Updates – RESILIENT

UNIPI

18/12/2024 -- 12/01/2026

(OR1) Model characterization and mapping: PyPSA and SMS++

- Mathematical classification of PyPSA and SMS++
- Identification of mapping PyPSA components into SMS++ by aligning
  - Identification of suitable SMS++ Blocks to represent PyPSA components
  - Mathematical expressions of objective values and constraints
  - Meaning of input parameters
  - Objective functions
- Classification of necessary changes to represent selected PyPSA objects in SMS++ (integrated in OR2)
  - Introduction of marginal costs in relevant SMS++ blocks DCNetworkBlock, IntermittentUnitBlock, HydroUnitBlock
  - Integration of cycling notation to HydroUnitBlock
  - Introduction of efficiency in DC lines representation (DCNetworkBlock)
  - Introduction of multi-link representation in SMS++
  - Improved support of capacity expansion problems in UCNetworkBlock
  - Support negative loads and negative generators (e.g. SlackUnit and IntermittentUnitBlock)
  - Among others

Fioriti D., et al., “Enhancing Energy System Modelling with advanced mathematical decomposition techniques: feasibility of coupling SMS++ and PyPSA” 2025, EEEIC conference

(OR1) Model characterization and uncertainties

- Literature analysis on uncertainties
- Investigation of the role of selected uncertainties in model results
  - Methodology for analysis
  - Impact analysis on a selected system within Europe

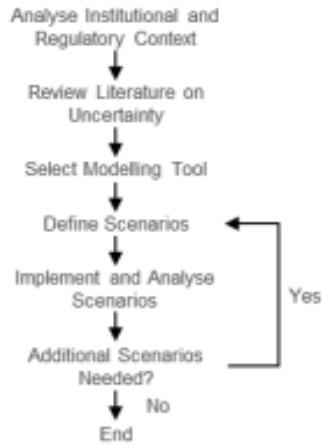


Fig. 1: Flowchart of the proposed framework.

1) **LED – Low Electricity Demand**: Annual demand is reduced to 404 TWh, based on the PNIEC-Slow scenario [20], representing a lower level of electrification and slower energy transition.
2) **WEO – World Energy Outlook Prices**: Fossil fuel prices are set according to the NZE pathway from the WEO 2024, with significantly lower values than the DE-IT scenario: 12.84 €/MWh for natural gas, 5.56 €/MWh for coal, and 16.90 €/MWh for oil.
3) **NCL – Open Nuclear Deployment**: Nuclear technologies are allowed to be installed in all countries, regardless of current national policies or existing infrastructure.
4) **BIO – Biomass Expansion**: No upper bound is imposed on biomass deployment, allowing the model to explore its maximum contribution to the energy mix, given the growing relevance of biomass for future scenarios.
5) **NOCO2 – Zero Emissions**: A strict constraint is imposed that prevents any CO_2 emissions from the power sector. This represents an ambitious and highly idealised target, beyond the scope of Italy's current 2050 decarbonisation plans [20].
6) **BIOCO2 – Biomass Expansion under Zero Emissions**: A combination of the BIO and NOCO2 assumptions, to assess the role of biomass as a dispatchable, carbon-neutral technology in a fully decarbonised power system.
7) **NCLCO2 – Open Nuclear Deployment under Zero Emissions**: A combination of the NCL and NOCO2 scenarios, exploring the potential role of nuclear energy in achieving a zero-emission system.

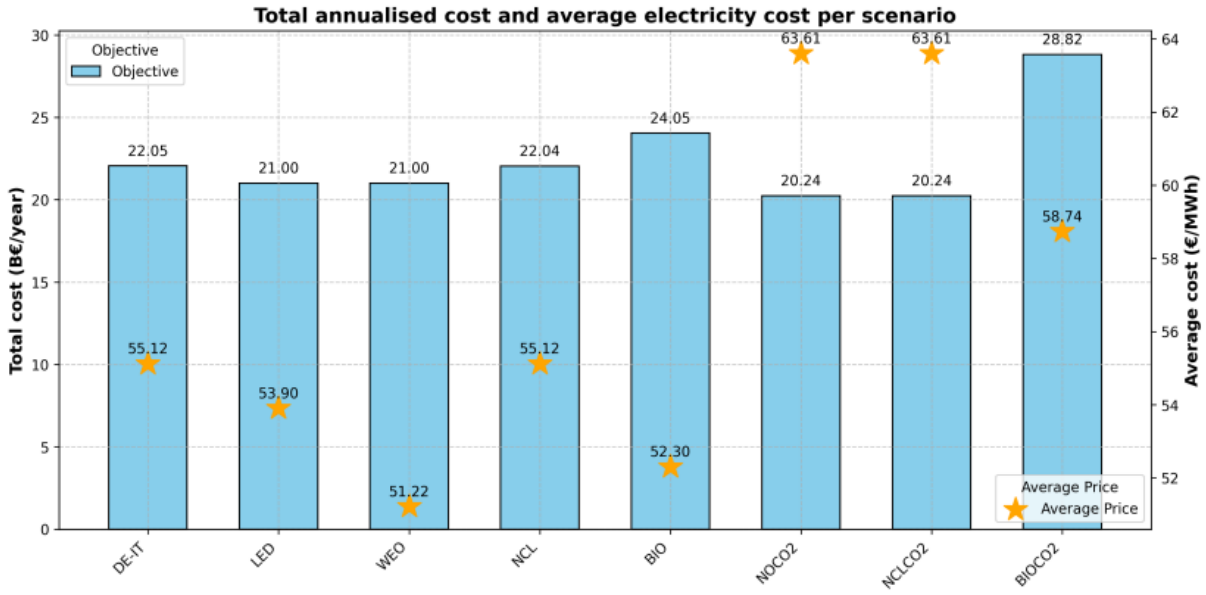


Fig. 6: Optimal total cost (bars) and average electricity cost (stars) for the different scenarios.

