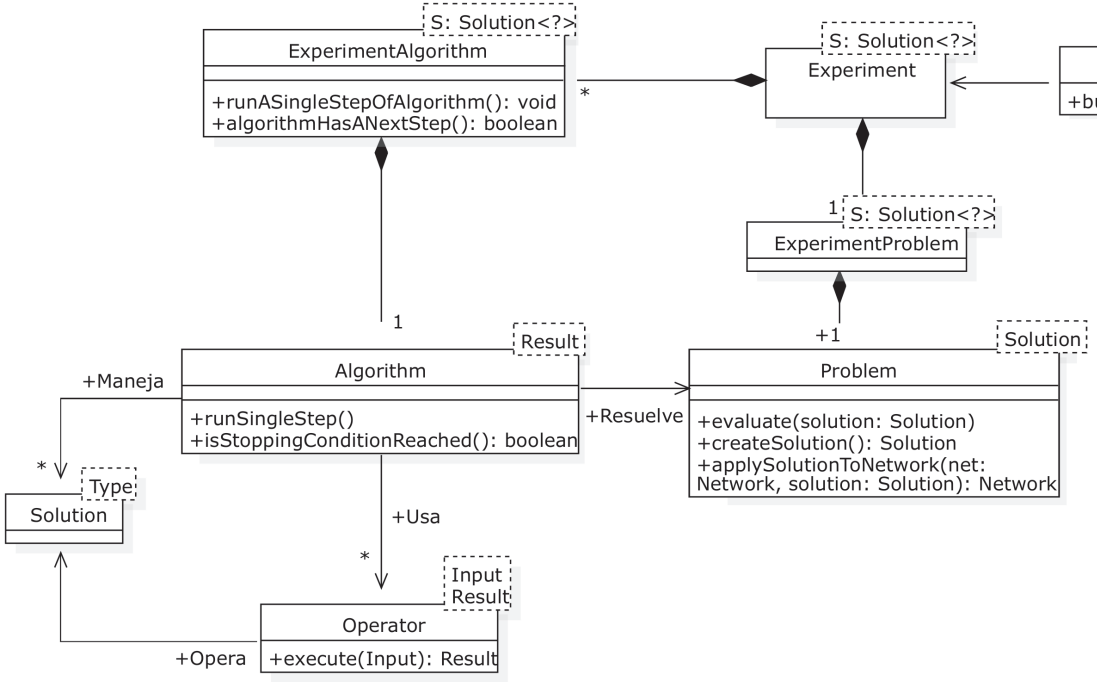
Ventana de simulación hidráulica



Node ID	Demand	Head	Pressure	Quality
N10	145.8332977294922	50.64276123046875	50.64276123046875	0.0
N11	138.88890075683594	50.25870895385742	50.25870895385742	0.0
N12	155.55560302734375	49.97396469116211	49.97396469116211	0.0
N13	261.1111145019531	49.624454498291016	49.624454498291016	0.0
N14	170.8332977294922	50.72154235839844	50.72154235839844	0.0
N15	77.77777862548828	50.84730529785156	50.84730529785156	0.0
N16	86.1111068725586	51.03553009033203	51.03553009033203	0.0
N17	240.27780151367188	54.605525970458984	54.605525970458984	0.0
N18	373.6111145019531	57.96050262451172	57.96050262451172	0.0
N19	16.666669845581055	60.41902542114258	60.41902542114258	0.0
N2	247.22219848632812	97.14077758789062	97.14077758789062	0.0
N20	354.16668701171875	54.26154708862305	54.26154708862305	0.0
N21	258.33331298828125	53.94206619262695	53.94206619262695	0.0
N22	134.72219848632812	53.927406311035156	53.927406311035156	0.0
N23	290.2778015136719	51.09099578857422	51.09099578857422	0.0
N24	227.77780151367188	50.821048736572266	50.821048736572266	0.0
N25	47.22222137451172	50.761329650878906	50.761329650878906	0.0
N26	250.0	50.77581787109375	50.77581787109375	0.0
N27	102.77780151367188	50.8275032043457	50.8275032043457	0.0
N28	80.55555725097656	50.88719177246094	50.88719177246094	0.0
N29	100.0	50.732093811035156	50.732093811035156	0.0

Implementación

Java Reflection y *Java Annotation*



Implementación

- Problema de diseño de RDA basado en el costo de tuberías

Objetivo

$$\text{Costo de inversión} = \sum_{i=1}^{N} (C_i \times D_i \times L_i)$$

Implementación

Problema de operación basado en el Régimen de bombeo

Objetivo 1

$$C_E(S) = \sum_{n=1}^{NP} \sum_{t=0}^{NT-1} (P_c(t) \times E_c(n,t) \times S(n,t))$$

