technische universität
dortmund

Institute for Energy systems, Energy efficiency und Energy economics

Determination of Optimal Flexibility Potential for an Electrical Distribution Network

Agenda

- Motivation
- Challenges due to Energy Transformation
- Introduction to i-Automate
- Optimal Power flow implementation
- Case studies and results
- Conclusion and Outlook

Motivation

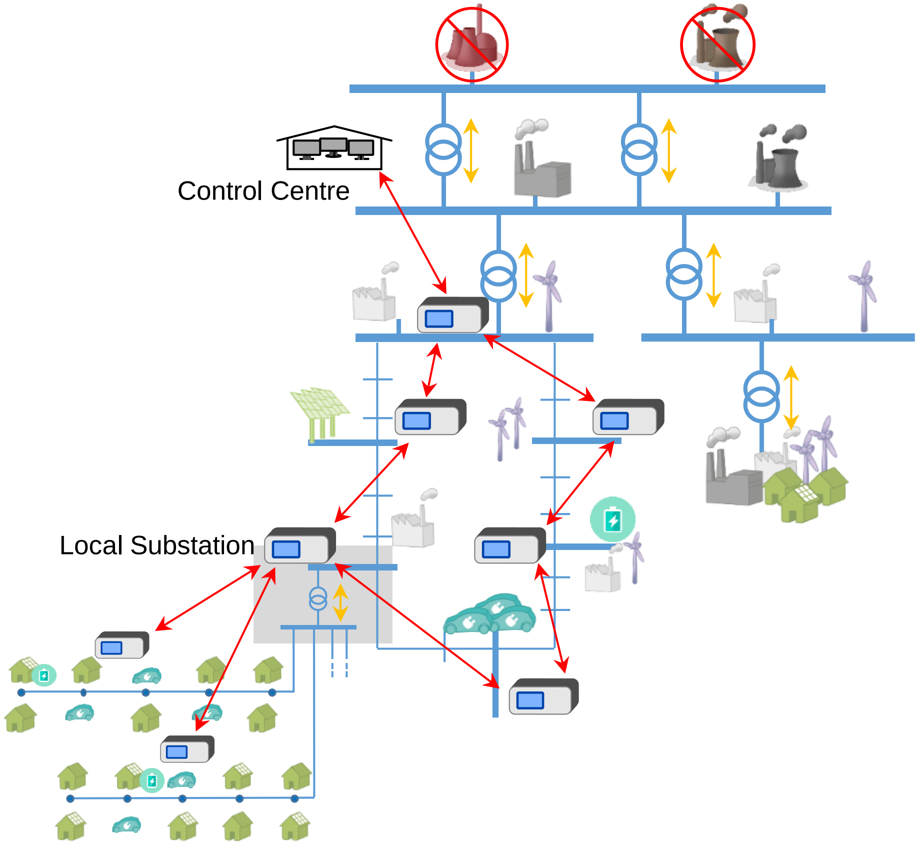
What and how the flexibilities in the MV grid could be used most efficiently to avoid violations while maximizing the integration of DEG infeed

Method

An optimization algorithm to calculate the economical set points for remotely controllable actuators such that all the constraint violations are within limits and the power output from DEGs is at maximum

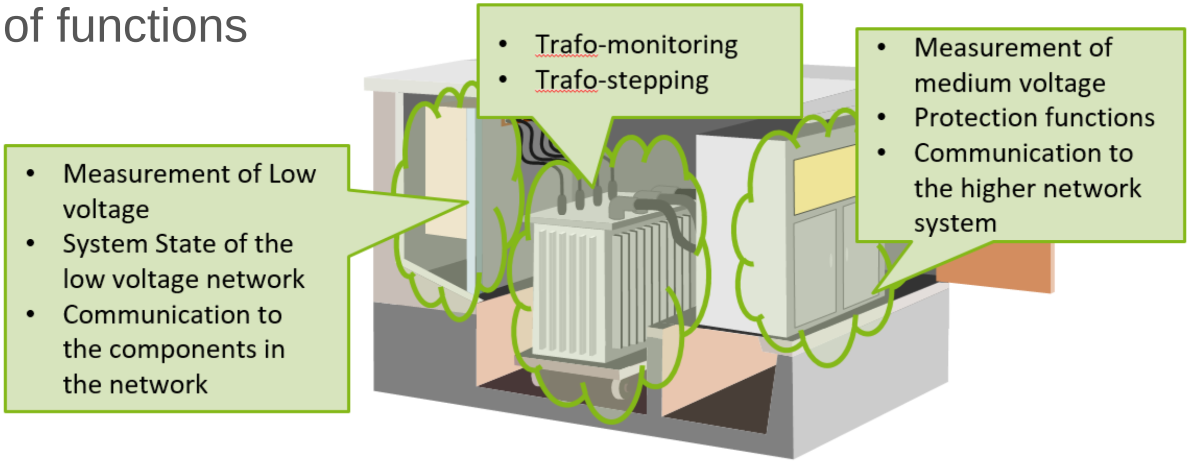
Challenges due to the Energy Transformation

- Conversion of the electrical energy system
  - Integration of new energy resources, loads and batteries
  - Elimination of conventional large-scale power plants
  - Bidirectional power flows
- Integration of measurement, automation and control possibilities
- From passive Distribution grid towards a Smart Grid



