Lecture Notes in Electrical Engineering 1203

Yongzhao Hua
Yishi Liu
Liang Han *Editors*

Proceedings of 2023 7th Chinese Conference on Swarm Intelligence and Cooperative Control

Swarm Optimization Technologies

Springer

Lecture Notes in Electrical Engineering 1203

Series Editors

The book series *Lecture Notes in Electrical Engineering* (LNEE) publishes the latest developments in Electrical Engineering—quickly, informally and in high quality. While original research reported in proceedings and monographs has traditionally formed the core of LNEE, we also encourage authors to submit books devoted to supporting student education and professional training in the various fields and applications areas of electrical engineering. The series cover classical and emerging topics concerning:

- Communication Engineering, Information Theory and Networks
- Electronics Engineering and Microelectronics
- Signal, Image and Speech Processing
- Wireless and Mobile Communication
- Circuits and Systems
- Energy Systems, Power Electronics and Electrical Machines
- Electro-optical Engineering
- Instrumentation Engineering
- Avionics Engineering
- Control Systems
- Internet-of-Things and Cybersecurity
- Biomedical Devices, MEMS and NEMS

For general information about this book series, comments or suggestions, please contact leontina.dicecco@springer.com.

To submit a proposal or request further information, please contact the Publishing Editor in your country:

China

Jasmine Dou, Editor (jasmine.dou@springer.com)

India, Japan, Rest of Asia

Swati Meherishi, Editorial Director (Swati.Meherishi@springer.com)

Southeast Asia, Australia, New Zealand

Ramesh Nath Premnath, Editor (ramesh.premnath@springernature.com)

USA, Canada

Michael Luby, Senior Editor (michael.luby@springer.com)

All other Countries

Leontina Di Cecco, Senior Editor (leontina.dicecco@springer.com)

**** This series is indexed by EI Compendex and Scopus databases. ****

Yongzhao Hua · Yishi Liu · Liang Han
Editors

Proceedings of 2023 7th Chinese Conference on Swarm Intelligence and Cooperative Control

Swarm Optimization Technologies

Springer

Editors
Yongzhao Hua
Institute of Artificial Intelligence
Beihang University
Beijing, China

Yishi Liu
School of Cyber Science and Technology
Beihang University
Beijing, China

Liang Han
Sino-French Engineer School
Beihang University
Beijing, China

ISSN 1876-1100 ISSN 1876-1119 (electronic)
Lecture Notes in Electrical Engineering
ISBN 978-981-97-3323-1 ISBN 978-981-97-3324-8 (eBook)
https://doi.org/10.1007/978-981-97-3324-8

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Contents

Distributed Nash Equilibrium Seeking Algorithm for Aggregative Games with Time-Varying Directed Communication Networks 1
Rui Zhu, Fuyong Wang, Zhongxin Liu, and Zengqiang Chen

Decentralized Alternating Direction Method of Multipliers for Constrained Optimization over Directed Networks 12
Jing Yan, Xinli Shi, Luyao Guo, and Ying Wan

Optimal Choice of Parameters for Coronavirus Herd Immunity Optimizer 25
Chao Liu, Xiangbo Qi, Leyu Yu, Xiaolong Tan, Yihui Su, and Bowen Liu

Multi-mode Composite Guidance Data Fusion Algorithm Based on Optimized Convex Combination Theory 37
Shiyi Zhao, Shuxin Liu, Daihua Wang, and Chen Si

Differential Game for Satellite Swarm Confrontation System with Strength Supplement 52
Lingwei Li and Bing Xiao

Research on Attack-Defense Game Evolution of Aircraft Swarms Considering Weapon Operational Effectiveness 64
Shiguang Hu, Le Ru, Maolong Lv, Xiaolin Zhao, Bo Lu, Wenfei Wang, Hailong Xi, and Gangshuo Yuan

Distributed Convergence to Nash Equilibria in a Zero-Sum Resource Allocation Game 77
Jie Hou and Xianlin Zeng

Event-Triggered Distributed NE Seeking Over Directed Networks 89
Qianle Tao, Chengxin Xian, and Yu Zhao

Optimal Leader-Following Tracking Control for Multiagent Systems: A Fully Distributed Approach 101
Guan Huang, Zhuo Zhang, Weisheng Yan, Rongxin Cui, and Shouxu Zhang

Variance-Reduced Distributed Splitting Schemes for Stochastic Generalized Nash Equilibrium Seeking 111
Haochen Tao, Shisheng Cui, and Jian Sun

Collective Foraging Mechanisms and Optimization Algorithms: A Review 123
Yinjie Ren, Zhan Xu, Jian Zhao, Jincun Liu, Yang Liu, and Jiahui Cheng

Optimal Strategies for the Target-Attacker-Defender Differential Game in 3D Space 136
Huiru Ye, Wenjing Hou, Li Liang, and Youqing Wang

Soft Actor-Critic Based Multi-drones Pursuit-Evasion Differential Game with Obstacles 148
Chengqiao Zhao, Zhicheng Deng, and Xiao Guo

Design and Optimization of a Multi-AMR Scheduling System Based on QT ... 162
Shuoxian Tang, Yongzheng Cong, Zhijie Cheng, Haibo Du, and Wenwu Zhu

Optimal Pricing Algorithm of EV Cluster Demand Response Considering Time-Space Coupling and User Bounded Rationality 173
Xinran Yan, Yuanrui Chen, Jun Zeng, Junfeng Liu, and Junhao Ma

Partial Information Defense Game with an Active Target 185
Qiyong He, Xiuxian Li, Min Meng, and Shuai Liu

Hierarchical Energy Optimization Management for Multi-microgrid Based on Potential Game and Nash Bargaining Game 198
Junhao Ma, Jun Zeng, Jingyao Zeng, Junfeng Liu, and Peng Zhou

Multiple Equal-Speed Players in Reach-Avoid Differential Games: Non-point Capture 210
Jiali Wang and Yang Tang

Pursuit-Evasion Game of Multiple Pursuers and Evaders with Intelligent Cooperation and Obstacle Avoidance in a Complex Environment 222
Maoxing Chen, Chunxi Yang, Xiufeng Zhang, and Gengen Li

Distributed Online Optimization with Long-Term Constraints and Bandit Feedback: An Event-Triggered Approach 235
Difeng Zhang, Zhangcheng Feng, Shaofu Yang, and Wenying Xu

A Novel GNN-Based Node Importance Ranking Method in a Heterogeneous Network 246
Shuyu Li, Dedong Ji, Jingwei Wang, and Weiguo Xia

Multi-objective Robust Optimization for Integrated Energy Systems Including Economics and Carbon Emissions 258
Enjie Li, Feilong Zhou, Haotian Xu, and Shuai Liu

Privacy Preserving Distributed Optimization via Paillier Encryption and Randomness Injection 270
Xinyan Cheng, Huan Gao, Yongfeng Zhi, and Shu Zhang

Task-Oriented Game Antagonism System Architecture of Heterogeneous Clusters 283
Dawei Liu, Mutian Guo, Lan Wei, Junda Zhai, Pinhua Huang, and Xiaoyue Wang

PSO-GA Based Fuel Optimization Algorithm for High Orbit One-to-Many Spacecraft Rendezvous 297
Jiukai Zhu, Yeqing Zhang, Jianhui Yu, Guangtao Ran, and Yanning Guo

Improving Interval Kalman Filtering Through Interval Optimization Strategy 310
Xuanchen Li, Xiaoling Wang, Juan Qian, and Guo-Ping Jiang

Flexible Terminal Constraints Based Optimal Strategy for Multiple UAVs Formation Transformation 322
Kaidan Li, Kewei Xia, and Jianan Wang

Optimal Strategies of Orbital Target-Attacker-Defender Game with a Non-maneuvering Target 334
Yifeng Li, Xi Liang, and Zhaohui Dang

Elman Neural Network Optimized by Swarm Intelligence for SOC Estimation of Lithium-Ion Battery 347
Dezhi Shen, Jie Ding, and Min Xiao

Transition Trajectory Planning for a VTOL Morphing UAV Based on Improved Pigeon-Inspired Optimization 359
Duo Gao, Boyi Chen, Teng Liao, Wenjun Xu, and Yanbin Liu

Research on Optimization Method for Configuration Connection of Multi UAV Cluster Collaborative Navigation 373
Chen Gu, Rong Wang, Weicheng Zhao, Zhi Xiong, and Jianye Liu

Multiparameter Optimization in Formation Control of Uncertain Swarm Systems with Collision Avoidance 385
Caiyun Wang, Cui Wei, Quanling Zhang, and Ye-Hwa Chen

Enhancing Flexibility and Cost-Effectiveness of Virtual Power Plants Through Multi-market Profit Optimization and Interruptible Loads Integration 398
Xiaohui Chang, Wei Chen, and Chunquan Mi

Renewable Energy System Optimization Decision Method Considering Wind Power Prediction 409
Yulong Han, Licai Guo, and Haibo He

The Design of an Efficient Distributed Collaborative Scheduling Method and the Optimal Planning Strategy for Providing Photovoltaic Access Capacity 418
Fan Bai, Pengcheng Guo, Luda Wang, Liwei Sun, Haifeng Chu, Siyu Lv, Hongzhe Liu, and Wenwu Yu

Observer-Based Positive Edge Consensus of Multi-agent Systems on Directed Graphs 430
Pengyu Zhang, Wei Zhang, Li Mao, Jiayi Chen, and Zhi Hu

A New Comparison Function Based Direct Multisearch Method for Derivative-Free Multi-objective Optimization Problems 440
Fuyu Zhao, Hui Lv, Yongxia Liu, and Dandan Liu

Learning-Based Optimal Pursuing Strategies Against Random Orbital Evaders 453
Zhaoyang Liu, Xi Liang, Zhaohui Dang, and Ganghua Su

Spacecraft Droplet Hover Trajectory Optimization Based on Chaotic Genetic Algorithm 465
Rui Shao, Xinglong Li, Bin Song, Zhengyang Liu, and Guangfu Ma

Impulsive Orbital Pursuit-Evasion Game Considering Reaction Time 477
Wenyuan Xie, Yutao Hao, and Zhaohui Dang

