MACRO: Incentivizing Multi-leader Game-based Pareto-efficient Crowdsourcing for Video Analytics

Yu Chen[1], Sheng Zhang*[1], Ziying Zhou[2], Xiaokun Wang[1], Yu Liang[3], Ning Chen[1] and Yuting Yan[1],
Mingjun Xiao[4], Jie Wu[5], Zhuzhong Qian[1], Harry Xu[6]
[1]State Key Lab. for Novel Software Technology, Nanjing University, P.R. China
[2]School of Systems Science, Beijing Normal University, China
[3]School of Computer and Electronic Information / School of Artificial Intelligence, Nanjing Normal University, China
[4]School of Computer Science and Technology, University of Science and Technology of China, P.R. China
[5]Center for Networked Computing, Temple University, USA
[6]Computer Science Department, Samueli School of Engineering, University of California, Los Angeles, USA

***Abstract*—In recent years, many crowdsourcing platforms have emerged, using the resources of recruited workers to perform diverse outsourcing tasks, where the video analytics attracts much attention due to its practical implications. For maximum profits, platforms carefully choose the workers and determine the video analytics configurations to ensure accuracy; meanwhile, workers possess the flexibility to tailor the configurations for their individual gains, which makes it hard for platforms to optimize their profits considering the platform-worker conflicts. In this paper, we design an incentive mechanism for Multi-leader game-based video Analytics upon CROwdsourcing, named MACRO, to overcome the above situation. Under that mechanism, we first formulate the utility optimization problems for platforms and workers, respectively. We then propose a dual ascent-based method to optimally determine the video analytics configurations for a multi-platform game, ensuring Pareto efficiency. Moreover, in the context of a multi-leader game involving platform-worker conflicts, we design an incentive function with its incentive factor update strategy and propose an ADMM-based approach for maximizing incentives that motivate workers to contribute to the platforms' profits. Rigorous proofs demonstrate the linear convergence of the MACRO to the multi-leader Stackelberg equilibrium. Trace-driven experiments show that MACRO improves the Pareto efficiency by 26.3%, outperforming other approaches.**

I. Introduction

In recent years, mobile crowdsourcing [1–3] has emerged as a prominent paradigm, resulting in the proliferation of various platforms (e.g., CrowdFlower [4], AMT [5]). They publish diverse tasks, including mobile sensing [6], traffic prediction [7] and image labeling [8], and employ a large number of crowdsourcing workers with mobile devices to complete them. The low cost [9], high flexibility [10], and extensive coverage [11] provided by mobile crowdsourcing have spurred numerous research efforts, covering aspects such as data aggregation [12, 13], worker recruitment [14, 15], and task assignment [16, 17].

Meanwhile, for crowdsourcing workers, the performance of mobile devices (e.g., smartphones, tablets, and laptops) is continuously improving [18, 19], and computer vision technologies [20, 21] are evolving steadily. Benefiting from the aforementioned trends, among various crowdsourcing tasks, video analytics [22–24], such as object classification, content understanding, and behavior recognition has garnered much attention [25, 26] due to its practical implications. To complete the video analytics tasks, platforms have been actively engaging in hiring proper workers, sending video data to them, and strategically selecting the configurations [27] (i.e., frame rates, resolutions and models) to maximize their accuracy-based profits.

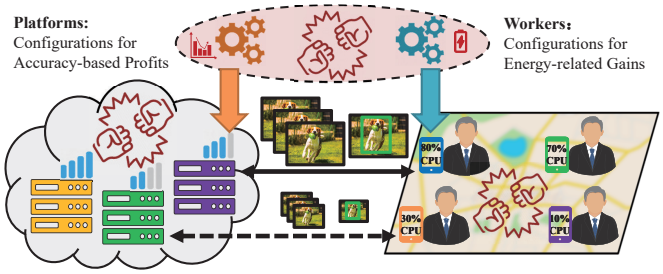


Fig. 1. Conflicts among Platforms, among Workers, and between Platforms and Workers in Video Analytics Crowdsourcing Scenarios

Additionally, workers possess the flexibility [28] to accept the video analytics tasks or not. Upon receiving the tasks, they utilize their mobile devices equipped with video analytics models [29, 30] to complete them. During the task execution process, workers may tailor the configurations for their individual gains. Specifically, different video analytics configurations result in varying energy consumption (e.g., for transmission and computation) [31, 32] on their mobile devices, based on which workers are compensated with corresponding gains [25, 26]. Notably, conflicts can emerge between platforms and workers in determining video analytics configurations, due to their distinct optimization goals. Therefore, as demonstrated in Fig. 1, it is nontrivial for platforms to optimally choose the recruited workers and determine the video analytics configurations for maximum accuracy-based profits while considering the workers' energy-related gains. We will face these challenges:

Firstly, due to the heterogeneity of the video analytics upon crowdsourcing, it is hard to appropriately decide on the configurations to maximize task completion benefits. As shown in

*The corresponding author is Sheng Zhang (sheng@nju.edu.cn). This work was supported in part by NSFC (62202233), Double Innovation Plan of Jiangsu Province (JSSCBS20220409), the Science and Technology Project of State Grid Co., LTD (Research on data aggregation and dynamic interaction technology of enterprise-level real-time measurement data center, 5108-202218280A-2-399-XG), Collaborative Innovation Center of Novel Software Technology and Industrialization, and Postgraduate Research & Practice Innovation Program of Jiangsu Province (KYCX23_0158).

our case studies later, the variation in video data and task types among platforms introduces complexity to the optimization of their profits based on video analytics accuracy. Moreover, platforms' bandwidth budgets [33] further complicate the pursuit of accuracy-related profits, requiring a delicate trade-off between accurate results and efficient data transmission. In addition, considering the benefits for workers, it is essential to pay attention to their diverse computing capabilities and choices of configurations for video analytics. Besides, the energy consumption, directly determined by the configurations on workers' mobile devices, significantly impacts their compensation gains [25, 26], necessitating its inclusion for maximum benefits.

Secondly, since the computation resources (e.g., CPU/GPU) of the workers' mobile devices are limited [34], conflicts arise among platforms when aiming to maximize the profits. Specifically, multiple platforms may hire the same worker for video analytics, but the worker's capacity constraints may hinder the platforms from achieving their desired optimal configurations. Furthermore, the above mentioned configuration of each platform is multi-dimensional, corresponding to multiple workers, which makes it more challenging to simultaneously optimize all platforms' objectives. Typically, the number of workers is quite large compared to that of platforms, which can result in significant computational overhead when determining the configurations [35]. The conflicts and computational complexities highlight the need for innovative approaches to effectively optimize the video analytics across multiple platforms.

Last but not least, the conflict between platforms and workers makes it fairly intractable to simultaneously optimize their respective benefits. Despite the success of profit maximization across all platforms, the optimal video analytics configurations for platforms may not align with those for the workers [36]. Thus, workers may prioritize their own interests over the platforms' and shift configuration decisions to maximize their own gains, which is detrimental to the platforms. The relationship between each platform and workers resembles that in single-leader multi-follower Stackelberg game [37], which is used to address the conflicts between the leader and followers. However, our scenario involves multiple platforms that compete for the workers, and thus the conventional Stackelberg game method, suitable for a single platform, does not work. Hence, an incentive-based mechanism is required to motivate the workers to contribute to maximizing the platforms' profits.

