Oemof
Workshop Week

Constraints

Martha Hoffmann

Session 4

RLI, 18.09.2019

Introducing words

REINER LEMOINE INSTITUT

Inner workings of oemof:

Linear optimization and constraints

All workshop contents at: https://github.com/smartie2076/oemof_workshop
Todays jupyter notebooks are stored in ./Day_3_Custom_Constraints_for_Oemof

Agenda

REINER LEMOINE INSTITUT

- Introduction to linear optimization
- Linear equation systems of oemof models
- In-build oemof constraints
- Introducing own constraints
- Further examples for constraints

Agenda

- Introduction to linear optimization
- Linear equation systems of oemof models
- In-build oemof constraints
- Introducing own constraints
- Further examples for constraints

An introduction to linear optimization

- Example: ./3_LP_general_example.ipynb

- Linear Problem (LP) / Mixed Integer (Linear) Problem (MI(L)P) consists of :
  - a target function
  - a set of constraints and balances
- Solver searches on the edges of the solutions space for the optimal solution
- Available solvers: CBC, GLPK, Gurobi, …
- Time steps adjustable (e.g. 15 mins, hourly)

Agenda

- Introduction to linear optimization
- Linear equation systems of oemof models
- In-build oemof constraints
- Introducing own constraints
- Further examples for constraints

Linear equation system generated by oemof

A set of linear equations fully describes an energy system model as a whole

The „lp-file“

- Linear equation system describes energy system fully
  - generated using Pyomo package
  - can be stored in „lp-file“

- The „lp-file“ is transferred to solver for optimization
  - Recommended solver: coinor-cbc
  - Same file can be optimized with different solvers

- Lp-file can help to verify and debug your code

- Example: ./micro_grid_fixed_cap_basic.ipynb

The „lp-file“: Objective value

- The objective value should be minimized:

```
min
objective:
+0.03955047913155272 GenericInvestmentStorageBlock_invest(storage)
+0.022175440918742569 InvestmentFlow_invest(genset_electricity_bus)
+0.034350422598754829 InvestmentFlow_invest(pv_electricity_bus)
+0.091601126930012891 InvestmentFlow_invest(wind_electricity_bus)
+0.063761955366631234 flow(diesel_fuel_bus_0)
+0.063761955366631234 flow(diesel_fuel_bus_1)
+0.063761955366631234 flow(diesel_fuel_bus_2)
+0.063761955366631234 flow(diesel_fuel_bus_3)
+0.063761955366631234 flow(diesel_fuel_bus_4)
```

Optimizing with oemof – Objective value

REINER LEMOINE INSTITUT

- Oemof generates a linear equation system describing the energy system model
- Solves for the minimal objective value (costs)
- Target function:

$$min \sum_i \left(Capex(i) * CRF(i) + Opex_{fix}(i)\right) * P_{inst}(i) + \sum_i \sum_t Opex_{var}(i) * E_{gen}(i,t)$$

$$i \in \{WEA, PV, BHKW, Speicher\}$$
$$t \in \{1 \ldots 8760\}$$

Capex	Capital expenditure	EUR/kW
CRF	Capital recovery factor	-
$Opex_{fix}$	Fixed operational expenditure	EUR/(kW*a)
$Opex_{var}$	Variable operational expenditure	EUR/kWh
P_{inst}	Capacity of component	kW
E_{gen}	Generated electricity per timestep	kWh
i	Index of system components	-
t	Index of time steps	-

The „lp-file“: Bus balances

- Each bus is by default balanced:
  - $\sum inputs = \sum outputs \; \forall \, t$
  - Ie. no energy can be lost or generated from nowhere
  - Can require „shortage“-Source or „excess“-Sink

```
c_e_Bus_balance(fuel_bus_0)_:
+1 flow(diesel_fuel_bus_0)
-1 flow(fuel_bus_genset_0)
= 0
```