Multi-aircraft Cooperative Horizontal Track Programming Algorithm Based on Convex Optimization Algorithm 489
Jun Jie Zhang, Cheng Xu , and Yi Ming Feng

A Novel Method for Optimizing the Electromagnetic Compatibility of the Multi-radio-Frequency System 502
Houpu Xiao, Youwei Meng, Haoting Yan, and Yaoyao Li

Delta-V-Based Strategies in Orbital Pursuit-Evasion-Interception Game 511
Sihan Xu, Xi Liang, and Zhaohui Dang

Swarm Intelligence Optimization Algorithm in JSP: A Review on Models, Methods and Applications 524
Penghao Ren, Zongfeng Wang, and Yuejing Pan

An Improved Genetic Algorithm with Tabu Tables for Flexible Job-Shop Scheduling Problems 532
Bao Jie and Zhu Zhenghu

Target Assignment in the Multi-UAV Combat Based on Distributed Compensation Auction Algorithm with Fixed-Time Consensus Protocol 539
Yiyang Chen, Yongzhao Hua, Jianglong Yu, and Xiwang Dong

Model Predictive Control Using Indirect Operator Splitting Quadratic Program Method and Its Application in PMSM 552
Gang-qiang Ma, Qing-hui Meng, Hui-long Zhu, Chao Lu, and Han-rui Wang

Minimal Delta-V-Based Orbital Rendezvous for Approaching a Non-cooperative Target on GEO 567
Yutao Hao, Zhaoyang Liu, and Sihan Xu

A Hierarchical Formation-Containment Control Scheme for Second-Order Multi-agent Systems 580
Yue Cao, Xiangyu Wang, and Yangyang Chen

Recommendation Based on Knowledge Graph and Graph Neural Network 593
Hongjie Shen, Qiuhao Guo, Hongfeng Xu, Jixiang Jiang, and Linxiu Chen

Implicit Cooperative Decision-Making for Unknown Area Exploration of Multi-agent Systems 604
Yong Wang, Yuanning Zhu, and Qingkai Yang

Underwater Object Detection Based on Transformer Query Optimization 616
Hongyang Sun and Baojie Fan

Design of Bending Machine Control System Based on Embedded Soft PLC . . . 628
Jun Yan, Yue Yuan, Dawei Ding, Jin Tao, Fengyu Xu, and Chaobin Hu

Spacecraft Chase and Escape Game Based on PPO Algorithm 641
Ruiqing Ye, Wudong Deng, Yanning Guo, and Chuanjiang Li

Formation Control Against Attack: An Integrated Design of Stackelberg and Zero-Sum Games 652
Shuxian Yang, Hao Yang, and Yuhang Xu

A Sparse Semantic Reconstruction Method for Spacecraft for Collaborative Awareness 661
Bin Wu, Yuhong Zeng, Xuemei Zhao, Zhengyang Gao, and Weixin Wang

A Mixed-Integer Trajectory Optimization Method for Unpowered Aerial Vehicles Formation Reconfiguration 674
Yang Chen, Zhuo Liang, and Rui Lv

Optimal Evasion Strategy of Three-Dimensional Differential Game Based on RHC 685
Shenjia Zhao, Hairui Zhang, and Rui Lv

Author Index 697

Distributed Nash Equilibrium Seeking Algorithm for Aggregative Games with Time-Varying Directed Communication Networks

Rui Zhu[1,2], Fuyong Wang[1,2(✉)], Zhongxin Liu[1,2], and Zengqiang Chen[1,2]

[1] College of Artificial Intelligence, Nankai University, Tianjin 300350, China
[2] TianjinKey Laboratory of Brain Science and Intelligent Rehabilitation, Nankai University, Tianjin 300350, China
wangfy@nankai.edu.cn

Abstract. A distributed discrete Nash equilibrium (NE) seeking algorithm is designed for aggregative games (AGs) through multi-round communications under the restricted strongly monotone assumption. Every agent can observe its own cost function and strategy, and access information only of neighbors according to the time-varying directed communication networks. Then, the proposed algorithm where the number of communications per iteration is fixed turns out to converge to a unique NE point and the rate of convergence is linear. The complexity of the algorithm in this paper is lower compared with others of increasing communication rounds. Finally, a networked Nash-Cournot game is considered to show the accuracy of the algorithm.

Keywords: Distributed NE seeking · Aggregative game · Time-varying directed networks · Linear convergence · Restricted strongly monotone

1 Introduction

It is simple to note that there exist a lot of cases around us that all agents of a multi-agent system are in a competitive relationship such as the congestion control of communication networks [1]. Every agent in this time can be regarded as a player of non-cooperative game and they only want to minimize their own cost function. Thereupon, the concepts of game theory were applied to multi-agent systems, which had made a lot of progress in the research of multi-agent systems [2,3]. The NE is an important concept in non-cooperative games. At the NE point any agent unilaterally changing its strategy will not reduce its own costs, it is an optimal solution for all agents. Hence, how to seek the NE arouses the research interest of many experts and scholars.

At present, there exist many algorithms about Nash equilibrium seeking, and with the deepening of research, the algorithms develop from centralized [4] to semi-decentralized[5,6], then to distributed [7,8]. The distributed setup contains

Y. Hua et al. (Eds.): CCSICC 2023, LNEE 1203, pp. 1–11, 2024.
https://doi.org/10.1007/978-981-97-3324-8_1

many advantages comparing with the others, such as communication cost savings, privacy protection and robustness. Tatarenko *et al.* [9] have proposed a kind of Nesterov type accelerated distributed gradient play algorithm for convex networked Nash games with strongly monotone mappings and verified the geometric convergence rate of this discrete algorithm. Distributed algorithm designs for different game models are distinct. Aggregative games, as an important subclass of non-cooperative game, are taken as the research objects in many literatures to design corresponding distributed algorithms [10–12]. Every agent in it will be influenced by the aggregate of all agents rather than just other agents' strategies. Reference [10] has considered a distributed continuous-time algorithm to seek the generalized NE of aggregative games with coupled constraints. In [11], a distributed algorithm with multi-round constant communications has been investigated for strongly monotone AGs based on the undirected connected graph. This algorithm reaches the linear convergence rate and needs less communication rounds per iteration. These advantages inspire us to extend this algorithm to directed time-varying topological graphs so that it can be applicable to a wider range of scenarios.

Several attempts have been made to research aggregation games on time-varying topological graphs including the continuous [13,14] and discrete [15,16] time. Ye *et al.* [14] proposed a distributed continuous privacy-preserving Nash equilibrium seeking algorithm which combined a gradient algorithm with the perturbed average consensus protocol when the objective function of every agent cannot be leaked. There exist some differences between continuous and discrete algorithm, while this paper focuses on discrete forms. A distributed discrete algorithm is derived to seek the generalized NE of AGs over undirected time-varying jointly connected graphs in [15]. Reference [16] introduces a momentum term into the update of agents' strategies and this term can accelerate the convergence rate of designed distributed discrete-time NE seeking algorithms with vanishing step-sizes over time-varying jointly strongly connected graphs. However, current proposed discrete algorithms can not be verified to be linearly convergent directly for AGs over time-varying topological graphs. Hence, this makes us more determined to apply multi-round communications to the algorithm design of this model.

Compared with the literatures above, the followings are the main contributions of this paper in summary. Firstly, the multi-round communications are applied to design the discrete time algorithm for AGs over time-varying directed communication networks. Secondly, the assumption about strongly monotone mapping is relaxed as the restricted strongly monotone one, which make the assumption milder. Thirdly, this algorithm is proved to converge linearly to the NE.

The rest of the paper is arranged as follows: Sect. 2 introduces the distributed NE seeking problem for AGs with time-varying topological graphs and makes some common assumptions, while the algorithm for solving this problem is established in Sect. 3. Section 4 analyzes the convergence property of this algorithm. Then a numerical example is provided in Sect. 5. Finally, a brief conclusion is given in Sect. 6.

2 Problem Statement

Consider the following AG including the agent set, strategy sets and cost functions. The agent set $\mathcal{M}$ consists of m agents, i.e. $\mathcal{M} = \{1, 2, \ldots, m\}$. Let $\Omega_j \in \mathbb{R}^l$ denote the strategy set of the agent j, where $\mathbb{R}^l$ is l-dimensional real space. Define $x = (x_1^T, \ldots, x_m^T)^T \triangleq col(x_1, \ldots, x_m) \in \Omega = \prod_{j=1}^m \Omega_j$ as the strategy profile, where $x_j \in \Omega_j$ stands for the strategy variable of the agent j and $\prod_{j=1}^m \Omega_j$ is the Cartesian product of the strategy sets of all agents. The cost function $J_j(x_j, \delta(x))$ of agent j is subject to its own strategy and an aggregate function $\delta(x) = \frac{1}{m}\sum_{j=1}^m x_j$ depending on the strategies of all agents. Every agent in this game takes minimizing its own cost function as the goal, that is

$$\min_{x_j \in \Omega_j} J_j(x_j, \frac{1}{m}x_j + \delta(x_{-j})) \quad \forall j \in \mathcal{M} \tag{1}$$

where x_{-j} are all rival agents other than the agent j, $\delta(x_{-j})$ expresses all rival agents' aggregate and $J_j : \Omega_j \times \mathbb{R}^l \to \mathbb{R}$.

Definition 1. *A strategy profile x^* is called a NE of the game (1), if $J_j(x_j^*, \frac{1}{m}x_j^* + \delta(x_{-j}^*)) \leq \inf\{J_j(x_j', \frac{1}{m}x_j' + \delta(x_{-j}^*))|x_j' \in \Omega_j\}$ holds for every agent $j \in \mathcal{M}$.*

For subsequent convergence analysis, some common assumptions are enumerated.