However, existing works have certain limitations in tackling the aforementioned challenges. Some studies [22–24] focus on optimizing video analytics configurations, but their approaches are not directly applicable to addressing those conflicts between platforms and workers for video analytics. Other research [25, 38–40] studies the selection of suitable workers for crowdsourcing tasks, but few consider the complexities of simultaneously recruiting workers for video analytics from multiple platforms. The remaining studies [28, 36, 41, 42] explore incentive mechanisms with game theory, but they do not account for the difficulty of efficiently optimizing incentive decisions when dealing with a substantial number of workers.

In this paper, we design an incentive mechanism for Multi-leader game-based video Analytics upon CROwdsourcing, denoted as MACRO, which addresses the above challenges. Under mechanism MACRO, we firstly formulate the utility optimization problems for platforms and workers, respectively. For each platform, its profit mainly depends on the video analytics accuracy while the configurations, such as frame rates, should be constrained by bandwidth budget. For each worker, determining the optimal video analytics configuration is also essential to maximize its energy consumption-related gains, which, however, is limited by the availability of computing resources.

TABLE I
RELATED WORKS CONSIDERING THE CONFLICTS AMONG WORKERS, AMONG PLATFORMS, AND BETWEEN WORKERS AND PLATFORMS.

Related Works	Workers	Platforms	Platform&Worker	Task Category
MCTA[2]	√	√	×	Mobile Sensing
ME-UCB[38]	√	×	×	Truth Discovery
LOL[25]	√	×	×	Video Analytics
AIAI[28]	√	×	√	Mobile Sensing
DDIM[41]	√	√	√	Mobile Sensing
Crowd^2[26]	√	√	×	Video Analytics
MACRO	√	√	√	Video Analytics

In order to achieve the goal of maximizing platforms' profits while taking into account the gains of workers, we analyze and tackle the inter-platform and platform-worker conflicts for heterogeneous video analytics tasks upon crowdsourcing. We propose a dual ascent-based method to determine appropriate video analytics configurations for a multi-platform game, ensuring Pareto efficiency. Moreover, in the context of a multi-leader game involving platform-worker conflicts, we design an incentive function with its incentive factor updating strategy, and propose an ADMM-based incentive maximization method that motivates workers to contribute to the platforms' Pareto efficiency. Rigorous proofs show that MACRO exhibits linear convergence to the multi-leader Stackelberg equilibrium.

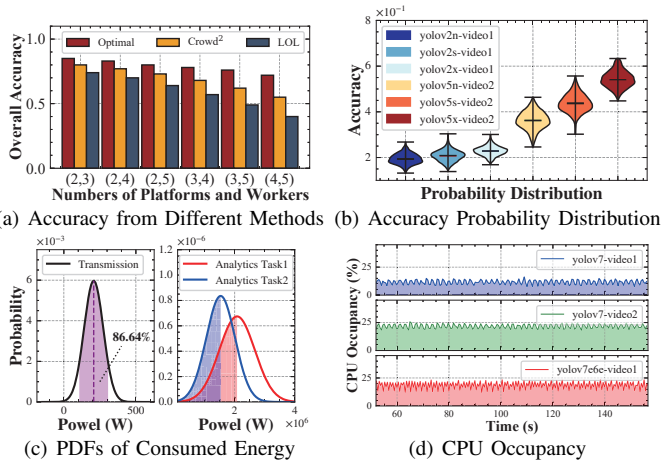
Extensive trace-driven experiments conducted on the video dataset PANDA [43] show MACRO's superiority in achieving the multi-leader Stackelberg equilibrium. Concretely, MACRO improves the Pareto efficiency for yolov7-based video analytics crowdsourcing by 26.3% on average, outperforming others. To summarize, we make the following contributions:

- To our knowledge, this is the first work to optimize the crowdsourcing utilities for video analytics scenarios, simultaneously considering the conflicts among platforms, among workers, and between platforms and workers.
- We design an ADMM-based incentive mechanism called MACRO, coordinating the video analytics configurations for platforms and workers, which reaches the multi-leader Stackelberg equilibrium and ensures Pareto efficiency.
- Trace-driven experiments evaluate the effectiveness, efficiency, scalability, and overhead of MACRO, comparing it with other existing works.

II. SYSTEM MODEL AND PROBLEM FORMULATION

A. Background and Motivation

Crowdsourcing platforms recruit workers to accomplish various crowdsourcing tasks, such as mobile sensing [2, 28, 41], truth discovery [38], and video analytics [25, 26]. As shown in Table I, existing research has considered different conflicts in



(a) Accuracy from Different Methods (b) Accuracy Probability Distribution

(c) PDFs of Consumed Energy (d) CPU Occupancy

Fig. 2. Motivation and Case Studies for Mechanism MACRO

crowdsourcing scenarios involving three categories as follows. *Conflicts among workers* [25, 38] are the competitions among workers when platforms recruit them for task completion, and some workers may not be selected. *Conflicts among platforms* [2, 26] arise when multiple platforms face the challenge of simultaneously selecting the same workers for the limited computational resources. *Platform-worker conflicts* [28, 41] occur when platforms and workers have divergent interests in completing of crowdsourcing tasks, such as in the choice of configurations, leading to conflicts between the two parties.

Although previous studies have explored various types of conflicts, we notice that for video analytics tasks, a research gap exists in simultaneously considering the conflicts among workers, among platforms, and between platforms and workers, warranting further investigation. Moreover, through experiments shown in Fig. 2(a), we validate that existing approaches [25, 26] for video analytics crowdsourcing solely addressing the conflicts among workers or platforms have a negative impact on the profits of platforms (defined as Pareto efficiency [44] later), compared to an optimal approach that additionally considers the conflicts between platforms and workers. As a result, in order to overcome the aforementioned situation and bridge this gap in research, we design an incentive mechanism called MACRO for video analytics upon crowdsourcing which incentivizes workers to contribute to the platforms' Pareto efficiency and tackles the platform-worker conflicts.

B. System Settings and Models

Video Analytics upon Crowdsourcing. In our mechanism MACRO, the set of platforms is denoted as $\boldsymbol{\mathcal{N}} = \{1, ..., N\}$, and the set of workers for video analytics is $\boldsymbol{\mathcal{M}} = \{1, ..., M\}$. Each platform receives a variety of video analytics tasks from the requestors and then recruits the proper workers for them. During worker selection for video analytics, the configurations including the video frame rate, resolution and analytics model should be considered. Due to the limited capacity of mobile devices, we consider that each worker m's device is equipped with a single CNN model for analytical simplicity [45], and the input resolution of this model is denoted as s_m. However, our proposed mechanism can also be extended for multiple CNNs on each mobile device. Once the worker m is chosen, the resolution is set as s_m to match the input resolution of the equipped CNN model on its mobile device [46].