Consumo energético

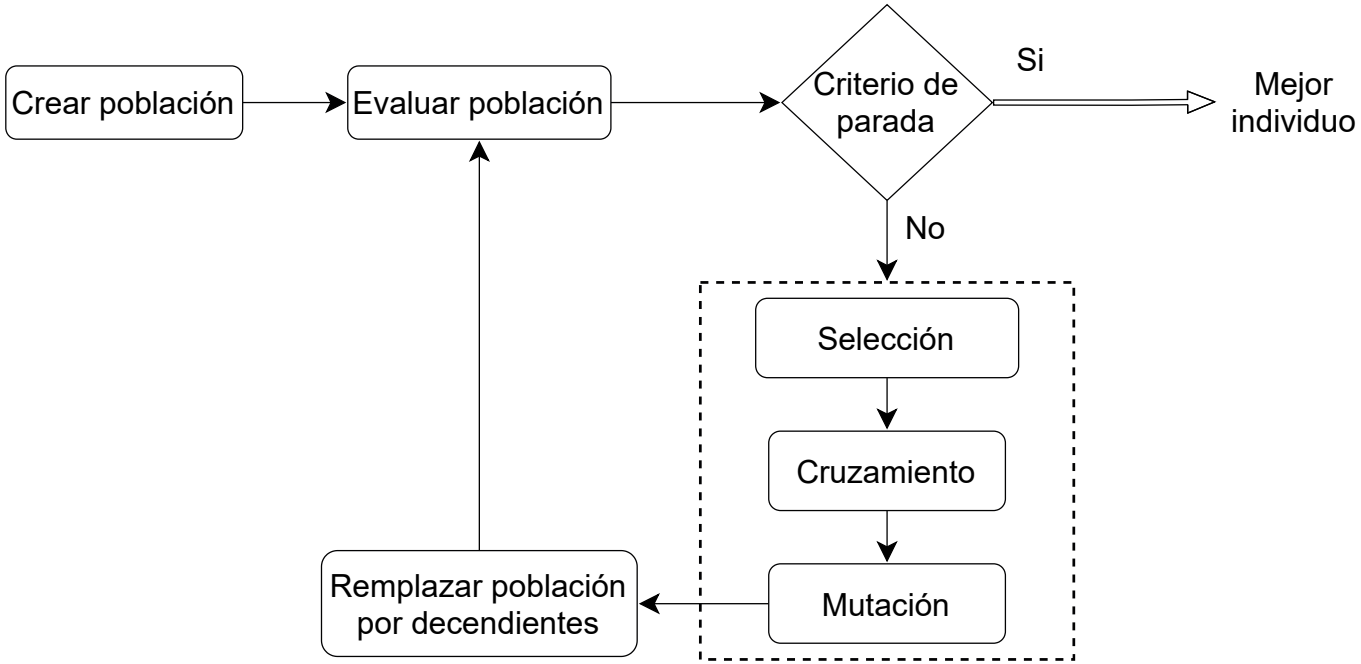
$$E_c(n,t) = \frac{10^{-3} \times \gamma \times Q(n,t) \times h(n,t)}{e(n,t)}$$