The „lp-file“: Bus balances

- Each bus is by default balanced:
  - $\sum inputs = \sum outputs \; \forall \; t$
  - Ie. no energy can be lost or generated from nowhere
  - Can require „shortage“-Source or „excess“-Sink

```
c_e_Transformer_relation(genset_fuel_bus_electricity_bus_0)_:
+1 flow(fuel_bus_genset_0)
-3.0303030303030303 flow(genset_electricity_bus_0)
= 0

c_e_InvestmentFlow_fixed(wind_electricity_bus_0)_:
-0.31556899999999999 InvestmentFlow_invest(wind_electricity_bus)
+1 flow(wind_electricity_bus_0)
= 0
```

The „lp-file“: Bus balances

- Each bus is by default balanced:
  - $\sum inputs = \sum outputs \; \forall \; t$
  - Ie. no energy can be lost or generated from nowhere
  - Can require „shortage“-Source or „excess“-Sink

```
c_e_Bus_balance(electricity_bus_0)_:
-1 flow(electricity_bus_excess_0)
-1 flow(electricity_bus_storage_0)
+1 flow(genset_electricity_bus_0)
+1 flow(pv_electricity_bus_0)
+1 flow(storage_electricity_bus_0)
+1 flow(wind_electricity_bus_0)
= 279.53099120000002
```

Agenda

- Introduction to linear optimization
- Linear equation systems of oemof models
- In-build oemof constraints
- Introducing own constraints
- Further examples for constraints

In-build bounds of flows

- Bounds limit Flows to an interval
- Decreases search area for valid optimization results
- Examples:
  - Component parameters: `min_storage_capacity`, `max_storage capacity`
  - Flow parameters: `nominal_value`
  - Investment parameters: `maximum`
- Example: ./micro_grid_system_inbuilt_bounds.ipynb

The „lp-file“: Bounds

```
0 <= flow(wind_electricity_bus_4) <= +inf
0 <= InvestmentFlow_invest(electricity_bus_storage) <= +inf
0 <= InvestmentFlow_invest(genset_electricity_bus) <= +inf
0 <= InvestmentFlow_invest(pv_electricity_bus) <= 800
0 <= InvestmentFlow_invest(storage_electricity_bus) <= +inf
0 <= InvestmentFlow_invest(wind_electricity_bus) <= 500
0 <= GenericInvestmentStorageBlock_capacity(storage_0) <= +inf
```

```
c_u_GenericInvestmentStorageBlock_min_capacity(storage_0)_:
-1 GenericInvestmentStorageBlock_capacity(storage_0)
+0.20000000000000001 GenericInvestmentStorageBlock_invest(storage)
<= 0
```

In-built constraints

- Limit the sum of a Flow: $\sum Flow \cdot variable = const$
- Indirectly decreases search area of optimization, acts like a „exit criterion of a loop“
- Examples:
  - `summed_max`
  - `emission_limit`

- Example: ./micro_grid_fixed_inbuild_sum.ipynb

The „lp-file“: Constraints

```
c_u_emission_limit_:
+0.59999999999999998 flow(genset_electricity_bus_0)
+0.59999999999999998 flow(genset_electricity_bus_1)
+0.59999999999999998 flow(genset_electricity_bus_2)
+0.59999999999999998 flow(genset_electricity_bus_3)
+0.59999999999999998 flow(genset_electricity_bus_4)
<= 1.5
```

Agenda

- Introduction to linear optimization
- Linear equation systems of oemof models
- In-build oemof constraints
- Introducing own constraints
- Further examples for constraints

Guidelines for own constraints

- Rules for own constraints:
  - Linearized behaviour
  - No no „if-then-relation“ with other decision variables
- If-then relations can be implemented when accessing a definite timeseries of `actual_value`

Process of writing constraints

1. Simplify real-world boundary to valid constraint

2. Determine structure of constraint:
   - Does the constraint have to be applied each time step individually?
   - Does the constraint concern Investment objects?