Assumption 1. *The strategy set Ω_j is convex and compact for every agent $j \in \mathcal{M}$. Furthermore, at arbitrarily fixed $x_{-j} \in \Omega_{-j}$, the cost function $J_j(x_j, \delta(x))$ is convex in $x_j \in \Omega_j$ and continuously differentiable in x_j.*

Based on Assumption 1, Two mappings $G_j : \Omega_j \times \mathbb{R}^l \to \mathbb{R}^l$ and $\Psi_j : \Omega \to \mathbb{R}^l$ are defined respectively as

$$G_j(x_j, y_j) := (\nabla_{x_j} G_j(\cdot, \delta) + \frac{1}{m}\nabla_\delta G_j(x_j, \cdot)) \mid_{\delta = y_j}$$
$$\Psi_j(x) := \nabla_{x_j} J_j(x_j, \delta(x)).$$

Let $y = col(y_1, \ldots, y_m)$, $G(x, y) = col(G_1(x_1, y_1), \ldots, G_m(x_m, y_m))$ and $\Psi(x) = col(\Psi_1(x), \ldots, \Psi_m(x))$, it is obviously obtained that $\Psi(x) = G(x, \mathbf{1}_m \otimes \mathbf{I}_l \delta(x))$.

The following lemma is the famous result of associating the NE with the solution to a variational inequality.

Lemma 1 ([17]). *A NE x^* of the game (1) is the same as a solution of the variational inequality problem $VI(\Omega, \Psi)$, that is, $x^* = \Pi_\Omega[x^* - \alpha\Psi(x^*)], \forall \alpha > 0$, where $\Pi_\Omega(z) := argmin_{x \in \Omega}\{\| x - z \|\}$ is the Euclidean projection of $z \in \mathbb{R}^{ml}$ onto a set $\Omega \subset \mathbb{R}^{ml}$.*

Some assumptions are needed about the mappings $\{G_j\}_{j=1}^m$ and Ψ to aid in convergence analysis of the algorithm.

Assumption 2. *The map $G_j(x_j, y)$ is L_j-Lipschitz continuous on $y \in \mathbb{R}^l$ for any fixed $x_j \in \Omega_j$, i.e., $\| G_j(x_j, y') - G_j(x_j, y'') \| \leq L_j \| y' - y'' \|$, $\forall y', y'' \in \mathbb{R}^l$. Denote $L_G := \max_{j \in \mathcal{M}} L_j$.*

Assumption 3. *The map $\Psi(x)$ is L_Ψ-Lipschitz continuous on Ω and restricted γ_Ψ-strongly monotone in regard to any NE $x^* \in \Omega$, that is,*

$$\| \Psi(x) - \Psi(\widehat{x}) \| \leq L_\Psi \| x - \widehat{x} \| \qquad \forall x, \widehat{x} \in \Omega;$$
$$(\Psi(x) - \Psi(x^*)^T (x - x^*) \geq \gamma_\Psi \| x - x^* \|^2 \quad \forall x \in \Omega.$$

Remark 1. *It is worth noticing that there exists a unique NE $x^* \in \Omega$ for the considered game under Assumption 1 and 3 [9].*

Assumption 4. *The directed communication network is strongly connected at every time $k \in \mathbb{N}$, where $\mathbb{N}$ denotes the set of natural numbers. Then, the weighted adjacency matrix $W(k) \in \mathbb{R}^{m \times m}$ of this network satisfies:*

- *Self-loops: $w(k)_{j,j} > 0$ for any $j \in \mathcal{M}$, where $w(k)_{j,j}$ represents the element on row j and column j of $W(k)$;*
- *Double stochastic(column and row stochastic): $W(k)\mathbf{1}_m = \mathbf{1}_m$ and $\mathbf{1}_m^T W(k) = \mathbf{1}_m^T$.*

It is obvious that $\lambda_{m-1}(W(k)) < 1$ for any k under Assumption 4, where $\lambda_{m-1}(W(k))$ indicates the second largest eigenvalue of $W(k)$. Further, the following assumption can be established when the networks are selected among a finite family [18].

Assumption 5. *There exists $\widehat{\lambda} \in (0, 1)$ which makes $\lambda_{m-1}(W(k)) \leq \widehat{\lambda}$ hold for any k.*

3 Algorithm Design

For solving the NE of considered aggregative game, a distributed multi-round communication NE seeking algorithm is put forward in Algorithm 1.

Algorithm 1. Distributed multi-round communication NE seeking algorithm

Initialization: for every agent j, set the initial strategy and aggregate estimate as $x_j(0) \in \Omega_j$ and $s_j(0) = x_j(0)$, respectively. Let the step-size be $\alpha > 0$ and μ be a positive integer signifying the constant communication rounds at every iteration.

Update of $x_j(k)$. For every agent $j \in \mathcal{M}$, the update form of the strategy estimate is

$$x_j(k+1) := \Pi_{\Omega_j}[x_j(k) - \alpha G_j(x_j(k), s_j(k))]. \tag{2}$$

Update of $s_j(k)$. For every agent $j \in \mathcal{M}$, the first update form of aggregate estimate is

$$\widetilde{s}_j(k+1) := \sum_{i=1}^{m} w_{j,i}(s_i(k) + x_i(k+1) - x_i(k)); \tag{3}$$

then set $\widetilde{s}_j^1(k+1) = \widetilde{s}_j(k+1)$ and the following update is repeated for $v = 1, \ldots, \mu - 1$,

$$s_j^v(k+1) := \sum_{i=1}^{m} w_{j,i}\widetilde{s}_i^v(k+1); \quad \widetilde{s}_j^{v+1}(k+1) := s_j^v(k+1). \tag{4}$$

Finally, $s_j(k) := \widetilde{s}_j^{\mu}(k+1)$.

Let $x(k) := col(x_1(k), \ldots, x_m(k))$, $\widetilde{s}(k) := col(\widetilde{s}_1(k), \ldots, \widetilde{s}_m(k))$ and $s(k) := col(s_1(k), \ldots, s_m(k))$. Then, the proposed algorithm is transformed into the following matrix form:

$$\begin{cases} x(k+1) &= \Pi_{\Omega}[x(k) - \alpha G(x(k), s(k))] \\ \widetilde{s}(k+1) &= (W(k) \otimes \mathbf{I}_l)(s(k) + x(k+1) - x(k)). \\ s(k+1) &= (W(k) \otimes \mathbf{I}_l)^{\mu-1}\widetilde{s}(k+1) \end{cases} \tag{5}$$

According to the communication network of iteration k, every agent j can receive the difference of the strategy estimate $x_q(k+1) - x_q(k)$ and aggregate estimate $s_q(k)$ from its neighbors $q \in \mathcal{M}$, moreover, it can also transmit these information of its own to agents whose neighbors include the agent j.

4 Convergence Analysis

Firstly, the relationship between $x(k)$ and $s(k)$ is demonstrated.

Lemma 2. *Based on Assumption 4, the formula* $\sum_{j=1}^{m} s_j(k) = \sum_{j=1}^{m} x_j(k)$ *holds for any* $k \geq 0$.

Proof. The formula can be verified by the mathematical induction. Because $s_j(0) = x_j(0)$ for all agents j, it is easy to know that the equation holds when $k = 0$. Then, suppose the equation holds for $k > 0$, there exists

$$\begin{aligned} \sum_{j=1}^{m} s_j(k+1) &= (\mathbf{1}_m^T \otimes \mathbf{I}_l)s(k+1) \\ &= (\mathbf{1}_m^T \otimes \mathbf{I}_l)(W(k) \otimes \mathbf{I}_l)^{\mu}(s(k) + x(k+1) - x(k)). \end{aligned}$$

On account of $\mathbf{1}_m^T W(k) = \mathbf{1}_m^T$, we have

$$\begin{aligned} \sum_{j=1}^{m} s_j(k+1) &= (\mathbf{1}_m^T \otimes \mathbf{I}_l)(s(k) + x(k+1) - x(k)) \\ &= \sum_{j=1}^{m} s_j(k) + \sum_{j=1}^{m} x_j(k+1) - \sum_{j=1}^{m} x_j(k) = \sum_{j=1}^{m} x_j(k+1). \end{aligned} \tag{6}$$

Therefore, this formula holds for any $k \geq 0$. The proof is completed.

Next, the convergence of the algorithm is given.

Theorem 1. *Let Assumptions 1–5 hold, the sequence* $\{x(k)\}_{k\geq 0}$ *generated by the designed Algorithm 1 will converge to the NE* x^* *at the rate of linear convergence if the step-size* α *and communication rounds* μ *per iteration satisfy the following conditions, respectively,*

$$0 < \alpha < min\{\frac{1-\widehat{\lambda}^{\mu}}{2\widehat{\lambda}^{\mu}L_G}, \frac{4(1-\widehat{\lambda}^{\mu})\widehat{\lambda}^{\mu}L_G - 2\gamma_{\Psi}(1-\widehat{\lambda}^{\mu})^2}{4\widehat{\lambda}^{2\mu}L_G^2 - L_{\Psi}^2(1-\widehat{\lambda}^{\mu})^2}\}, \tag{7}$$

$$\mu > \log_{\widehat{\lambda}} \frac{\gamma_{\Psi}}{2L_G + \gamma_{\Psi}}. \tag{8}$$

Proof. It follows from Lemma 2 that the estimation error variable can be expressed as

$$h(k) := s(k) - \mathbf{1}_m \otimes \mathbf{I}_l \delta(x(k)) = (\mathbf{I}_m - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^T) \otimes \mathbf{I}_l s(k).$$

Therefore, the (5) can be rewrote as based on $W(k)\mathbf{1}_m = \mathbf{1}_m$,

$$\begin{cases} x(k+1) &= \Pi_{\Omega}[x(k) - \alpha G(x(k), h(k) + \mathbf{1}_m \otimes \mathbf{I}_l \delta(x(k)))] \\ h(k+1) &= (W(k) \otimes \mathbf{I}_l)^{\mu} h(k) + [W(k)^{\mu}(\mathbf{I}_m - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^T)] \otimes \mathbf{I}_l (x(k+1) - x(k)). \end{cases} \tag{9}$$

Then, define $T(x) = \Pi_{\Omega}[x - \alpha\Psi(x)]$ and $\varepsilon(x,h) = \Pi_{\Omega}[x - \alpha G(x, h + \mathbf{1}_m \otimes \mathbf{I}_l\delta(x))] - T(x)$ to facilitate the subsequent analysis.