Pampado A., Fioriti D., Poli D., "Exploring uncertainties in future power systems: methodology and analysis for Italy 2040" 2025, EEEIC conference

(OR2) SMS++ developments

Activities done

- Defined ScenarioGenerator, implemented DiscreteScenarioSet and ScenarioReductionSolver
- Finalised development of TwoStageStochasticBlock
- Improved data output: all relevant :Block now have :Solution
- Multi-energy features designed, implemented, tested
- Developed PrimalProximalSolver
- Developed crucial capability in LagBFunction and LagrangianDualSolver (objective changes in sub-Block)
- Added design variables to DCNetworkBlock
- Many bugfixes and minor features added
- Developed CONDA package for ease of integration https://github.com/conda-forge/staged-recipes/pull/29920

Planned activities

- Extensive tests of large-scale, stochastic problems with different approaches (off-the-shelf, InvestmentSolver, LagrangianDualSolver) to identify the most appropriate Configuration for each setting
- Development of nested decomposition approaches for two-stage problems
- Analysis of BendersDecompositionSolver
- Analysis of the extension of TwoStageStochasticBlock to Benders' reformulation
- Draft development of MultiStageStochasticBlock
- Analysis of BundleSolver 2.0 (MasterProblemBlock)
- Development of other features as needed by PyPSA

Developments available at https://gitlab.com/smspp/smspp-project

(OR3) PyPSA – SMS++ interface

Investigated strategies

1. Notebook implementation
   https://github.com/SPSUnipi/SMSpp_PyPSA

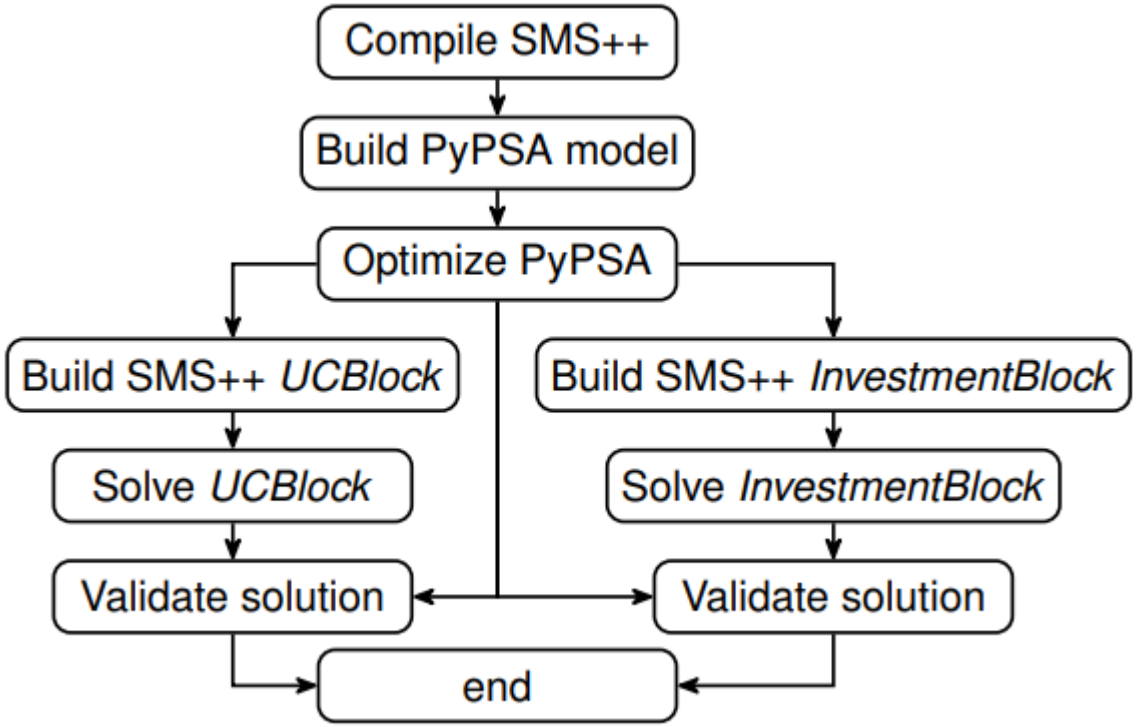
2. No auxiliary packages
   https://github.com/SPSUnipi/SMSpp_builder

