

Oemof
Workshop Week

Constraints

Martha Hoffmann

Session 4

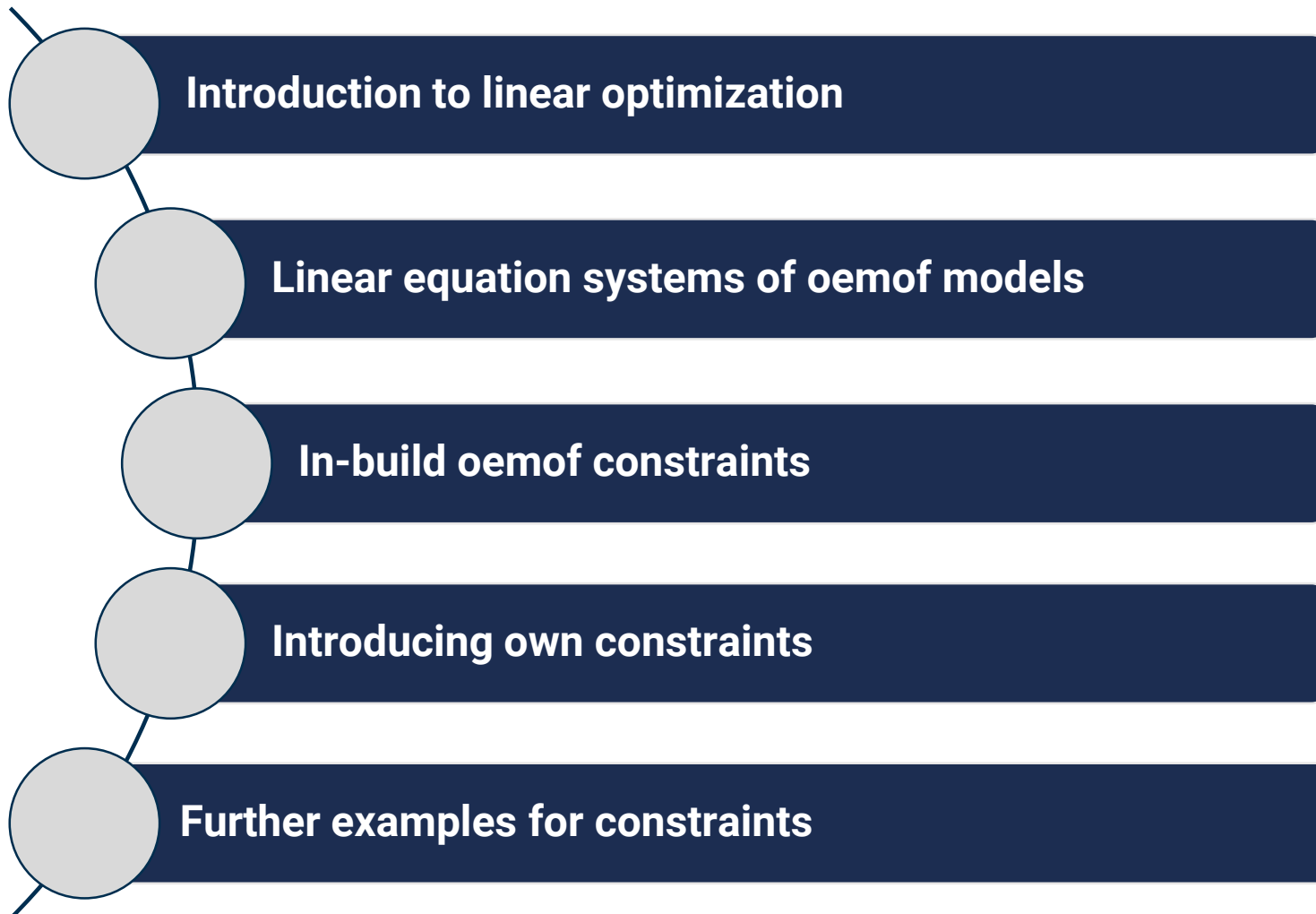
RLI, 18.09.2019