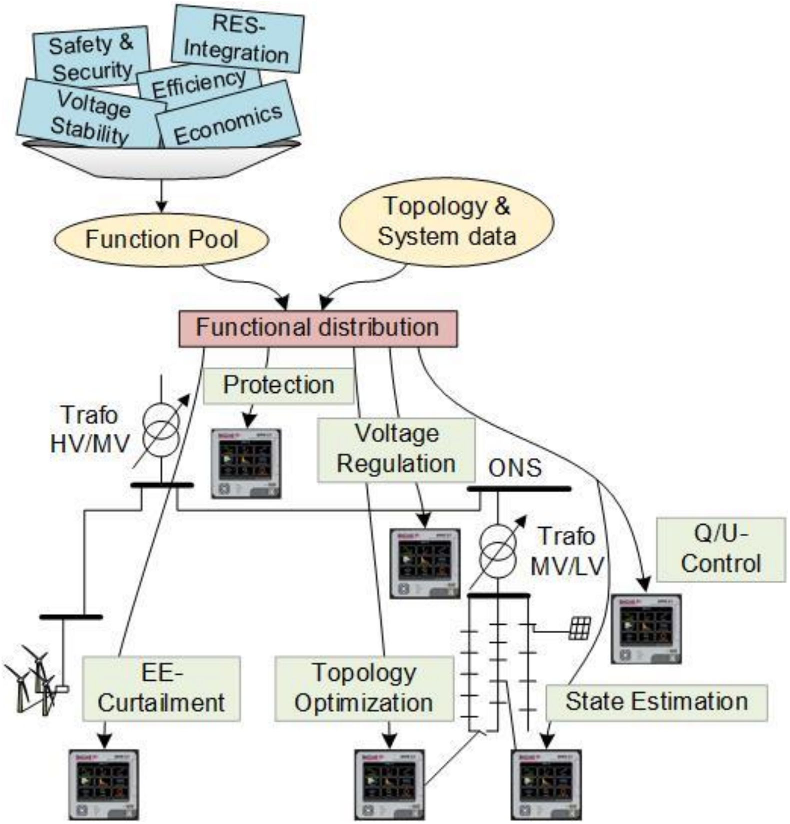
State of the art technology

- No measurement or automation in the secondary substations until now
- Pilot projects: iONS, iNES, GridEye…
- But:
  - No Modularity as in i-Automate
  - No high functional protection functions
- Innovation: Accumulation of functions
  - Monitoring
  - Protection
  - Control
  - Automation



Our Solution: i-Automate

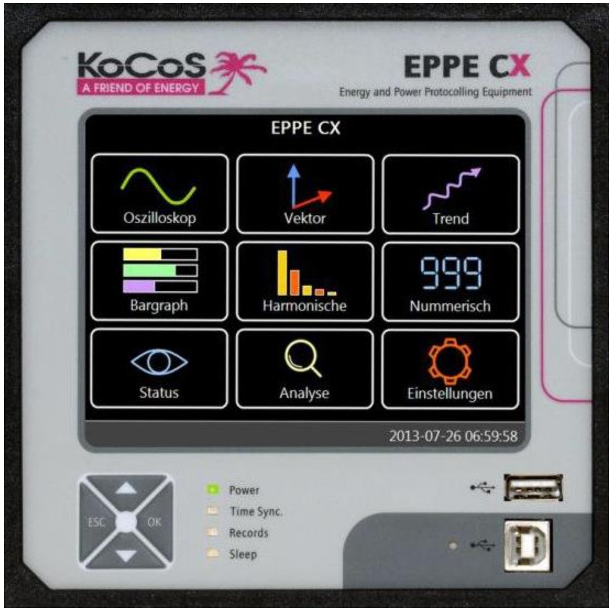
- Modular and flexible system architecture combined
    - Local-autonomous, state-of-the-art protection and control functions
    - higher-level, aggregated and coordinated smart grid automation functions
- Usage of a uniform, proven system platform
- Modular implementation
    1. Observe (Measure)
    2. Manual Intervention (Control)
    3. Automation (Regulate)



The Hardware

- Established system in the field of power quality measurement
    - Integrated data acquisition and processing
    - Data storage, communication and user interface
- Protection functions with hard real-time requirements are outsourced to a controller
    - Distance Protection
    - Differential Protection
    - Overcurrent Protection

☺ Function conflicts are avoided by the separation of functions using a co-processor

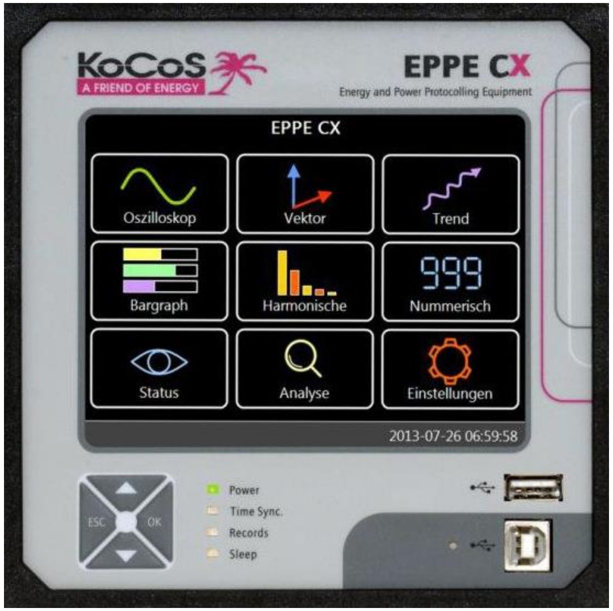


Control functions

- Automation and control functions are implemented modularly on the central processing unit
  - State Estimation
  - Voltage Regulation

- Problems due to voltage regulation
  - Non-optimal usage of available resources
  - Higher costs due to non-optimal use

- Because of the available system state information, optimization algorithm is the next way to go on



Optimal Power Flow

- Objective function
  - Minimize the cost of generation $$Min\ F_g = \sum_{i=1}^{n} (a_i {P_{gi}}^2 + b_i P_{gi} + c_i + d_i {Q_{gi}}^2 + e_i Q_{gi} + f_i)$$
- Equality constraints
  - Power balance at each node - power flow equations

$$P_i^G - P_i^L = \sum_{j=1}^{N} V_i V_j \ [G_{ij} \cos(\theta_i - \theta_j) + B_{ij} \sin(\theta_i - \theta_j) \qquad i = 1\ to\ N$$

