Appendix of "Combined Carbon Capture and Utilization with Peer-to-Peer Energy Trading for Multi-Microgrids Using Multi-Agent Proximal Policy Optimization"

Ming Chen, Zhirong Shen, Lin Wang, *Senior Member, IEEE* and Guanglin Zhang, *Member, IEEE*

APPENDIX

[1] In this Appendix, we provide the detailed implementation process of the ADMM method compared in Fig. 14. Due to the highly time-coupled storage system dynamic limits (2a), (5a) and (11a), i.e., the next storage level states depend on the current states and actions. It leads to an increase in computational complexity and time complexity when the time horizon T becomes larger. Moreover, there is a variety of uncertain system parameters such as intermittent renewable generation, unpredictable power loads, and fluctuating electricity prices, which are unpredictable in practical implementation. Therefore, we adopt the one-step ADMM method. In each time-step t, the ADMM observes the current system state and solve the problem

$$\mathbf{P1}': \begin{aligned} &\text{minimize}: && \sum_{m=1}^{M}\{f_{m,t}^{P}+f_{m,t}^{H}-f_{m,t}^{E}\} \\ &\text{s.t.} && (1)-(12), \text{and } (14), \\ & && x_{m,m',t}+x_{m',m,t}=0, \end{aligned} \tag{a1}$$

Here, the equation (13) for resolving conflicts of trading intentions is transformed into the trading constraint $x_{m,m',t}+x_{m',m,t}=0$, and the ADMM method can direct solve the trading decision in an iterative manner. Besides, for the non-convex "*ln*" cost term in constraint (15c), we use the *fmincon* solver to solve it. Then, the control policy variable in time slot t can be denoted as $\mathbf{u}_t=\{B_{m,t},G_{m,t}^{I},G_{m,t}^{O},P_{m,t}^{H},\boldsymbol{x}_t^m\}_{m=1}^{M}$, where $\boldsymbol{x}_t^m=\{x_{m,m',t},\forall m'\in M\backslash m\}$. We introduce auxiliary variables $\widehat{\boldsymbol{x}}_{m,t}=\{\widehat{x}_{m,m',t},\forall m'\in M\backslash m\}$. Then, for $\forall m\in M,\forall m'\in M\backslash m$, we replace the trading constraint $x_{m,m',t}+x_{m',m,t}=0$ with

$$\begin{cases}\widehat{x}_{m,m',t}=x_{m,m',t},\\ \widehat{x}_{m,m',t}+x_{m',m,t}=0.\end{cases} \tag{a2}$$

Therefore, the augmented Lagrangian function for problem $\mathbf{P1}'$ is

$$\begin{aligned}\mathcal{L}=\sum_{m\in M}\Big[f_{m,t}^{P}+f_{m,t}^{H}-f_{m,t}^{E}+\sum_{m'\neq m}\Big(\lambda_{m,m',t}(\widehat{x}_{m,m',t}-x_{m,m',t})\\ +\frac{\rho}{2}\|\widehat{x}_{m,m',t}-x_{m,m',t}\|_2^2\Big)\Big],\end{aligned} \tag{a3}$$

where $\boldsymbol{\lambda}=\{\lambda_{m,m',t},\forall m'\in M\backslash m\}$ is the Lagrangian multiplier of problem $P1'$, and $\rho>0$ is the quadratic penalty parameter of the trading constraint. The ADMM solution method involves iterations between a local level problem and a global level problem. Specifically, the local level problem involves microgrids solving their local optimization problems in parallel based on fixed dual variables $\boldsymbol{\lambda}$ and auxiliary variables $\widehat{\boldsymbol{x}}_{m,t}$. The global level problem involves updating the auxiliary variables and dual variables using the results from the local level problems. The detailed optimization process of the ADMM algorithm is given as follows:

1) Update local energy scheduling and trading policies for each microgrid $\boldsymbol{u}_t^m=\{B_{m,t},G_{m,t}^{I},G_{m,t}^{O},P_{m,t}^{H},\boldsymbol{x}_t^m\},\forall m\in M$:

$$\begin{aligned}\boldsymbol{u}_t^m(k+1)=\underset{\boldsymbol{u}_t^m(k)}{\arg\min}\Big[f_{m,t}^{P}+f_{m,t}^{H}-f_{m,t}^{E}+\\ \sum_{m'\neq m}\Big(\frac{\rho}{2}\|\widehat{x}_{m,m',t}(k)-x_{m,m',t}\|_2^2-\lambda_{m,m',t}x_{m,m',t}\Big)\Big].\end{aligned} \tag{a4}$$

2) Update auxiliary variables $\widehat{\boldsymbol{x}}_{m,t},\forall m\in M$:

$$\begin{aligned}\widehat{\boldsymbol{x}}_{m,t}(k+1)=\underset{\widehat{\boldsymbol{x}}_{m,t}(k)}{\arg\min}\sum_{m\in M}\sum_{m'\neq m}\\ \Big[\frac{\rho}{2}\|\widehat{x}_{m,m',t}-x_{m,m',t}(k+1)\|_2^2+\lambda_{m,m',t}(\widehat{x}_{m,m',t})\Big].\end{aligned} \tag{a5}$$

3) Update the Lagrangian multipliers $\forall\lambda_{m,m',t}\in\boldsymbol{\lambda}$:

$$\begin{aligned}\lambda_{m,m',t}(k+1)=\lambda_{m,m',t}(k)\\ +\rho\Big(\widehat{x}_{m,m',t}(k+1)-x_{m,m',t}(k+1)\Big).\end{aligned} \tag{a6}$$

The ADMM method will repeat the above iterations until the following conditions are met, i.e.,

$$\sum_{m'\neq m}^{M}\|\widehat{x}_{m,m',t}(k)+\widehat{x}_{m',m,t}(k)\|_2\leq\epsilon_1, \tag{a7}$$

$$\sum_{m'\neq m}^{M}\|\widehat{x}_{m,m',t}(k+1)-\widehat{x}_{m,m',t}(k)\|_2\leq\epsilon_2. \tag{a8}$$

where k is the iteration index, ϵ_1 is the original residual threshold, and ϵ_2 is the dual residual threshold. After continuous iterations, the ADMM method optimizes and outputs the energy scheduling policy for the proposed energy coordination problem.

[1]This is the appendix of the paper titled "Combined Carbon Capture and Utilization with Peer-to-Peer Energy Trading for Multi-Microgrids Using Multi-Agent Proximal Policy Optimization", which is under review by the IEEE Transactions on Control of Network Systems.