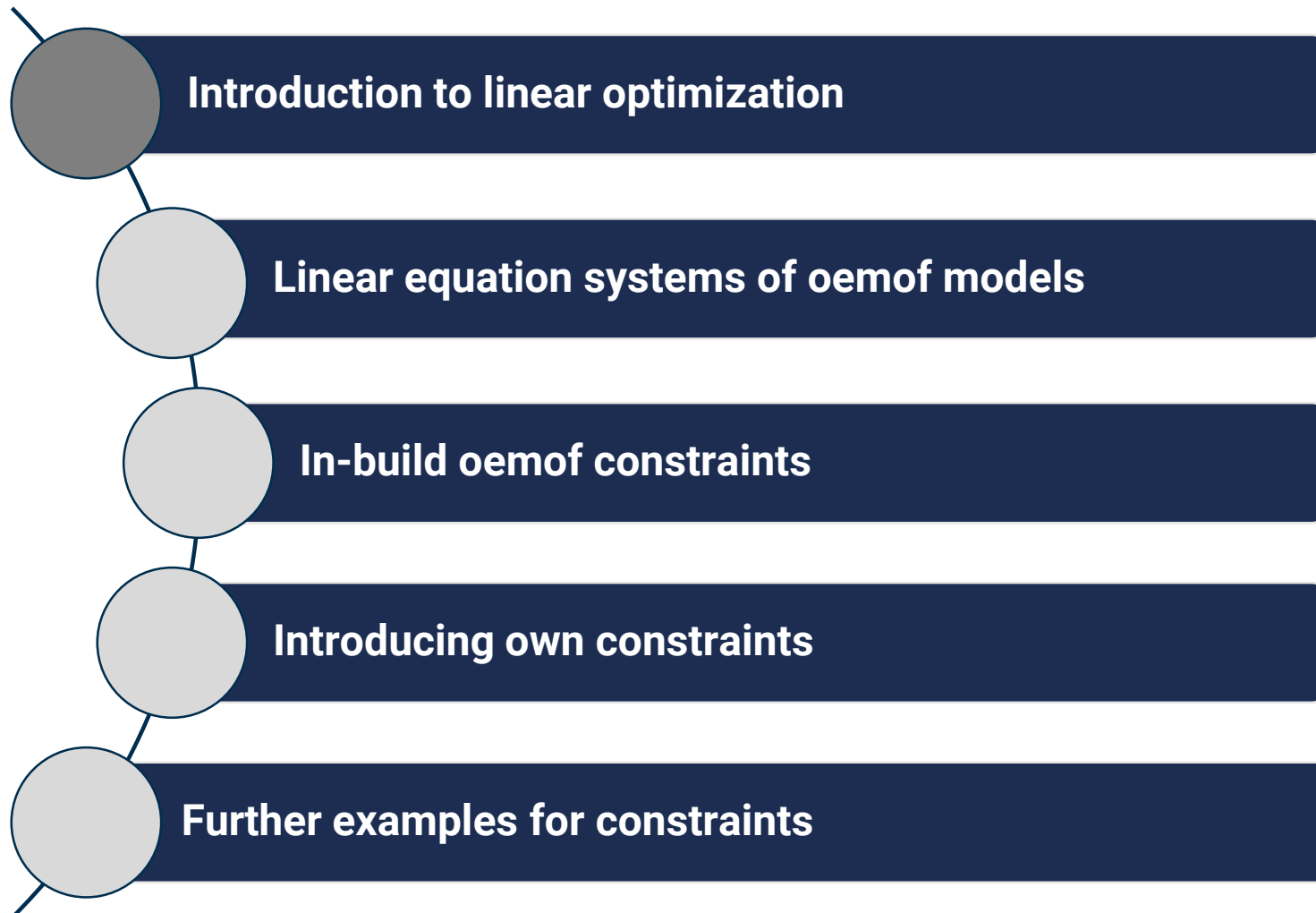
o e m o f
['ø:mɔi]

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](#)