Furthermore, platform n needs to determine the frame rate decision $f_{n,m}$, satisfying $f_{n,m} \in [0, F_n], \forall n \in \boldsymbol{\mathcal{N}}, m \in \boldsymbol{\mathcal{M}}$, where F_n is the maximum frame rate that can be set for video analytics, and it is decided by the video data provided from requestors to platform n. When $f_{n,m}$ is set as 0, it means that the platform n does not recruit worker m for video analytics. Otherwise, platform n dispatches the video analytics task to worker m, and the frame rate is determined as $f_{n,m}$. Similar to [25], since there exists a large amount of to-be-analyzed video data in the crowdsourcing scenario, the frame rate decision can be approximately regarded as a continuous variable between 0 and $F_{n,m}$. For ease of expression later, we denote the frame rate decision related to platform n as $\boldsymbol{f}_{n,\cdot} = \{f_{n,1}, ..., f_{n,M}\}$, and $\boldsymbol{f}_{\cdot,m} = \{f_{1,m}, ..., f_{N,m}\}$ related to worker m.

Utility for Platforms. The revenue of each platform n is affected by the accuracy $a_{n,m}$ of the video analytics task on each worker m's device. To evaluate the video analytics accuracy, we employ the commonly used metric F1-score [47], and we present its specific calculation details in Section V.A. However, platforms need to decide the video analytics configurations before videos are fed into DNN models. Fortunately, following some existing works [26, 48], we observe that the F1-score based accuracy depends on video analytics configurations, and it can be formulated as

$$a_{n,m} = \phi_{n,m}(f_{n,m})\epsilon_{n,m}(s_m), \tag{1}$$

where the concave functions $\phi_{n,m}(f_{n,m}) = c_1(1 - e^{-c_2 f_{n,m}})$ and $\epsilon_{n,m}(s_m) = c_3(1 - e^{-c_4 s_m})$ represent the accuracy with respect to frame rate $f_{n,m}$ and resolution s_m. Related works [26, 48] only consider the varying video content or the heterogeneity of workers' models when fitting the parameters c_1, c_2, c_3 and c_4. We take both of them into consideration when modeling Eq. (1) inspired by the results shown in Fig. 2(b).

Based on marginal effects [49] in economics, the revenue for each platform n depending on video analytics accuracy $a_{n,m}$ can be modeled as a concave function. It intuitively means that as accuracy increases closer to 1, the gains for each platform increase more slowly. Following the existing work [26], we set the revenue function $G_n(a_{n,m})$ with respect to accuracy as a logarithmic form function in our system model. Overall, we model the utility function for platform n as

$$U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}) = \sum\nolimits_{m=1}^{M} u_{n,m}^{\mathrm{p}}(f_{n,m}) = \sum\nolimits_{m=1}^{M} G_n(a_{n,m}), \tag{2}$$

where $u_{n,m}^{\mathrm{p}}(f_{n,m})$ is calculated as $G_n(a_{n,m})$, and it represents platform n's utility resulting from worker m.

When the platform n dispatches video analytics tasks to the recruited workers, the to-be-analyzed video data may occupy its bandwidth resource. However, the bandwidth budget B_n is usually considered limited [33], which is represented as

$$\sum\nolimits_{m=1}^{M} \frac{f_{n,m}}{F_n} R_n b_{n,m} \leq B_n, \tag{3}$$

where R_n is the original bitrate of video data from platform n. Without loss of generality, we consider that the data trans-

mission paths (e.g., Wi-Fi [50], 4G LTE [51] and 5G [52]) between the platforms and workers may be different. Thus, the monetary cost for each unit of bandwidth occupancy is unique for each pair of platform n and worker m, denoted as $b_{n,m}$.

Utility for Workers. According to existing works [31, 32], for crowdsourcing workers equipped with mobile devices, the battery becomes a significant concern due to the inconvenience of recharging. Thus, workers' gains can be measured based on compensation for energy consumption [25, 26], which mainly results from transmission and computation for video analytics.

Transmission energy consumption arises when the worker downloads video data from crowdsourcing platforms, and it is proportional to the size of the downloaded video data. Inspired from [48], the data size of a frame with resolution s_m can be calculated as $\alpha(s_m)^2$, where α is a constant. Thus, the energy consumption incurred by worker m's downloading video data from crowdsourcing platforms is modeled as

$$e_m^{\mathrm{d}} = \sum\nolimits_{n=1}^{N} \gamma_m^{\mathrm{d}} \alpha(s_m)^2 f_{n,m}, \tag{4}$$

where γ_m^{d} represents the energy consumption on worker m's device for downloading one bit of video data from platforms. To verify the validity of the above Eq. (4) and profile for the energy consumption rate γ_m^{d}, experiments on 1080p and 5ps videos are conducted with power analyzer (AITEK AWE2101 [53]), as shown in Fig. 2(c). Besides, energy consumption incurred by video analytics on worker n's device [26] is

$$e_m^{\mathrm{c}} = \sum\nolimits_{n=1}^{N} \gamma_{n,m}^{\mathrm{c}} f_{n,m}, \tag{5}$$

where $\gamma_{n,m}^{\mathrm{c}}$ denotes the energy consumption on worker m's device for analyzing each video frame from platform n. Different from the existing work [26], when profiling for $\gamma_{n,m}^{\mathrm{c}}$, we further take into account the differences in video analytics tasks across platforms, as shown in Fig. 2(c).

Following [25, 26], workers' earnings mainly cover their execution cost of energy consumption as compensation. Based on it, we model the utility function for each worker m as

$$U_m^{\mathrm{w}}(\boldsymbol{f}_{\cdot,m}) = \sum\nolimits_{n=1}^{N} u_{n,m}^{\mathrm{w}}(f_{n,m}) = \omega_m(e_m^{\mathrm{d}} + e_m^{\mathrm{c}}), \tag{6}$$

where ω_m is priced [54] by worker m for consuming one unit of energy, and $u_{n,m}^{\mathrm{w}}(f_{n,m}) = \omega_m(\gamma_m^{\mathrm{d}}\alpha(s_m)^2 + \gamma_{n,m}^{\mathrm{c}})f_{n,m}$ means the utility of worker m when executing the video analytics task from platform n. It is noted that workers' benefits are affected by compensation for energy consumption, which is further determined by the configurations of video analytics.

Obviously from Eq. (6), workers can increase their video analysis workload by raising the frame rate, thereby boosting their earnings. However, on each worker's device, the computing resources (e.g., CPU/GPU) [55] are constrained as

$$\sum\nolimits_{n=1}^{N} f_{n,m} c_{n,m} \leq C_m, \tag{7}$$

where $c_{n,m}$ corresponds to the computation demand for one video frame analyzed on worker m's device. It is worth noting that the computation demand $c_{n,m}$ for each platform n's video analytics task executed with worker m's model is considered distinctive, following the CPU occupancy results in Fig. 2(d).

