

Implementación actual

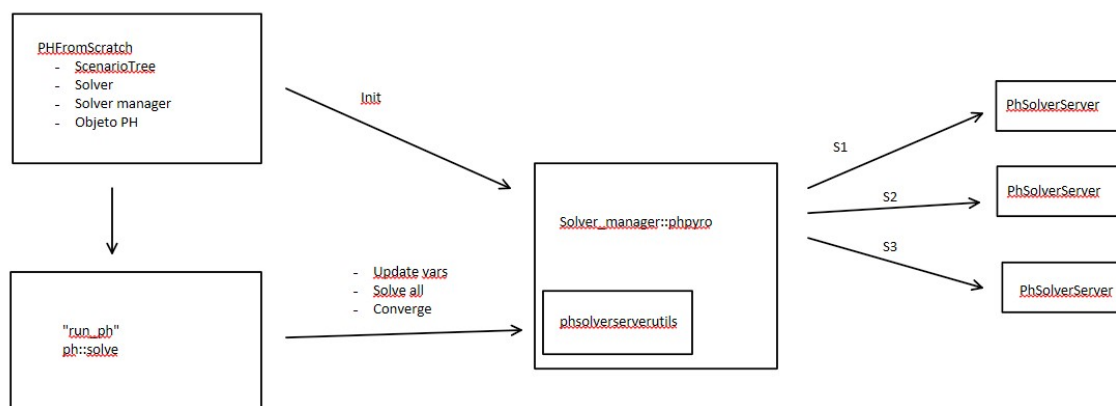
En el estado actual del proyecto se puede ejecutar el algoritmo PH haciendo uso de *mpi* y *pyro*, utilizando objetos remotos que ejecutarán el algoritmo.

Podemos diferenciar 3 capas en la ejecución del algoritmo.

En primer lugar, contamos con un contexto *PHFromScratch* que manejará todos los recursos necesarios para la ejecución. Se encargará de inicializar los modelos y el resto de objetos para resolver el problema. El más importante será el objeto *PH* que contiene los datos de la ejecución actual, así como el algoritmo base.

En un segundo lugar, tenemos el *solver_manager* que se encarga de distribuir el trabajo entre los nodos paralelos y mantener la consistencia de los datos.

Por último, los objetos *PHSolverServer* son los que se despliegan de forma paralela y cada uno resuelve un escenario concreto en cada una de las iteraciones.



Adaptación propuesta

Para adaptar esta solución a *Spark* debemos conseguir ejecutar el solver seleccionado en cada uno de los nodos. A cada nodo se le debe comunicar el objeto PH (con todas las variables necesarias) así como las variables de sincronización.

Una posible implementación sería crear un nuevo `solver_manager` que utilice el mismo sistema de cola de acciones para reducir al máximo la modificación en el resto del código.

También sería necesario modificar el archivo *phsolverserverutils* que se encarga de enviar la mayoría de comandos a la cola de trabajos en paralelo. Sería adecuado hacerlo similar al existente para que sea fácilmente interoperable.

Por último, el cálculo de convergencia al final de cada iteración parece que se ejecuta de forma local. Puede ser interesante considerar paralelizarlo si se consigue alguna ganancia significativa en problemas con muchos escenarios.