For any $x' \in \Omega$, there exists through Assumption 3

$$\begin{aligned} \| T(x') - T(x^*) \| &\leq \| x' - x^* - \alpha(\Psi(x') - \Psi(x^*)) \| \\ &= \sqrt{\| x' - x^* - \alpha(\Psi(x') - \Psi(x^*)) \|^2} \\ &\leq \sqrt{1 - 2\alpha\gamma_{\Psi} + \alpha^2 L_{\Psi}^2} \| x' - x^* \| . \end{aligned}$$

Similarly, we can obtain by Assumption 2

$$\begin{aligned} \| \varepsilon(x,h) \| &\leq \alpha \| G(x, h + \mathbf{1}_m \otimes \mathbf{I}_l\delta(x)) - G(x, \mathbf{1}_m \otimes \mathbf{I}_l\delta(x)) \| \\ &\leq \alpha \sqrt{\sum_{j=1}^{m} L_j^2 \| h_j \|^2} \leq \alpha L_G \| h \| . \end{aligned}$$

Due to Lemma 1 and the above analysis, we have

$$\begin{aligned} \| x(k+1) - x^* \| &\leq \| T(x(k)) - T(x^*) \| + \| \varepsilon(x(k), h(k)) \| \\ &\leq \sqrt{1 - 2\alpha\gamma_{\Psi} + \alpha^2 L_{\Psi}^2} \| x(k) - x^* \| + \alpha L_G \| h(k) \| \end{aligned} \tag{10}$$

and

$$\begin{aligned}\| x(k+1)-x(k) \| \leq & \| x(k+1)-x^* \| + \| x(k)-x^* \| \\ & \leq (\sqrt{1-2\alpha\gamma_\Psi+\alpha^2 L_\Psi^2}+1)\times \| x(k)-x^* \| +\alpha L_G \| h(k) \| .\end{aligned}$$

Next, a series of orthogonal matrices $\mathbf{S}(k)=(\mathbf{S}_1(k),\mathbf{S}_2(k))$ are introduced, where every column of $\mathbf{S}_1(k)\in\mathbb{R}^{m\times(m-1)}$ consists of the eigenvector of $W(k)$ and $\mathbf{S}_2(k)=\frac{\mathbf{1}_m}{\sqrt{m}}$ corresponds to the eigenvalue 1 of $W(k)$ for any k, that is,

$$S(k)^T W(k) S(k)=\begin{bmatrix}\mathbf{E(k)} & \mathbf{0}\\ \mathbf{0} & 1\end{bmatrix}$$

where $\mathbf{E(k)}=diag\{\lambda_1(W(k)),\ldots,\lambda_{m-1}(W(k))\}$ and $0<\lambda_1(W(k))\leq\ldots\leq\lambda_{m-1}(W(k))<1$.

Hence,

$$\begin{aligned}(W(k)\otimes\mathbf{I}_l)^\mu h(k) &= S(k)\begin{bmatrix}\mathbf{E(k)} & \mathbf{0}\\ \mathbf{0} & 1\end{bmatrix}^\mu S(k)^T\otimes\mathbf{I}_l h(k)\\ &= S_1(k)E(k)^\mu S_1(k)^T\otimes\mathbf{I}_l h(k),\end{aligned}$$

then we can obtain

$$\| (W(k)\otimes\mathbf{I}_l)^\mu h(k) \| \leq \lambda_{m-1}^\mu(W(k)) \| h(k) \| \leq \widehat{\lambda}^\mu \| h(k) \| \tag{11}$$

follows from the Assumption 5 and the properties $\| S_1(k) \|=1$ of orthogonal matrices.

Similarly, owing to $\| \mathbf{I}_m-\frac{1}{m}\mathbf{1}_m\mathbf{1}_m^T \|=1$,

$$\begin{aligned}&\| [W(k)^\mu(\mathbf{I}_m-\frac{1}{m}\mathbf{1}_m\mathbf{1}_m^T)]\otimes\mathbf{I}_l(x(k+1)-x(k)) \|\\ &\leq \widehat{\lambda}^\mu \| (x(k+1)-x(k)) \|\\ &\leq \widehat{\lambda}^\mu[(\sqrt{1-2\alpha\gamma_\Psi+\alpha^2L_\Psi^2}+1) \| x(k)-x^* \| +\alpha L_G \| h(k) \|].\end{aligned} \tag{12}$$

Based on the above analysis, it is easy to know that

$$\begin{aligned}\| h(k+1) \| \leq & \widehat{\lambda}^\mu(1+\alpha L_G) \| h(k) \| +\\ & \widehat{\lambda}^\mu(\sqrt{1-2\alpha\gamma_\Psi+\alpha^2L_\Psi^2}+1) \| x(k)-x^* \|\end{aligned} \tag{13}$$

Therefore, there yields combining (10) with (13)

$$\begin{pmatrix}\| x(k+1)-x^* \|\\ \| h(k+1) \|\end{pmatrix} \leq Q\begin{pmatrix}\| x(k)-x^* \|\\ \| h(k) \|\end{pmatrix} \tag{14}$$

where

$$Q = \begin{pmatrix} \sqrt{1-2\alpha\gamma_\Psi+\alpha^2 L_\Psi^2} & \alpha L_G \\ \widehat{\lambda}^\mu(\sqrt{1-2\alpha\gamma_\Psi+\alpha^2 L_\Psi^2}+1) & \widehat{\lambda}^\mu(\alpha L_G+1) \end{pmatrix}.$$

Let $\Xi := \sqrt{1-2\alpha\gamma_\Psi+\alpha^2 L_\Psi^2}$ for ease of understanding. It is noted that the matrix Q is a non-negative matrix. So $\rho(Q) < 1$ can be demonstrated if all the sequential principal minors of $\mathbf{I}_2 - Q$ is positive. In other words, $1-\Xi > 0$ and $det(\mathbf{I}_2 - Q) > 0$ need to be proved when (7) and (8) hold.

Firstly, we demonstrate $1-\Xi > 0$. It follows from (8) that $2\widehat{\lambda}^\mu L_G - \gamma_\Psi(1-\widehat{\lambda}^\mu) < 0$. Because $\gamma_\Psi \le L_\Psi$, there exists $2\widehat{\lambda}^\mu L_G - L_\Psi(1-\widehat{\lambda}^\mu) < 0$ which ensures the second term of the upper bound for α is positive. Then, we can find that in view of (7)

$$\alpha(2\gamma_\Psi(1-\widehat{\lambda}^\mu)^2 - 4(1-\widehat{\lambda}^\mu)\widehat{\lambda}^\mu L_G) + \alpha^2(4\widehat{\lambda}^{2\mu}L_G^2 - L_\Psi^2(1-\widehat{\lambda}^\mu)^2) > 0$$

which is rearranged as

$$\frac{4\widehat{\lambda}^{2\mu}L_G^2}{(1-\widehat{\lambda}^\mu)^2}\alpha^2 - \frac{4\widehat{\lambda}^\mu L_G}{1-\widehat{\lambda}^\mu}\alpha + 1 > \alpha^2 L_\Psi^2 - 2\gamma_\Psi\alpha + 1.$$

It is obvious that the right side of the above inequality is positive, so the two sides of it are squared as

$$1 - \frac{2\widehat{\lambda}^\mu}{1-\widehat{\lambda}^\mu}\alpha L_G > \sqrt{1-2\alpha\gamma_\Psi+\alpha^2 L_\Psi^2}. \tag{15}$$

Therefore, we can obtain $1-\Xi > \frac{2\widehat{\lambda}^\mu}{1-\widehat{\lambda}^\mu}\alpha L_G > 0$.

Next, $det(\mathbf{I}_2 - Q) > 0$ is verified. The form of $det(\mathbf{I}_2 - Q)$ is as follows,

$$\begin{aligned} det(\mathbf{I}_2 - Q) &= (1-\Xi)[1-\widehat{\lambda}^\mu(1+\alpha L_G)] - \alpha L_G\widehat{\lambda}^\mu(\Xi+1) \\ &= (1 - \frac{2\alpha L_G}{1-\Xi+2\alpha L_G} - \widehat{\lambda}^\mu)(1-\Xi+2\alpha L_G). \end{aligned} \tag{16}$$

Based on (15), we have

$$1-\Xi+2\alpha L_G > \frac{2\alpha L_G}{1-\widehat{\lambda}^\mu} > 0.$$

that is, $1-\widehat{\lambda}^\mu > \frac{2\alpha L_G}{1-\Xi+2\alpha L_G}$. Then, it is easy to see that $det(\mathbf{I}_2 - Q) > 0$.

As a result, we can obtain $\rho(Q) < 1$, which implies the algorithm (5) will converge to the NE x^* at a linear rate when (7) and (8) hold. The proof is completed.

5 Numerical Example

A networked Nash-Cournot game, as in [19], is considered in this section to show this designed algorithm will converge to the NE. This game consists of 15 firms which sell their own products to a unique market. All firms in the game exchange information based on the time-varying directed graphs. The graph of every iteration is generated randomly and satisfies Assumption 4 and 5, where $\widehat{\lambda} = 0.9$ is set as the upper bound of the second largest eigenvalue λ_{m-1} for a series of graphs' adjacency matrices i.e. $\{W(k)\}_{k=1}^{\infty}$.

Denote the quantity of products sold by the firm j to the market by $x_j \in \Omega_j$, where $\Omega_j = [0, \frac{j}{10} + \frac{1}{j^2}]$ is taken. And Let $J_j(x_j, \delta(x)) = c_j(x_j) - p(m\delta(x))x_j$ be the cost function of the firm j, where

$$c_j(x_j) = e_j x_j^2; \quad p(m\delta(x)) = 5 - r\delta(x).$$

Choose $e_j = 0.002 + 0.001j^2$ and $r = 0.005$. Let $e = col(e_1, \ldots, e_m)$, it is obvious that

$$G_j(x_j, s_j) = 2e_j x_j - 5 + rs_j + \frac{r}{m}x_j,$$
$$\Psi(x) = 2diag\{e\}x - 5\mathbf{1}_m + \frac{r}{m}(x + \mathbf{1}_m^T x \mathbf{1}_m).$$

Then, it can be calculated that $L_\Psi = 0.4547, \gamma_\Psi = 0.0066, L_G = 0.005$.