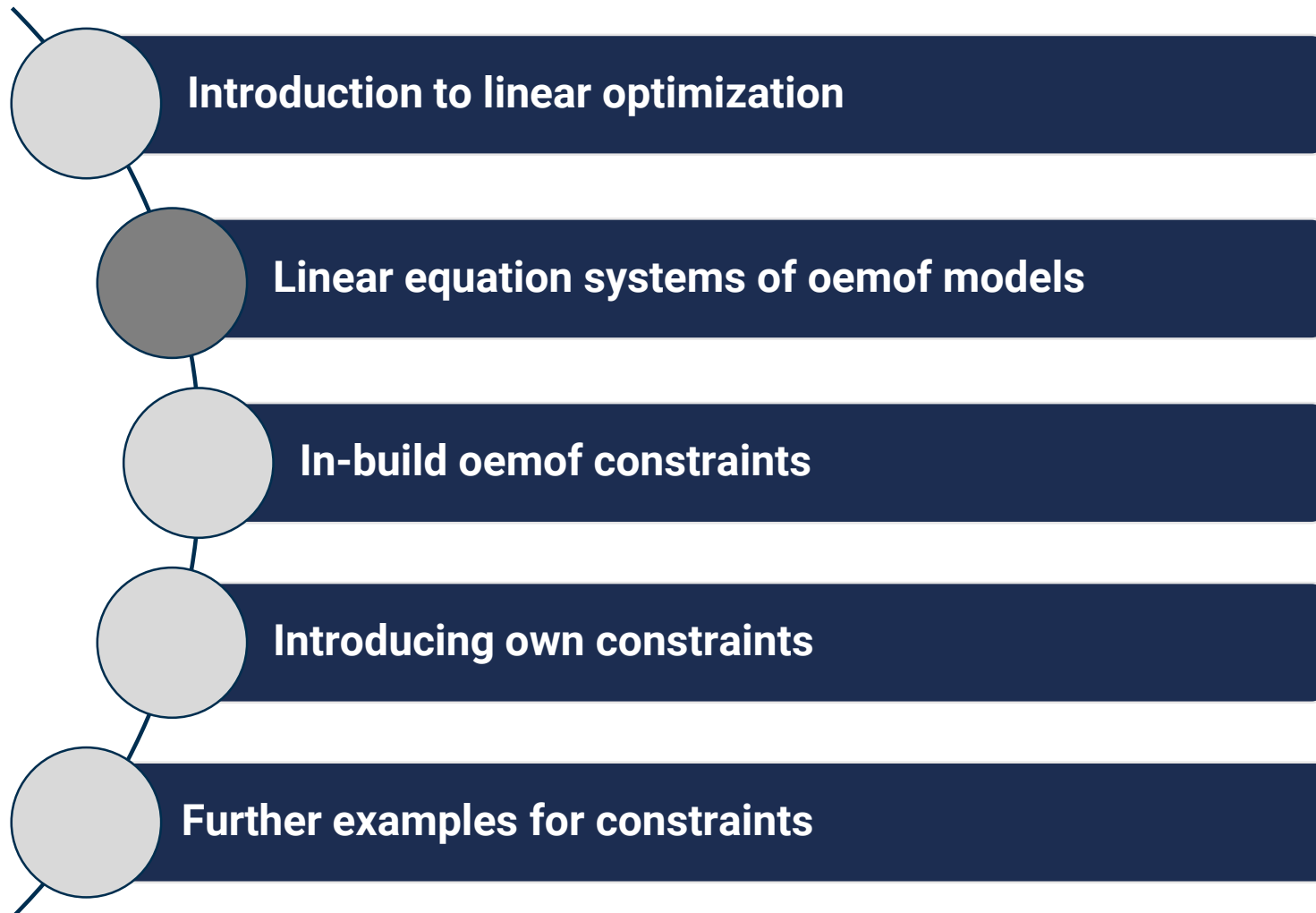


Agenda



- ▶ 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



We learnt:

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

All workshop contents at: https://github.com/smartie2076/oemof_workshop
Todays jupyter notebooks are stored in [./Day_3_Custom_Constraints_for_Oemof](#)

The „lp-file“

- ▶ Oemof generates a set of linear equations based on the created model, using Pyomo
- ▶ Linear equation system can be stored in „lp-file“
- ▶ lp-file 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
 - Includes all necessary data of your energy system
- ▶ Example: `./micro_grid_fixed_cap_basic.ipynb`

The „lp-file“: Main balance

- ▶ The objective value should be minimized:

Image for dispatch problem

Optimizing with oemof – Objective value

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

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

$i \in \{WEA, PV, BHKW, Speicher\}$

$t \in \{1...8760\}$

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“: Objective value

- ▶ The objective value should be minimized:

Image for investment problem

The „lp-file“: Bus balances

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

Image for bus balance easy

The „lp-file“: Bus balances

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

Image for bus balance easy

The „lp-file“: Bus balances

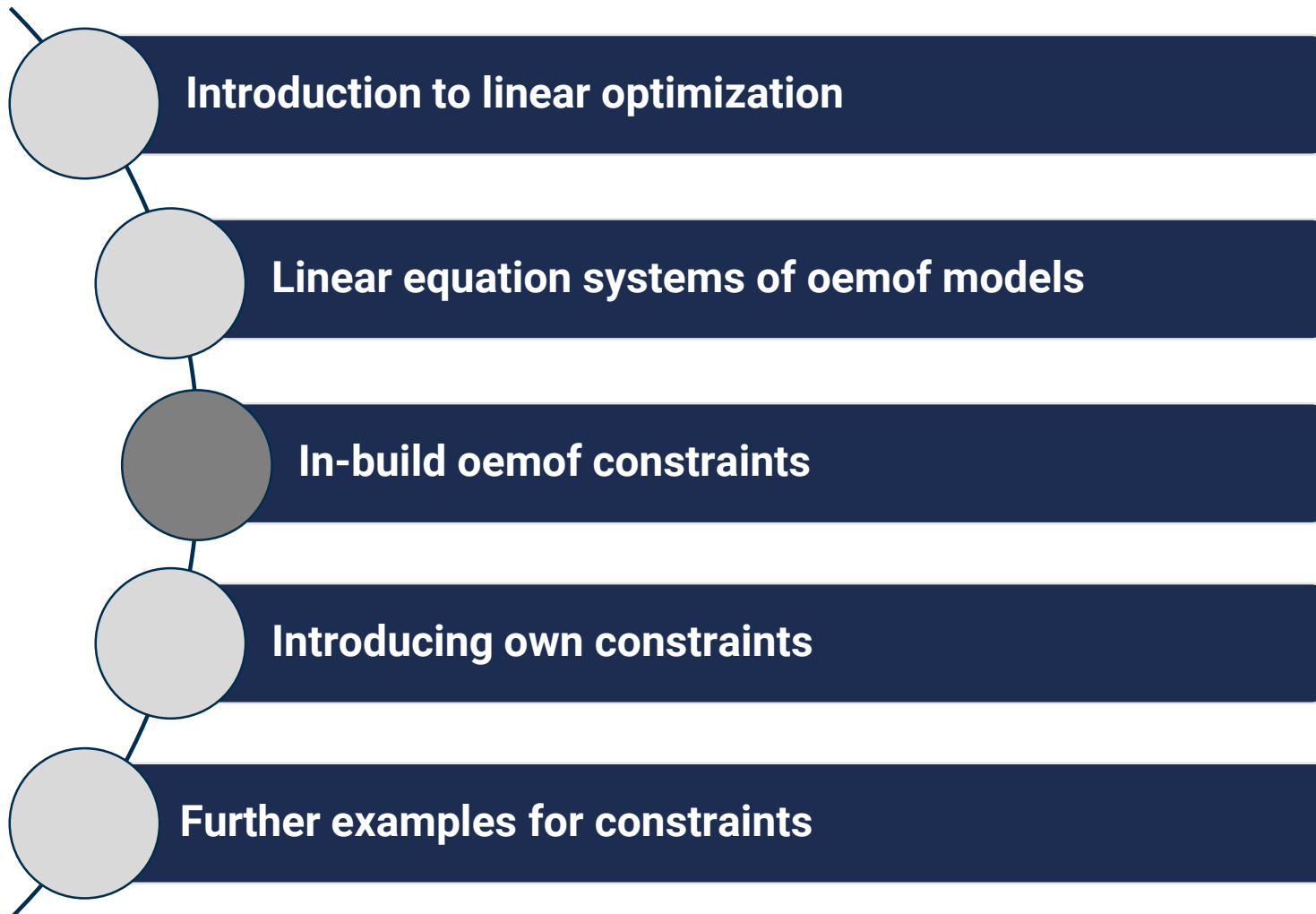
- ▶ Each bus is by default balanced:
 - ▶ $\sum inputs = \sum outputs \quad \forall t$
 - ▶ i.e. no energy can be lost or generated from nowhere
 - ▶ Can require „shortage“-Source or „excess“-Sink

