plan4res installation

1 Quick Install

We summarize here the commands to install plan4res (on an empty repository):

- git clone https://github.com/openENTRANCE/plan4res.git
- git clone --recursive https://gitlab.com/cerl/plan4res/p4r-env.git
- cd p4r-env
- bin/p4r
- exit
- bin/p4r add-on stopt
- bin/p4r add-on sms++ install CPLEX=cplex_studioXXXXXX.bin
- cd scripts
- mkdir python
- cd python
- git clone https://github.com/openENTRANCE/openentrance.git
- git clone https://github.com/openENTRANCE/plan4res-scripts.git
- cd ../..
- mv Launch* scripts/
- mv BSPar-Investment.txt config/
- mv sddp_greedy_investment.txt config/
- mv sddp_solver.txt config/
- mv uc_solverconfig.txt config/
- mv sddp_greedy.txt config/

You are now ready to play!

2 Introduction

Installing plan4res requires to :

- **first install the p4r-env environment (container)** (see CERL / Plan4Res / p4r-env · GitLab). This environment will allow to very simply install and run all plan4res solvers and tools. Anyway, all parts of plan4res can also run without this environment but it will then require to install all dependencies manually, which often proves time-consuming and tricky.
  The p4r-env environment can be installed on Linux, Windows, Mac and even CRAY machines. Once the environment is installed, the behavior will be exactly the same on all kind of systems (apart from the performances of course….)
  p4r-env is a singularity container, which means that it requires singularity to be installed.
  Instructions to install are sumarised in section 3 but you may look at detailed instructions here https://gitlab.com/cerl/plan4res/p4r-env/-/tree/master#p4r-environment-installation-execute-once
  Windows as some specific prerequisite which are detailed here https://gitlab.com/cerl/plan4res/p4r-env/-/tree/master#prerequisite

- **then install the softwares** :
  - StOpt (see Stochastic Control / StOpt · GitLab) : stochastic optimisation library used for solving the SSV problem
  - SMS++ (see SMS++ / The SMS++ Project · GitLab) : modelling and optimisation library including the 3 main solvers of plan4EU
  - Formatting tool (except if you are creating the NetCDF input data files in the format required by SMS++ on your own) ; this tool creates input data file with the format required by SMS++ taking into input user friendly CSV and XLSX files
  - (Optional) Preprocessing tool: this tool allows to create plan4res dataset out of long-term pluriannual energy system scenarios (eg from GENeSYS-MOD)
  - (Optional) Postprocessing tool: this tool trnasforms the files created by the solver (SMS++) into more user-friendly files and creates some synthetized results.
  - (Optional) IAMC format transformation: this tool transforms the results from the postprocessing tool into files within the IAMC format, which can then be uploaded to a Scenario Explorer
  - (Optional) Visualisation tool: creates graphs

- Then install the python scripts for creating and formatting datasets as well as prsttreating
- Get the configuration files for SMS++
- And finally get the launching scripts if wished

3 Get documentation, launching scripts and configuration files

git clone https://github.com/openENTRANCE/plan4res.git

This allows to download :

- The documentation (including this document)
- The scripts for launching plan4rs Launch* Once p4r-env is created, move them to p4r-env/scripts
- The configuration files of SMS++ : Once p4r-env is created, move them to p4r-env/config

4 Install p4r-env

4.1 New instal

1. Choose an empty directory (eg P4R)
2. If a former install of p4r exists, clean :
   p4r-env/bin/p4r -c
   delete p4r-env directory
3. Download p4r-env:
   git clone --recursive https://gitlab.com/cerl/plan4res/p4r-env.git
4. cd p4r-env
5. bin/p4r

4.2 Update existing Instal

1. cd p4r-env
2. bin/p4r

⚠ bin/p4r downloads an updated SIF image, whose size is very big, and can thus take a long time…

bin/p4r -t checks if the install is ok

In case you wish to avoid updating (and thus downloading a very big file) each time you run p4r-env, you can change the config file: p4r-env/config/plan4res.conf

P4R_SINGULARITY_IMAGE_PRESERVE=1 instead of P4R_SINGULARITY_IMAGE_PRESERVE=0

4.3 P4r-env structure

The p4r-env repository contains the following sub-repositories:

- bin: contains all software executables. Those are in the user's search PATH
- config: contains configuration files:
  - p4r-env configuration files
  - sms++ configuration files
- data: this repository is meant for storing all data and results. This repository contains 4 sub-repositories:
  - cache: for data cached from external sources
  - local: for storing user's data
  - scratch: used as working directory of tools and can be used for volatile data during program runs
  - staging: for storing data that need to be uploaded to external repositories
- scripts:
  - scripts contains the main workflow scripts, which are executing different tasks repeteadly or sequencially. In particular it should contain:
    - LaunchSSV: which will run data formatting, followed by execution of SSV
    - LaunchSIM: which will run data formatting, followed by execution of SIM followed by post-treatment, visualization and writing IAMC files
    - LaunchCEM: which will run data formatting, followed by execution of CEM followed by post-treatment, visualization and writing IAMC files
  - Scripts/add-ons contains the scripts for installing stopt and sms++

- site: used by p4r-env users to install their own local software

4.4 Using p4r-env

To launch the plan4res environment, go in the p4r-env repository and type bin/p4r. This will start a shell in the plan4res containerized environment such as:

[P4R-ENV] /home/Plan4Res/P4Rtest/p4r-env >

In classical uses of plan4res, you will not need to work from the shell as scripts for running all different workflows are available which include launching the environment and all softwares.

5 Install Softwares

In order to run plan4res you need to install :

- StOpt
- SMS++ which will create the 3 executables for the CEM, SSV and UC models
- The python scripts:
    - Formatting tool
    - Preprocessing tool
    - Postprocessing tool
    - Visualisaiton tool
    - IAMC writing tool

5.1 SMS++

SMS++ uses its own solving libraries as well as external libraries for solving linear problems. It is able to use both CPLEX, GUROBI, SCIP, HIGHS, and COIN-OR.

Instructions for installation of these softwares can be found here https://gitlab.com/smspp/smspp-project/-/wikis/Installing-SMS++#sms

As an example, in case you are using cplex for solving LPs, you need to copy your cplex installer file (cplex_studioxxx.bin) in the p4r_env directory, and ensure your cplex licence is ok.

StOpt must be installed first.

6. If you had previously installed (even if it failed), remove StOpt and sms++:
   bin/p4r add-on stopt uninstall
   bin/p4r add-on sms++ unistall

7. Install StOpt :
   bin/p4r add-on stopt

8. Install SMS++ (from Master version)
   bin/p4r add-on sms++ install CPLEX=cplex_studioxxx.bin (if using cplex)

You may prefer to use the dev version of SMS++ , in this case you need to do the following:

bin/p4r add-on sms++ getdev
bin/p4r add-on sms++ compile CPLEX=cplex_studioxxx.bin (if using cplex)
bin/p4r add-on sms++ install

see SMS++ / The SMS++ Project · GitLab for more options.

5.2 Python scripts

The python scripts need to be installed by the user in p4r-env/scripts/python (after creation of this repo)

cd p4r-env/scripts
mkdir python
cd python

- **openENTRANCE nomenclature definitions :**

This python package contains definitions of variables and regions, it is used by the other python scripts.

git clone https://github.com/openENTRANCE/openentrance.git

- **Python scripts for pre processing, postprocessing, formatting, visualitation and IAMC writing**

  git clone https://github.com/openENTRANCE/plan4res-scripts.git

5.3 Get documentation, configuration files and launching scripts

- **General workflow scripts**
  These are the scripts LaunchSSV, LaunchSIM and LaunchCEM, which are part of the https://github.com/openENTRANCE/plan4res.git repository. These scripts are meant to be in p4r-env/scripts
  - LaunchSSV proceeds to the following steps:
    - Create netcdf files
    - Run sddp_solver (computes BellmanValues)
  - LaunchSIM proceeds to the following steps:
    - Create netcdf files
    - Run sddp_solver (compute simulations)
    - Post-treat results, write IAMC files and create graphs
  - LaunchCEM proceeds to the following steps:
    - Create netcdf files
    - Run investment_solver (compute best investment and run simulations)
    - Post-treat results, write IAMC files and create graphs

- **SMS++ configuration files** are also in the https://github.com/openENTRANCE/plan4res.git repository. They need to be moved to p4r-env/config