Solving Unit Commitment Problems with Demand Responsive Loads

Devon Sigler*, Juliette Ugirumurera*, Jose Daniel Lara*,+, Sourabh Dalvi*, and Clayton Barrows*

*National Renewable Energy Lab, +University of California Berkeley

Overview

This work focuses on using variations of the Frank-Wolfe (FW) algorithm [1] for solving unit commitment problems with high volumes of demand responsive loads on the power grid. We present a formulation of the unit commitment problem with demand responsive loads and subsequently show, through reformulation and relaxations, that variations of the Frank-Wolfe algorithm can be used to determine time series decisions for demand responsive loads. Using computational experiments on the IEEE Reliability Test System, we demonstrate that the timeseries of demand responsive load decisions obtained through our approach is near optimal and describe how large-scale parallel implementations of our approach can achieve a high degree of computational efficiency.

Problem Formulation and Structure

Like [2], we present a generic version of the UC problem, but with a set of demand responsive agents A and their respective flexible loads, which we call the UCDR problem. This is a standard DCOPF power flow formulation where we add variables $s_{a,i}^t$ to represent demand responsive loads. The sets Π_a represent constraints on a demand responsive load.

$$\min_{p,x,y^{\pm}} \sum_{g\in G, t\in T} c_g^t(p_g^t, x_g^t) + \sum_{i\in\Phi, t\in T} c_i^t(y_i^{t,+}, y_i^{t,-}) \quad (1)$$

$$\text{s. t.} \sum_{g\in G_i} p_g^t + \sum_{e\in\mathcal{E}_{in}(i)} f_e^t - \sum_{e\in\mathcal{E}_{out}(i)} f_e^t \quad (2)$$

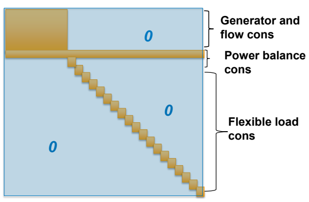
$$= \hat{d}_i^t + \sum_{a\in A} s_{a,i}^t + y_i^{t,+} - y_i^{t,-} \quad \forall i\in\Phi, t\in T$$

$$\underline{F}_e \leq f_e^t \leq \overline{F}_e \quad \forall e\in\mathcal{E} \quad (3)$$

$$B_e\left(\theta_i^t - \theta_j^t\right) = f_e^t \quad \forall e\in\mathcal{E}, t\in T. \quad (4)$$

$$(p_g, x_g) \in \Pi_g \quad \forall g\in G \quad (5)$$

$$(s_a, z_a) \in \Pi_a \quad \forall a\in A \quad (6)$$



Solution Approach

Due to problem structure of the UCDR problem we can solve a relaxed version using variations of the FW algorithm. These relaxations can be used to approximate optimal flexible load profiles. They can also be used in a branch and bound scheme.

Our Approach:
1.) Solve a relaxed version of the UCDR using a variation of the FW algorithm
2.) Project each s_a onto Π_a , which we denote as $\hat{s}_a$
3.) Solve UCDR with $\hat{s}_a$ values fixed

Problem Reformulation

To apply a variation of FW algorithm, we must reformulate the UCDR problem and relax certain constraints to obtain a relaxation suitable for FW algorithmic approaches.

$$\min_{d} \; H(d) \quad (9)$$

$$\text{s. t.} \; d_i^t = \hat{d}_i^t + \sum_{a\in A} s_{a,i}^t \quad (10)$$

$$(s_a, z_a) \in \Pi_a \quad \forall a\in A \quad (11)$$

where $H(d)$ is defined as

$$H(d) =$$

$$\min_{p,x,y^{\pm}} \sum_{g\in G, t\in T} c_g^t(p_g^t, x_g^t) + \sum_{i\in\Phi, t\in T} c_i^t(y_i^{t,+}, y_i^{t,-}) \quad (12)$$

