

Paralelización del algoritmo Progressive Hedging para la resolución de problemas estocásticos

Grao en Ingeniería Informática
Universidad de Santiago de Compostela

Autor: Cristofer Canosa Domínguez

Director: Juan Carlos Pichel Campos

2 de septiembre de 2018

Tabla de contenidos

- 1 Introducción
 - Programación Estocástica
 - Pyomo
 - Objetivos
- 2 Gestión del proyecto
 - Alcance y entregables
 - Planificación temporal
- 3 Análisis
 - Algoritmo *Progressive Hedging*
 - Herramientas
- 4 Diseño
- 5 Implementación
- 6 Pruebas
- 7 Conclusiones

Programación estocástica

- Problemas de optimización

Programación estocástica

- Problemas de optimización
- Existe un nivel de incertidumbre

Programación estocástica

- Problemas de optimización
- Existe un nivel de incertidumbre
- Generan múltiples escenarios

Programación estocástica

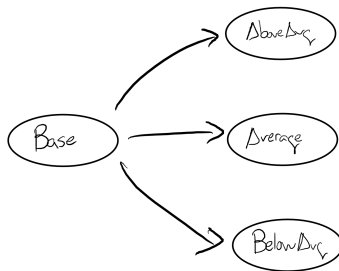


Figura: Árbol de escenarios en un problema estocástico

Ejemplos de aplicación

- 1 TFM: “Problemas de rutas de vehículos”
- 2 Modelo de optimización de la oferta de generación eléctrica para compañías eléctricas que participan en el mercado eléctrico liberalizado MIBEL.
- 3 “Progressive Hedging aplicado a coordinación hidrotérmica”

Pyomo

- Formulación y solución de modelos de optimización
- Uso de solucionadores de terceros (CPLEX, GLPK)
- *Sandia National Laboratories y University of California*
- Python



Objetivos

- 1 Estudiar y analizar el funcionamiento del algoritmo *Progressive Hedging* en PySP.
- 2 Análisis de las diferentes alternativas de paralelización disponibles que mejor se adapten al problema. Se tendrán en cuenta tecnologías Big Data (Apache Spark) o modelos tradicionales de paralelización.
- 3 Diseño e implementación del nuevo módulo e integración con Pyomo.
- 4 Análisis y evaluación del rendimiento.

Alcance

- Adaptar el módulo de programación estocástica (PySP) a una nueva implementación paralela.
- Nueva implementación más escalable que permita abordar problemas de mayor tamaño.
- Realizar un análisis de rendimiento.

Entregables

- Código de Pyomo actualizado con el módulo de ejecución de PH paralelo.
- Estudio de rendimiento.
- Memoria de realización del proyecto.
- Otra documentación asociada a la realización del proyecto.

Metodología de desarrollo

- Límite temporal estricto

Metodología de desarrollo

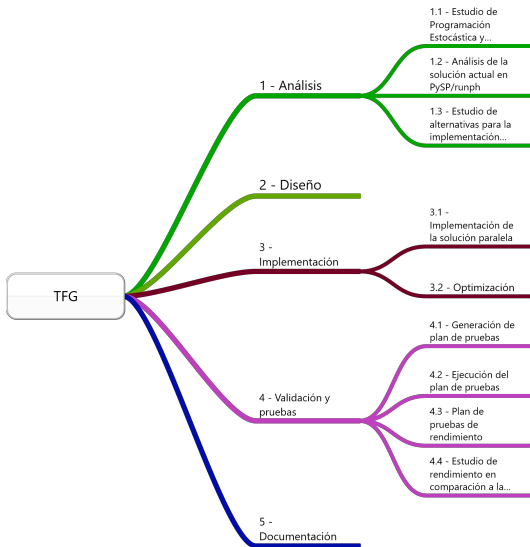
- Límite temporal estricto
- Poco peso de la fase de implementación

Metodología de desarrollo

- Límite temporal estricto
- Poco peso de la fase de implementación
- Mayor énfasis en análisis y documentación

Metodología de desarrollo

- Límite temporal estricto
- Poco peso de la fase de implementación
- Mayor énfasis en análisis y documentación
- Metodología en cascada