3. Auxiliary repositories

   – Python input/output for SMS++
   https://github.com/SPSUnipi/pySMSpy

   – Transformation PyPSA – SMS++
   https://github.com/SPSUnipi/pypsa2smspp

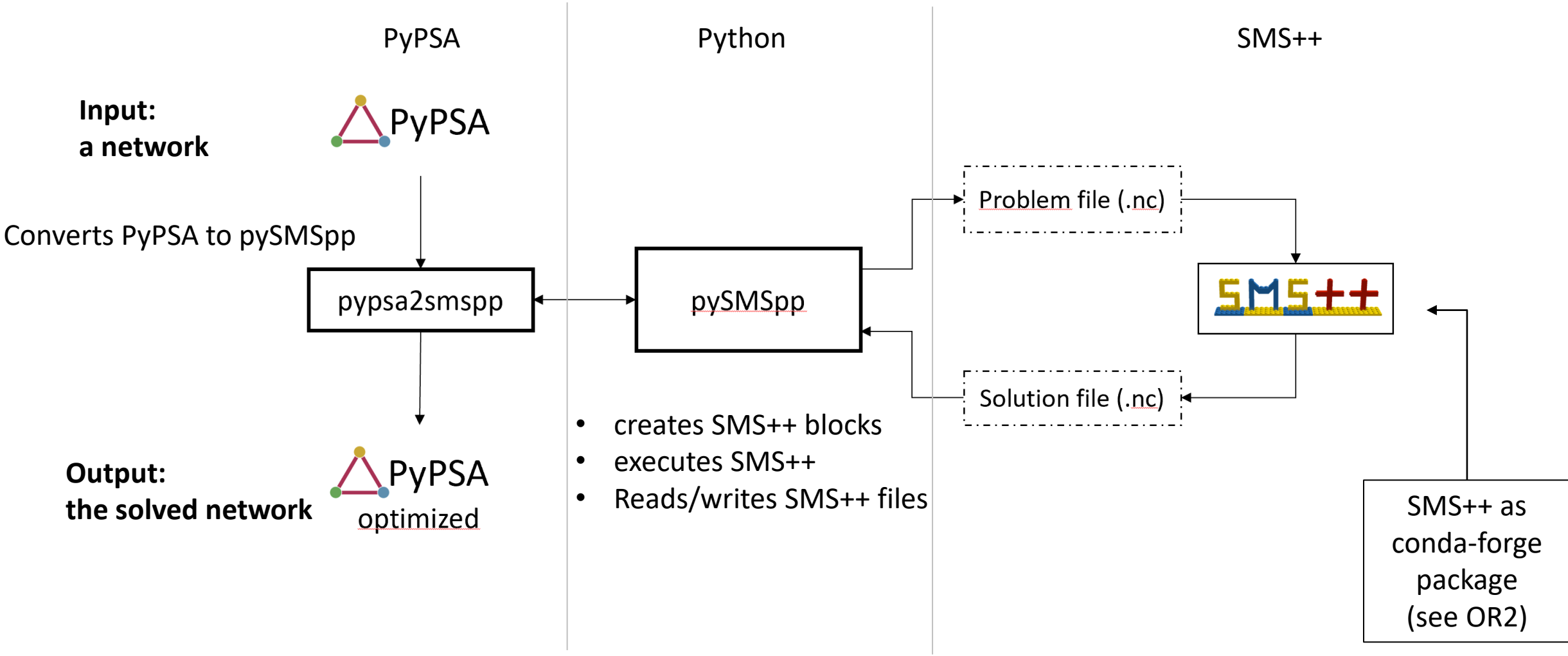
Prototype implementation



Implemented in https://github.com/SPSUnipi/SMSpp_builder

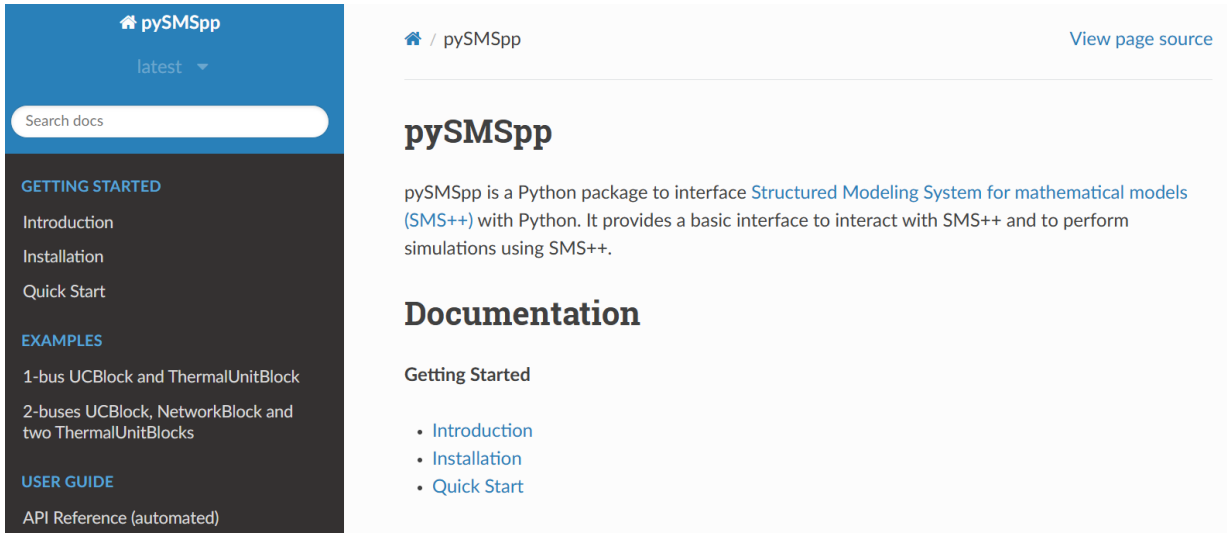
Fioriti D., et al., "Enhancing Energy System Modelling with advanced mathematical decomposition techniques: feasibility of coupling SMS++ and PyPSA" 2025, EEEIC conference

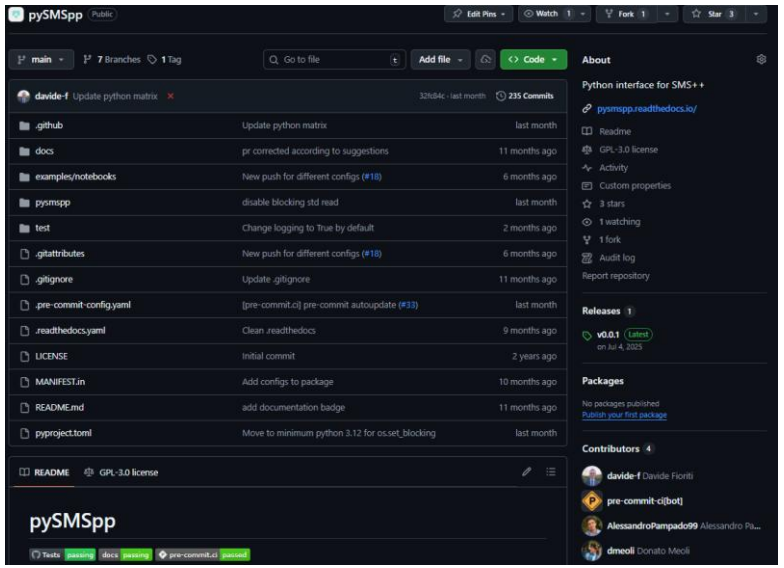
(OR3) PyPSA – SMS++: Proposed interface



(OR3) pySMSpp

- **Goal:** enable bidirectional interaction with SMS++ through input-output file
- **Structure:** modular structure to represent SMS++ blocks and utilities to save, load and optimize the blocks
- **Installation**: package available in pip
- Includes automated testing using Github Continuous Integration