Objetivo 2

$$C_N(S) = \sum_{n=1}^{NP} \sum_{t=0}^{NT-1} r_t$$

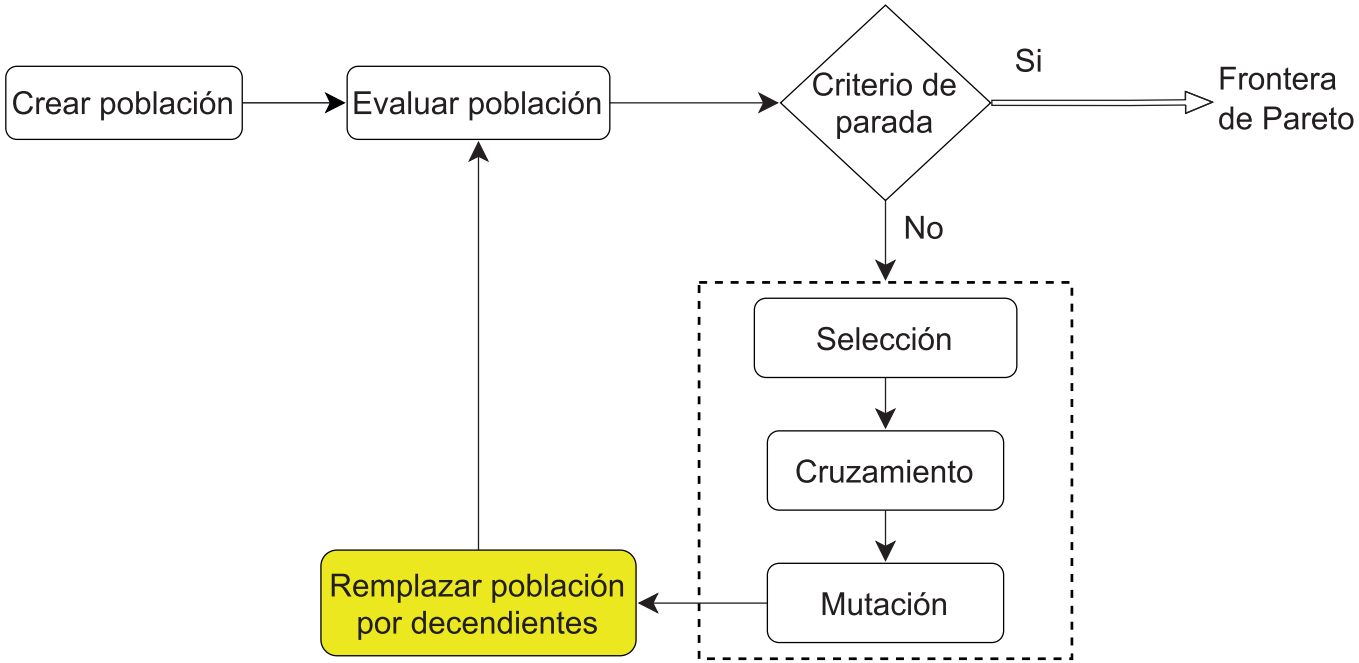
Implementación

Algoritmo Genético



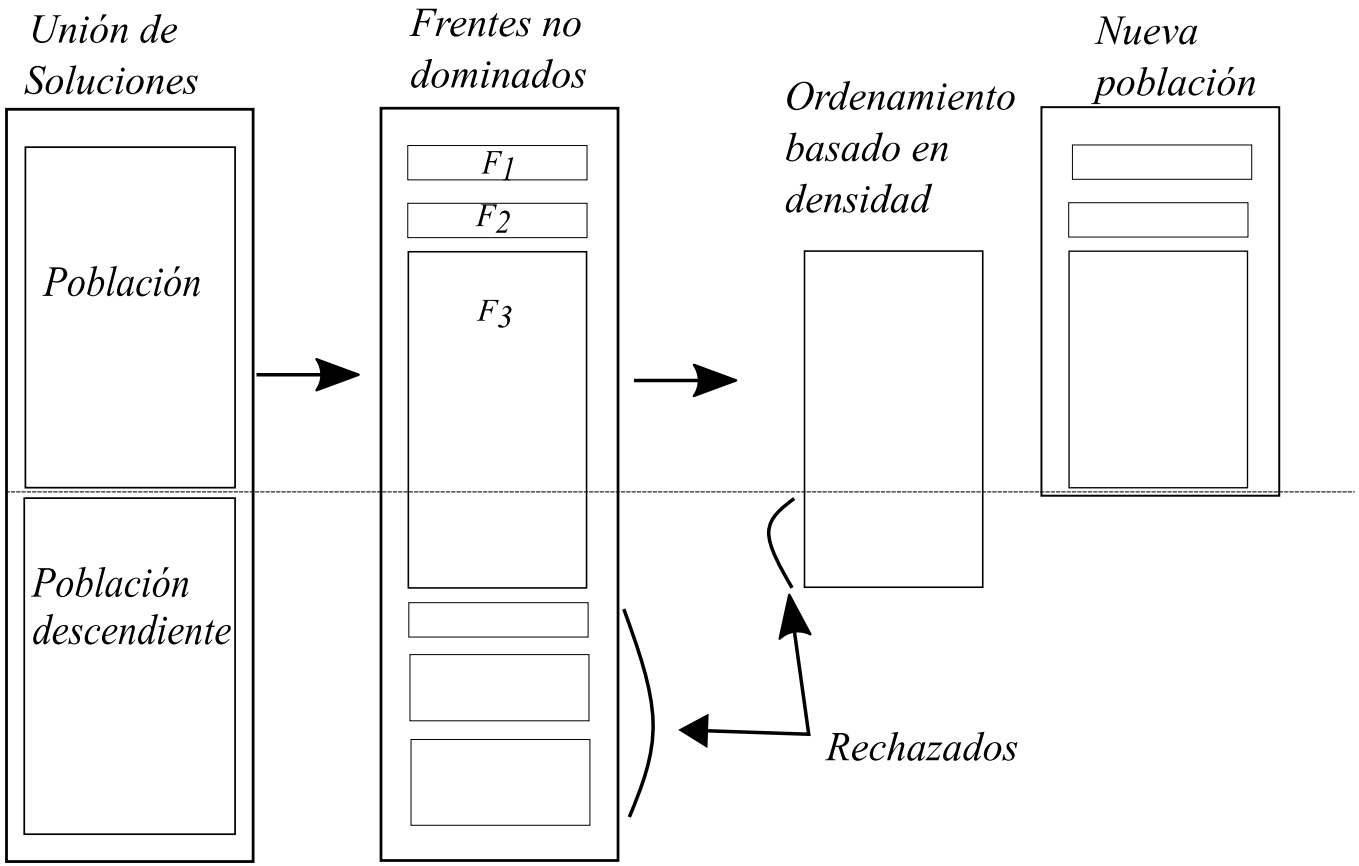
Implementación

Non-dominated Sorting Genetic Algorithm-II (INSGA-II)



Implementación

Non-dominated Sorting Genetic Algorithm-II (INSGA-II)



Pruebas

- Realización de pruebas y su especificación formal
- 39 Pruebas automatizadas
- 16 Pruebas manuales
- Documento de especificación formal de pruebas

Pruebas

Prueba Automatizada

Test ID :	AT009
Titulo:	Número máximo de evaluaciones no válido.
Característica:	Validar parámetro *maxEvaluations*.
Objetivo:	Validar que el parámetro *maxEvaluations* no sea negativo. Si el parámetro es negativo debe lanzarse una excepción.
Configuración:	Instancia de la clase *GeneticAlgorithm* inicializada.
Datos de prueba:	Cualquier entero menor que 0.
Acciones de prueba:	Llamar al método *setMaxEvaluations* con un argumento negativo.
Resultados esperados:	Una excepción.

Pruebas

Prueba Manual

Test ID :	MT001
Titulo:	Visualización de la red.
Característica:	Mostrar visualización de la red.
Objetivo:	Confirmar que la red puede ser leida desde un archivo con extensión ".inp" y ser visualizada en la aplicación.
Configuración:	El equipo tiene la aplicación JHawanetFramework lista para ejecutar.
Datos de prueba:	inp: "hanoi-Frankenstein.inp".
Acciones de prueba:	1. Abrir JHawanetFramework. 2. Cargar archivo de red.
Resultados esperados:	El sistema muestra la red leida desde un archivo inp gráficamente en la aplicación.

Diseño del caso

- Elección del caso
  - La aplicación desarrollada
- Objetivos de la investigación
  - ¿Cómo el sistema desarrollado trabaja en la práctica?
- Características a evaluar
  - Funcionalidad
  - Usabilidad
  - Utilidad
  - Utilidad del manual de usuario