$$\text{s. t.} \sum_{g\in G_i} p_g^t + \sum_{e\in\mathcal{E}_{in}(i)} f_e^t - \sum_{e\in\mathcal{E}_{out}(i)} f_e^t \quad (13)$$

$$= d_i^t + y_i^{t,+} - y_i^{t,-} \quad \forall i\in\Phi, t\in T$$

$$\underline{F}_e \leq f_e^t \leq \overline{F}_e \quad \forall e\in\mathcal{E} \quad (14)$$

$$B_e\left(\theta_i^t - \theta_j^t\right) = f_e^t \quad \forall e\in\mathcal{E}, t\in T \quad (15)$$

$$(p_g, x_g) \in \Pi_g \quad \forall g\in G. \quad (16)$$

The formulation presented here is equivalent to the UCDR, however it casts the problem as a function of the flexible loads, with local constraints on each flexible load.

We must ensure our objective function is convex, as well as our constraint set. Towards this end we consider a relaxed version of $H(d)$ and use the convex hull of Π_a,Conv(Π_a) .

Problem Relaxation

When all integer variables in (16) are relaxed it is well known that H becomes a convex piece-wise linear function of d, see[3], [4]. We denote the resulting convex function as $\hat{H}$. We consider Conv(Π_a) in the constraints to complete the relaxation.

$$\min_{d} \; \hat{H}(d) \quad (17)$$

$$\text{s. t.} \; d_i^t = \hat{d}_i^t + \sum_{a\in A} s_{a,i}^t \quad (18)$$

$$s_a \in \text{Conv}(\Pi_a) \quad \forall a\in A \quad (19)$$

This relaxation has a convex L-Lipschitz continuous objective function and a convex compact feasible region. This problem we solve with a parallelizable variation of the FW algorithm.

Proposed Algorithm (FW-Sample)

We apply a modified version of the FW based algorithm presented in [5], which minimizes the max over all sub-gradients of points in small neighborhood around the current iterate.

Instead, we use a sample-based set Υ = Conv($\pi_1, \ldots, \pi_N$) which provides only a rough approximation of all nearby sub-gradients. However, our results indicate that using this set results in good quality steps and is computationally efficient.

Highlighted in red are the parts of the algorithm that can be warm-started based on previous iterations or can be done in parallel. The other parts of the algorithm are extremely cheap to compute.

1) initialization: $0 \leftarrow k$, ϵ, n, N, and d_0 feasible for (10) and (11)
2) solve $\hat{H}(d_0)$ and collect dual variables π_i^t from constraints (13).
3) solve

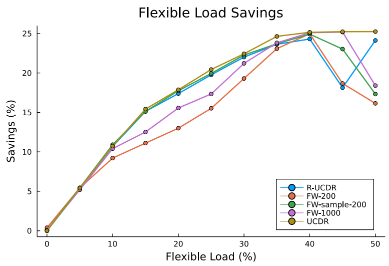
$$\min_{\substack{s_a\in\Pi_a \\ a\in A}} \max_{\pi\in\Gamma} \hat{H}(d_k) + \sum_{i\in\Phi}\sum_{t\in T}\left(\hat{d}_i^t + \sum_{a\in A} s_{a,i}^t - d_{i,k}^t\right)\pi_i^t \quad (20)$$

Step 3 parallel solve over A

4) update step size $\alpha \leftarrow \frac{2}{k+2}$
5) update solution $d_{i,k+1}^t \leftarrow d_{i,k}^t + \alpha(\hat{d}_i^t + \sum_{a\in A} s_{a,i}^t - d_{i,k}^t)$
6) compute $\hat{H}(d_{k+1})$ and collect dual variables π_i^t from constraints (13). ***Step 6 warm start via dual-simplex***
7) for $j = 1, \ldots, N$ ***step 7 Parallelize loop***
   sample $d_{k+1,j} \in \mathcal{B}(d_{k+1}, \epsilon)$
   solve $\hat{H}(d_{k+1,j})$ ***Step 7 warm start via dual-simplex***
   collect dual variables $\pi_{i,j}^t$ from constraints (13)
8) $k+1 \leftarrow k$
9) if: $k > n$
   break
   else:
   return to step 3.