C. Problem Formulation

Based on the above models, we give the problem formulation for each crowdsourcing platform and worker as follows. For each platform n, the optimization problem $\mathbb{P}_{1,n}^{\mathrm{p}}$ is:

$$\mathbb{P}_{1,n}^{\mathrm{p}}: \max_{f_{n,m}\in[0,F_n]} U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}) = \sum\nolimits_{m=1}^{M} u_{n,m}^{\mathrm{p}}(f_{n,m}) \quad \textit{s.t.} \text{ Ineq. (3)},$$

where each platform n needs to select the proper workers and determines the frame rate decision $\boldsymbol{f}_{n,\cdot}$ within its bandwidth budget to maximize the analytics accuracy-based revenue. In the meantime, the optimization problem $\mathbb{P}_{1,m}^{\mathrm{w}}$ for worker m is:

$$\mathbb{P}_{1,m}^{\mathrm{w}}: \max_{f_{n,m}\in[0,F_n]} U_m^{\mathrm{w}}(\boldsymbol{f}_{\cdot,m}) = \sum\nolimits_{n=1}^{N} u_{n,m}^{\mathrm{w}}(f_{n,m}) \quad \textit{s.t.} \text{ Ineq. (7)},$$

where each worker m can also adjust its frame rate decision $\boldsymbol{f}_{\cdot,m}$ to maximize its energy consumption-related utility under the computing resource constraint C_m.

III. Multi-platform Game For Pareto Efficiency

A. Solution Space Reduction for Original Problems

Since the bandwidth budget for each platform n is limited, if its frame rate decisions $\boldsymbol{f}_{n,\cdot}$ are all set to F_n, the cost of the occupied bandwidth will exceed B_n, i.e., $\sum_{m=1}^{M} R_n b_{n,m} > B_n$. Based on the above reasonable assumption, we present the following **Proposition 1** and convert the original problem $\mathbb{P}_{1,n}^{\mathrm{p}}$ to

$$\mathbb{P}_{2,n}^{\mathrm{p}}: \max_{f_{n,m}\in[0,F_n]} U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}) = \sum\nolimits_{m=1}^{M} u_{n,m}^{\mathrm{p}}(f_{n,m}) \tag{8}$$

$$\textit{s.t.} \quad \sum\nolimits_{m=1}^{M} (R_n b_{n,m}/F_n) f_{n,m} - B_n = 0. \tag{9}$$

Proposition 1. *For each crowdsourcing platform n, the optimization problem $\mathbb{P}_{1,n}^{\mathrm{p}}$ can be equivalently converted to $\mathbb{P}_{2,n}^{\mathrm{p}}$.*

Proof. We prove it by contradiction. Assume the optimal $\bar{\boldsymbol{f}}_{n,\cdot}$ for $\mathbb{P}_{1,n}^{\mathrm{p}}$ satisfies $\sum_{m=1}^{M}(R_n b_{n,m}/F_n)\bar{f}_{n,m} < B_n$. When each worker m's frame rate is F_n, the cost for bandwidth will exceed B_n, i.e., $\sum_{m=1}^{M} R_n b_{n,m} > B_n$. Besides, the optimization goal $U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot})$ is strictly increasing, so there must exist a worker m and its increased frame rate $f'_{n,m} \in (\bar{f}_{n,m}, F_n]$, such that constraint (3) is met and the optimization goal $U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot})$ is improved. Thus, the existence of $f'_{n,m}$ contradicts the optimality of $\bar{\boldsymbol{f}}_{n,\cdot}$, which means $\sum_{m=1}^{M}(R_n b_{n,m}/F_n)\bar{f}_{n,m} = B_n$. □

Similar to **Proposition 1**, due to the constrained computing capacity on each worker n's mobile device, the original problem $\mathbb{P}_{1,m}^{\mathrm{w}}$ is transformed to $\mathbb{P}_{2,m}^{\mathrm{w}}$ as follows:

$$\mathbb{P}_{2,m}^{\mathrm{w}}: \max_{f_{n,m}\in[0,F_n]} U_m^{\mathrm{w}}(\boldsymbol{f}_{\cdot,m}) = \sum\nolimits_{n=1}^{N} u_{n,m}^{\mathrm{w}}(f_{n,m}) \tag{10}$$

$$\textit{s.t.} \quad \sum\nolimits_{n=1}^{N} c_{n,m} f_{n,m} - C_m = 0. \tag{11}$$

Proposition 2. *For each worker m, the optimization problem $\mathbb{P}_{1,m}^{\mathrm{w}}$ can be equivalently converted to $\mathbb{P}_{2,m}^{\mathrm{w}}$.*

Proof. Omitted due to similarity to **Proposition 1**. □

Via problem transformation, all the less-than-equal signs in original problems are converted to the equal signs in $\mathbb{P}_{2,n}^{\mathrm{p}}$ and $\mathbb{P}_{2,m}^{\mathrm{w}}$, which reduces the solution space for the overall frame rate decision $\boldsymbol{f}$, defined as $\{\boldsymbol{f}_{n,\cdot} | n \in \boldsymbol{\mathcal{N}}\}$ or $\{\boldsymbol{f}_{\cdot,m} | m \in \boldsymbol{\mathcal{M}}\}$.

B. Pareto Efficiency in Multi-platform Game

We first consider the conflicts between the platforms in recruiting workers for video analytics due to the limited computing resource of workers, modeled as a multi-platform game:

Definition 1 (Multi-platform Game)**.** Multi-platform game for video analytics upon crowdsourcing consists of the following:

- Players: Platforms $\boldsymbol{\mathcal{N}}$.
- Strategies: $\boldsymbol{f}_{n,\cdot}, \forall n \in \boldsymbol{\mathcal{N}}$.
- Payoffs: $U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}) = \sum_{m=1}^{M} u_{n,m}^{\mathrm{p}}(f_{n,m}), \forall n \in \boldsymbol{\mathcal{N}}$.
- Additional Constraints: (9), (11), $\forall n \in \boldsymbol{\mathcal{N}}, m \in \boldsymbol{\mathcal{M}}$.

For each platform n, it needs to not only select the proper workers (i.e., whether $f_{n,m}$ is 0, $\forall m \in \boldsymbol{\mathcal{M}}$) but also determine the optimal frame rate decision $\boldsymbol{f}_{n,\cdot}$ for the selected workers. Besides, each platform has its own bandwidth budget, which makes it harder to simultaneously maximize the payoffs of all platforms in multi-platform game. To balance the objectives among different platforms, we leverage the concept of Pareto efficiency [44], which guarantees that resources are allocated in the most efficient manner within the given constraints. We define the Pareto efficiency in multi-platform game as follows.

Definition 2 (Pareto Efficiency (PE) in Multi-platform Game)**.** The overall strategy $\boldsymbol{f}^{\mathrm{PE}}$ for video analytics reaches the Pareto efficiency if and only if there exists no other overall strategy $\boldsymbol{f}$, such that for at least one platform n, $U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}) > U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}^{\mathrm{PE}})$, and for any other platform i except n, $U_i^{\mathrm{p}}(\boldsymbol{f}_{i,\cdot}) \geq U_i^{\mathrm{p}}(\boldsymbol{f}_{i,\cdot}^{\mathrm{PE}})$.

We notice that PE is an ideal outcome, implying that no platform can change its strategy to increase its payoff without decreasing others'. However, based on **Definition 2**, it is hard to directly obtain the overall strategy $\boldsymbol{f}^{\mathrm{PE}}$ satisfying PE.