Let $\alpha = 0.003$ and $\mu = 9$ which can ensure (7) and (8) hold. The following Fig. 1 is trajectory of the strategy and consensus error, which verifies the effectiveness of Theorem 1 and shows the designed algorithm will converge to the NE linearly.

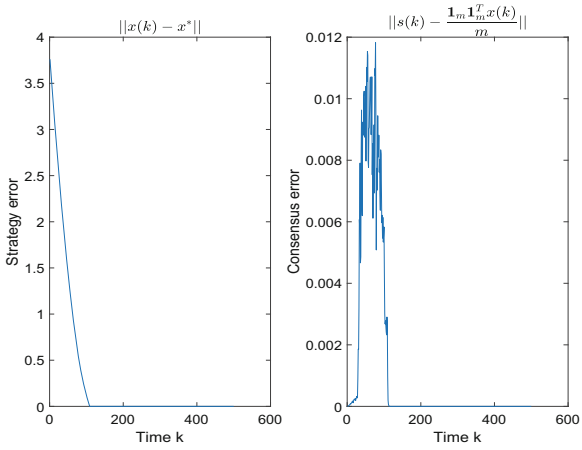

Fig. 1. Trajectory of the strategy and consensus error

6 Conclusion

A distributed discrete multi-round communication algorithm is designed to solve the NE seeking problem for AGs with time-varying directed communication networks in this paper. The common assumption about strongly monotone mapping is relaxed as the restricted strongly monotone one. Based on this, the convergence of the algorithm can be proved and the rate is linear. The work that the communication networks are jointed strongly connected graphs will be investigated in the future.

Acknowledgements. This work was supported in part by the National Natural Science Foundation of China under Grant 62103203 and 61973175; and in part by the Fundamental Research Funds for the Central Universities, Nankai University under Grant 63231230.

References

1. Barrera J, Garcia A (2015) Dynamic incentives for congestion control. IEEE Trans Automat Control 60(2):299–310
2. Pendharkar P (2012) Game theoretical applications for multi-agent systems. Exp Syst Appl 39(1):273–279
3. Hong Y, Wang J, Wang J et al (2022) Cooperative and competitive multi-agent systems: from optimization to games. IEEE/CAA J Automat Sinica 9(5):763–783
4. Marden J, Arslan G, Shamma J (2009) Joint strategy fictitious play with inertia for potential games. IEEE Trans Automat Control 54(2):208–220
5. Semsar-Kazerooni E, Khorasani K (2009) Multi-agent team cooperation: a game theory approach. Automatica 45(10):2205–2213
6. Grammatico S (2017) Dynamic control of agents playing aggregative games with coupling constraints. IEEE Trans Automat Control 62(9):4537–4548
7. Salehisadaghiani F, Pavel L (2016) Distributed Nash equilibrium seeking: a gossip-based algorithm. Automatica 72:209–216
8. Ye M, Hu G (2017) Distributed Nash equilibrium seeking by a consensus based approach. IEEE Trans Automat Control 62(9):4811–4818
9. Tatarenko T, Shi W, Nedic A (2021) Geometric convergence of gradient play algorithms for distributed Nash equilibrium seeking. IEEE Trans Automat Control 66(11):5342–5353
10. Liang S, Yi P, Hong Y (2017) Distributed Nash equilibrium seeking for aggregative games with coupled constraints. Automatica 85:179–185
11. Huang S, Lei J, Hong Y (2023) A linearly convergent distributed Nash equilibrium seeking algorithm for aggregative games. IEEE Trans Automat Control 68(3):1753–1759
12. Fang X, Wen G, Huang T, Fu Z, Hu L (2022) Distributed Nash equilibrium seeking over Markovian switching communication networks. IEEE Trans Cybernet 52(6):5343–5355
13. Wang X, Sun X, Andrew R, Liu K (2020) Distributed robust Nash equilibrium seeking for aggregative games under persistent attacks: a hybrid systems approach. Automatica 122:109255
14. Ye M, Hu G, Xie L, Xu S (2022) Differentially private distributed Nash equilibrium seeking for aggregative games. IEEE Trans Automat Control 67(5):2451–2458

15. Belgioioso G, Nedic A, Grammatico S (2021) Distributed generalized Nash equilibrium seeking in aggregative games on time-varying networks. IEEE Trans Automat Control 66(5):2061–2075
16. Fang X, Wen G, Zhou J, Lü J, Chen G (2022) Distributed Nash equilibrium seeking for aggregative games with directed communication graphs. IEEE Trans Circuits Syst I: Regular Papers 69(8):3339–3352
17. Facchinei F, Pang J-S (eds) (2004) Finite-Dimensional Variational Inequalities and Complementarity Problems. Springer, New York. https://doi.org/10.1007/b97543
18. Bianchi M, Grammatico S (2021) Fully distributed Nash equilibrium seeking over time-varying communication networks with linear convergence rate. IEEE Control Syst Lett 5(2):499–504
19. Yi P, Pavel L (2019) An operator splitting approach for distributed generalized Nash equilibria computation. Automatica 102:111–121

Decentralized Alternating Direction Method of Multipliers for Constrained Optimization over Directed Networks

Jing Yan[1], Xinli Shi[1(✉)], Luyao Guo[2], and Ying Wan[2]

[1] School of Cyber Science and Engineering, Southeast University, Nanjing 210096, China
xinli_shi@seu.edu.cn

[2] School of Mathematics, Southeast University, Nanjing 210096, China

Abstract. In this paper, we consider the decentralized constrained optimization problem in which the objective is to minimize the sum of convex functions subject to equality and set constraints over a directed network. To tackle the optimization problem, we introduce a new algorithm that integrates finite-time weighted average consensus with the Alternating Direction Method of Multipliers (ADMM). Most decentralized optimization algorithms for solving this problem over directed networks use a column-stochastic weight matrix, which necessitates that each agent be aware of its own out-degree. However, the proposed algorithm eliminates this requirement but uses a row-stochastic weight matrix. Additionally, the provided algorithm is proven to achieve a sublinear convergence rate. Finally, the efficacy of our algorithm is confirmed through the numerical simulations performed on a least squares problem subject to local equality and set constraints.

Keywords: Decentralized constrained optimization · directed networks · convergence rate · alternating direction method of multipliers (ADMM)

1 Introduction

In this article, we explore the decentralized constrained optimization (DCO) problem defined in a directed network, where N agents collaborate to find the optimal solution. The DCO problem can be described as follows:

$$\min_{x\in\mathbb{R}^n} \sum_{i=1}^{N} f_i(x) \quad \text{s.t. } A_i x = b_i,\ i = 1,\ldots,N,\ x \in \bigcap_{i=1}^{N} \mathcal{X}_i, \tag{1}$$

This work was supported by the National Natural Science Foundation of China under Grant No. 62106042, and the Natural Science Foundation of Jiangsu Province of China under Grant No. BK20210211.

Y. Hua et al. (Eds.): CCSICC 2023, LNEE 1203, pp. 12–24, 2024.
https://doi.org/10.1007/978-981-97-3324-8_2

where the local function $f_i : \mathbb{R}^n \to \mathbb{R}$ is convex, closed, proper and known only by agent i; $A_i x = b_i$ with $A_i \in \mathbb{R}^{m\times n}$, $b_i \in \mathbb{R}^m$ is the local equality constraint and $\mathcal{X}_i$ represents the local convex constraint set related to the decision variables of agent i. The objective of this work is to devise a decentralized algorithm that allows every agent to obtain a consensus optimal solution to problem (1) through local computation and information exchange with its neighbor agents. Such decentralized optimization problems are common in many engineering fields, including machine learning [1,2], and energy system control [3].

Decentralized optimization methods have been extensively investigated in literature. To tackle the decentralized optimization problem over directed networks, [4] proposes the subgradient-push algorithm. Building upon the Alternating Direction Method of Multipliers (ADMM) [5], the authors in [6,7] propose D-DistADMM and D-ADMM-FTERC, respectively. However, the above methods are only applicable to unconstrained optimization problems. To solve DCO problems, several continuous- and discrete-time algorithms have been proposed [8–10]; however, these algorithms are designed over undirected networks. In [11], DC-DistADMM is proposed to solve the DCO problem in directed networks based on a column-stochastic weight matrix. DC-DistADMM achieves an ergodic sublinear convergence rate of $O(1/k)$ under the general convexity assumption. However, DC-DistADMM relies on the information of each agent's out-degree, which is unrealistic within the broadcasting environment.

In this article, to solve problem (1) over directed networks, we propose a Directed Constrained Decentralized ADMM (DCD-ADMM), which utilizes a row-stochastic weight matrix and thus can be adapted to more communication environments than the algorithms that use column-stochastic weight matrices. Compared to the previously proposed algorithms using column-stochastic weight matrices, such as [4,6,7,11], DCD-ADMM does not require agents to possess information about their own out-degree to construct the column-stochastic matrices. This is more practical in networks based on broadcast communication. In addition, during the communication process, each agent has to transmit two variables to its neighbors in DC-DistADMM [11]. However, in DCD-ADMM, only one variable needs to be transmitted, thereby reducing the communication cost. We prove that DCD-ADMM converges to an exact optimal solution and achieves a sublinear convergence rate of $O(1/k)$.

The rest of this article is structured by: In Sect. 2, necessary notations and preliminaries for the algorithm development and analysis are presented. In Sect. 3, we propose the DCD-ADMM algorithm. In Sect. 4, the convergence analysis and rate of DCD-ADMM are shown. In Sect. 5, we illustrate the numerical simulation results of DCD-ADMM. Finally, in Sect. 6, we conclude the article.