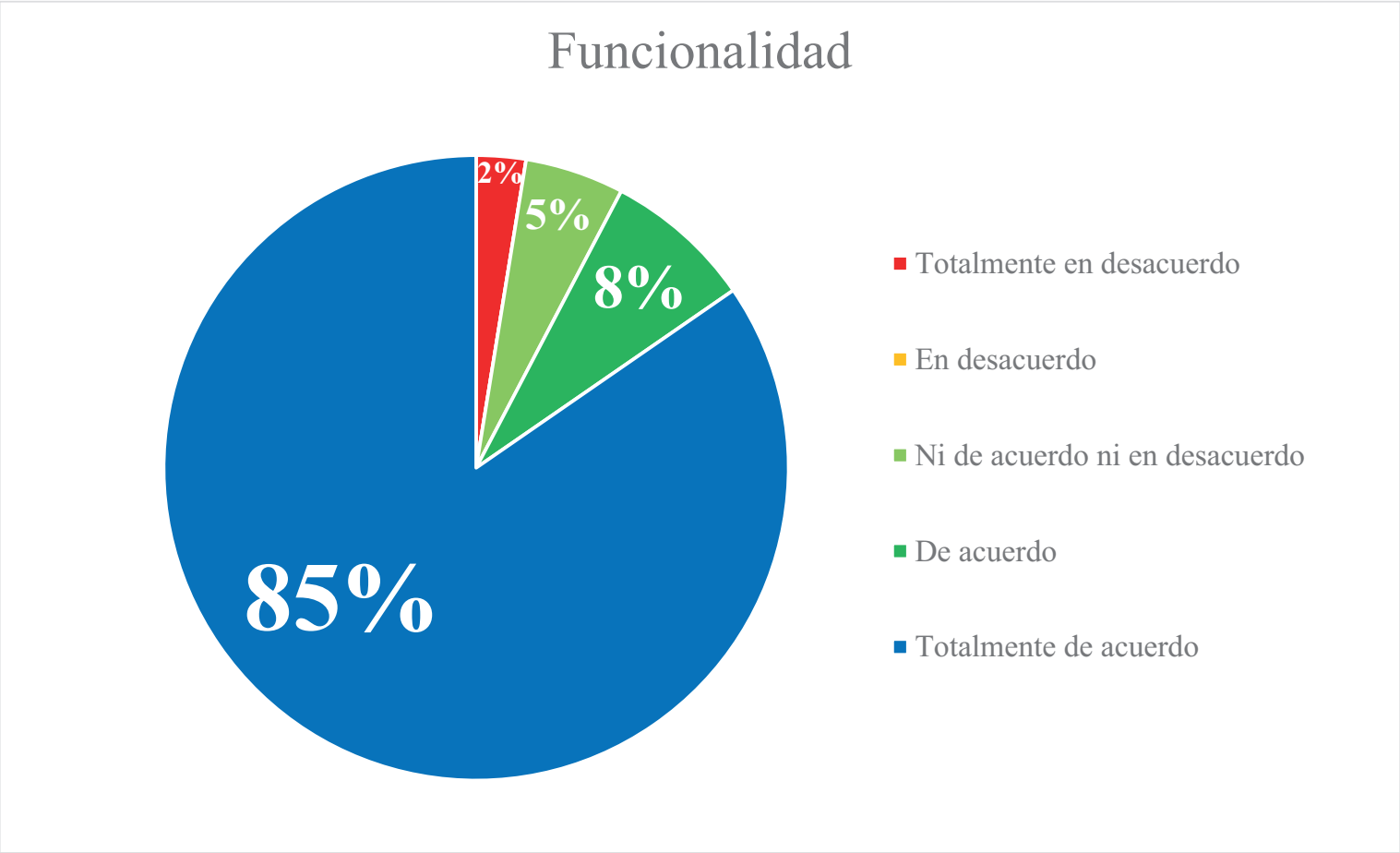
Diseño del caso

- Protocolo para conducir el estudio de caso
- Unidad de análisis
  - Profesionales con conocimientos en computación, hidráulica y meta-heurísticas
- Consideraciones técnicas
  - Window 64bits
  - Java 1.8
- Consideraciones para los usuarios
  - Conocimientos básicos en hidráulica y metaheurísticas para saber interpretar los resultados de la aplicación. Si se desea incorporar nuevos algoritmos debe tener conocimientos en programación.

Recolección de datos

- Participantes
  - Yamisleydi Salgueiro
  - Marco Alsina
  - Sergio Silva
- Instrumentos para la recolección de los datos
  - Encuesta utilizando la escala de Likert
  - Test de usabilidad SUS