C. Dual Ascent-based Approach for Pareto Efficiency

For Pareto efficiency in multi-platform game, we propose a dual ascent-based [56] approach shown in **Alg. 1**. The idea of **Alg. 1** is to reach Pareto efficiency by maximizing the social welfare [26] of all platforms, i.e., $\sum_{n=1}^{N} U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot})$, with the constraints in (9) and (11) satisfied. We prove the effectiveness of **Alg. 1** in **Thm. 1** later. To realize the idea, we leverage the dual ascent method [56] for the above maximization problem with constraints. We present the technical details as follows:

1) Lagrangian Function Construction: Due to the presence of constraints, directly solving the social welfare maximization problem is not trivial. To transform it into an unconstrained optimization problem, we build the Lagrangian function [57] as $\mathcal{L}(\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ for the platforms' social welfare maximization problem with equality constraints (9), (11) as $\mathcal{L}(\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu}) =$

$$\begin{aligned} &-\sum\nolimits_{n=1}^{N} U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}) - \sum\nolimits_{n=1}^{N} \lambda_n \Big(\sum\nolimits_{m=1}^{M} \frac{R_n b_{n,m}}{F_n} f_{n,m} - B_n\Big) \\ &-\sum\nolimits_{m=1}^{M} \mu_m \Big(\sum\nolimits_{n=1}^{N} c_{n,m} f_{n,m} - C_m\Big), \end{aligned} \tag{12}$$

where $\boldsymbol{\lambda} = \{\lambda_1, ..., \lambda_N\}$ and $\boldsymbol{\mu} = \{\mu_1, ..., \mu_M\}$ are Lagrange multipliers, which are used to incorporate the equality constraints into $\mathcal{L}(\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ involving platforms' social welfare.

2) Lagrangian Function Minimization: Building on the aforementioned Lagrange function $\mathcal{L}(\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu})$, we convert the constrained social welfare maximization problem into a problem of minimizing function $\mathcal{L}(\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ as

$$\mathbb{P}_3^{\mathrm{p}} : \min\nolimits_{\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu}} \mathcal{L}(\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu}).$$

Intuitively, based on the definition of $\mathcal{L}(\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ in Eq. (12), obtaining the optimal solution to problem $\mathbb{P}_3^{\mathrm{p}}$ entails simultaneously optimizing platform's social welfare and satisfying the constraint conditions in (9) and (11).

3) Dual Ascent for Problem $\mathbb{P}_3^{\mathrm{p}}$: Given that problem $\mathbb{P}_3^{\mathrm{p}}$ involves three interdependent decision variables $\{\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu}\}$, deriving their optimal solutions concurrently is challenging. Hence, we employ a dual ascent-based approach to solve it. Concretely, after randomly initializing $\{\boldsymbol{\lambda}^0, \boldsymbol{\mu}^0\}$, we firstly solve for the optimal value of $\boldsymbol{f}^1$ as

$$\boldsymbol{f}^1 \leftarrow \arg\min\nolimits_{\boldsymbol{f}} \mathcal{L}(\boldsymbol{f}, \boldsymbol{\lambda}^0, \boldsymbol{\mu}^0). \tag{13}$$

Once $\boldsymbol{f}^1$ is decided, we sequentially update $\{\boldsymbol{\lambda}^1, \boldsymbol{\mu}^1\}$ as

$$\lambda_n^1 \leftarrow \lambda_n^0 - \eta\Big(\sum\nolimits_{m=1}^{M} \frac{R_n b_{n,m}}{F_n} f_{n,m}^1 - B_n\Big), \forall n \in \boldsymbol{\mathcal{N}}, \tag{14}$$

$$\mu_m^1 \leftarrow \mu_m^0 - \eta\Big(\sum\nolimits_{n=1}^{N} c_{n,m} f_{n,m}^1 - C_m\Big), \forall m \in \boldsymbol{\mathcal{M}}, \tag{15}$$

where the updates of $\{\boldsymbol{\lambda}^1, \boldsymbol{\mu}^1\}$ are performed in the direction of gradient descent, and η denotes the step size for the updates. Following the above outlined update process, **Alg. 1** iteratively updates the overall frame rate decision $\boldsymbol{f}^t$ and Lagrange multipliers $(\boldsymbol{\lambda}^t, \boldsymbol{\mu}^t)$ in each iteration round t, shown in lines 3-5. For the optimal solution $(\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ to problem $\mathbb{P}_3^{\mathrm{p}}$, adhering to the dual ascent framework, T_{max} is set to a sufficiently large number of rounds to allow the objective function to converge.

Algorithm 1: Dual Ascent for PE in Multi-platform Game

Input: $R_n, b_{n,m}, F_n, B_n, c_{n,m}, C_m, \forall n \in \boldsymbol{\mathcal{N}}, m \in \boldsymbol{\mathcal{M}}$

1 $t \leftarrow 0$, and Randomly Initialize $\boldsymbol{f}^0$, $\boldsymbol{\lambda}^0$ and $\boldsymbol{\mu}^0$;
2 **while** $t < T_{max}$ **do**
3 $\quad \boldsymbol{f}^{t+1} \leftarrow \arg\min_{\boldsymbol{f}} \mathcal{L}(\boldsymbol{f}, \boldsymbol{\lambda}^t, \boldsymbol{\mu}^t)$;
4 $\quad \lambda_n^{t+1} \leftarrow \lambda_n^t - \eta(\sum_{m=1}^{M} \frac{R_n b_{n,m}}{F_n} f_{n,m}^{t+1} - B_n), \forall n \in \boldsymbol{\mathcal{N}}$;
5 $\quad \mu_m^{t+1} \leftarrow \mu_m^t - \eta(\sum_{n=1}^{N} c_{n,m} f_{n,m}^{t+1} - C_m), \forall m \in \boldsymbol{\mathcal{M}}$;
6 $\quad t \leftarrow t + 1$;

Output: $\boldsymbol{f}^{T_{max}}$.

D. Theoretical Analysis

Furthermore, we show the effects of **Alg. 1** in **Thm. 1**:

Theorem 1. *The overall frame rate decision $\boldsymbol{f}^{T_{max}}$ obtained from **Alg. 1**, which maximizes the social welfare of platforms, reaches Pareto efficiency in multi-platform game.*

Proof. Following [56], dual ascent-based **Alg. 1** can converge linearly to the optimal $\boldsymbol{f}^{sw}$ for $\mathbb{P}_3^{\mathrm{p}}$ and maximizes platforms' social welfare, which proves that $\boldsymbol{f}^{T_{max}} = \boldsymbol{f}^{sw}$. Next, we prove $\boldsymbol{f}^{sw}$ reaches PE by contradiction as follows. If $\boldsymbol{f}^{sw}$ does not reach PE, according to **Def. 2**, there must exist another frame rate decision $\boldsymbol{f}'$, such that for at least one platform n, $U_n^{\mathrm{p}}(\boldsymbol{f}'_{n,\cdot}) > U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}^{sw})$, and for any other platform i except n, $U_i^{\mathrm{p}}(\boldsymbol{f}'_{i,\cdot}) \geq U_i^{\mathrm{p}}(\boldsymbol{f}_{i,\cdot}^{sw})$. Therefore, it can be easily obtained that $\sum_{n=1}^{N} U_n^{\mathrm{p}}(\boldsymbol{f}'_{n,\cdot}) > \sum_{n=1}^{N} U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}^{sw})$, which contradicts the optimality of $\boldsymbol{f}^{sw}$ for $\mathbb{P}_3^{p}$. Thus, **Thm. 1** is proved. □

However, in addition to inter-platform conflicts, platform-worker conflicts can make it more difficult to achieve Pareto efficiency because workers can unilaterally change frame rate

decisions to maximize their own utility, i.e., $U_m^{\mathrm{w}}(\boldsymbol{f}_{\cdot,m}), \forall m \in \boldsymbol{\mathcal{M}}$. Moreover, when the number of workers M increases to a relatively large size, each platform n needs to update massive frame rate decisions $\{f_{n,1}^t, ..., f_{n,M}^t\}$, which is highly computationally intensive. We address the above challenges later.