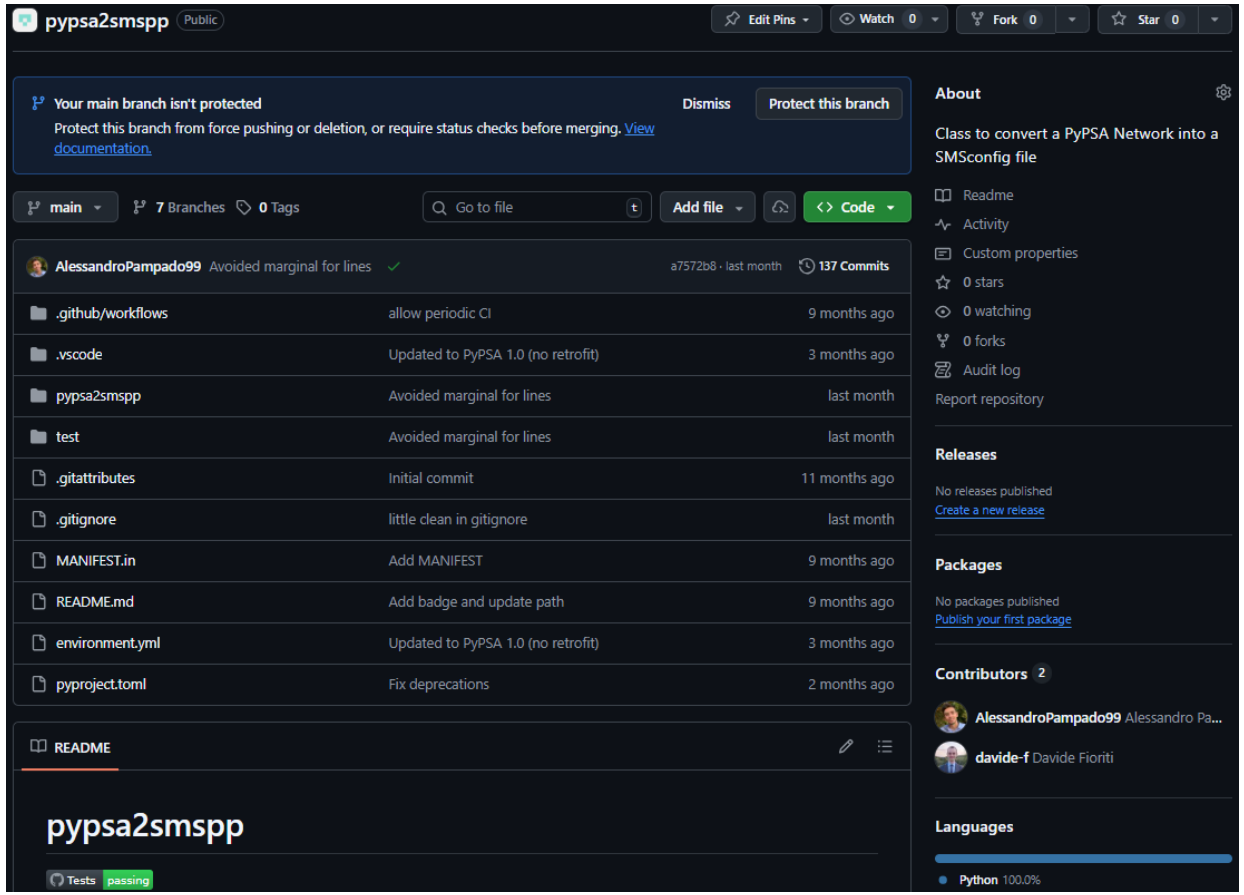




https://github.com/SPSUnipi/pySMSpp

(OR3) pypsa2smspp

- **Goal:** enable bidirectional conversion from a PyPSA object to a pySMSpp object to allow solving a PyPSA network using SMS++
- **Structure:** modular structure to transform PyPSA networks into SMS++ networks, while allowing selected decomposition techniques
- **Installation**: package available in pip
- Includes automated testing using Github Continuous Integration

