Technical University Berlin

Masterthesis

Development of a framework to model a decentralised, multi-period optimal power flow based on the Alternating direction Method of Multipliers

Eric Rockstädt

supervised by
Dr. Richard Weinhold

Berlin, January 30, 2022

Abstract

Executive Summary

Contents

List of Figures

List of Tables

List of Symbols

In the following, several symbols are described that will be later used within the document.

Parameters

$\overline{E_s}$	Maximum energy level of storage s	1 kW h
$\overline{L_l}$	Maximum line capacity of transmission line l	1 kW
$\overline{P_g}$	Maximum power of generator g	1 kW
$\overline{P_s^c}$	Maximum charging power of storage s	1 kW
$\overline{P_s^d}$	Maximum withdrawal power of storage s	1 kW

Sets

$\mathcal{G}$	Set of generators
$\mathcal{L}$	Set of transmission lines
$\mathcal{N}$	Set of nodes
$\mathcal{S}$	Set of storages
$\mathcal{T}$	Set of hourly timesteps

Variables

D_n	Demand at node n	1 kW
E_s	Storage level of storage s	1 kW
P_g	Power generation of generator g	1 kW
P_s^c	Charging power of storage s	1 kW
P_s^d	Withdrawal power of storage s	1 kW

Abbreviations

ADMM	Alternating Directon Method of Multipliers
ALR	Augmented Lagrangian Relaxation

1 Introduction

1.1 Motivation

1.2 Research objectives

2 Theoretical Approach

2.1 ADMM

Alternating Directon Method of Multipliers (ADMM) is very well explained in Boyd (2010).

2.2 Decentralized energy systems

2.3 Dispatch

2.4 Optimal power flow

3 Methodology

3.1 Problem formulations

3.1.1 General problem

The following optimization problem describes a basic economic dispatch with network restrictions.

$$
\begin{aligned}
\min \quad & \sum_{t\in\mathcal{T}}\sum_{g\in\mathcal{G}} c_g * P_g(t) + \sum_{s\in\mathcal{S}} c_s * (P_s^d(t) + P_s^c(t)) && && \text{(1a)}\\
\text{s.t.} \quad & 0 \leq P_g(t) \leq \overline{P_g} && \forall g \in \mathcal{G}, t \in \mathcal{T} && \text{(1b)}\\
& 0 \leq P_s^d(t) \leq \overline{P_s} && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(1c)}\\
& 0 \leq P_s^c(t) \leq \overline{P_s} && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(1d)}\\
& 0 \leq E_s(t) \leq \overline{E_s} && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(1e)}\\
& E_s(t) - E_s(t-1) - P_s^c(t) + P_s^d(t) = 0 && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(1f)}\\
& I_n(t) = \sum_{g\in\mathcal{G}} P_g(t) + \sum_{s\in\mathcal{S}}(P_s^d(t) - P_s^c(t)) - D_n(t) = 0 && \forall t \in \mathcal{T}, n \in \mathcal{N} && \text{(1g)}\\
& -\overline{L_l} \leq PTDF * I_n(t) \leq \overline{L_l} && \forall l \in \mathcal{L}, t \in \mathcal{T}, n \in \mathcal{N} && \text{(1h)}
\end{aligned}
$$

3.1.2 Replace inequality line constraint

Since the ADMM can not cope with inequality constraints, equation (1h) is replaced by two equality constraints by introducing two slack variables R_{ref} and R_{cref}. The problem formulation evolves to the following:

$$
\begin{aligned}
\min \quad & \sum_{t\in\mathcal{T}}\sum_{g\in\mathcal{G}} c_g * P_g(t) + \sum_{s\in\mathcal{S}} c_s * (P_s^d(t) + P_s^c(t)) && && \text{(2a)}\\
\text{s.t.} \quad & 0 \leq P_g(t) \leq \overline{P_g} && \forall g \in \mathcal{G}, t \in \mathcal{T} && \text{(2b)}\\
& 0 \leq P_s^d(t) \leq \overline{P_s} && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(2c)}\\
& 0 \leq P_s^c(t) \leq \overline{P_s} && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(2d)}\\
& 0 \leq E_s(t) \leq \overline{E_s} && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(2e)}\\
& E_s(t) - E_s(t-1) - P_s^c(t) + P_s^d(t) = 0 && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(2f)}\\
& I_n(t) = \sum_{g\in\mathcal{G}} P_g(t) + \sum_{s\in\mathcal{S}}(P_s^d(t) - P_s^c(t)) - D_n(t) = 0 && \forall t \in \mathcal{T}, n \in \mathcal{N} && \text{(2g)}\\
& PTDF * I_n(t) + R_{ref} - \overline{L_l} = 0 && \forall l \in \mathcal{L}, t \in \mathcal{T}, n \in \mathcal{N} && \text{(2h)}\\
& R_{cref} - PTDF * I_n(t) - \overline{L_l} = 0 && \forall l \in \mathcal{L}, t \in \mathcal{T}, n \in \mathcal{N} && \text{(2i)}
\end{aligned}
$$

The complicating constraints are equations (2g), (2h) and (2i). If these constraints are relaxed, the main problem decomposes into a generator and a storage subproblem.