IV. Incentivized Multi-leader Game via ADMM

A. Multi-leader Game for Incentivized Crowdsourcing

Platform-worker conflicts mean that the workers have their own optimization goals in Eq. (10), which may not be consistent with those of the platforms in Eq. (8). Consequently, the worker will change the frame rate decision to maximize its own utility, resulting in the platforms' social welfare not being maximized, and thus Pareto efficiency may not be achieved. Naturally, we think of designing an incentive-based method to encourage workers to contribute to Pareto efficiency. Besides, the conventional Stackelberg game [37] only works for single-leader scenarios, and it cannot be directly applied to our mechanism MACRO, where multiple platforms recruit workers for video analytics. Therefore, based on the above analysis, we present the incentive-based multi-leader game as follows.

Definition 3 (Incentive-based Multi-leader Game)**.** The multi-leader game for video analytics upon incentivized crowdsourcing consists of the following:

- Players: platforms $\boldsymbol{\mathcal{N}}$ and workers $\boldsymbol{\mathcal{M}}$.
- Strategies:
  - ▷ Platforms: $\boldsymbol{\theta}_{n,\cdot} = \{\theta_{n,1}, ..., \theta_{n,M}\}, \forall n \in \boldsymbol{\mathcal{N}}$.
  - ▷ Workers: $\boldsymbol{f}_{\cdot,m}, \forall m \in \boldsymbol{\mathcal{M}}$.
- Payoffs:
  - ▷ Platforms: $U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}), \forall n \in \boldsymbol{\mathcal{N}}$.
  - ▷ Workers: $I_m(\boldsymbol{\theta}_{\cdot,m}, U_m^{\mathrm{w}}(\boldsymbol{f}_{\cdot,m})), \forall m \in \boldsymbol{\mathcal{M}}$.
- Additional Constraints: (9), (11), $\forall n \in \boldsymbol{\mathcal{N}}, m \in \boldsymbol{\mathcal{M}}$.

As shown in the above incentive-based multi-leader game model, the incentive function $I_m(\boldsymbol{\theta}_{\cdot,m}, U_m^{\mathrm{w}}(\boldsymbol{f}_{\cdot,m}))$ is used to replace each worker m's original optimization goal $U_m^{\mathrm{w}}(\boldsymbol{f}_{\cdot,m})$, and it describes the interactions between worker m and other platforms. Notably, the incentive function $I_m(\boldsymbol{\theta}_{\cdot,m}, U_m^{\mathrm{w}}(\boldsymbol{f}_{\cdot,m}))$ is influenced by the incentive factors $\boldsymbol{\theta}_{\cdot,m} = \{\theta_{1,m}, ..., \theta_{N,m}\}$ decided by platforms. Besides, the platforms' payoffs are the same as in multi-platform game, following the classic idea in incentive mechanisms, i.e., the platforms motivate workers to adjust the strategies as platforms' expectation with bonus [58].

Similar to the single-leader Stackelberg game, the incentive factors $\boldsymbol{\theta}$, defined as $\{\boldsymbol{\theta}_{n,\cdot} | n \in \boldsymbol{\mathcal{N}}\}$ or $\{\boldsymbol{\theta}_{\cdot,m} | m \in \boldsymbol{\mathcal{M}}\}$ will have an impact on workers' frame rate strategies. Specifically, the platforms need to determine the proper incentive factors, which motivate the workers to adjust their frame rate strategies for Pareto efficiency defined in **Definition 2**. Furthermore, it is necessary to ensure that none of the platforms and workers have incentives to change their strategies when PE is reached.

To summarize the above requirements in MACRO, we follow the concept of Nash equilibrium [59, 60] for Stackelberg game and define the multi-leader Stackelberg equilibrium as:

Definition 4 (Multi-leader Stackelberg Equilibrium (MSE))**.** A strategy $(\boldsymbol{\theta}^*, \boldsymbol{f}^*)$ reaches MSE if and only if the following 2 conditions for platforms and workers, respectively, are met:

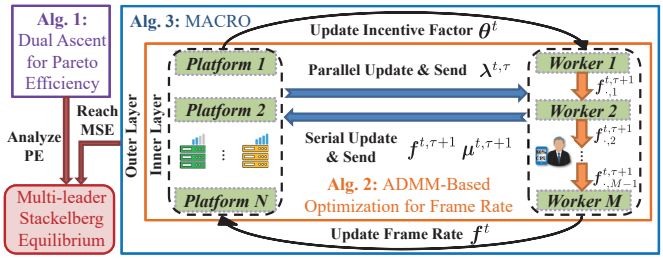


Fig. 3. MACRO for Video Analytics upon Incentivized Crowdsourcing

(a) When the frame rate strategy $\boldsymbol{f}^*$ is determined by workers, there exists no other strategy $(\boldsymbol{\theta}, \boldsymbol{f})$ changed by platforms such that $\sum_{n=1}^{N} U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}) > \sum_{n=1}^{N} U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}^*)$.

(b) When the incentive factor strategy $\boldsymbol{\theta}^*$ is determined by platforms, there exists no other strategy $(\boldsymbol{\theta}^*, \boldsymbol{f})$ changed by workers such that for at least one worker m,

$$I_m(\boldsymbol{\theta}_{\cdot,m}^*, U_m^{\mathrm{w}}(\boldsymbol{f}_{\cdot,m})) > I_m(\boldsymbol{\theta}_{\cdot,m}^*, U_m^{\mathrm{w}}(\boldsymbol{f}_{\cdot,m}^*)), \tag{16}$$

and for any other worker j except m,

$$I_j(\boldsymbol{\theta}_{\cdot,j}^*, U_j^{\mathrm{w}}(\boldsymbol{f}_{\cdot,j})) \geq I_j(\boldsymbol{\theta}_{\cdot,j}^*, U_j^{\mathrm{w}}(\boldsymbol{f}_{\cdot,j}^*)). \tag{17}$$

As illustrated in the above definition, the condition (a) is consistent with **Thm. 1**, which guarantees the Pareto efficiency for platforms when MSE is reached, and thus platforms will not change their strategies. Besides, the condition (b) ensures that given a strategy $(\boldsymbol{\theta}^*, \boldsymbol{f}^*)$ satisfying MSE, no worker can deviate from $\boldsymbol{f}^*$ to increase its incentive-based payoff without decreasing any other workers'. In short, none of the platforms and workers have incentives to alter their strategies for higher payoff when others' keep unchanged, which reaches the MSE.

To motivate the workers to contribute to Pareto efficiency for platforms and then reach multi-leader Stackelberg equilibrium, we carefully design the incentive function I_m as follows.