3. Create a constraint with a constraint rule, add directly to the linear model of the energy system using Pyomo

4. Verify your constraint by checking…
   - …the lp-file (for few timesteps)
   - …the results (fow a higher number of timesteps)

Renewable share constraint

- Type: Summed minimum
- Based on: Minimum renewable share limit (constant)

$$\sum P_{PV} + \sum P_{Wind} - r_{lim} \cdot \sum P_{demand} \geq 0$$

- Custom constraint with summed limit: ./micro_grid_custom_constraint_summed_limit.ipynb

Micro grid stability constraint

- Type: Minimum bound per timestep
- Based on: Minimum stability limit (constant)

$$P_{DG}(t) + P_{pcc,cons} + (SOC(t) - SOC_{min}) \cdot CAP_{storage,kWh} \cdot Crate \cdot \eta_{discharge} \cdot \eta_{inv} >= L_s \cdot (P_D(t) - P_{short}(t)) \quad \forall t$$

$$P_{DG}(t) + P_{pcc,cons} + CAP_{storage,kW} \cdot \eta_{inv} >= L_s \cdot (P_D(t) - P_{short}(t)) \quad \forall t$$

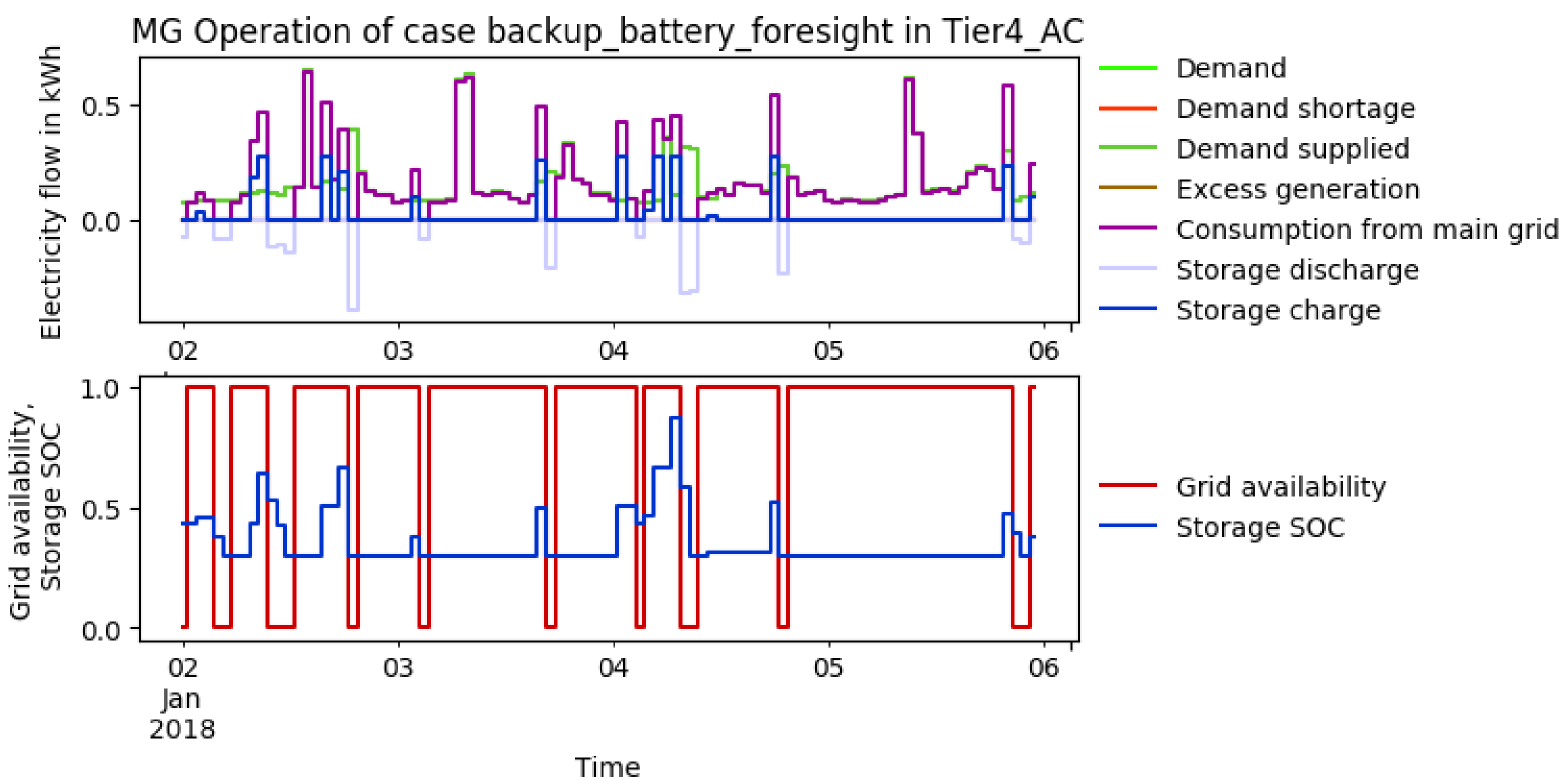
- Custom constraint with bounded flows: ./micro_grid_custom_constraint_flows.ipynb