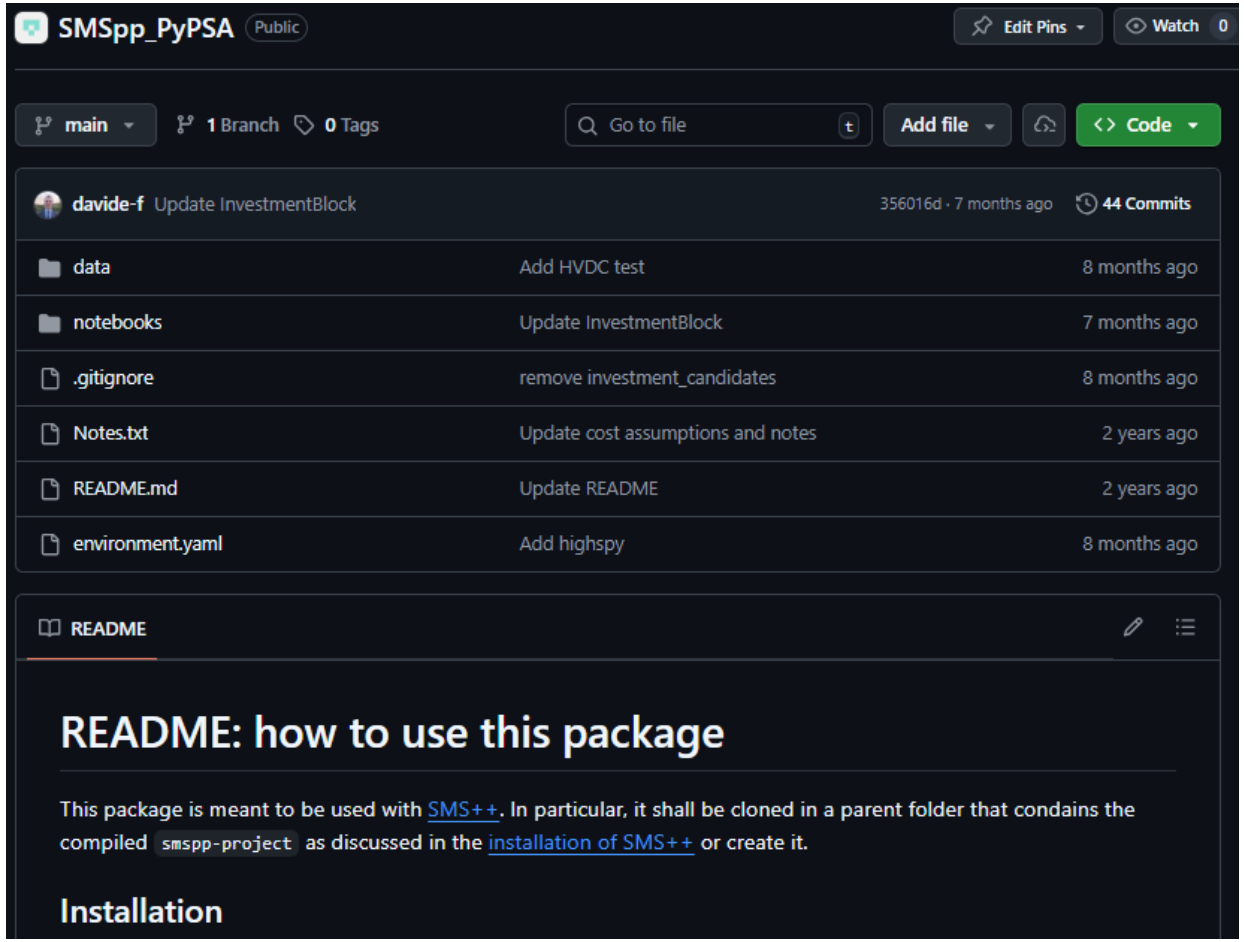


https://github.com/SPSUnipi/pypsa2smspp

(OR3) SMSpp_PyPSA repository for investigation and testing

- **Goal:** repository that contains raw implementation and visualization of PyPSA to SMS++ interface

- **Structure:** contains notebooks and scripts for file writing suitable for SMS++ using PyPSA inputs



https://github.com/SPSUnipi/SMSpp_PyPSA

Previous report on activities 2024

Updates on SMS++ – PyPSA interface

UNIPI

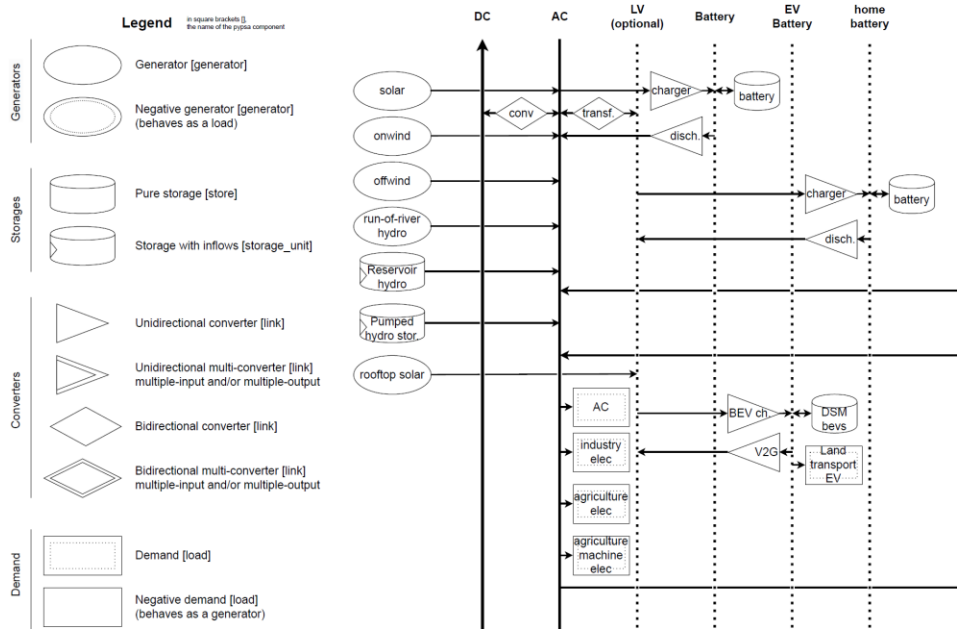
18/12/2024

Model characterization – PyPSA-Eur energy system

- **Mapping of PyPSA-Eur structure**
  - Carriers
  - Technologies and components
  - Types of parameters by component

	A	B	C	D	E
1	**Technology name**	**Category**	**Physical compone**	**Option**	**Carrier**
2	co2 atmosphere	co2	N		co2
3	Co2 storage	co2	Y		co2 stored
4	Sequestration link	co2	N		co2 sequestred
5	Sequestration store (e.g. underground)	co2	Y/N		co2 sequestred
6	CO2 vent co2 from storages	co2	?	co2_vent	co2 vent
7	CO2 pipelines	co2	Y	co2_network	CO2 pipeline
8	Allam (gas) cycle	electricity	Y	allam	allam
9	Direct Air Capture	co2	Y	dac	co2
10	Conventional generators	electricity	Y	conventional_generation	electricity
11	Haber-Bosch process	ammonia	Y	ammonia	Haber-Bosch
12	Ammonia cracker	ammonia	Y	ammonia	ammonia cracker
13	Ammonia storage	ammonia	Y	ammonia	ammonia store
14	Electricity distribution	electricity	Y	electricity_distribution_grid	low voltage
15	rooftop solar	electricity	Y	electricity_distribution_grid	solar rooftop