- First and foremost, since the incentive function is meant for each platform n to motivate the worker m to select the proper frame rate decision $f_{n,m}$ for Pareto efficiency, the increase in each worker's own utility $u_{n,m}^{\mathrm{w}}(f_{n,m})$ needs to be hindered. As $u_{n,m}^{\mathrm{w}}(f_{n,m})$ increases with respect to frame rate $f_{n,m}$, we set the incentive value, which should be negative to $f_{n,m}$, as $\hat{I} - \theta_{n,m} f_{n,m}$ [61]. The parameter $\hat{I}$ is large enough to make the incentive value positive.
- Second, with the incentive value, each worker m should pay attention to platform n's utility function $u_{n,m}^{\mathrm{p}}(f_{n,m})$.
- Last, each worker m's own utility function $u_{n,m}^{\mathrm{w}}(f_{n,m})$ will naturally be taken into consideration.

In summary, we design the incentive function, simultaneously considering the above 3 aspects as $I_m(\boldsymbol{\theta}_{\cdot,m}, \boldsymbol{f}_{\cdot,m}) =$

$$\sum_{n=1}^{N} (\underbrace{u_{n,m}^{\mathrm{w}}(f_{n,m})}_{\textit{worker's utility}} + \underbrace{u_{n,m}^{\mathrm{p}}(f_{n,m})}_{\textit{platform's utility}} + \underbrace{(\hat{I} - \theta_{n,m} f_{n,m})}_{\textit{incentive value}}). \tag{18}$$

B. Incentive-based Configuration Update upon ADMM

To reach the multi-leader Stackelberg equilibrium based on the designed incentive function, we propose an efficient two-layer iteration-based method as shown in Fig. 3. The inner layer focuses on updating workers' frame rates $\boldsymbol{f}$ and the outer layer addresses the update of platforms' incentive factors $\boldsymbol{\theta}$,

which together constitute MACRO. Concretely, the inner layer guarantees the fulfillment of condition (b) for MSE, while the outer layer ensures condition (a). For their technical details, we sequentially present the *Overview of MACRO*, *Update* $\boldsymbol{f}$ *in Inner Layer* and *Update* $\boldsymbol{\theta}$ *in Outer Layer* as follows.

1) Overview of MACRO: For ease of expression, we denote the outer layer's iterative round as $t \in \{0, 1, 2, ...\}$, and denote the inner layer's iterative step as $\tau \in \{0, 1, 2, ...\}$. We then illustrate the general process of transitioning from iterative round t to $t+1$, which includes the updates of frame rates $\boldsymbol{f}$ in the inner layer and incentive factors $\boldsymbol{\theta}$ in the outer layer:

- At each iterative round t, given the incentive factors $\boldsymbol{\theta}^t$ from platforms, the workers optimize their incentive functions by adjusting their frame rate decisions $\boldsymbol{f}^t$ in an iterative manner (i.e., $\tau = 0, 1, 2, ...$).
- Then at the next iterative round $t+1$, the platforms adjust their incentive factors $\boldsymbol{\theta}^{t+1}$ based on the updated frame rate decisions $\boldsymbol{f}^t$, to motivate the workers to pay more attention to the platforms' objectives. After that, workers further adjust their frame rate decision $\boldsymbol{f}^{t+1}$ in an iterative manner (i.e., $\tau = 0, 1, 2, ...$).

Subsequently, we further elucidate the details of the updates in the inner layer and outer layer, which ensures the condition (b) and (a), respectively, for MSE.

2) Update $\boldsymbol{f}$ *in Inner Layer:* Given the incentive factors $\boldsymbol{\theta}^t$ from platforms, to fulfil the condition (b) in **Definition 4** for MSE, our idea is to follow **Thm. 1**, maximizing the sum of all workers' incentive utilities $\sum_{m=1}^{M} I_m(\boldsymbol{\theta}_{\cdot,m}, \boldsymbol{f}_{\cdot,m})$ with the constraints (9) and (11) satisfied. However, as previously noted, the number of workers M can be very large in comparison to platforms, leading to a high computational overhead. Thus, we turn to Alternating Direction Method of Multipliers (ADMM) [62], which can decompose the complex problem into multiple simplified subproblems.

Under the ADMM method framework, we first construct an augmented Lagrangian function [57] for our considered maximization problem with equality constraints. After that, to optimize the augmented Lagrangian function, we further spread its high computational cost to workers and platforms, respectively. We present the technical details as follows:

- *Augmented Lagrangian Function Construction:* Based on ADMM framework [62], we build the augmented Lagrangian function [57] as $\mathcal{L}_\rho(\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu}) =$
$$-\sum_{m=1}^{M} I_m(\boldsymbol{\theta}_{\cdot,m}, \boldsymbol{f}_{\cdot,m}) - \sum_{n=1}^{N} \lambda_n \Big(\sum_{m=1}^{M} \frac{R_n b_{n,m}}{F_n} f_{n,m} - B_n\Big)$$
$$-\sum_{m=1}^{M} \mu_m \Big(\sum_{n=1}^{N} c_{n,m} f_{n,m} - C_m\Big) + \frac{\rho}{2} \sum_{n=1}^{N} \Big|\sum_{m=1}^{M} \frac{R_n b_{n,m}}{F_n} f_{n,m} - B_n\Big|^2,$$
where $\boldsymbol{\lambda} = \{\lambda_1, ..., \lambda_N\}$, $\boldsymbol{\mu} = \{\mu_1, ..., \mu_M\}$ are Lagrange multipliers, the same as the Lagrangian function in Eq. (12), which are used to incorporate the constraints (9) and (11) into $\mathcal{L}_\rho(\boldsymbol{f}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ involving workers' incentive utilities. Besides, we use the damping factor ρ [57] to control the convergence rate and stability for ADMM.
- *Augmented Lagrangian Function Minimization:* Upon the above augmented Lagrange function, we transform the incentive utility maximization problem with constraints in round t to a problem of minimizing $\mathcal{L}_\rho(\boldsymbol{f}^t, \boldsymbol{\lambda}^t, \boldsymbol{\mu}^t)$ as
$$\mathbb{P}_4^{\mathrm{w}} : \min\nolimits_{\boldsymbol{f}^t, \boldsymbol{\lambda}^t, \boldsymbol{\mu}^t} \mathcal{L}_\rho(\boldsymbol{f}^t, \boldsymbol{\lambda}^t, \boldsymbol{\mu}^t).$$
Notably, according to the definition of $\mathcal{L}_\rho(\boldsymbol{f}^t, \boldsymbol{\lambda}^t, \boldsymbol{\mu}^t)$, finding the best solution for problem $\mathbb{P}_4^{\mathrm{w}}$ means that we have to improve the platform's social welfare and meet the constraints (9) and (11) at the same time.