Image for bus balance transformer

The „lp-file“: Bus balances

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

Image for bus balance electricity bus



- ▶ 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`

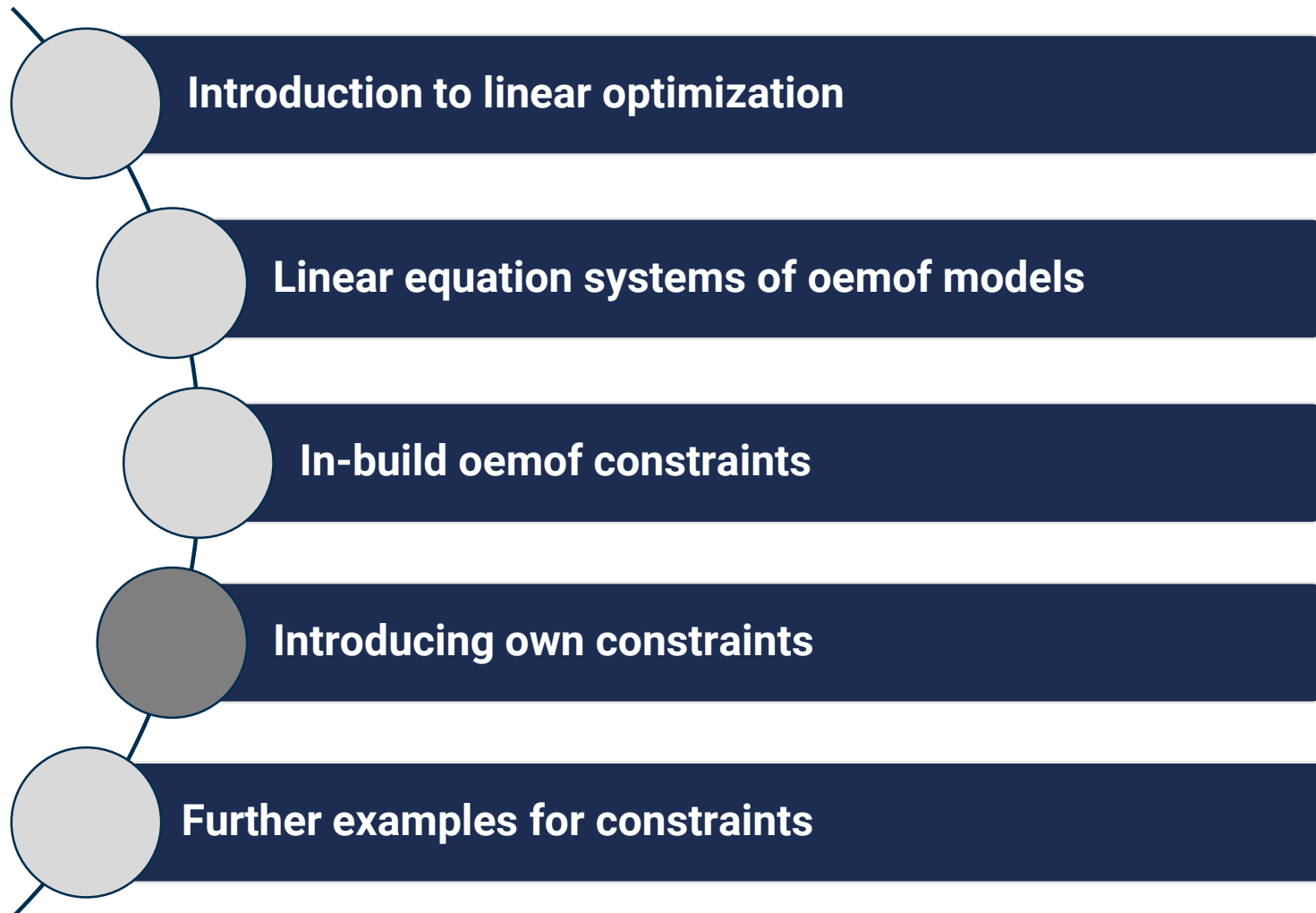
- ▶ Example: `./micro_grid_system_inbuilt_bounds.ipynb`