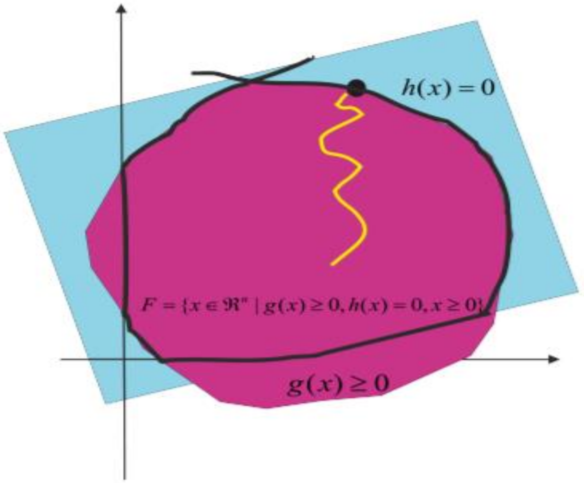
$$Q_i^G - Q_i^L = \sum_{j=1}^{N} V_i V_j \ [G_{ij} \sin(\theta_i - \theta_j) + B_{ij} \cos(\theta_i - \theta_j) \qquad i = 1\ to\ N$$

- Inequality constraints
  - Network operating limits (line loading, voltages, transformer loading)
  - Limits on control variables (Active and Reactive power limits for DEGs and external grid)

OPF using Quadratic interior point method

- Best known technique when a full alternating current (AC) solution is needed
- Handle inequality constraints using barrier functions
- Start from a point in the “interior” of the solution space
- Efficient solution engines are available (e.g MatPower)

$$
\begin{aligned}
&\min_{x} f(x) \\
&s.t. \\
&\qquad g_i(x) \geq 0, i = 1, 2, \ldots, m_1; \\
&\qquad h_j(x) = 0, j = 1, 2, \ldots, m_2; \\
&\qquad x \geq 0,
\end{aligned}
$$



Source: https:// tu-ilmineau.de/Lecture_materials_Abebe/IPM_Slides

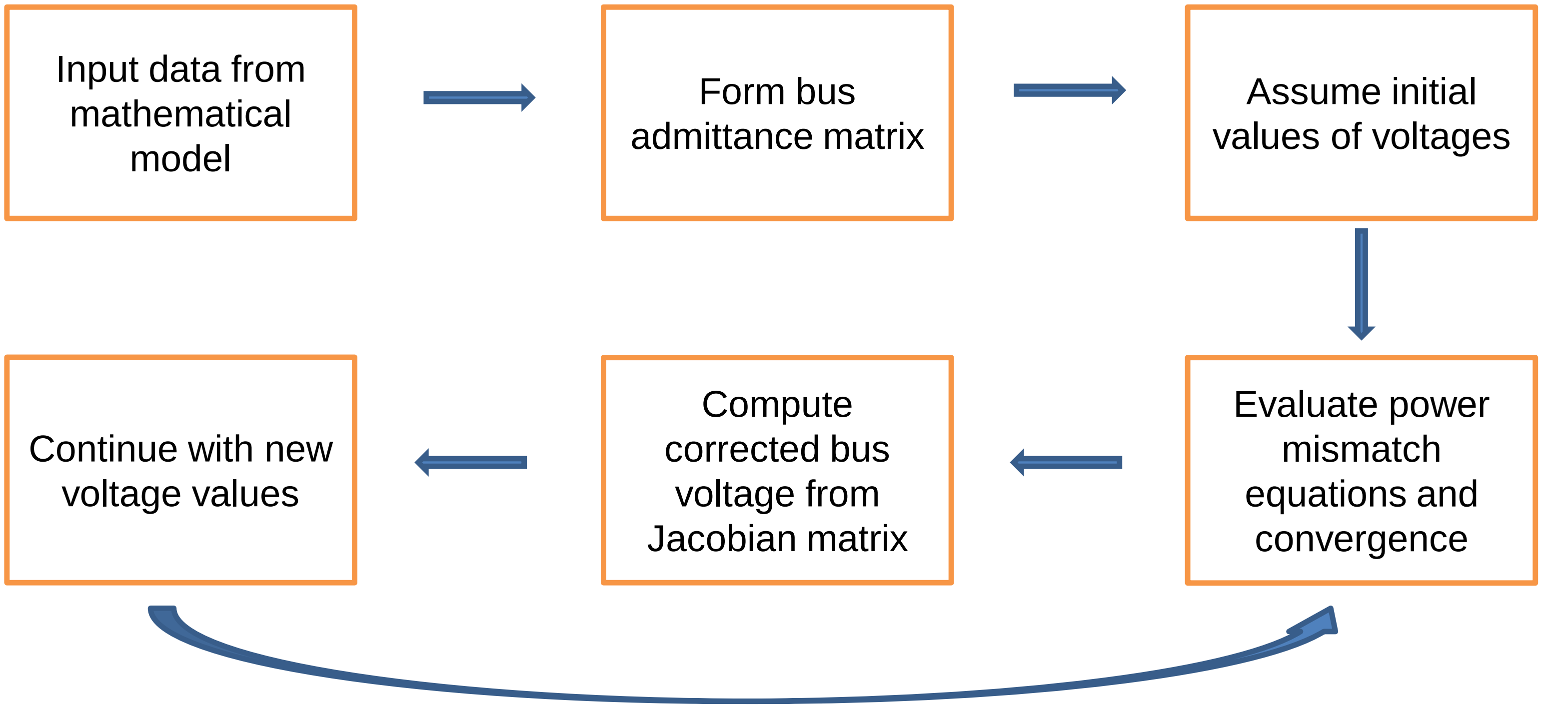
Solution Approach

- Multi step approach that includes
  - mathematical modeling and power flow analysis using Python and Pandapower
  - controller design and optimization
  - Case studies for proof of concept

