BERKELEY LAB
Bringing Science Solutions to the World

IBPSA Project 2

BOPTEST

U.S. DEPARTMENT OF ENERGY
Office of Science

IBPSA Project 2 Expert Meeting Task 3

Ettore Zanetti ezanetti@lbl.gov

11-20/21-2024

Task 3: Test Cases Progress, Breakout sessions

Session 1

- Updates to current test cases
- New test case development status
- New test case ideas and contributions
- Changing the test case naming scheme

Session 2

- Test case compilation and execution tools
- Test case development and review process

Task 3: Test Cases, Session 1
Updates to current test cases

All test cases

- Added activate “<input>_activate” signals to test cases documentation

BESTEST Air and BESTEST Hydronic

- Change cooling setpoint "con_oveTSetCoo_u” min and max from [23 30] to [5 35] and heating setpoint from [15 23] to [5 35]. This is to avoid unreasonable limits for controllers.

BESTEST Hydronic and BETEST Hydronic Heat Pump

- Change dependency by removing them from IDEAS library and moving them to BOPTEST repository. This will help update and maintain the models more easily.

Multizone Office Simple Air

- Correct typo in documentation for cooling setback from 12 to 30.

Task 3: Test Cases, Session 1
Updates to current test cases

Multizone Residential Hydronic

- Update Modelica code so that overwrite input ”oveTSetPum” to ”oveTSetPumBoi” so that this set point change will control thermostat activating both the boiler and the circulation pump. This allows supervisory control of the HVAC with only set points.
- Pump baseline control logic is changed from PI following error on set point to on/off depending on thermostat control signal. This helps simplify baseline controller making numerical simulation faster and more robust.
- A safety on boiler control is added, allowing it to turn on only if there is flow through the boiler. This safety is bypassed if controlling the boiler directly via ”boi_oveBoi_u”. This makes baseline controller safer and more robust. However, if low lever control is implemented by supervisory algorithm it could crash the simulation.

Task 3: Test Cases, Session 1
Updates to current test cases

Single zone commercial hydronic

(Updates from Adrenalin)

Task 3: Test Cases, Session 1
Updates to new test cases

- Large Office [Yan, Xing]

No news. Need 2025 budget to start working on it again. Ettore will take a look at t = 0 initialization error.

- Twozone Apartment Hydronic [Ettore]

No updates. Two zone model with DER and storage was developed in a separate effort.

- DTU emulators [Matthias]

No updates. They are going to do experimental campaign on the real buildings and hopefully start development

- ETS emulator [Kun]

No updates, Ali still waiting on Visa.

Task 3: Test Cases, Session 1
New test cases Ideas and Contributions

Open discussion:

- What test cases do you think are missing?
- Who wants to contribute a new test case?

