Appendix A

2-stoch-prog-IPM

Download

The 2-stoch-prog-IPM.zip is available in Jordi Castro's web page `http://www-eio.upc.es/~jcastro/`. Section *Data*

Requirement

For making the comparison you will need the CPLEX software application.

Installation

The application runs on Linux. The 2-stoch-prog-IPM.zip file must be unzip. For unzip the file from the terminal, the command used is:

```
> unzip 2-stoch-prog-IPM.zip
```

The next step is to compile `BlockIP`, from the terminal, inside the directory 2-stoch-prog-IPM, the commands are:

```
> make clean
```

```
> make
```

In this step `BlockIP` will be ready to work, with all the associated problems.

Content

2-stoch-prog-IPM application has all the folders of the two general instances that are created to measure the performance of `BlockIP` and CPLEX with large variables in the first-stage. Also, the `BlockIP` library.

- Supply Chain

- Electric

The directories can be divided into executable folders and the ones for the results.

In the first group, we can find two subgroups, one for each type of problem:

- Supply Chain
  - SCPrimal. It is the primal linear version.
  - SCDual. It is the linear dual version of the problem.
  - SCPQuad for the primal quadratic version.
  - SCDQuad for the dual quadratic version

And the second subgroup is:

- Electric
  - ElPrimal. It is the primal linear version of this problem (Electric)
  - ElDual, with the linear dual version of the problem (ElectricDual)
  - ElPQuad for the primal Quadratic version
  - ElDQuad for the dual Quadratic version

Inside each folder, we can find a namefile.C file with the code in C++ that generate and solve the problem with BlockIP. Each problem required some input data, which generate a specific instance. These values can be given directly or using the `Comp` script (explained later) from the Results folders. Beside, each folder contains the Makefile file.

Executing directly

After compiling `BlockIP`, you need to be inside the directory that you want to run. For example

```
> cd SCPrimal
```

Then, execute with some parameters: <number of Supplies> <number Pfailities> <number of clients> and <number of blocks>. Before this data you need the `-SupplyChainEsc` flag. For example to run an instance with 4 suppliers, 3 processing facilities, 3 Clients and 3 scenarios you need the following command:

```
> ./SCPrimal -SupplyChainEsc 4 3 3 3
```

For the group of Electric, the `-ElectricEsc` flag is need, for example:

```
> cd ElPrimal
> ./Electric -ElectricEsc 2 20 10 10
```

This command runs the instance with 2 stages, 20 number of technologies, 10 different modes, and 10 scenarios. The problems must have more than 2 scenarios. (more details below)

The second group, Results, are folders to manage the outcomes. Here, each folder organizes the results with the aid of a scripts file for each problem:

- Supply Chain
  - SCResults, for the linear performances.
  - SCQResults, for the quadratics problems.
- Electric
  - ElResults, for linear experiments.
  - ElQResults, for quadratic instances.

Inside each one of those folders, we can find the script with all the commands that help you to create new instances and make the comparison with CPLEX. Also, the folders with the outcomes.

For example, in SCResults we can find the next folders:

- ResultsPrimal. For the outcomes of BlockIP.
- ResultDual. For generate the problem dual and create the MPS file for CPLEX.
- ResultsMPS. For storage the MPS file resulting from BlockIP and then read it by CPLEX.
- ResultsCplexBarrier. Here we save all the results of the CPLEX barrier algorithm.
- ResultsCplexDual. Only store files if comparisons with dual CPLEX are activated.

The scripts inside SCResults are:

- Comp. Create different instances and help you to make a comparison between BlockIP and CPLEX. Then, send the outcomes to the corresponding folder of results.
- ScriptCplexBarrier.txt with the commands to runs CPLEX with the barrier algorithm (baropt).
- ScriptCplexDual.txt with the commands to runs CPLEX with the Dual algorithm.

Modifying the size of the instances

The instances can be manipulated in order to increase, mainly, the first stage variables.

For creating the Supply Chain instances, you must edit the `Comp` script. Here, we have four variables.

- S1. It is the number of suppliers that we what to consider
- P1. It is the number of process facilities
- C1. The number of clients and
- B1. The number of scenarios.

For the Electricity problem, we have four variables too.

- S1. The number of stages, always 2.
- P1. The number of technologies
- C1. The different modes
- B1. The number of scenarios.

How to generate and solve new instances

For creating a new set of instances, for example, of Supply Chain with various scenarios.

- Step 1. Go inside SCResults folder
- Step 2. Open the Comp script
- Step 3. Check the files and directions for the outcomes files: ResultPrimal, ResultsMPS, and ResultsCplexBarrier (ResultsCplexDual, only if is activated)
- Step 4. Edit the variables (S1, P1, C1) for creating the specific instance. For B1 variable, that refers to the number of scenarios, you can use the for loop with the different numbers.
- Step 5. Execute the Comp file from SCResults.
- Step 6. Look for the outcomes inside their corresponding directories.