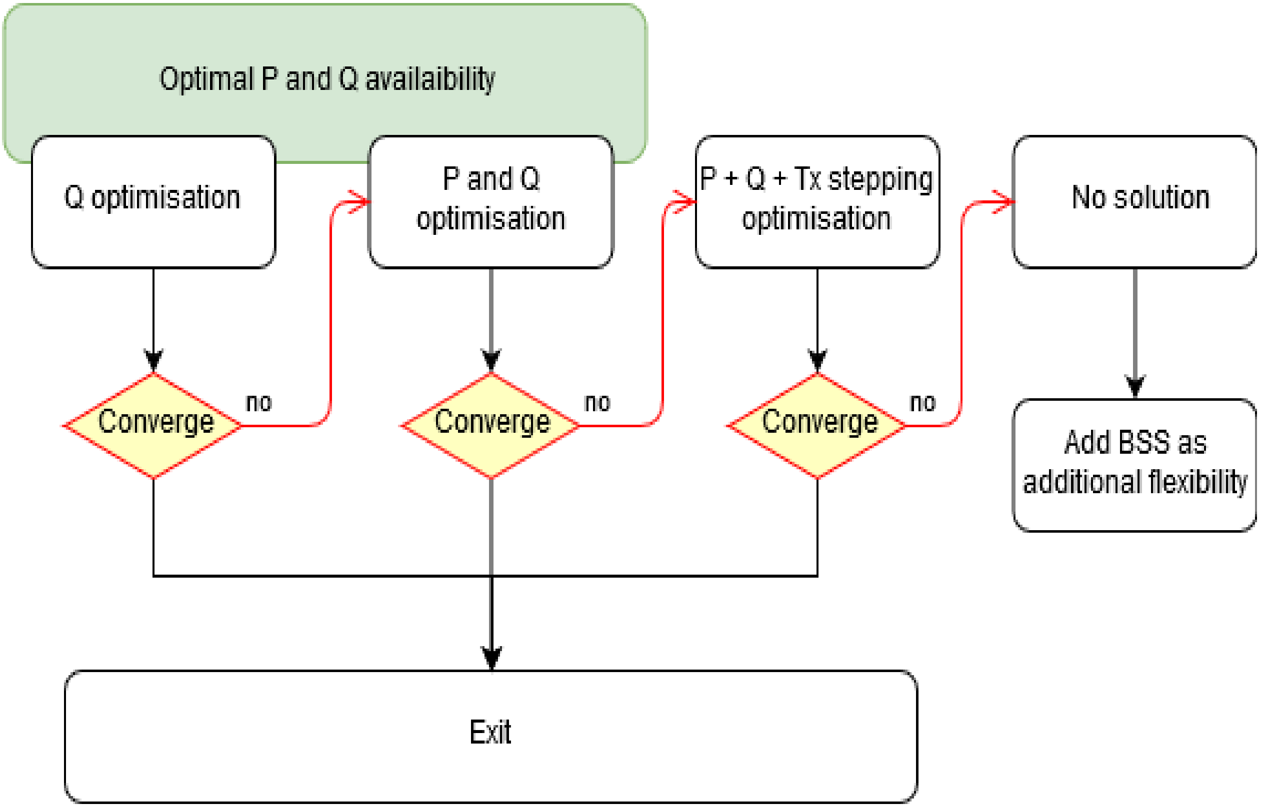
Model the distribution network elements for power flow analysis → Assign costs to individual actuators to be used in optimization → Induce network violations in case study networks and evaluate the performance of optimization algorithm

Mathematical modeling and load flow

- Network elements (lines, buses, generators, transformer etc.) modelling using Python programming language
- Power flow analysis in Python using Newton Raphson Method