Análisis de datos



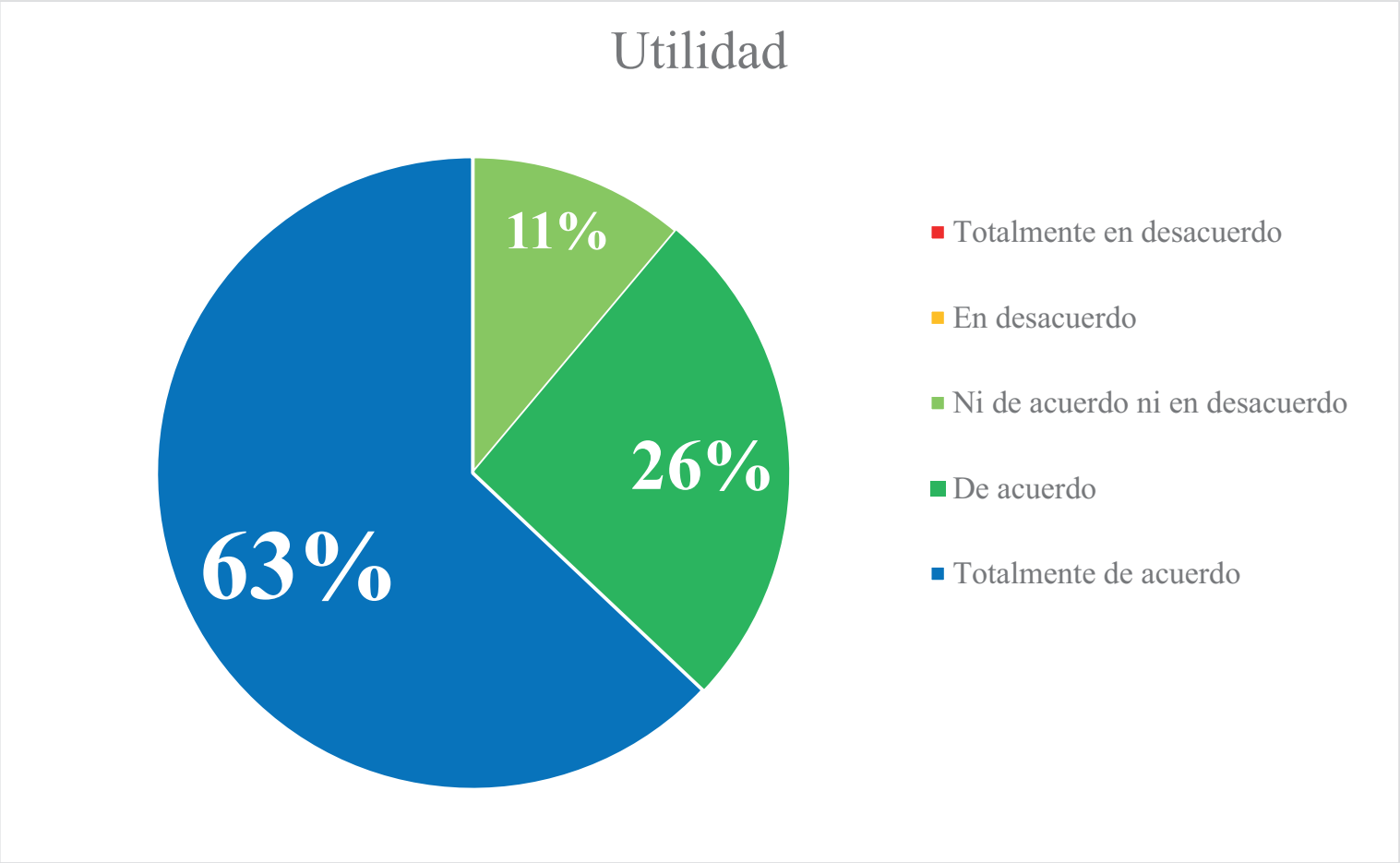
Análisis de datos

$\bar{x}_{SUS} = 79{,}17$

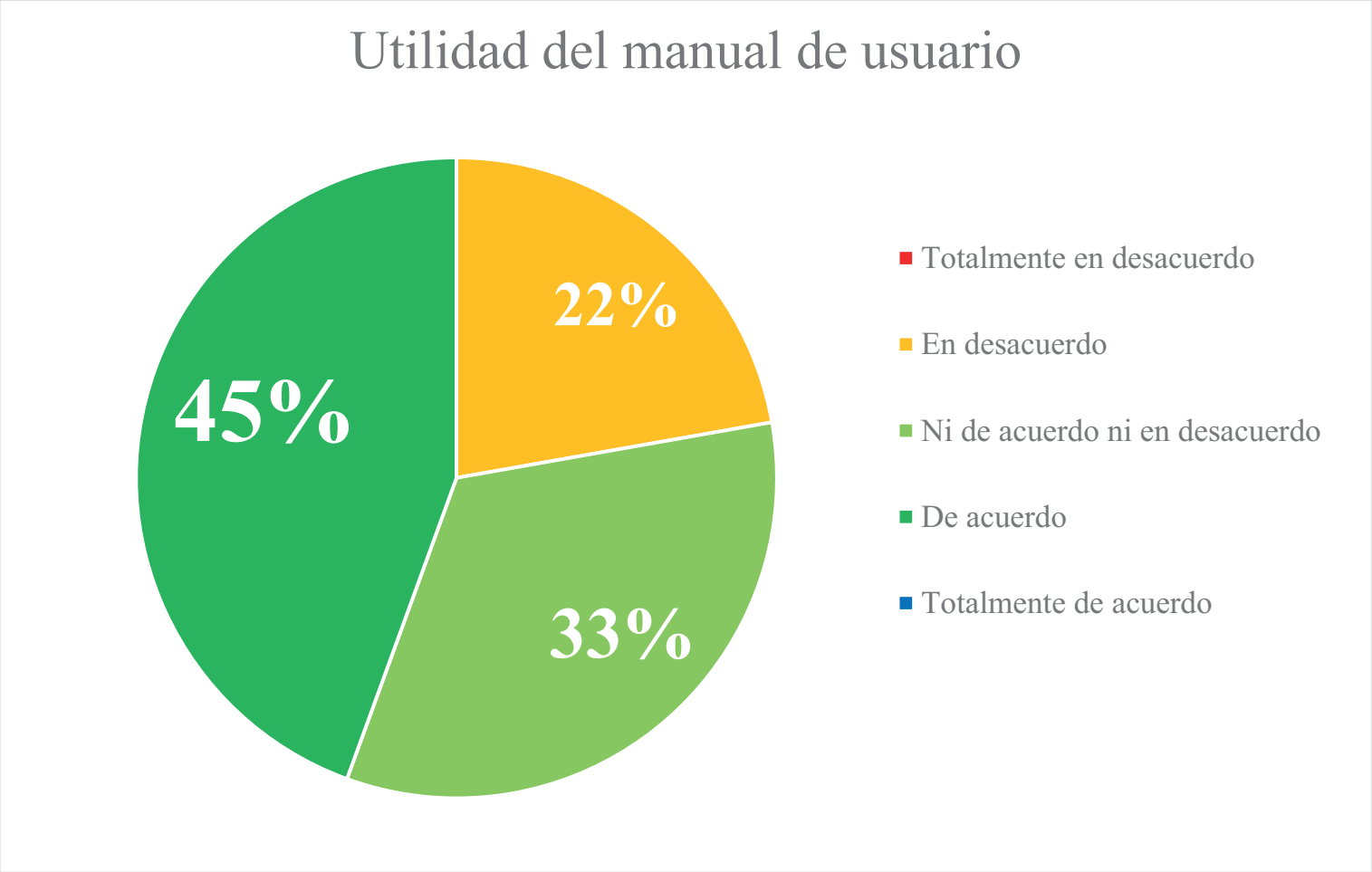
$s_{SUS} = 9{,}46$

SUS Score	Calificación
$> 80{,}3$	Excelente
68 - 80,3	Buena
68	Regular
51 - 68	Mala
< 51	Terrible

Análisis de datos



Análisis de datos



Conclusiones estudio de caso

- Criterios de funcionalidad, usabilidad y utilidad cumplidos, concluyendo que el sistema trabaja bien en la práctica.
- Criterio de utilidad del manual de usuario no cumplido por lo que se deben realizar cambios para mejorar la información y ayuda que se entrega con éste.

Conclusiones

- Software para la optimización de los procesos de diseño y operación de RDA.
- Todos los objetivos establecidos fueron logrados cumplidos