Agenda

- Introduction to linear optimization
- Linear equation systems of oemof models
- In-build oemof constraints
- Introducing own constraints
- Further examples for constraints

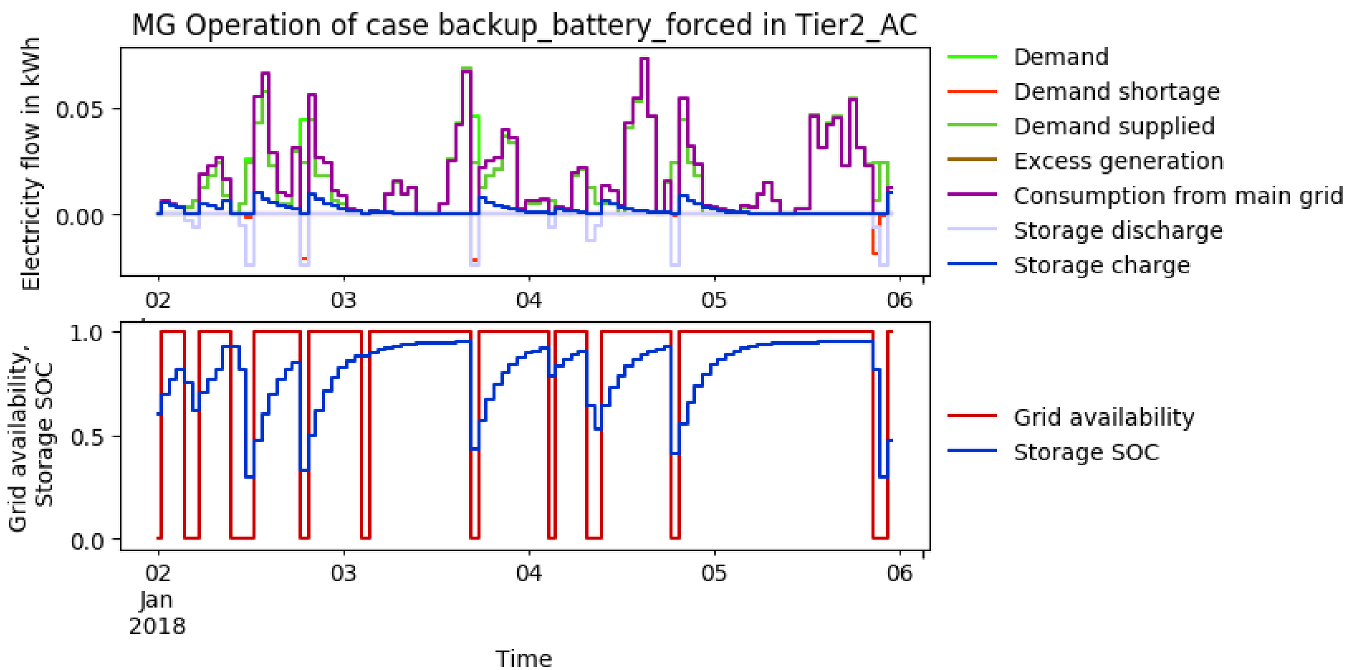
Intermittantly switching off a component

- Type: Setting flow value in timesteps
- Based on: External boolean timeseries



Forced battery charge

- Type: Setting flow per timestep
- Based on:
  - External boolean timeseries
  - Linerarized formular for value of flow



THANK YOU FOR YOUR ATTENTION !

REINER LEMOINE INSTITUT

How to follow Oemof's activities?

Website: https://oemof.org/
Github: https://github.com/oemof
Or join our mailing list!

Tel: +49 (0)30 1208 434 88
E-Mail: martha.hoffmann@rl-institut.de
Web: http://www.rl-institut.de
Twitter: @rl_institut