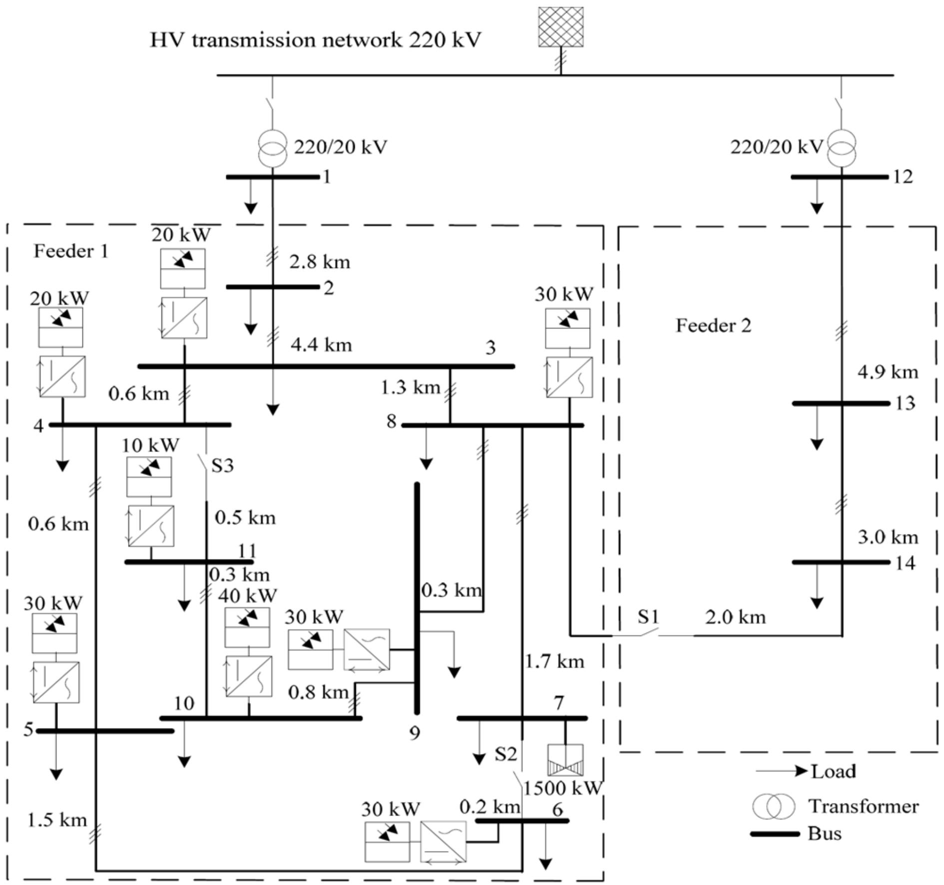


Control Design and Optimization



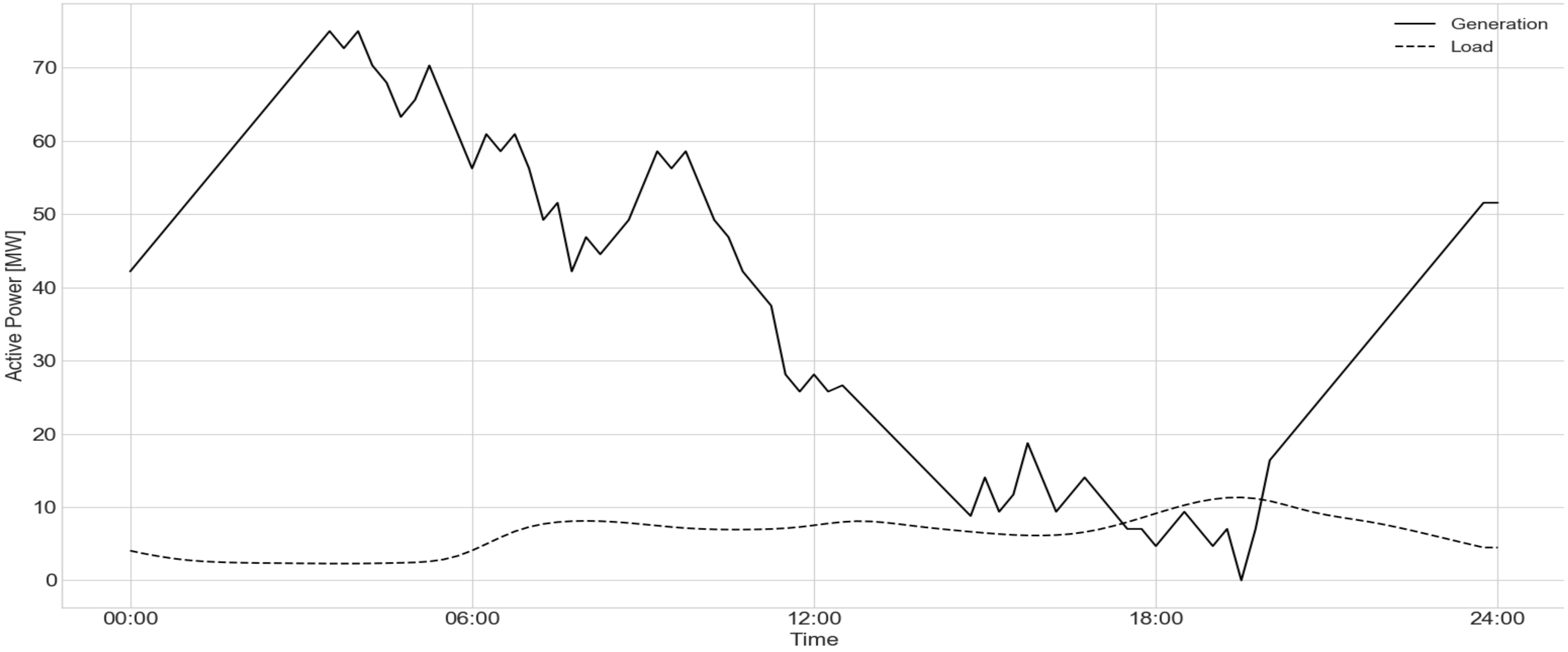
Case Study

CIGRE MV Benchmark Network



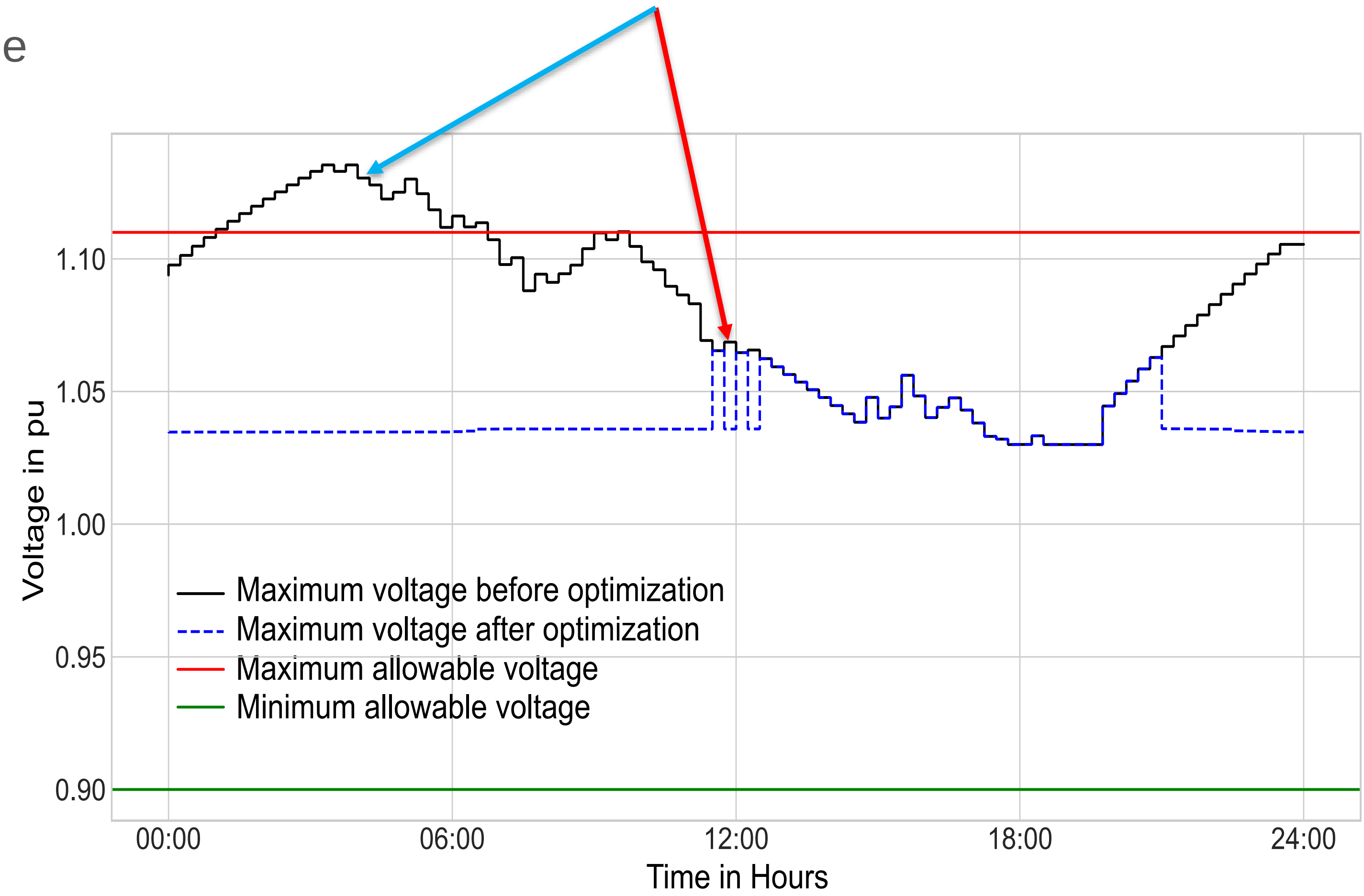
Case Study

- 24 hour Load Generation profile to induce violations



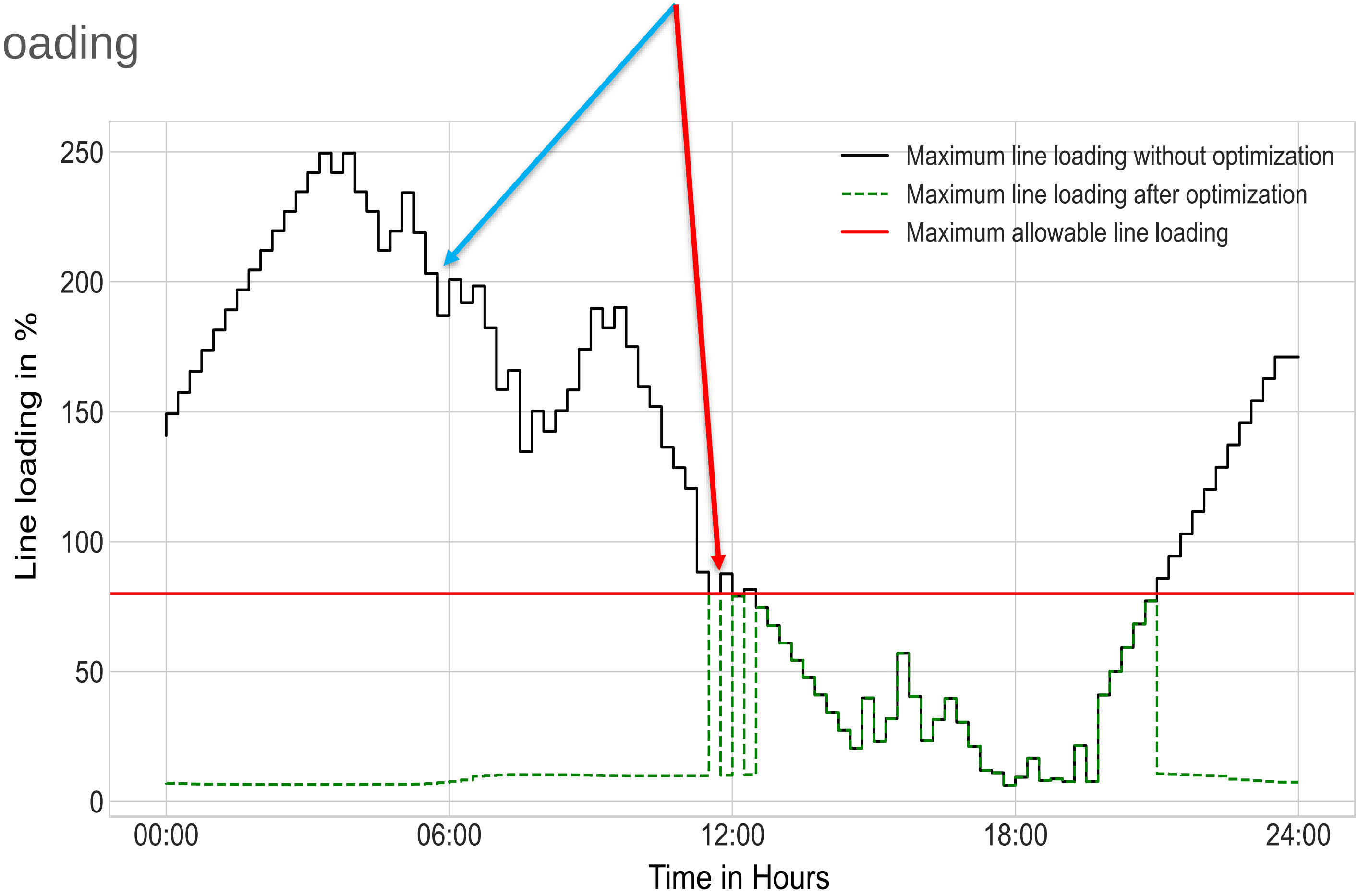
Results

- Voltage



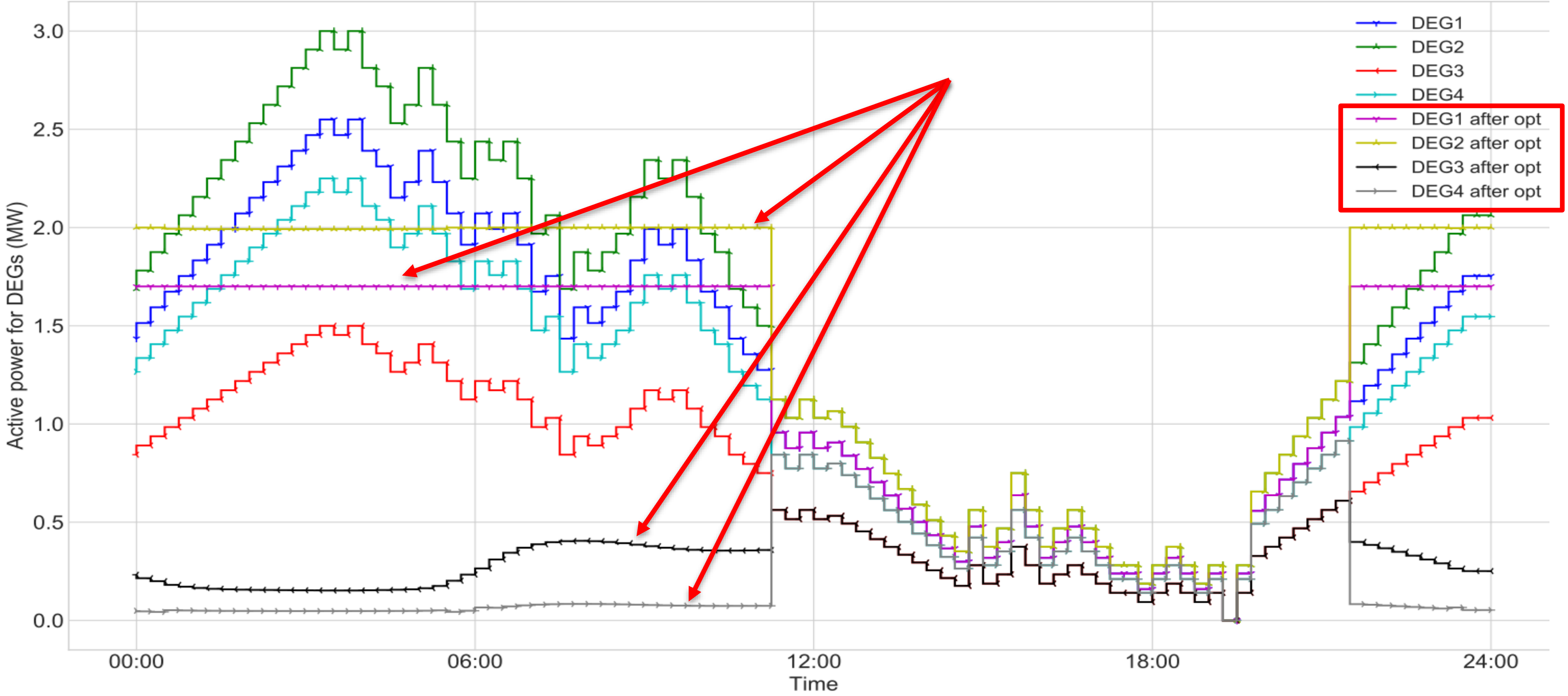
Results

- Line loading



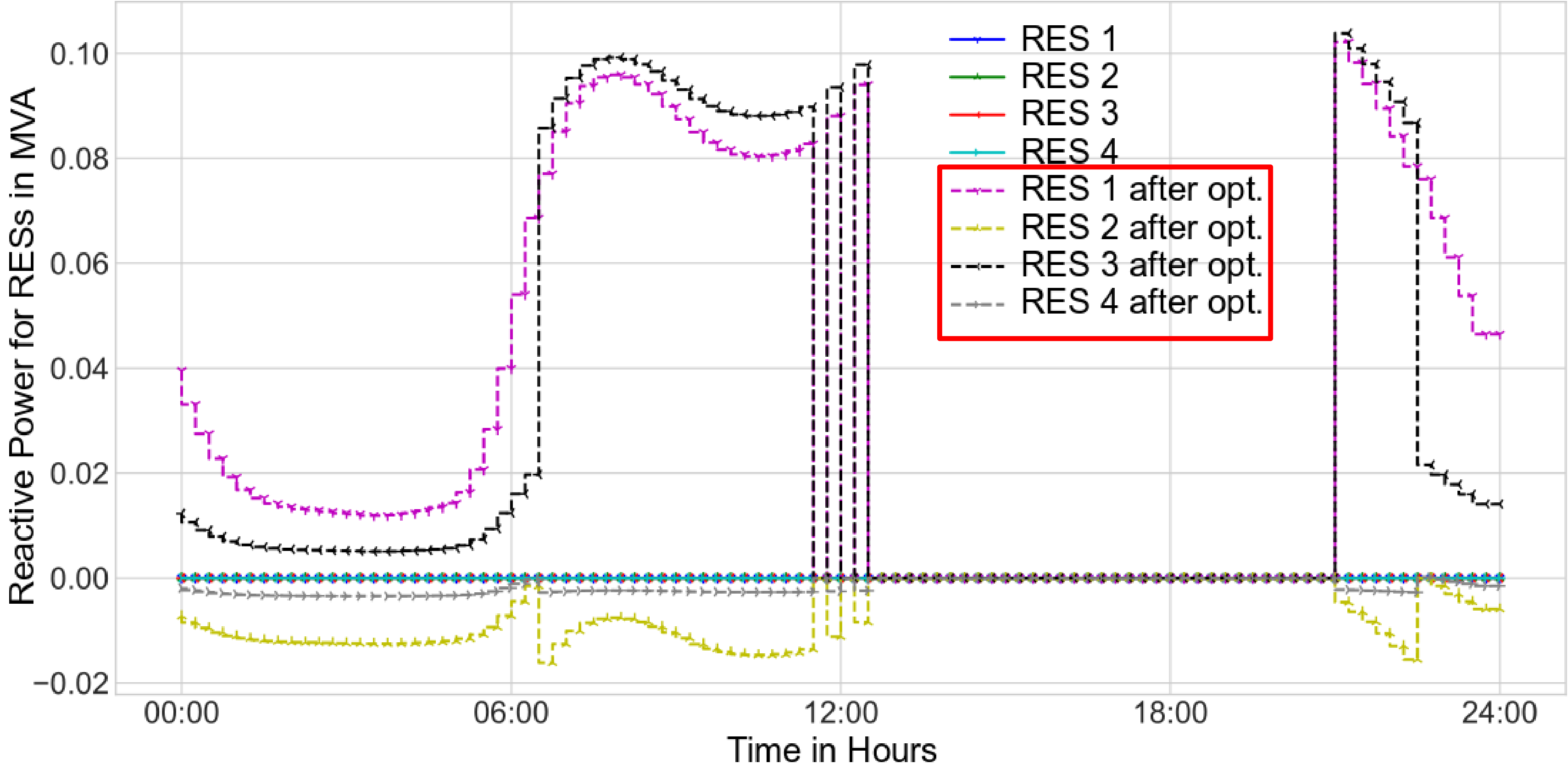
Results