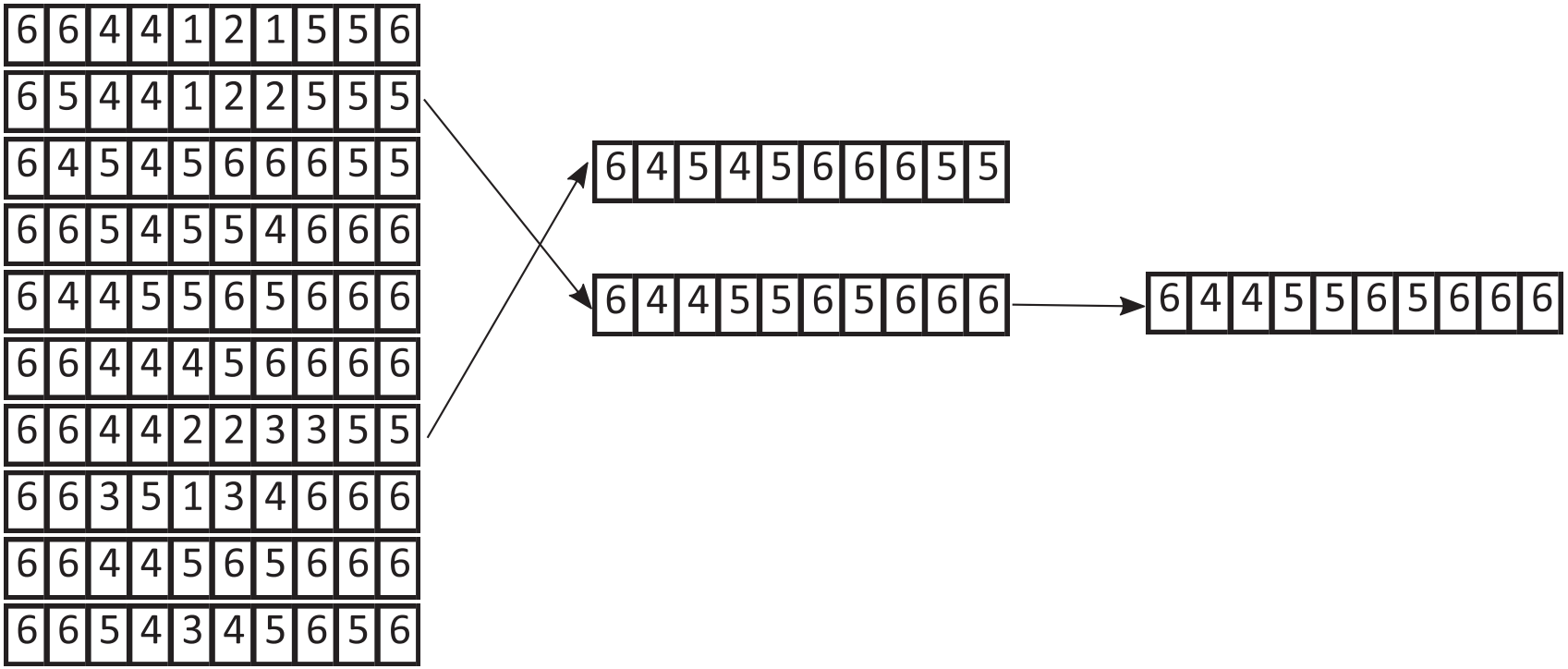
Trabajos Futuros

- Agregar nuevos algoritmos, operadores y problemas
- Recortar el numero de decimales utilizados
- Permitir exportar la imagen de la red
- Guardar las imágenes exportadas en un formato vectorial
- Cambiar tamaño de los iconos cuando el cursor este encima
- Permitir agregar formulas LaTeX en la descripción del algoritmo
- Permitir utilizar distintos algoritmos en un mismo Experimento
- Añadir métricas de comparación entre algoritmos
- Añadir en el menú contextual los patrones de bombeo o demanda en caso de que la red cargada los especifique.
- Permitir resetear a los valores por defecto en la ventana de configuración del problema