2 Notations and Preliminaries

2.1 Notations

A^T represents the transpose of matrix A. Let $e_i = [0, \ldots, 1_i, \ldots, 0]^T \in \mathbb{R}^N$. The all-ones vector and identity matrix of appropriate dimensions are denoted by $\mathbf{1}$

and I, respectively. The operator $\otimes$ denotes the Kronecker product. $\|\cdot\|$ denotes the l_2-norm. For $x_i \in \mathbb{R}^n$, $i = 1, 2, \ldots, N$, we define $\text{col}\{x_1, x_2, \ldots, x_N\} = [x_1^T, x_2^T, \ldots, x_N^T]^T$. $Z \succ 0$ denotes that the matrix Z is positive definite. Given $Z \succ 0$ and $y \in \mathbb{R}^n$, we denote $\|y\|_Z = \sqrt{\langle y, Zy \rangle}$. The indicator function of a set S is defined as $\mathcal{I}_S(y) = 0$ if $y \in S$, and $\mathcal{I}_S(y) = +\infty$, otherwise.

Let $\mathcal{G}(\mathcal{V}, \mathcal{E})$ denote a directed graph, where $\mathcal{V} = \{1, 2, \ldots, N\}$ is the set of agents, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of directed edges. If a directed edge $(i, j) \in \mathcal{E}$ exists, it signifies that agent j can send information directly to agent i, where j is treated as an in-neighbor of i and, conversely, i is considered as an out-neighbor of j. Define $\mathcal{N}_i^{in} = \{j \in \mathcal{V} | (i, j) \in \mathcal{E}\}$ as the set of in-neighbors of i. The directed graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is called strongly connected if a directed path can be found from any agent i to any other agent j, where $i, j \in \mathcal{V}$, and $i \neq j$.

2.2 Weighted Average Consensus

In a directed graph $G(\mathcal{V}, \mathcal{E})$ with N agents, the non-negative matrix $R = [r_{ij}] \in \mathbb{R}^{N \times N}$ is the weighted adjacency matrix related to the graph G, defined as $r_{ij} > 0$ if $(i, j) \in \mathcal{E}$ or $i = j$, and $r_{ij} = 0$, otherwise. For agent $i \in \mathcal{V}$, we denote u_i^k as the local information of agent i at time k, and each agent i has an initial information u_i^0. Each agent updates its information by the following update:

$$u_i^{k+1} = r_{ii} u_i^k + \sum_{j \in \mathcal{N}_i^{in}} r_{ij} u_j^k, \ k \geq 0. \tag{2}$$

Assumption 1. *The directed graph G is strongly connected and the related weighted adjacency matrix R is row-stochastic.*

Lemma 1 *([12]). Under Assumption 1, let $\{u_i^k\}_{k \geq 0}$ denote the sequence generated by (2) at each agent $i \in \mathcal{V}$. Define $\bar{u} = \sum_{j=1}^N \pi_j u_j^0$, where $[\pi_1, \pi_2, \ldots, \pi_N]^T \in \mathbb{R}^N$ is the normalized left eigenvector of the matrix R related to the eigenvalue 1. Then, u_i^k asymptotically converges to $\bar{u}$ for all $i \in \mathcal{V}$, i.e., $\lim_{k \to \infty} u_i^k = \bar{u} = \sum_{j=1}^N \pi_j u_j^0$, $\forall i \in \mathcal{V}$.*

2.3 Finite-Time Consensus

In the following, we briefly describe the decentralized algorithm proposed in [13,14], with which each agent can obtain $\bar{u} = \sum_{j=1}^N \pi_j u_j^0$ after a finite number of steps.

Define the vector $u_{2k}^T = [u_i^0, u_i^1, \ldots, u_i^{2k}]$ as $2k+1$ successive values at agent i and define the associated matrix:

$$\Gamma\{u_{2k}^T\} = \begin{bmatrix} u_i^0 & u_i^1 & \ldots & u_i^k \\ u_i^1 & u_i^2 & \ldots & u_i^{k+1} \\ \vdots & \vdots & \ddots & \vdots \\ u_i^k & u_i^{k+1} & \ldots & u_i^{2k} \end{bmatrix}.$$

The vector that represents the differences between successive values of u_i^{2k} is expressed as $\bar{u}_{2k}^T = [u_i^1 - u_i^0, u_i^2 - u_i^1, \ldots, u_i^{2k+1} - u_i^{2k}]$. By increasing the dimension k of the matrix $\Gamma\{\bar{u}_{2k}^T\}$ until it loses rank, we obtain $\Gamma\{\bar{u}_{2k_i^o}^T\}$ as the first defective matrix of agent i. It has been shown in [13] that for any initial condition, the kernel $\boldsymbol{\beta}_i = [\beta_i^0, \beta_i^1, \ldots, \beta_i^{M_i}]^T$ of the matrix $\Gamma\{\bar{u}_{2k_i^o}^T\}$, where $M_i = k_i^o$, can be utilized to calculate the weighted average consensus, except a set of initial conditions with Lebesgue measure zero. Then, $\bar{u}$ can be computed by

$$\phi_u(i) = \frac{u_{M_i}^T \boldsymbol{\beta}_i}{\mathbf{1}^T \boldsymbol{\beta}_i} = \bar{u}, \tag{3}$$

where $u_{M_i}^T = [u_i^0, u_i^1, \ldots, u_i^{M_i}]$.

Lemma 2 (*[13]*). *Let Assumption 1 hold, and* $\{u_i^k\}_{k\geq 0}$ *denote the sequence generated by (2) at each agent* $i \in \mathcal{V}$. *The weighted average consensus* $\bar{u} = \sum_{j=1}^N \pi_j u_j^0$ *can be distributively obtained by (3) at each agent* $i \in \mathcal{V}$ *within a finite number of steps no greater than* $2N$.

2.4 Max-Consensus Algorithm

The maximum value in the network can be calculated in a decentralized manner by the max-consensus algorithm [15]. Each agent $i \in \mathcal{V}$ has an initial state $x_i^0 \in \mathbb{R}$ and updates its state by $x_i^{k+1} = \max_{j \in \mathcal{N}_i^{in} \cup \{i\}} \{x_j^k\}$. All states converge to the maximum value $x_{\max} = \max\{x_1^0, \ldots, x_N^0\}$ within a finite number of steps no greater than N [16].

3 DCD-ADMM Algorithm Development

Consider a directed network $G(\mathcal{V}, \mathcal{E})$ of N agents. Let $R = [r_{ij}] \in \mathbb{R}^{N\times N}$ be the weighted adjacency matrix correlated to G. Under Assumption 1, the weighted matrix R satisfies the properties of being irreducible and row-stochastic, with positive diagonals. By the Perron-Frobenius theorem [17], one has that there exists a strictly positive left eigenvector $\pi = [\pi_1, \pi_2, \ldots, \pi_N] \in \mathbb{R}^N$ (i.e., $\pi_i > 0,\ \forall i \in \mathcal{V}$) of R related to the eigenvalue 1 such that $\pi^T \mathbf{1}_N = 1$.

To solve problem (1) in a distributed way, we introduce local copies $x_i \in \mathbb{R}^n$ of the global decision variable x for all $i \in \mathcal{V}$ and impose the consensus constraint $x_i = x_j,\ \forall i, j \in \mathcal{V}$ to ensure that the local copies of every agent are identical. Then, problem (1) is equivalent to

$$\min \sum_{i=1}^N f_i(x_i) \ \text{ s.t. } A_i x_i = b_i,\ x_i \in \mathcal{X}_i,\ \forall i \in \mathcal{V},\ x_i = x_j,\ \forall i, j \in \mathcal{V}. \tag{4}$$

Define the agreement subspace $\mathcal{C} = \{[y_1^T, y_2^T, \ldots, y_N^T]^T \in \mathbb{R}^{Nn} |\ y_i = y_j, 1 \leq i, j \leq N\}$. Let $\bar{\pi} = [\pi_1^{\frac{1}{2}}, \pi_2^{\frac{1}{2}}, \ldots, \pi_N^{\frac{1}{2}}] \in \mathbb{R}^N$, $\Pi = \text{diag}\{\pi\} \otimes I_n$ and $\bar{\Pi} = \text{diag}\{\bar{\pi}\} \otimes I_n$. Using the indicator function $\mathcal{I}_{\mathcal{X}_i}$ of the set $\mathcal{X}_i$, (4) transforms into

$$\min \sum_{i=1}^{N}(f_i(x_i)+\mathcal{I}_{\mathcal{X}_i}(x_i)) \ \text{ s.t. } A_i x_i = b_i,\ \forall i \in \mathcal{V},\ \bar{\Pi}(\mathbf{x}-\mathbf{y})=0,\ \mathbf{y}\in\mathcal{C}, \quad (5)$$

where $\mathbf{x}=[x_1^T, x_2^T, \ldots, x_N^T]^T \in \mathbb{R}^{Nn}$, $\mathbf{y}=[y_1^T, y_2^T, \ldots, y_N^T] \in \mathbb{R}^{Nn}$. Then, utilizing the indicator function $\mathcal{I}_{\mathcal{C}}$ of the set $\mathcal{C}$, (5) becomes

$$\min F(\mathbf{x})+\mathcal{I}_{\mathcal{C}}(\mathbf{y}) \ \text{ s.t. } \mathbf{A}\mathbf{x}=\mathbf{b},\ \bar{\Pi}(\mathbf{x}-\mathbf{y})=0, \quad (6)$$

where $F(\mathbf{x})=\sum_{i=1}^{N}(f_i(x_i)+\mathcal{I}_{\mathcal{X}_i}(x_i))$, $\mathbf{A}=\text{blkdiag}(A_1, A_2, \ldots, A_N) \in \mathbb{R}^{Nm\times Nn}$, and $\mathbf{b}=[b_1^T, b_2^T, \ldots, b_N^T]^T \in \mathbb{R}^{Nm}$.