- **Graphical representation:**
  - **PyPSA-Symbols-drawio :**
    repository with drawio symbols for PyPSA objects
    https://github.com/SPSUnipi/PyPSA-symbols-drawio
  - **PyPSA-Eur-drawio:**
    repository with drawio graph for PyPSA-Eur
    https://github.com/SPSUnipi/PyPSA-Eur-drawio



Model characterization – Mathematical representation

PyPSA

Objective function	Symbol	Generator	Link	Line	Storage unit	Store
Capital cost	CAP	X	X	X	X	X
Marginal cost	MC	X	X	X	X	X
Marginal cost energy storage	MCE				X	X
Stand-By costs	SB	X	X			
Start up/ shut down cost	SC	X	X			
Splillment costs	SC				X	

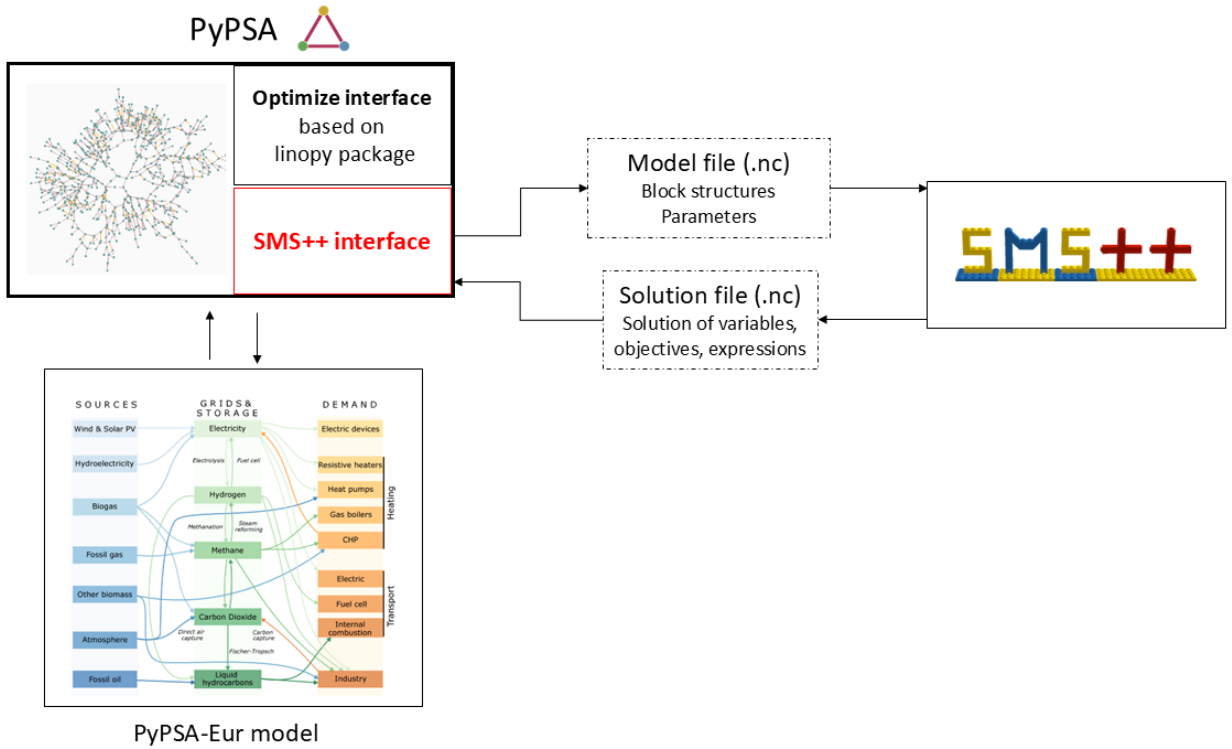
Equation	Symbol	Generator	Link	Line	Storage unit	Store	Condition	Example
Size bound	SB	X	X	X	X	X		$\underline{G}_{i,r} \leq G_{i,r} \leq \overline{G}_{i,r}$
Modularity	MD	X	X	X	X	X		$G_{i,r} = \overline{G}^{mod}_{i,r} n_{i,r}$
Power bound	PB	X	X	X	X	X		$\underline{g}_{i,r,t} G_{i,r} \leq g_{i,r,t} \leq \overline{g}_{i,r,t} G_{i,r}$
Power unit commitment	PB_{UC}	X	X				committable	$\delta_{i,r,t} \underline{g}_{i,r,t} G_{i,r} \leq g_{i,r,t} \leq \delta_{i,r,t} \overline{g}_{i,r,t} G_{i,r}$
Minimum time	MT	X	X					$\sum_{t'=t}^{t+T^{min_{up}}} \delta_{k,t'} \geq T_{min_{up}} (\delta_{k,t} - \delta_{k,t-1})$
Total energy produced	PSUM	X	X					$E^{sum}_{min} \leq \sum_{t \in T} w^G_t g_{i,r,t} \leq E^{sum}_{max}$
Start up/shut down cost	SC	X	X					$suc_{k,t} \geq suc_k (\delta_{k,t} - \delta_{k,t-1})$
Rump up/down	RUD	X	X					$(g_{i,r,t} - g_{i,r,t-1}) \leq \tau u_{i,r} G_{i,r}$
Kirchhof's law	KL			X				$\sum_l C_{l,c} x_l p_{l,t} = 0$
Line losses	LL			X				$P^{loss}_l = \alpha_l + \beta_l p_{l,t}$
Energy storage level	ESL				X	X		$e_{i,s,t} = e_{i,s,t-1} + w^S_t h_{i,s,t}$
Energy storage bound	ESB				X	X		$0 \leq e_{i,s,t} \leq E_{i,s} e^{max}_{i,s}$
Initial energy level	IEL_S				X	X		$e_{i,s,0} = e_{i,s,init}$
Cyclic energy level	IEL_{CS}				X	X	cyclic_state_of_charge	$e_{i,s,0} = e_{i,s,\|T\|}$