Fin

Operadores de Selección

Tournament Selection



Operadores de Selección

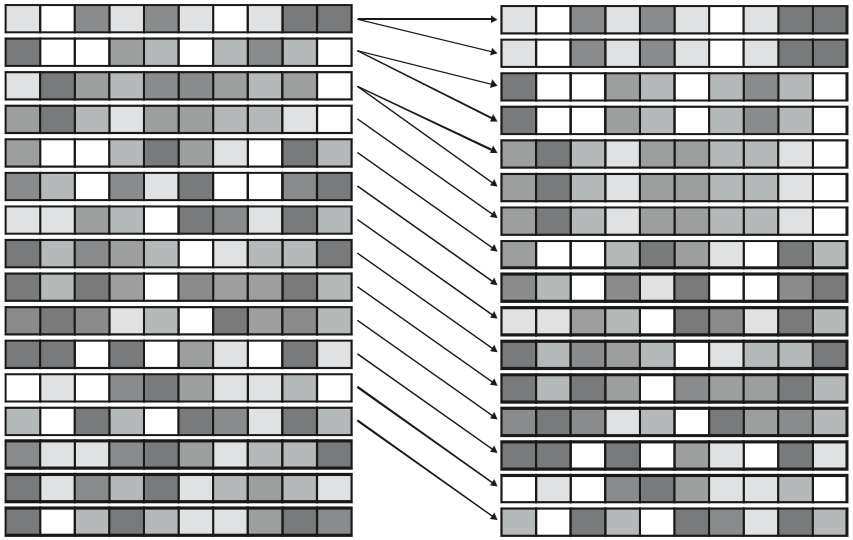
Uniform Selection

$$p_{max} = \frac{\beta}{N_c}$$

$$p_{min} = \frac{2-\beta}{N_c}$$

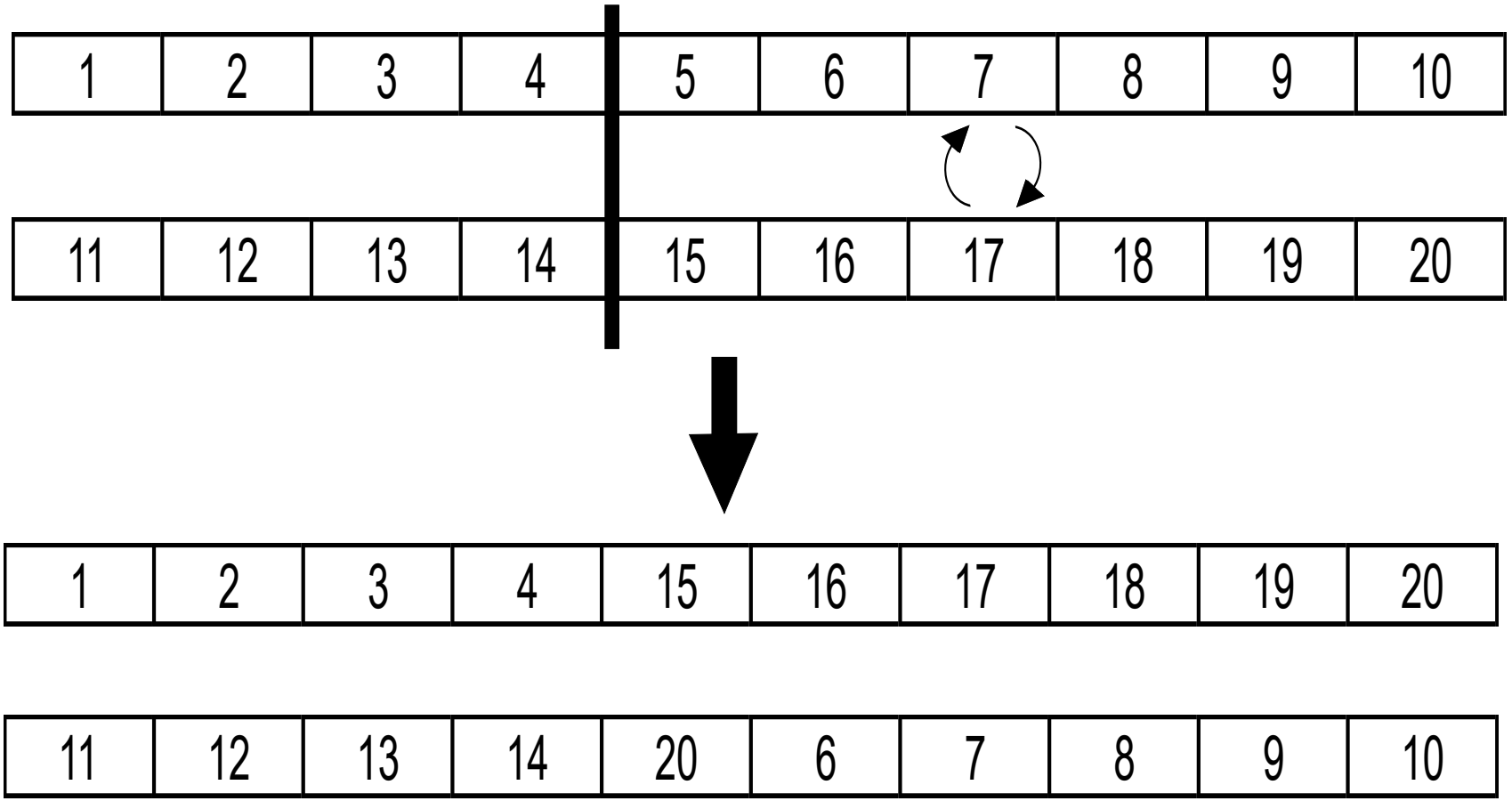
$$1{,}5 <= \beta <= 2$$

$$p_i = p_{min} + (p_{max} - p_{min}) \times \frac{N_c - i}{N_c - 1}$$



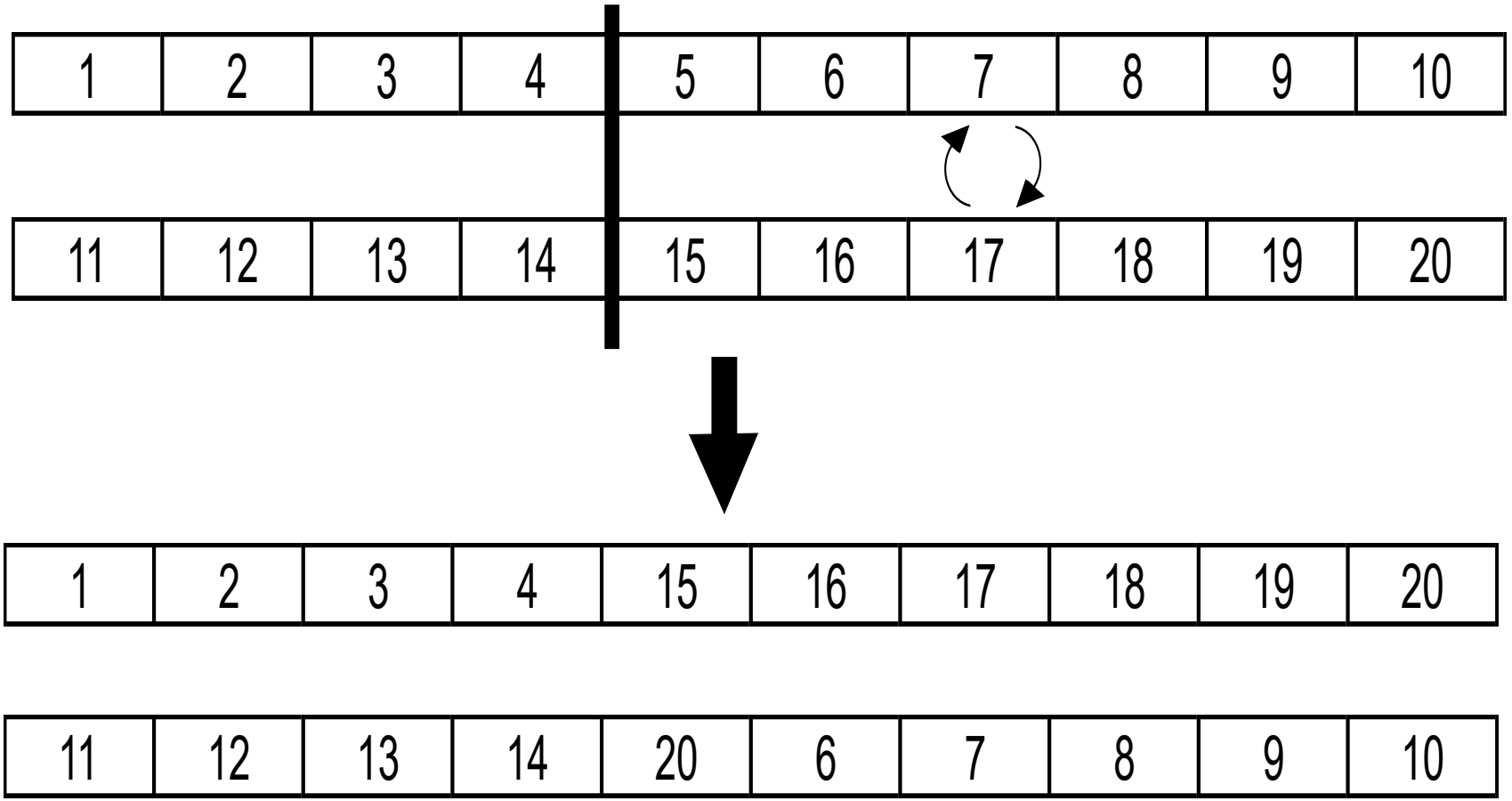
Operadores de Selección

SinglePointCrossover



Operadores de Selección

SinglePointCrossover



Operadores de Mutación

SBXCrossover

$$\begin{cases} \beta_q = \sqrt[d_i+1]{r\alpha} & \text{si } r \leq \frac{1}{\alpha} \\ \beta_q = \sqrt[d_i+1]{\frac{1}{2-r\alpha}} & \text{si } r > \frac{1}{\alpha} \end{cases}$$