3.1.3 Augmented Lagrangian Relaxation

The complicated constraints are relaxed by implementing a max-min problem using the dual variables of the complicated constraints. Hereby, λ is the dual of the energy balance constraint, μ and ρ are the duals of the upper and lower flow constraint respectively. Since the objective function is linear, the relaxation is implemented by using the Augmented Lagrangian Relaxation (ALR). Thus, a penalty term per dual variable is added whose value equals zero in the optimality point.

$$
\begin{aligned}
&\max\,(\lambda,\mu,\rho) && \text{(3a)}\\
&\quad \min\,(P_g, P_s^d, P_s^c) \quad \sum_{t\in\mathcal{T}}\sum_{g\in\mathcal{G}} c_g * P_g(t) + \sum_{s\in\mathcal{S}} c_s * (P_s^d(t) + P_s^c(t))\\
&\quad + \lambda * \left[\sum_{g\in\mathcal{G}} P_g(t) + \sum_{s\in\mathcal{S}} (P_s^d(t) - P_s^c(t)) - D_n(t))\right]\\
&\quad + \frac{\gamma}{2} * \left\|\sum_{g\in\mathcal{G}} P_g(t) + \sum_{s\in\mathcal{S}} (P_s^d(t) - P_s^c(t)) - D_n(t))\right\|_2^2\\
&\quad + \mu * \left[PTDF * I_n(t) + R_{ref} - \overline{L_l}\right]\\
&\quad + \frac{\gamma}{2} * \left\|PTDF * I_n(t) + R_{ref} - \overline{L_l}\right\|_2^2\\
&\quad + \rho * \left[R_{cref} - PTDF * I_n(t) - \overline{L_l}\right]\\
&\quad + \frac{\gamma}{2} * \left\|R_{cref} - PTDF * I_n(t) - \overline{L_l}\right\|_2^2
\end{aligned}
$$

$$
\begin{aligned}
\text{s.t.}\quad & 0 \le P_g(t) \le \overline{P_g} && \forall g \in \mathcal{G}, t \in \mathcal{T} && \text{(3b)}\\
& 0 \le P_s^d(t) \le \overline{P_s} && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(3c)}\\
& 0 \le P_s^c(t) \le \overline{P_s} && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(3d)}\\
& 0 \le E_s(t) \le \overline{E_s} && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(3e)}\\
& E_s(t) - E_s(t-1) - P_s^c(t) + P_s^d(t) = 0 && \forall s \in \mathcal{S}, t \in \mathcal{T} && \text{(3f)}
\end{aligned}
$$

3.1.4 Matrix Form

Typically, ADMM solves problems in the form:

$$
\begin{aligned}
\min\,(x,z) \quad & f(x) + g(z) && \text{(4a)}\\
\text{s.t.} \quad & \mathbf{A}x + \mathbf{B}z = c && \text{(4b)}
\end{aligned}
$$

If applied to the formulation in the section 3.1.1, the generator problem looks like:

$$f(x) = f(\mathbf{P_G}) = \mathbf{P_G} * \vec{\mathbf{c_G}} \tag{5a}$$
$$= \begin{bmatrix} P_{g_1}(t_1) & P_{g_2}(t_1) \\ P_{g_1}(t_2) & P_{g_2}(t_2) \end{bmatrix} * \begin{bmatrix} c_{g_1} \\ c_{g_2} \end{bmatrix} \tag{5b}$$
$$= \begin{bmatrix} P_{g_1}(t_1) * c_{g_1} + P_{g_2}(t_1) * c_{g_2} \\ P_{g_1}(t_2) * c_{g_1} + P_{g_2}(t_2) * c_{g_2} \end{bmatrix} \tag{5c}$$

In addition, the storage problem yields:

$$g(z) = g(\mathbf{P_S^d}, \mathbf{P_S^c}) \tag{6a}$$
$$= \left(\mathbf{P_S^d} + \mathbf{P_S^c}\right) * \vec{\mathbf{c_S}} \tag{6b}$$
$$= \left(\begin{bmatrix} P_{s_1}^d(t_1) & P_{s_2}^d(t_1) \\ P_{s_1}^d(t_2) & P_{s_2}^d(t_2) \end{bmatrix} + \begin{bmatrix} P_{s_1}^c(t_1) & P_{s_2}^c(t_1) \\ P_{s_1}^c(t_2) & P_{s_2}^c(t_2) \end{bmatrix} \right) * \begin{bmatrix} c_{s_1} \\ c_{s_2} \end{bmatrix} \tag{6c}$$
$$= \begin{bmatrix} P_{s_1}(t_1) * c_{s_1} + P_{s_2}(t_1) * c_{s_2} \\ P_{s_1}(t_2) * c_{s_1} + P_{s_2}(t_2) * c_{s_2} \end{bmatrix} \tag{6d}$$

Only the energy balance constraint and the constraints for the power flow are part of the ADMM formulation. All the other constraints are either part of the generator problem or of the storage problem and can be easily decomposed.

The energy balance constraint in matrix form yields:

$$a = 1 \tag{7a}$$

3.1.5 Scaled Form

According to Boyd (2010), ADMM is often written in a shorter, scaled form. In the scaled form the linear and quadratic terms of the objective function are combined and the dual variables are scaled. This yields a much shorter formulation. For each dual variable, a residual is defined:

$$r = \sum_{g \in \mathcal{G}} P_g(t) + \sum_{s \in \mathcal{S}} (P_s^d(t) - P_s^c(t)) - D_n(t) \tag{8}$$

$$s = PTDF * I_n(t) + R_{ref} - \overline{L_l} \tag{9}$$

$$t = R_{cref} - PTDF * I(t) - \overline{L_l} \tag{10}$$

4 Results

References

Boyd, S. (2010). "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers". In: *Foundations and Trends® in Machine Learning* 3.1, pp. 1–122.