SMS++

Equation	Symbol	Intermittent	Thermal	Battery	Hydro	DC Network
Maximum reserves power	RMAX	X	X	X	X	
Minimum reserves power	RMIN	X	X	X	X	
Power bound	PB	X				
Power bound unit commitment	PB_{UC}	X				
Binary relation	BINR		X			
Binary start-up	BINU		X			
Binary shut-down	BIND		X			
Rump up/down	RUD		X	X	X	
Primary reserves	RPR		X	X		
Secondary reserves	RSC		X	X		
Power bound $\tau = 2$	$PB_{\tau=2}$		X			
Power bound $\tau = 1$	$PB_{\tau=1}$		X			
Power balance	PBAL			X		
Power bound discharge	PB^+			X		
Power bound charge	PB^-			X		
Power bound discharge unit commitment	PB^+_{UC}			X		
Power bound converter	PBC			X		
Energy storage level	ESL			X		
Energy storage level simplified	ESLS			X		
Energy storage bound	ESB			X		
Energy storage discharge	ESD			X		
Energy storage charge	ESC			X		
Volume bound	VB				X	
Primary reserves turbine	RPR_T				X	
Primary reserves pump	RPR_P				X	
Secondary reserves turbine	RSC_T				X	
Secondary reserves pump	RSC_P				X	
Power-to-flow function turbine	PTF_T				X	
Power-to-flow function pump	PTF_P				X	
Volume level	VL				X	
Power bound DC network	PB_{DC}					X
Power bound AC network	PB_{AC}					X
Power bound AC-DC network	PB_{AC-DC}					X
Network cost	NC					X
Energy balance	EBAL					X

PyPSA – SMS++ interface

The goal



PyPSA-Eur model

Possible strategies

1. Notebook implementation
   https://github.com/SPSUnipi/SMSpp_PyPSA
2. No auxiliary packages
   https://github.com/SPSUnipi/SMSpp_builder
3. Auxiliary repositories
   - Python input/output for SMS++
     https://github.com/SPSUnipi/SMSpy
   - (optionally) transformation PyPSA – SMS++
     https://github.com/SPSUnipi/PyPSA_SMS_interface

SMSpp_builder: compile and test SMS++ PyPSA interface

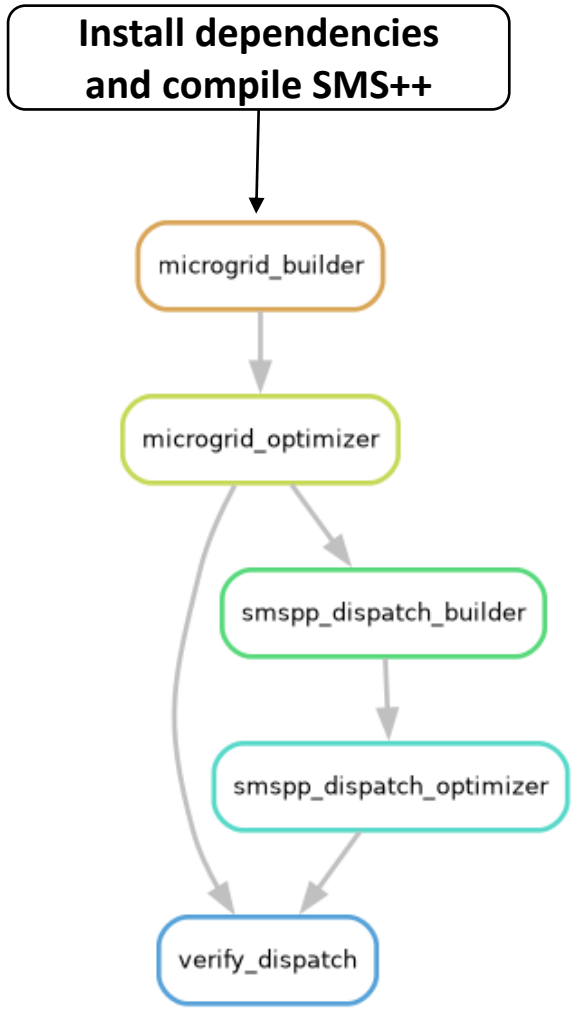
- **Goal:**
  Test compilation and execution of PyPSA - SMS++

- **Automation:**
  - Github Action
  - Snakemake (except installation and compilation)