The augmented Lagrangian function of (6) is

$$\begin{aligned}\mathcal{L}_c(\mathbf{x},\mathbf{y},\lambda,\mu) =& F(\mathbf{x})+\mathcal{I}_{\mathcal{C}}(\mathbf{y})+\lambda^T\bar{\Pi}(\mathbf{x}-\mathbf{y})+\mu^T(\mathbf{A}\mathbf{x}-\mathbf{b})\\ &+\frac{c}{2}\|\bar{\Pi}(\mathbf{x}-\mathbf{y})\|^2+\frac{c}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2, \end{aligned} \quad (7)$$

where $\lambda \in \mathbb{R}^{Nn}$ and $\mu \in \mathbb{R}^{Nm}$ are dual variables, and $c>0$ is a constant. Based on ADMM, the primal and dual updates are respectively given as

$$\begin{aligned}\mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}}\{F(\mathbf{x})+(\lambda^k)^T\bar{\Pi}(\mathbf{x}-\mathbf{y}^k)+(\mu^k)^T(\mathbf{A}\mathbf{x}-\mathbf{b})\\ &\qquad +\frac{c}{2}\|\bar{\Pi}(\mathbf{x}-\mathbf{y}^k)\|^2+\frac{c}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2\}\\ &= \arg\min_{\mathbf{x}}\{F(\mathbf{x})+\frac{c}{2}\|\bar{\Pi}(\mathbf{x}-\mathbf{y}^k)+\frac{1}{c}\lambda^k\|^2+\frac{c}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}+\frac{1}{c}\mu^k\|^2\}, \quad (8a)\\ \mathbf{y}^{k+1} &= \arg\min_{\mathbf{y}}\{\mathcal{I}_{\mathcal{C}}(\mathbf{y})+(\lambda^k)^T\bar{\Pi}(\mathbf{x}^{k+1}-\mathbf{y})+\frac{c}{2}\|\bar{\Pi}(\mathbf{x}^{k+1}-\mathbf{y})\|^2\}\\ &= \arg\min_{\mathbf{y}}\{\mathcal{I}_{\mathcal{C}}(\mathbf{y})+\frac{c}{2}\|\bar{\Pi}(\mathbf{x}^{k+1}-\mathbf{y})+\frac{1}{c}\lambda^k\|^2\}, \quad (8b)\\ \lambda^{k+1} &= \lambda^k+c\bar{\Pi}(\mathbf{x}^{k+1}-\mathbf{y}^{k+1}), \quad (8c)\\ \mu^{k+1} &= \mu^k+c(\mathbf{A}\mathbf{x}^{k+1}-\mathbf{b}). \quad (8d)\end{aligned}$$

One can derive an explicit expression for the optimal solution $\mathbf{y}^{k+1}$ of (8b), i.e., $\mathbf{y}^{k+1}=[(y^{k+1})^T,\ldots,(y^{k+1})^T]^T \in \mathbb{R}^{Nn}$, where $y^{k+1}=\sum_{i=1}^{N}\pi_i(x_i^{k+1}+\frac{1}{c\bar{\pi}_i}\lambda_i^k)$. The proof is omitted here as it closely resembles the one presented in Appendix B of [18]. Thus, (8b) reduces to a weighted average consensus problem.

Since the normalized left eigenvector π of the matrix R is used in the algorithm development, we need a preprocessing process to compute π at each node in a distributed manner, as shown in Algorithm 1. The preprocessing process performs $2N+N$ iterations, of which $2N$ iterations are used for the finite-time consensus algorithm to calculate the normalized left eigenvector π and β_i, and N iterations are used for the max-consensus algorithm to calculate $t_{\max}$ at each agent $i\in\mathcal{V}$. After the preprocessing process, each node $i\in\mathcal{V}$ obtains π, β_i and $t_{\max}$. Finally, we give DCD-ADMM in Algorithm 2.

4 Convergence Analysis

In this section, we analyze the convergence of DCD-ADMM and prove that it achieves an ergodic sublinear convergence rate of $O(\frac{1}{k})$.

Algorithm 1: Preprocessing process

Input:
Weighted adjacency matrix R, number of nodes N.

Initialize:
$s_i = e_i \in \mathbb{R}^N$, $flag_i = 1$, $\forall i \in \mathcal{V}$.

`/* finite-time consensus */`

for $k = 0, 1, 2, \ldots, 2N-1$, *each node* $i = 1, ..., N$, *(In parallel)* **do**
  $s_i^{k+1} = r_{ii} s_i^k + \sum_{j \in \mathcal{N}_i^{in}} r_{ij} s_j^k$
  if k *is even and* $flag_i = 1$ **then**
    $t = k/2$
    $\bar{s}_{2t}^T = [s_i^1 - s_i^0, \ldots, s_i^{2t+1} - s_i^{2t}]$
    if $\Gamma\{\bar{s}_{2t}^T\}$ *loses rank* **then**
      $M_i = t$
      Compute the kernel $\boldsymbol{\beta}_i$ of the matrix $\Gamma\{\bar{s}_{2t}^T\}$
      $flag_i = 0$
    end
  end
end

$s_{M_i}^T = [s_i^0, \ldots, s_i^{M_i}] \in \mathbb{R}^{N \times (M_i+1)}$ and $\pi = s_i^* = \frac{s_{M_i}^T \boldsymbol{\beta}_i}{\mathbf{1}^T \boldsymbol{\beta}_i}$, $\forall i \in \mathcal{V}$

`/* max-consensus */`

$t_i^0 = M_i$, $\forall i \in \mathcal{V}$
for $k = 0, 1, 2, \ldots, N-1$, *each node* $i = 1, ..., N$, *(In parallel)* **do**
  $t_i^{k+1} = \max_{j \in \mathcal{N}_i^{in} \cup \{i\}} \{t_j^k\}$
end
$t_{\max} = t_i^N$

4.1 Convergence Analysis of DCD-ADMM

In this subsection, we provide a convergence proof of DCD-ADMM, demonstrating that it converges to the exact optimal solution.

The Lagrange function of (6) is

$$\mathcal{L}(\mathbf{x}, \mathbf{y}, \lambda, \mu) = F(\mathbf{x}) + \mathcal{I}_{\mathcal{C}}(\mathbf{y}) + \lambda^T \bar{\Pi}(\mathbf{x} - \mathbf{y}) + \mu^T(\mathbf{A}\mathbf{x} - \mathbf{b}). \tag{9}$$

Assumption 2. *The Lagrange function* $\mathcal{L}$ *has a saddle point* $(\mathbf{x}^*, \mathbf{y}^*, \lambda^*, \mu^*)$, *i.e., for all* $\mathbf{x} \in \mathbb{R}^{Nn}$, $\mathbf{y} \in \mathbb{R}^{Nn}$, $\lambda \in \mathbb{R}^{Nn}$ *and* $\mu \in \mathbb{R}^{Nm}$,

$$\mathcal{L}(\mathbf{x}^*, \mathbf{y}^*, \lambda, \mu) \leq \mathcal{L}(\mathbf{x}^*, \mathbf{y}^*, \lambda^*, \mu^*) \leq \mathcal{L}(\mathbf{x}, \mathbf{y}, \lambda^*, \mu^*) \tag{10}$$

holds.

In accordance with the Karush-Kuhn-Tucker (KKT) conditions of (6), $(\mathbf{x}^*, \mathbf{y}^*, \lambda^*, \mu^*)$ satisfies

Algorithm 2: DCD-ADMM Algorithm

Input:
Weighted adjacency matrix R.
Initialize:
$x_i^0 \in \mathbb{R}^n$, $y_i^0 \in \mathbb{R}^n$, $\lambda_i^0 = \mathbf{0}_n$, $\mu_i^0 \in \mathbb{R}^m$, $\forall i \in \mathcal{V}$.
for $k = 0, 1, 2, \ldots,$ *each node* $i = 1, ..., N$, *(In parallel)* **do**
$x_i^{k+1} =$
$\arg\min_{x_i}\{f_i(x_i) + \mathcal{I}_{\mathcal{X}_i}(x_i) + \frac{c}{2}\|\bar{\pi}_i(x_i - y_i^k) + \frac{1}{c}\lambda_i^k\|^2 + \frac{c}{2}\|A_i x_i - b_i + \frac{1}{c}\mu_i^k\|^2\}$
$z_i^0 = x_i^{k+1} + \frac{1}{c\bar{\pi}_i}\lambda_i^k$
for $t = 0, 1, 2, \ldots, t_{\max}$ **do**
$z_i^{t+1} = r_{ii} z_i^t + \sum_{j \in \mathcal{N}_i^{in}} r_{ij} z_j^t$
end
$y_i^{k+1} = \frac{z_{M_i}^T \boldsymbol{\beta}_i}{\mathbf{1}^T \boldsymbol{\beta}_i}$, where $z_{M_i}^T = [z_i^0, \ldots, z_i^{M_i}] \in \mathbb{R}^{n \times (M_i+1)}$
$\lambda_i^{k+1} = \lambda_i^k + c\bar{\pi}_i(x_i^{k+1} - y_i^{k+1})$
$\mu_i^{k+1} = \mu_i^k + c(A_i x_i^{k+1} - b_i)$
end

$$-\bar{\Pi}\lambda^* - \mathbf{A}^T\mu^* \in \partial F(\mathbf{x}^*), \tag{11a}$$
$$\bar{\Pi}\lambda^* \in \partial\mathcal{I}_{\mathcal{C}}(\mathbf{y}^*), \tag{11b}$$
$$\mathbf{A}\mathbf{x}^* - \mathbf{b} = 0, \tag{11c}$$
$$\mathbf{x}^* = \mathbf{y}^*, \tag{11d}$$

where $\partial F(\mathbf{x}^*)$ and $\partial\mathcal{I}_{\mathcal{C}}(\mathbf{y}^*)$ are the set of sub-gradients of F at $\mathbf{x}^*$, and $\mathcal{I}_{\mathcal{C}}$ at $\mathbf{y}^*$, respectively. To simplify the notations in analysis, we define $\mathbf{w}^k = \text{col}\{\mathbf{x}^k, \mathbf{y}^k, \lambda^k, \mu^k\}$, $\mathbf{w}^* = \text{col}\{\mathbf{x}^*, \mathbf{y}^*, \lambda^*, \mu^*\}$, $\mathbf{v}^k = \text{col}\{\mathbf{y}^k, \lambda^k, \mu^k\}$, $\mathbf{v}^* = \text{col}\{\mathbf{y}^*, \lambda^*, \mu^*\}$ and the positive definite matrix $H = \text{diag}\{\frac{c}{2}\Pi, \frac{I}{2c}, \frac{I}{2c}\}$.