Task 3: Test Cases, Session 1
Changing the test case naming scheme

ID_<building_type>_<n>zones_<hvac_system>_<location>

ID: Unique identifier of test case, could be number, could be word

<building_type>: residential,commercial,industrial, district

<n>zones: Number of thermal zones

<hvac_system> : have unique tags to identify VAV, FCU, RAD, RADF…

<location>: city-country

The composition of these values could make up the name of the test case

Task 3: Test Cases, Session 1
Changing the test case naming scheme

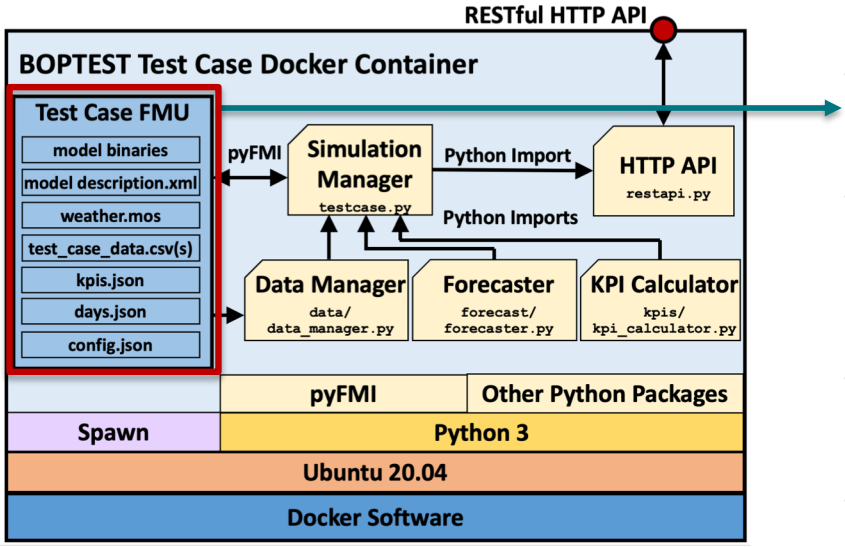
Previous name	Proposed name
BESTEST Air	TC1_BESTEST900_commercial_1zon_FCU_Denver-USA
BESTEST Hydronic	TC2_BESTEST900_residential_1zon_RAD_Brussel-Belgium
BESTEST Hydronic Heat Pump	TC3_BESTEST900hp_residential_1zon_RAD-AWHP_Brussel-Belgium
Single zone commercial Hydronic	TC4_University001_commercial_1zon_RAD-AHU_Copenhagen-Denmark
Two Zone Apartment Hydronic	TC5_Apartment001_residential_2zon_RADF-AWHP_Milan-Italy
Multizone Residential Hydronic	TC6_Singlefamily001_residential_6zon_RAD_Bordeaux-France
Multizone Office Simple Air	TC7_Office001_commercial_5zon_VAV_Chicago-USA
Single Zone Commercial Air	TC8_Retail001_commercial_1zon_VAV_NewYorkCity-USA
Flexible Research Platform	TC9_Testbed001_commercial_10zon_VAV_Tennessee-USA
Multizone Office Simple Hydronic	TC10_Office002_commercial_2zon_FCU-AWHP_Brussel-Belgium
Multizone Office Simple Hybrid	TC11_Office003_commercial_27zon_TABS-AHU-GSHP_Dilbeek-Belgium
Multizone Office Complex Air	TC12_Office004_commercial_5zon_VAV_Chicago-USA

Task 3: Test Cases, Session 2

- Test Case compilation and execution tools
- Test Case Development and review process

Task 3: Test Cases, Session 2
Test Case compilation and execution tools

History and Current State



- BOPTEST uses co-simulation FMUs to encapsulate test cases
- Most of the test cases are compiled using JModelica as at the time it was the most reliable open source Modelica compiler
- Unfortunately JModelica has been discontinued since Dec 2019
- Currently BOPTEST includes Test Case compilation in the unit tests for most test cases
- This means that most test cases are still using Modelica 3.2.3 and older versions of the Modelica libraries (Buildings, IDEAS)

Task 3: Test Cases, Session 2
Test Case compilation and execution tools

Current Developments

- The only open source compiler available currently is OpenModelica (OM)
- Currently in development branch 422 Test Cases have been updated to Modelica 4.0 and relative version of the libraries, and partially tested with OM

TestCase	Compile Dymola	Simulate Dymola	Compile OMEdit	Simulate OMEdit	OM FMU Compile	OM FMU Simulate in pyfmi
bestest_air	v	v	v	v	x	x
bestest_hydronic	x	v	v	v	o	o
bestest_hydronic_heat_pump	v	v	v	v	o	o
singlezone_commercial_hydronic	v	v	v	v	o	o
multizone_residential_hydronic	v	v	v	v	o	o
multizone_office_simple_air	v	v	v	o	o	o
twozone_apartment_hydronic	v	v	v	o	o	o

- Even though most models compile and simulate in OM, FMU compilation remains an open question because currently OM co-simulation FMUs are under development and only forward Euler solver is available.