- Active power for DEGs



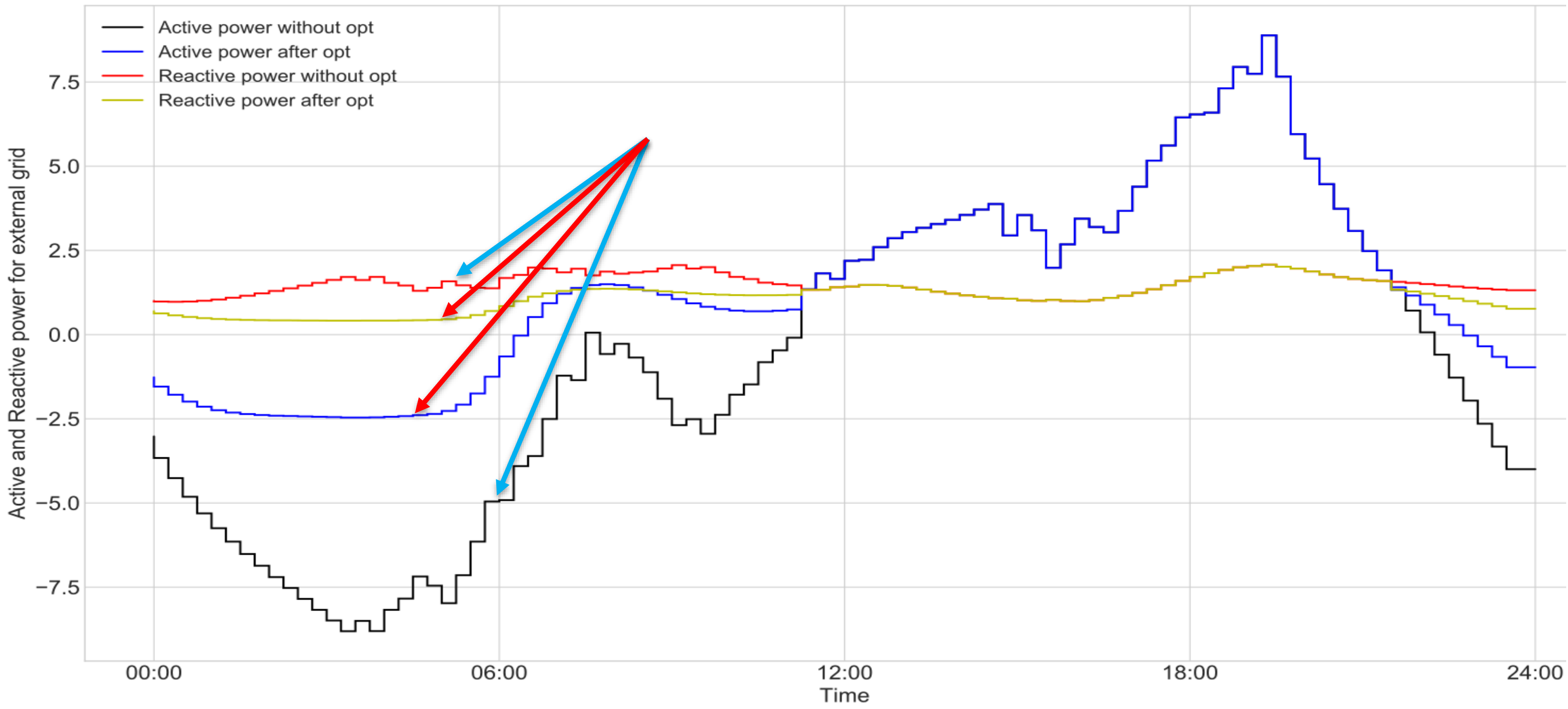
Results

- Reactive power for DEGs



Results

- Active and Reactive power (external grid)



Limitations

- Power flow analysis does not converge due to different reasons
- Size of the optimization problem
    - Bigger and complex networks with hundreds of controls in the real world
    - Number of constraints
- Optimization problem is non linear due to power balance equations and hence adds to the complexity and computing power and time required
- Number of discrete variables further adds to the complexity
    - Position of transformer taps

Conclusion and Outlook

- Power flow analysis is tested using Python with the help of case studies
- Results show that P, Q output of the controllable DEGs, the external grid, the transformer steppings and BESS can be optimally adjusted to avoid network limit violations

- Further case studies and load generation profiles could be used to validate the algortihm
- Load shedding could be part of cost function in case of non-convergence
- Additional continous control variables could be included in test cases e.g BESS
- The algorithm should be tested with bigger, complex networks using real time simulation

- Bring the OPF algorithm running on the hardware device, to be placed in the field!!!

Thank you for your attention ☺

ie3.tu-dortmund.de