Theorem 1. *Given Assumptions 1 and 2, the sequences* $\{\mathbf{x}^k\}$ *and* $\{\mathbf{v}^k\}$ *generated by (8a)-(8d) satisfy*

$$\|\mathbf{v}^{k+1} - \mathbf{v}^*\|_H^2 \leq \|\mathbf{v}^k - \mathbf{v}^*\|_H^2 - \|\mathbf{v}^k - \mathbf{v}^{k+1}\|_H^2. \tag{12}$$

Moreover, $\lim_{k\to\infty} \mathbf{v}^k = \mathbf{v}^*$ *and* $\lim_{k\to\infty} \mathbf{x}^k = \mathbf{x}^*$.

Proof. In accordance with the first-order optimality conditions of subproblems (8a) and (8b), we have

$$-(\bar{\Pi}\lambda^k + \mathbf{A}^T\mu^k + c\Pi(\mathbf{x}^{k+1} - \mathbf{y}^k) + c\mathbf{A}^T(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b})) \in \partial F(\mathbf{x}^{k+1}), \tag{13}$$
$$c\Pi(\mathbf{x}^{k+1} - \mathbf{y}^{k+1}) + \bar{\Pi}\lambda^k \in \partial\mathcal{I}_{\mathcal{C}}(\mathbf{y}^{k+1}). \tag{14}$$

Then, combining (13) with (8c) and (8d), and combining (14) with (8c), we obtain

$$-(\bar{\Pi}\lambda^{k+1} + c\Pi(\mathbf{y}^{k+1} - \mathbf{y}^k) + \mathbf{A}^T\mu^{k+1}) \in \partial F(\mathbf{x}^{k+1}), \tag{15}$$
$$\bar{\Pi}\lambda^{k+1} \in \partial\mathcal{I}_{\mathcal{C}}(\mathbf{y}^{k+1}). \tag{16}$$

Notice that the indicator function of a set is convex, but it is not differentiable. Since F and $\mathcal{I}_{\mathcal{C}}$ are convex functions, we have

$$\begin{aligned}
&F(\mathbf{x}^{k+1}) - F(\mathbf{x}^*) + \mathcal{I}_{\mathcal{C}}(\mathbf{y}^{k+1}) - \mathcal{I}_{\mathcal{C}}(\mathbf{y}^*) \\
&\leq -(\mathbf{x}^* - \mathbf{x}^{k+1})^T dF(\mathbf{x}^{k+1}) - (\mathbf{y}^* - \mathbf{y}^{k+1})^T d\mathcal{I}_{\mathcal{C}}(\mathbf{y}^{k+1}) \\
&= (\mathbf{x}^* - \mathbf{x}^{k+1})^T(\bar{\Pi}\lambda^{k+1} + c\Pi(\mathbf{y}^{k+1} - \mathbf{y}^k) + \mathbf{A}^T\mu^{k+1}) - (\mathbf{y}^* - \mathbf{y}^{k+1})^T\bar{\Pi}\lambda^{k+1} \\
&= (\mathbf{x}^* - \mathbf{x}^{k+1})^T\mathbf{A}^T\mu^{k+1} + c(\mathbf{x}^* - \mathbf{x}^{k+1})^T\Pi(\mathbf{y}^{k+1} - \mathbf{y}^k) + (\mathbf{y}^{k+1} - \mathbf{x}^{k+1})^T\bar{\Pi}\lambda^{k+1},
\end{aligned}$$

where $dF(\mathbf{x}^{k+1})$ can be any sub-gradient of F at $\mathbf{x}^{k+1}$ and $d\mathcal{I}_{\mathcal{C}}(\mathbf{y}^{k+1})$ can be any sub-gradient of $\mathcal{I}_{\mathcal{C}}$ at $\mathbf{y}^{k+1}$; in the first equality we use (15) and (16), and in the last equality we use (11d). Note that $\mathcal{I}_{\mathcal{C}}(\mathbf{y}^{k+1}) = \mathcal{I}_{\mathcal{C}}(\mathbf{y}^*) = 0$. Then, for any λ and μ, utilizing (11c) we obtain

$$\begin{aligned}
&F(\mathbf{x}^{k+1}) - F(\mathbf{x}^*) + \lambda^T\bar{\Pi}(\mathbf{x}^{k+1} - \mathbf{y}^{k+1}) + \mu^T(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}) \\
&\leq (\mu - \mu^{k+1})^T(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}) + (\lambda^{k+1} - \lambda)^T\bar{\Pi}(\mathbf{y}^{k+1} - \mathbf{x}^{k+1}) + c(\mathbf{x}^* - \mathbf{x}^{k+1})^T\Pi(\mathbf{y}^{k+1} - \mathbf{y}^k).
\end{aligned}$$

Then, utilizing (8c) and (8d), we obtain

$$\begin{aligned}
&F(\mathbf{x}^{k+1}) - F(\mathbf{x}^*) + \lambda^T\bar{\Pi}(\mathbf{x}^{k+1} - \mathbf{y}^{k+1}) + \mu^T(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}) \\
&\leq \frac{1}{c}(\mu - \mu^{k+1})^T(\mu^{k+1} - \mu^k) + \frac{1}{c}(\lambda - \lambda^{k+1})^T(\lambda^{k+1} - \lambda^k) + c(\mathbf{x}^* - \mathbf{x}^{k+1})^T\Pi(\mathbf{y}^{k+1} - \mathbf{y}^k) \\
&= \frac{1}{2c}(\|\mu - \mu^k\|^2 - \|\mu - \mu^{k+1}\|^2 - \|\mu^{k+1} - \mu^k\|^2) + \frac{1}{2c}(\|\lambda - \lambda^k\|^2 - \|\lambda - \lambda^{k+1}\|^2 - \|\lambda^{k+1} - \lambda^k\|^2) \\
&\quad + \frac{c}{2}(\|\bar{\Pi}(\mathbf{x}^* - \mathbf{y}^k)\|^2 - \|\bar{\Pi}(\mathbf{x}^* - \mathbf{y}^{k+1})\|^2 + \|\bar{\Pi}(\mathbf{x}^{k+1} - \mathbf{y}^{k+1})\|^2 - \|\bar{\Pi}(\mathbf{x}^{k+1} - \mathbf{y}^k)\|^2) \\
&= \frac{1}{2c}(\|\mu - \mu^k\|^2 - \|\mu - \mu^{k+1}\|^2 - \|\mu^{k+1} - \mu^k\|^2) + \frac{1}{2c}(\|\lambda - \lambda^k\|^2 - \|\lambda - \lambda^{k+1}\|^2 - \|\lambda^{k+1} - \lambda^k\|^2) \\
&\quad + \frac{c}{2}(\|\bar{\Pi}(\mathbf{y}^* - \mathbf{y}^k)\|^2 - \|\bar{\Pi}(\mathbf{y}^* - \mathbf{y}^{k+1})\|^2 - \|\bar{\Pi}(\mathbf{y}^{k+1} - \mathbf{y}^k)\|^2) \\
&\quad - \langle\lambda^{k+1} - \lambda^k, \bar{\Pi}(\mathbf{y}^{k+1} - \mathbf{y}^k)\rangle,
\end{aligned} \tag{17}$$

where in the first equality the identity $(a_1 - a_2)^T(a_3 - a_4) = \frac{1}{2}(\|a_1 - a_4\|^2 - \|a_1 - a_3\|^2) + \frac{1}{2}(\|a_2 - a_3\|^2 - \|a_2 - a_4\|^2)$ is used, and (8c) and (11d) are used in the last equality. As $\mathcal{I}_{\mathcal{C}}$ is convex, we have $(\mathbf{y}_1 - \mathbf{y}_2)^T(d\mathcal{I}_{\mathcal{C}}(\mathbf{y}_1) - d\mathcal{I}_{\mathcal{C}}(\mathbf{y}_2)) \geq 0$, using (16) and setting $\mathbf{y}_1 = \mathbf{y}^{k+1}$ and $\mathbf{y}_2 = \mathbf{y}^k$, then we obtain

$$\langle\bar{\Pi}(\lambda^{k+1} - \lambda^k), \mathbf{y}^{k+1} - \mathbf{y}^k\rangle \geq 0. \tag{18}$$

Since $\mathcal{I}_{\mathcal{C}}(\mathbf{y}^{k+1}) = \mathcal{I}_{\mathcal{C}}(\mathbf{y}^*) = 0$, with (11c) and (11d), one has

$$\begin{aligned}
&F(\mathbf{x}^{k+1}) - F(\mathbf{x}^*) + (\lambda^*)^T\bar{\Pi}(\mathbf{x}^{k+1} - \mathbf{y}^{k+1}) + (\mu^*)^T(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}) \\
&= \mathcal{L}(\mathbf{x}^{k+1}, \mathbf{y}^{k+1}, \lambda^*, \mu^*) - \mathcal{L}(\mathbf{x}^*, \mathbf{y}^*, \lambda^*, \mu^*) \geq 0.
\end{aligned} \tag{19}$$

When $\lambda = \lambda^*$ and $\mu = \mu^*$, substituting (18) and (19) into (17), we get

$$\begin{aligned}
&\frac{c}{2}\|\mathbf{y}^{k+1} - \mathbf{y}^*\|_{\Pi}^2 + \frac{1}{2c}\|\lambda^{k+1} - \lambda^*\|^2 + \frac{1}{2c}\|\mu^{k+1} - \mu^*\|^2 \\
&\leq \frac{c}{2}(\|\mathbf{y}^k - \mathbf{y}^*\|_{\Pi}^2 - \|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\Pi}^2) + \frac{1}{2c}(\|\lambda^k - \lambda^*\|^2 - \|\lambda^k - \lambda^{k+1}\|^2) \\
&\quad + \frac{1}{2c}(\|\mu^k - \mu^*\|^2 - \|\mu^k - \mu^{k+1}\|^2).
\end{aligned} \tag{20}$$