Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-27 (2016) 307–312

An Approximate Min-Sum Algorithm for Smart Grid Dispatch with Continuous Variables

Anna Elisabeth Kellerer * **Florian Steinke** **

* *Technical University Munich, Munich, Germany (e-mail: elisabeth.kellerer@tum.de)*
** *Siemens Corporate Technology, Munich, Germany (e-mail: florian.steinke@siemens.com)*

Abstract: We present a decentralized algorithm for solving economic dispatch problems in tree-shaped electrical distribution grids. Previous work used an extension of dynamic programming to trees, i.e. the min-sum algorithm, with discrete variables to solve these problems. We now present a novel, specially adapted, and efficient approximation scheme for continuous variables. Working with continuous variables is mandatory for many real world applications where line, consumption and generator capacities often have significantly different scales.

Keywords: Economic dispatch, dynamic programming, distribution grids, smart grid

1. INTRODUCTION

The widespread introduction of renewable energies is currently transforming the electric power sector in many countries. Few large power plants owned by few large operators produced almost all electric power centrally in the past. Today, there are often hosts of small and medium sized generation units throughout all levels of the electric distribution grid, with a great variety of owners. This trend towards an increasing number of actively controllable units at decentral locations and a diverse ownership structure will become even more prevalent in the future, when active demand side elements like e.g. electric vehicles (EV) and power-2-heat options are projected to grow more popular.

In the light of these developments, classic economic dispatch algorithms are now facing new challenges. Market-dominating centralized solutions like mixed-integer linear programming (MILP), e.g. Carrion and Arroyo (2006), are under pressure on the one hand regarding computational scaling. However, equally or more problematic is the necessity for participants to openly share their component models with the central dispatcher, since the models often represent important business secrets. Several decentral dispatch algorithms have therefore been proposed recently, as discussed below. This work is an extension of the work of Kellerer and Steinke (2015).

The economic dispatch problem is about scheduling a number of electricity generation and consumption units such that the overall welfare is maximized, while respecting generation and transportation constraints. The cost function is thus the sum of individual consumers' negative utility functions plus the generators' costs. For an electric grid with buses/nodes V, the problem reads

$$\min_{P_v \in \mathcal{P}_v,\, P_{vw} \in \mathcal{P}_{vw}} \sum_{v \in \mathrm{V}} C_v(P_v), \tag{1}$$

$$s.t.\ P_v = \sum_{w \neq v} P_{vw},\ P_{vw} = -P_{wv}. \tag{2}$$

In this expression, P_v denotes the net power injection at node v and P_{vw} the power transport from v to w. $C_v(P_v)$ is the cost or negative utility function associated with P_v, and $\mathcal{P}_v$, $\mathcal{P}_{vw}$ denote the feasibility sets for P_v and P_{vw}, respectively. This formulation of the economic dispatch problem corresponds to a simple loss-less power flow model with line capacity constraints. It is equivalent to the common DC approximation of the AC power flow equations for tree-structured grids.

Kellerer and Steinke (2015) propose to solve the economic dispatch problem for tree-shaped electrical grids via an extension of dynamic programming, that is known as belief propagation or the min-sum algorithm in the field of graphical models in statistics (Pearl, 1982; Koller and Friedman, 2009). Two messages are exchanged over each edge of the graph, one in each direction. They are computed as

$$m_{vw}(P_{vw}) = \min_{P_{uv} \in \mathcal{P}_{uv}, u \neq w} C_v\left(\sum P_{uv}\right) + \sum_{u \neq w} m_{uv}(P_{uv}), \tag{3}$$

where we have eliminated the nodal injection variables P_v and integrated the nodal domain constraints $\mathcal{P}_v$ into the cost functions (i.e. setting C_v to infinity if $P_v \notin \mathcal{P}_v$). For trees, these messages can be computed iteratively starting from the leaves where no incoming messages are to be considered. The final optimal transport/dispatch assignments are then determined as

$$P^*_{vw} = \underset{P_{vw} \in \mathcal{P}_{vw}}{\operatorname{argmin}}\ m_{vw}(P_{vw}) + m_{wv}(P_{vw}). \tag{4}$$

Kellerer and Steinke (2015) represent messages via the discretization of optimization variables P_{vw} as finite vectors. We will therefore denote this method as discrete message passing (DMP). This approach allows for an exact optimization on trees for arbitrary cost functions C_v. However, if the capacities of generators, consumers, and transmission lines are significantly different, it becomes computationally infeasible to use a common, sufficiently fine discretization. To understand the importance of this problem consider for example the scheduling of

2405-8963 © 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.10.709

a heat pump with one or two kW on a feeder with a capacity of several hundred kW. Capacity-dependent discretization and iterative re-discretization schemes can partially alleviate this problem (Kellerer and Steinke, 2015). However, a truly continuous approach is more suitable.

For continuous optimization variables P_{vw}, the messages (3) are one parameter mappings $m_{mv} : \mathcal{P}_{vw} \subset \mathbb{R} \to \mathbb{R}$. Their functional complexity will typically rise with the number and the complexity of incorporated cost functions and predecessor messages, second part in (3), since most function classes are not closed with respect to summation and minimization. The only notable exception is the class of quadratic functions on the whole of $\mathbb{R}$, corresponding to Gaussian family graphical models in statistics for which exact inference is possible (Koller and Friedman, 2009). For our problem, however, power limits are an important ingredient, and thus quadratic functions on $\mathbb{R}$ are unsuitable. In all other cases, we need to use approximate message calculation schemes, since if we use a finitely parametrized function class to represent our messages – the only practical possibility on a finite resource computer – then subsequent exact messages will eventually leave that function class, even if all cost functions are representable within the chose function class.