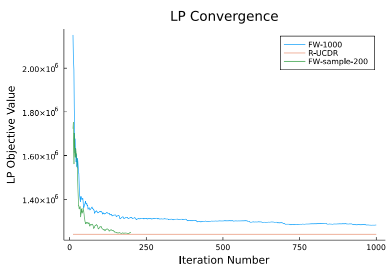
Results

To test the standard FW and FW-sample algorithm for computing solutions to the UCDR problem, we conducted a numerical study on the IEEE Reliability Test System (RTS-GMLC) [6]. To represent flexible loads on the system we allowed for a certain percentage of the load at each bus to be a flexible self scheduling load. Thus, there is one flexible load at each bus on the system. The percentage of self scheduling load was varied from 0-50%



Our implementations of the FW and FW-sample algorithms were written in the Julia language and used the PowerSystems.jl [7] and PowerSimulations.jl [8] ecosystem. The FICO Xpress Optimizer version 8.8.0. was used as the optimization solver.

To establish a baseline to compare methods against we solved the true UCDR problem. R-UCDR is the LP relaxation of the true problem. Numbers after algorithm names indicate the number of iterations the algorithm was run. The FW-sample used 10 samples.



Future Work

There are several avenues for future work. First, work to derive formal mathematical statements about the convergence of the FW-sample algorithm to understand under what conditions it converges with high probability should to be done. Second a more robust set of computational experiments to understand how the algorithm performs on different power systems and with different sets of demand responsive devices should be done. Third, testing a parallel implementation of the algorithm to understand how performant it.

References

1. M. Frank, P. Wolfe et al., "An algorithm for quadratic programming," Naval research logistics quarterly, vol. 3, no. 1-2, pp. 95–110, 1956.
2. B. Knueven, J. Ostrowski, and J.-P. Watson, "On mixed-integer programming formulations for the unit commitment problem," INFORMS Journal on Computing, vol. 32, no. 4, pp. 857–876, 2020.
3. D. Bertsimas and J. N. Tsitsiklis, Introduction to linear optimization. Athena Scientific Belmont, MA, 1997, vol. 6.
4. S. Boyd, S. P. Boyd, and L. Vandenberghe, Convex optimization. Cambridge university press, 2004.
5. S. N. Ravi, M. D. Collins, and V. Singh, "A deterministic nonsmooth frank wolfe algorithm with coreset guarantees," Informs Journal on Optimization, vol. 1, no. 2, pp. 120–142, 2019.
6. C. Barrows, A. Bloom, A. Ehlen, J. Ikaheimo, J. Jorgenson, D. Krishnamurthy, J. Lau, B. McBennett, M. O'Connell, E. Preston et al., "The ieee reliability test system: A proposed 2019 update," IEEE Transactions on Power Systems, vol. 35, no. 1, pp. 119–127, 2019.
7. J. D. Lara, C. Barrows, D. Thom, D. Krishnamurthy, and D. Callaway,"Powersystems.jl—a power system data management package for largescale modeling,"*SoftwareX*, vol. 15, p. 100747, 2021.
8. J. D. Lara, D. Krishnamurthy, C. Barrowset al., "Powersystems.jl and powersimulations.jl," National Renewable Energy Lab.(NREL),Golden, CO (United States), Tech. Rep., 2018.

This work was authored by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding was provided by the Laboratory Directed Research and Development (LDRD) Program at NREL. A portion of this research was performed using computational resources sponsored by the Department of Energy's Office of Energy Efficiency and Renewable Energy and located at the National Renewable Energy Laboratory. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government.

2022 IEEE Power & Energy Society General Meeting
Sheraton Denver Downtown Hotel Colorado , USA
July 17-21, 2022
NREL/PO-2C00-83433