- **Tested configurations (dispatch analysis)**

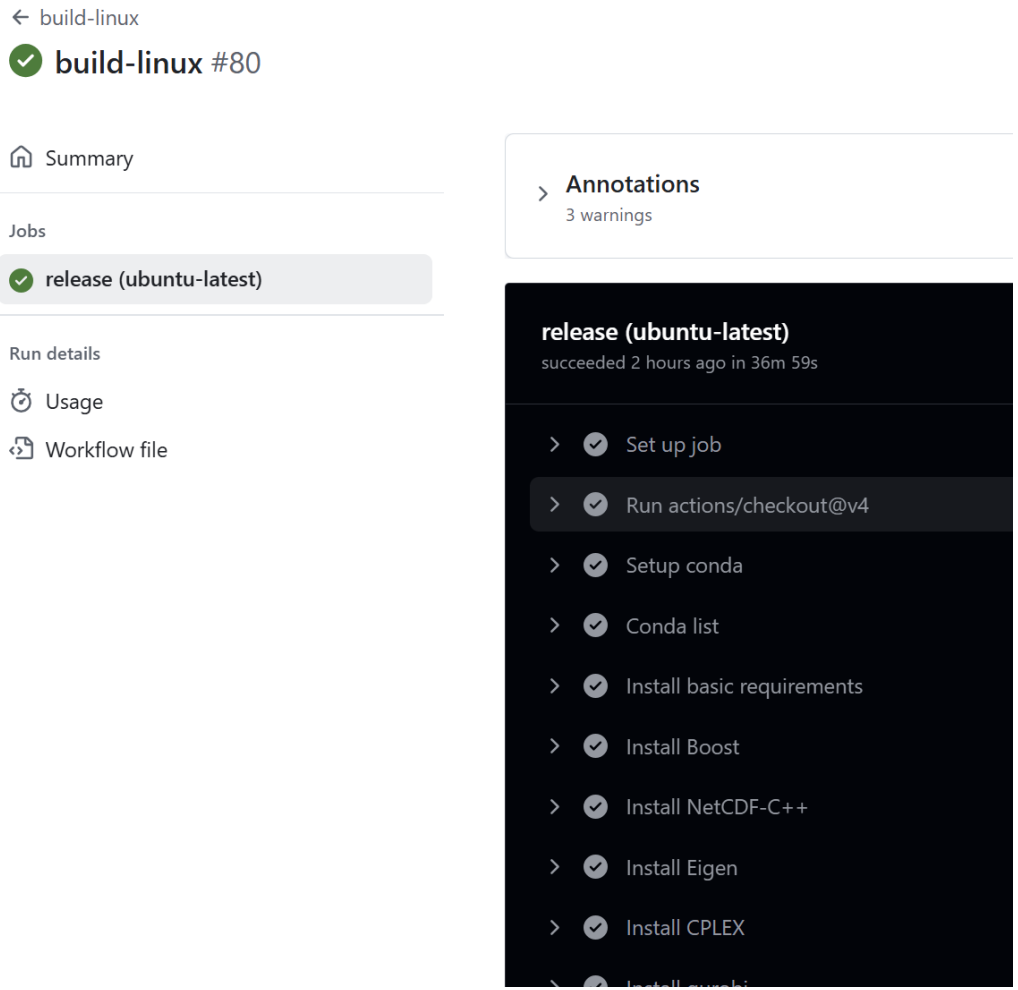
Test case	Network	Nodes	PyPSA components					
			Generator	StorageUnit	Store	Line	Link	Load
	[Yes/No]	[#]	[#]	[#]	[#]	[#]	[#]	[#]
1	No	1	1					1
2	No	1	3	2				1
3	Yes	2	1			1		2
4	Yes	5	3	2	1	3	1	4
5	No	2	1		1		1	2

Table 4.1: Test cases of *SMSpp_builder*



SMSpp_builder: compile and test SMS++ PyPSA interface

CI is successful



SMS++ results match PyPSA

```
[Tue Nov 19 18:47:51 2024]
localrule verify_dispatch:
    input: results/networks/microgrid_microgrid_ALL_4N_optimized.nc, result
    output: results/microgrid_microgrid_ALL_4N_complete.txt
    log: logs/verify_dispatch_microgrid_ALL_4N.log
    jobid: 0
    reason: Missing output files: results/microgrid_microgrid_ALL_4N_comple
    resources: tmpdir=/tmp

INFO:pypsa.io:Imported network microgrid_microgrid_ALL_4N_optimized.nc has
INFO:verify_dispatch:SMS++ obj                                : 19315.875900
INFO:verify_dispatch:PyPSA dispatch obj                       : 19315.875923
INFO:verify_dispatch:Relative difference SMS++ - PyPSA [%]: -0.00000
INFO:verify_dispatch:Absolute difference SMS++ - PyPSA [€] : -0.00002
INFO:verify_dispatch:Verification successful
Touching output file results/microgrid_microgrid_ALL_4N_complete.txt.
```

Ongoing priorities

Energy system representation

- **Finalize PyPSA and SMS++ description**
- **Define mapping of mathematical representations**
- **Define key priorities** and requirements for RESILIENT
- **Implement** in SMS++ the key missing options

Interface

- **Integrate Transformers**

  Added transformers! ×

  #4 opened 3 weeks ago by AlessandroPampado99 • Review required

- **Support capacity expansion problems**
- **Define software architecture of interface**
  - Notebook implementation (discarded)
  - No auxiliary packages (not recommended)
  - Auxiliary packages
    - comprehensive PyPSA-SMS++ conversion package
    - Python SMS++ input/output package
    - PyPSA-SMS++ conversion repository or PyPSA integration
- **Support more parameters and functionalities**

SMS++ developments

Activities done

- **Enhanced installation** file by a single bash file:
  - INSTALL.sh in Mac/Ubuntu
    https://gitlab.com/smspp/smspp-project/-/blob/develop/INSTALL.sh
  - INSTALL.ps1 for windows
    https://gitlab.com/smspp/smspp-project/-/blob/develop/INSTALL.ps1
- **Development of [MultiStage]ScenarioGenerator**
- **Initial development of TwoStageStochasticBlock**
- **Handling of quadratic constraints in MILPSolver**
- **Integration of AC load flow**
  https://gitlab.com/smspp/ucblock/-/blob/develop/src/ACNetworkBlock.cpp
- **Bug fixing**
- **Preliminary investigation of Multi-Energy design**

Planned activities

- **Define / implement OptimalTransportBlock and :OptimalTransportSolver** [scenario reduction, Q3 2025]
- **Finalise development of TwoStageStochasticBlock** [goal functional draft by Q1-Q2 2025]
- **Improve data output** [to support interface]
- **Multi-energy design** [After M18]
- **Improve BundleSolver** [decomposition]
- **Develop PrimalProximalSolver** [integer decomposition]
- **Bug fixing:**
  - InvestmentBlock solver for investment analyses
  - Hydro issue with storing capabilities
- **Feature inclusions to adapt to RESILIENT needs**