The „lp-file“: Bounds

- ▶ 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

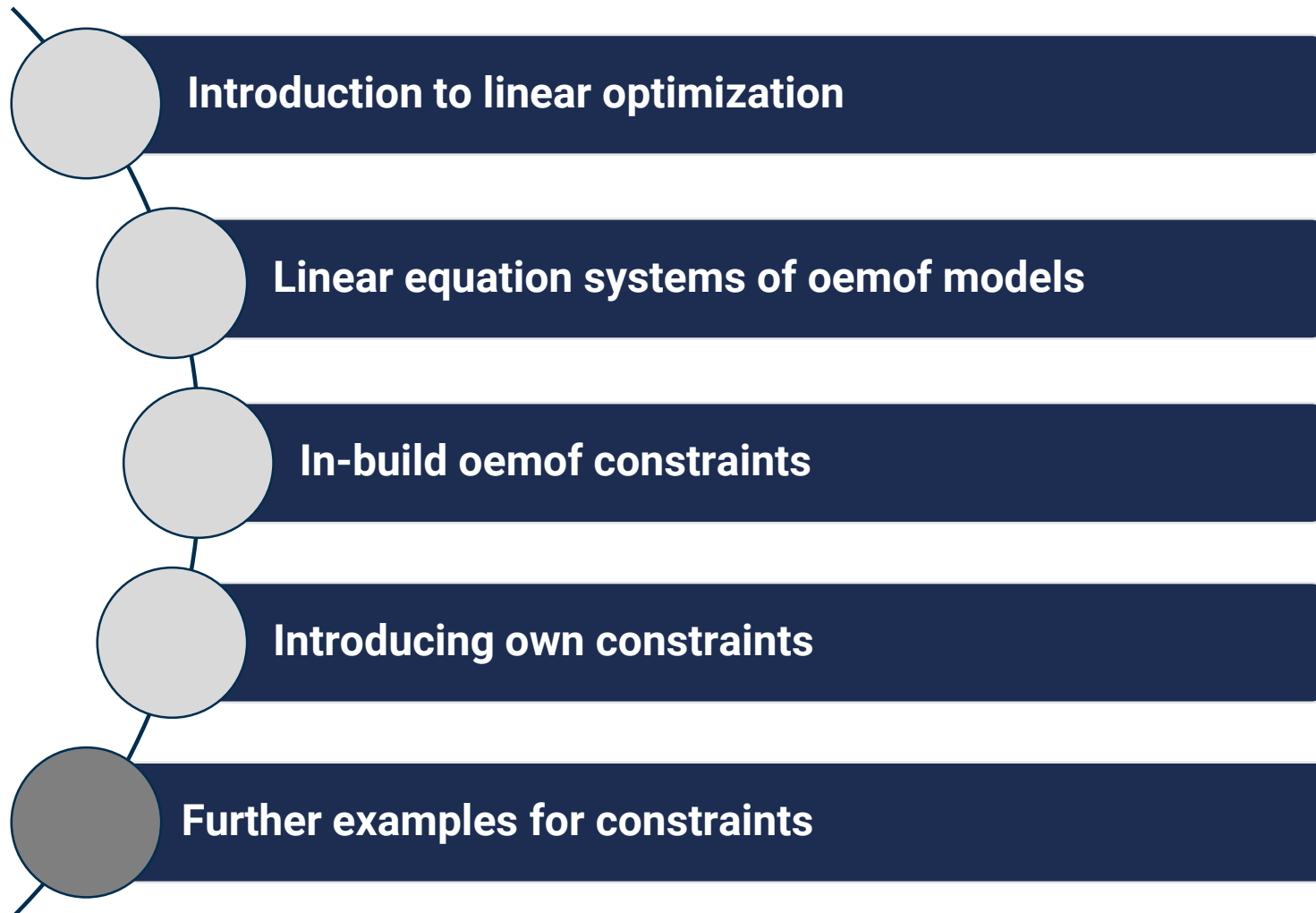
Agenda



- ▶ Rules for own constraints:
 - ▶ Linearized behaviour
 - ▶ Non-dependent on other decision variables, ie. no „if-then-relation“
- ▶ If-then relations can be implemented when accessing a definite timeseries of `actual_value`
- ▶ Calling...
 - ▶ A Flow of an component with nominal value:
 - ▶ A Flow of a component with an investment object
 - ▶ A fix parameter of a component

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 (for a higher number of timesteps)

Agenda



Renewable share constraint

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

Micro grid stability constraint

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

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
 - ▶ Linearized formular for value of flow

THANK YOU FOR YOUR ATTENTION !

How to follow Oemof's activities?

Website: <https://oemof.org/>

Github: <https://github.com/oemof>

Or join our mailing list!



License

Except where otherwise noted, this work and its content (texts and illustrations) are licensed under the Attribution 4.0 International (CC BY 4.0)

See license text for further information.



Tel: +49 (0)30 1208 434 88

E-Mail: martha.hoffmann@rl-institut.de

Web: <http://www.rl-institut.de>

Twitter: [@rl_institut](https://twitter.com/rl_institut)

Please quote as: "PRESENTATION TITLE" © [Reiner Lemoine Institut](#) | [CC BY 4.0](#)