We extend the work of Kellerer and Steinke (2015) with a novel approximate message calculation scheme for continuous P_v, P_{vw} variables, denoting the result as the continuous message passing algorithm (CMP). We assume costs $C_v(P_v)$ *and* messages $m_{vw}(P_{vw})$ to be convex quadratic functions on compact intervals. This choice of function class for the cost functions is motivated by generator cost curves that, in running mode, can often be approximated well using quadratic functions in the range between the allowed minimum, $\underline{P}_v$, and maximum, $\overline{P}_v$, power production. Using the same function class for the messages follows naturally, and the interval restrictions enable additionally the representation of transport capacity limitations (minimum $\underline{p}_{uv}$ and maximum $\overline{p}_{uv}$). The requirement of convexity of the cost functions is based on the idea that the chosen function class for the messages can only represent unimodal functions well.

After reviewing further related work in Section 2, we propose our novel, specially-adapted approximation scheme with an efficient message calculation mechanism in Section 3. Section 4 demonstrates our algorithm's features with a simulation experiment and we conclude in Section 5.

CMP features a strongly increased practicality in comparison to DMP, both in terms of the class of problems that can be solved (i.e. those with large capacity differences) and in terms of a significantly reduced communication load, since each message consist of only five numbers and not a vector of discretized values as in DMP, see also Section 4. The (partial) information security and the ability to evaluate finite step-size sensitivities remain the same as in DMP. CMP also scales linearly in the network size.

2. RELATED WORK

Several decentral economic dispatch methods have recently been proposed. The robust augmented Lagrangian method of Kim and Baldick (1997) coordinates optimal power flow (OPF) computations in different electrical regions via Lagrange multipliers. To obtain global optimality guarantees, Lam et al. (2012) propose a decomposable, convex relaxation of non-linear AC OPF (similar to Lavaei and Low (2012)) and show that the computed dispatch is identical to the optimal unrelaxed solution of the problem for tree grids. The alternating direction method of multipliers (ADMM) (Kraning et al., 2013) proposes an elegant formulation for convex OPFs and that can easily be implemented decentrally via exchanging primal and dual variables between the electrical network nodes. The main difference to our work is that all these methods perform iterative, distributed optimization via exchanging Lagrange multipliers, whereas our proposed methodology uses more complex messages but is not iterative.

A different approach is to use min-cost flow algorithms. For example, the theoretical work of Végh (2012) finds the global optimum in strongly polynomial time, but is again iterative in nature, requires a central solver for our problem setup, and does not yield finite step-size sensitivities.

Approximate inference in graphical models, the equivalent of approximate min-sum algorithms in statistics, is a major field in machine learning, see e.g. Wainwright and Jordan (2008) and the links therein. On the one hand many methods like e.g. expectation propagation (Minka, 2001) focus on marginalization of statistical models, i.e. the sum-product version of the min-sum algorithm. On the other hand loopy belief propagation is successfully applied also to optimization problems, see e.g. Frey and MacKay (1998). Projecting message updates onto a given class of functions in each iteration is done e.g. in Noorshams and Wainwright (2013).

However, we are not aware of literature using our proposed function class (quadratic plus interval constraints) for this purpose, nor the specific algorithm that we propose. Moreover, the application of approximate inference / approximate min-sum algorithms to power dispatch seems novel.

3. APPROXIMATE MESSAGE

We propose to derive (approximate) messages according to equation (3) by first computing the exact messages m_{vw} efficiently (Phase 1) and then re-projecting them onto our chosen function class (Phase 2), obtaining the approximate messages $\hat{m}_{uv}$. The resulting procedure is detailed in Algorithm 1 and graphically explained in Fig. 1. In this section, we describe the ideas behind the proposed algorithm and proof its correctness.

We start by characterizing the exact message m_{vw} on the attainable subset $\hat{\mathcal{P}}_{vw}$ as being a convex, piece-wise quadratic function in Proposition 1. This holds under the assumption that, as stated above, the incoming (approximate) messages $\hat{m}_{uv}$ and the local cost function C_v are convex quadratic functions on intervals. Note that the incoming messages are uniquely defined by five parameters, the quadratic coefficients $[\hat{\alpha}_{uv}, \hat{\beta}_{uv}, \hat{\gamma}_{uv}]$ and the interval limits $[\underline{p}_{uv}, \overline{p}_{uv}]$, but the exact message m_{vw} is not, since it is only piece-wise quadratic. In Proposition 2, we validate the proposed, efficient scheme for computing the parameters of the exact message m_{vw} from Algorithm 1 Phase 1. Finally, in Proposition 3, we show that the best quadratic approximation of m_{vw}, the new approximate message $\hat{m}_{vw}$, will again be convex. This result closes the circle to the convexity assumption on the incoming (approximate) messages $\hat{m}_{vw}$ for the next message computation step. It also ensures that the new message can again be communicated by transferring only five values.