- When Multizone Office Simple Hydronic was added to the available test cases Optimica was made available as compiler option for BOPTEST FMUs

Task 3: Test Cases, Session 2
Test Case compilation and evecution tools

Future Directions

Issues:

- Jmodelica not supported anymore and models stuck at Modelica 3.2.3
- OpenModelica (OM) only experimentally supports co-simulation FMUs
- PyFMI has a bug on event handling for model exchange FMUs

Possible solutions

Use Model exchange FMUs

- Use PyFMI do step and handle events in BOPTEST
- Change PyFMI with FMPy or OMSimulator
- Export model exchange FMUs with OM

Keep current implementation

- Use OM when possible
- Use Optimica compilation option for all other models

Task 3: Test Cases, Session 2
Test Case Development and review process

Comments from previous test case development efforts

- Let's face it, Modelica development is not easy. Especially if multiple tools are considered (Dymola, Optimica, OM)
- Volunteer effort comes in bursts of spare time, and monthly meetings may be too slow to quickly address issues and get feedback
- BOPTEST overhead on top of Modelica model:
  - Boundary conditions for forecasts, typical periods, documentation
  - Test case review
- Data availability and model calibration

Task 3: Test Cases, Session 2
Test Case Development and review process

How can we improve test case development and speed it up?

- Prepare onboarding process for new test case developments to help understand BOPTEST test case nuances
- Have quick feedback meetings outside monthly meetings upon request “office hours”
- Proposed cheat sheet with ”typical values” (buildings properties, internal gains, etc..)
- Proposed updated test case review document that can be also used by test case developer as general guideline
- Proposed test case stress test script that can be used by developer to quickly ”rattle” test case with different control sequences to check for robustness

Task 3: Test Cases, Session 2
Test Case Development and review process

How can we improve test case development and speed it up?

- Prepare onboarding process for new test case developments to help understand BOPTEST test case nuances

Help test case developer navigate BOPTEST test case development depending at what level they start:

- Have test case developer present existing Modelica model or reference case study if Modelica model needs to be developed
- Provide feedback on Modeling requirements using review document and point developer to useful models among Modelica libraries
- Set up discussion tab on GitHub that can be used during development
- Guide developer through BOPTEST reseources, utility scripts and test case structure

Task 3: Test Cases, Session 2
Test Case Development and review process

How can we improve test case development and speed it up?

- Have quick feedback meetings outside monthly meetings upon request “office hours”
- Use test case discussion tab to post quick questions and setup meeting in case questions cannot be addressed right away

- Proposed cheat sheet with ”typical values” (buildings properties, internal gains, etc..)

- Put together lists of “typical” values and best practices to help modelers. Help asked to take data from international / country standard to be put in this document

Task 3: Test Cases, Session 2
Test Case Development and review process

How can we improve test case development and speed it up?

- Proposed updated test case review document that can be also used by test case developer as general guideline
- New review document will be used also as checkbox by test case developer
- Review document became very detailed. Some questions might not be relevant for climate or HVAC considered in case study (i.e. moisture condensation in heating only radiating systems)
- Create examples of filled documents that can be shared with new test case developers

Task 3: Test Cases, Session 2
Test Case Development and review process

How can we improve test case development and speed it up?

- Proposed test case stress test script that can be used by developer to quickly ”rattle” test case with different control sequences to check for robustness

Input Json: "test_case_name": "bestest_air”
"compare_with_baseline": false
"simulation_step": 3600
"show_plots": true
”forecast_check": true

"scenarios": [{"time_period": "all", "electricity_price": "constant”,“custom_input”=true}, {”start_time": 0, “stop_time”: 86400, "electricity_price": "constant”}]

"input_list": ["con_oveTSetCoo_activate","con_oveTSetCoo_u"]

Input .csv: Time series of custom inputs that should be used in a specific scenarios

Task 3: Test Cases, Session 2
Test Case Development and review process

How can we improve test case development and speed it up?

- Proposed test case stress test script that can be used by developer to quickly ”rattle” test case with different control sequences to check for robustness
- The script will instantiate the test case and collect meta data in inputs needed (min and max)
- It will generate random sequence for given inputs in .json file using latin hypercubic sampling given time horizon and step to explore control space
- If model crashes during simulation it will return breaking sequence, FMU log, and .mat file
- It will generate plots of measurements and inputs. If baseline option is set to true it will compare with baseline
- If forecast_check set to true it will save forecast data for the scenarios that can be compared with data in Modelica (weather can be done automatically)