Cronograma

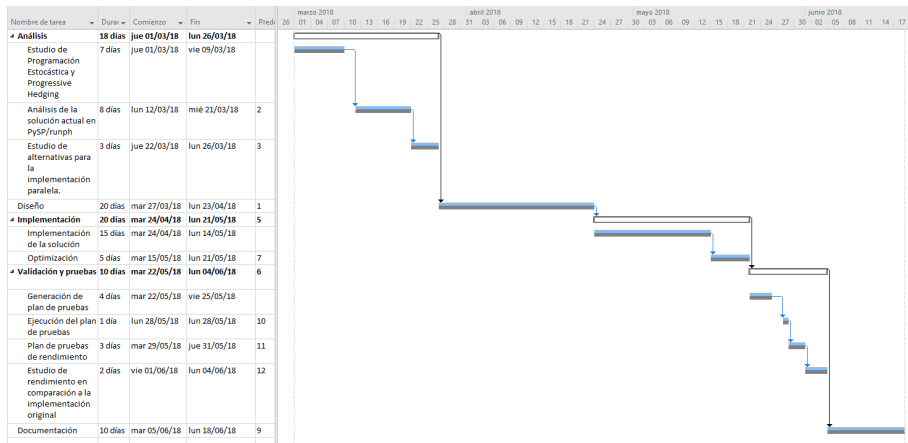


Figura: Línea base

Adaptación de la planificación

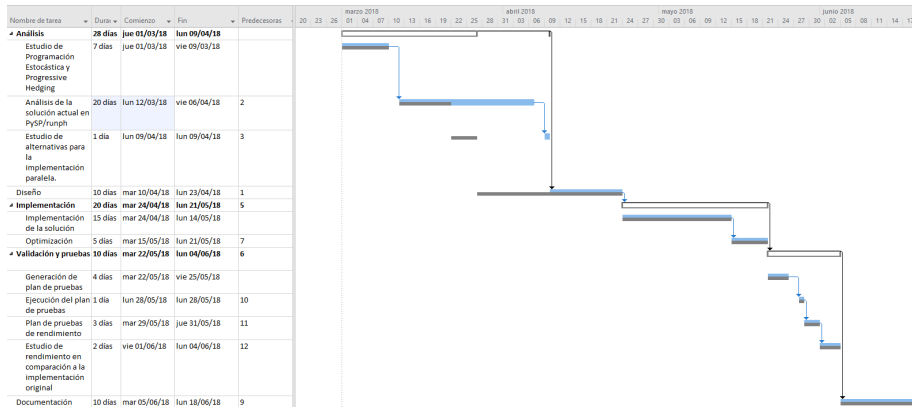


Figura: Primer retraso

Adaptación de la planificación

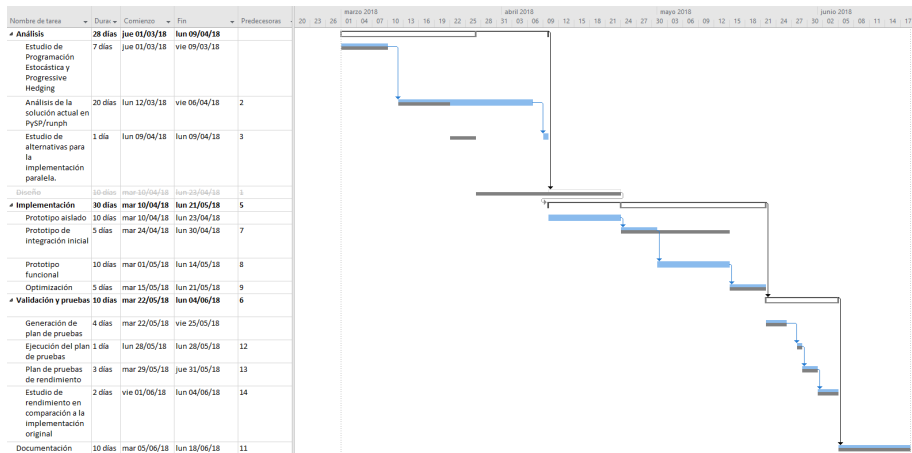


Figura: Línea base prototipos

Cronograma final

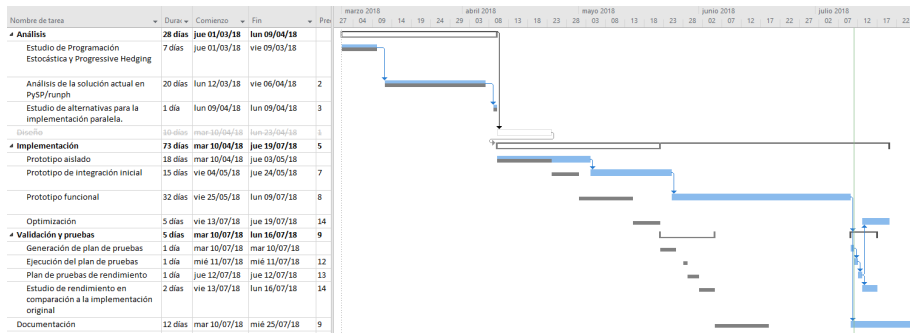


Figura: Línea base final

Progressive Hedging

Implementación en Pyomo

Spark

MPI

Diseño

Proceso

Funcionamiento

Implementación

Introducción
Gestión del proyecto
Análisis
Diseño
Implementación
Pruebas
Conclusiones

Recursos

Resultados

Lecciones aprendidas

Trabajo futuro

Conclusiones