$$\beta = 1 + 2\frac{y_1 - yL}{y_2 - y_1}$$

$$\alpha = 2 - \frac{1}{\beta^{di+1}}$$

$$\beta = 1 + 2 \cdot \frac{yU - y_2}{y_2 - y_1}$$

$$\alpha = 2 - \frac{1}{\beta^{di+1}}$$

$$C_1 = 0{,}5((y_1 + y_2) - \beta_q(y_2 - y_1))$$

$$C_2 = 0{,}5((y_1 + y_2) + \beta_q(y_2 - y_1))$$

Operadores de Mutación

PolynomialMutation

$$\Delta_1 = \frac{y - yL}{yU - yL}$$

$$\Delta_2 = \frac{yU - y}{yU - yL}$$

$$\begin{cases} \Delta_q = \sqrt[di+1]{2r + (1-2r)(1-\Delta_1)^{di+1}} - 1 & \text{si } r \leq 0{,}5 \\ \Delta_q = 1 - \sqrt[di+1]{2(1-r) + 2(r-0{,}5)(1-\Delta_2)^{di+1}} & \text{si } r > 0{,}5 \end{cases}$$

$$y = y + \Delta_q(yU - yL)$$

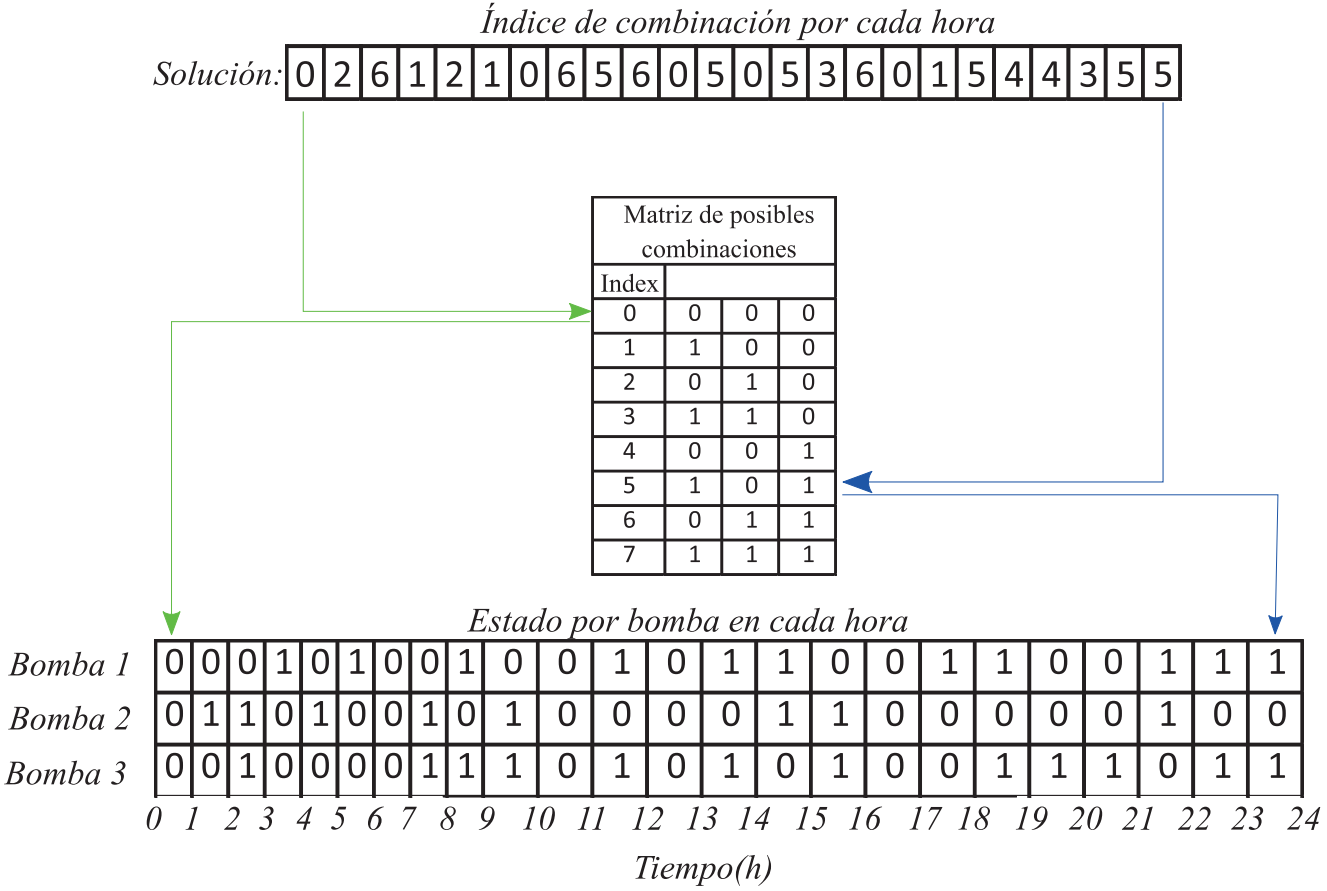
Representación de la solución del problema de diseño

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

	ID	Diámetro	Costo
1	DN12	304.80	45.73
2	DN16	406.40	70.40
3	DN20	508.00	98.39
4	DN24	609.60	129.33
5	DN30	762.00	180.75
6	DN40	1016.00	278.28

Solución:

4	1	1	2	3	4	6	1	6	4	2	6	5	4	2	6	6	5	1	1	5	2	6	3	6	2	6	5	5	5	4	6	4	1

Representación de la solucion problema operacional

Problema de operación basado en el Régimen de bombeo.