Algorithm 2: ADMM-based Optimization for Frame Rate

Input: $t, \boldsymbol{\theta}^t, \boldsymbol{f}^{t-1}, \boldsymbol{\lambda}^{t-1}, \boldsymbol{\mu}^{t-1}$
$\tau \leftarrow 0, (\boldsymbol{f}^{t,0}, \boldsymbol{\lambda}^{t,0}, \boldsymbol{\mu}^{t,0}) \leftarrow (\boldsymbol{f}^{t-1}, \boldsymbol{\lambda}^{t-1}, \boldsymbol{\mu}^{t-1})$;
while $\tau < \tau_{max}$ **do**
  `// Serial Update at Workers:`
  for *worker* $m = 1, 2, ..., M$ **do**
    Update $\boldsymbol{f}_{\cdot,m}^{t,\tau+1}$ and $\mu_m^{t,\tau+1}$ upon Eqs. (19), (20);
  `// Parallel Update at Platforms:`
  for *platform* $n = 1, 2, ..., N$ **do**
    Update $\lambda_n^{t,\tau+1}$ following Eq. (21);
  $\tau \leftarrow \tau + 1$;
Output: $\boldsymbol{f}^{t,\tau_{max}}, \boldsymbol{\lambda}^{t,\tau_{max}}, \boldsymbol{\mu}^{t,\tau_{max}}$.

However, when the number of workers M gets larger, the computational cost of solving problem $\mathbb{P}_4^{\mathrm{w}}$ will also become higher. According to ADMM framework, we solve problem $\mathbb{P}_4^{\mathrm{w}}$ in an iterative manner, i.e., iteratively updating frame rate decisions $\boldsymbol{f}^{t,\tau}$ and Lagrange multipliers $\boldsymbol{\lambda}^{t,\tau}$, $\boldsymbol{\mu}^{t,\tau}$ (i.e., $\tau = 0, 1, ...$). Furthermore, to spread the high computational cost, the computation for the frame rate decision $\boldsymbol{f}_{\cdot,m}^{t,\tau}$ and $\mu_m^{t,\tau}$ is distributed to worker m, and then each platform n coordinates the frame rate decisions $\boldsymbol{f}_{n,\cdot}^{t,\tau}$ by updating $\lambda_n^{t,\tau}$. We show the updates at workers and platforms as follows:

- *Serial Update at Workers:* When the iterative step is $\tau + 1$, each worker m updates its frame rate decision $\boldsymbol{f}_{\cdot,m}^{t,\tau+1}$ to minimize $\mathcal{L}_\rho(\boldsymbol{f}^t, \boldsymbol{\lambda}^t, \boldsymbol{\mu}^t)$. We further simplify $\mathcal{L}_\rho(\boldsymbol{f}^t, \boldsymbol{\lambda}^t, \boldsymbol{\mu}^t)$ by retaining the parts related to worker m, thereby obtaining $\boldsymbol{f}_{\cdot,m}^{t,\tau+1} =$
$$\begin{aligned}\underset{\boldsymbol{f}_{\cdot,m}^t}{\operatorname{argmin}} & -I_m(\boldsymbol{\theta}_{\cdot,m}^t, \boldsymbol{f}_{\cdot,m}^t) - \sum_{n=1}^{N} \Big(\lambda_n^{t,\tau} \frac{R_n b_{n,m}}{F_n} + \mu_m^{t,\tau} c_{n,m}\Big) f_{n,m}^t \\ & + \frac{\rho}{2} \sum_{n=1}^{N} \Big(\sum_{j \neq m} \frac{R_n b_{n,j}}{F_n} f_{n,j}^{t,\tilde{\tau}} + \frac{R_n b_{n,m}}{F_n} f_{n,m}^t - B_n\Big)^2,\end{aligned} \tag{19}$$
where $\tilde{\tau} = \tau + 1$ if $j < m$, otherwise $\tilde{\tau} = \tau$. Notably, in the serial update process for workers, each worker m requires the latest updated frame rate $\boldsymbol{f}_{\cdot,j}^{t,\tilde{\tau}}$ from another worker j as input to update its own frame rate $\boldsymbol{f}_{\cdot,m}^{t,\tau+1}$. Besides, each worker m further updates its Lagrange multiplier $\mu_m^{t,\tau+1}$ in the direction of gradient descent as
$$\mu_m^{t,\tau+1} = \mu_m^{t,\tau} - \eta\Big(\sum\nolimits_{n=1}^{N} c_{n,m} f_{n,m}^{t,\tau+1} - C_m\Big), \tag{20}$$
where η denotes the step size for the update [56] and it controls the update speed and stability.
- *Parallel Update at Platforms:* When the iterative step is $\tau + 1$, each platform n updates the Lagrange multiplier $\lambda_n^{t,\tau+1}$ in the direction of gradient descent as
$$\lambda_n^{t,\tau+1} = \lambda_n^{t,\tau} - \rho\Big(\sum\nolimits_{m=1}^{M} (R_n b_{n,m}/F_n) f_{n,m}^{t,\tau+1} - B_n\Big), \tag{21}$$
where we use the damping factor ρ as the step size for the update in accordance to the ADMM framework [62].

Observably, all the platforms update their corresponding Lagrange multipliers in a parallel manner.

As shown in **Alg. 2**, we implement the above procedure of updating $\boldsymbol{f}$ in the inner layer. In line 2, τ_{max} is set to a sufficiently large number of steps allowing the objective function to converge [63]. In lines 3-6, we spread the computational cost of solving $\mathbb{P}_4^{\mathrm{w}}$ across multiple workers and platforms.

3) Update $\boldsymbol{\theta}$ in Outer Layer: The outer layer is also an iteration structure for updating the incentive factors $\boldsymbol{\theta}^{t+1}$ after the frame rate $\boldsymbol{f}^t$ is updated in **Alg. 2**. To reach the condition (a) for MSE, which considers the Pareto efficiency for platforms, we need to optimize the social welfare of platforms. For the above goal, we show the details of updating incentive factors and judging the update termination condition as follows.

- *Update Incentive Factors:* Recall that each worker m's incentive function I_m consists of the worker's utility, platform's utility and incentive value. To maximize the social welfare of platforms, our idea is to update the incentive factors in a manner that 'offsets' the utility of workers. As the incentive value is set to $\hat{I} - \theta_{n,m} f_{n,m}$ in (18), we leverage the marginal utility of workers to update the incentive factor as
$$\theta_{n,m}^{t+1} = \mathrm{d}u_{n,m}^{\mathrm{w}}(f_{n,m}^t)/\mathrm{d}f_{n,m}, \forall n \in \boldsymbol{\mathcal{N}}, m \in \boldsymbol{\mathcal{M}}. \quad (22)$$
Intuitively, we update the incentive factor $\theta_{n,m}$ as the first order derivative of the worker's utility with respect to $f_{n,m}$, which 'offsets' the utility of workers and then ensures the social welfare of platforms.
- *Judge Termination Condition:* The iteration process in the outer layer continues until there is negligible improvement in the platforms' social welfare, i.e.,
$$|\sum\nolimits_{n=1}^{N} U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}^t) - \sum\nolimits_{n=1}^{N} U_n^{\mathrm{p}}(\boldsymbol{f}_{n,\cdot}^{t-1})| < \varepsilon, \quad (23)$$
where ε is the threshold to upper-bound the changes of social welfare between 2 consecutive iteration rounds, and it pertains to the convergence analysis of MACRO as detailed in **Thm. 4** later.

The process of updating $\boldsymbol{\theta}$ in the outer layer is illustrated in **Alg. 3**, where each platform n updates its incentive factor decision $\boldsymbol{\theta}_{n,\cdot}^{t+1}$ in lines 3-4. Additionally, the judgment of update termination condition lies in line 2.

C. Theoretical Analysis

Next we analyze how MACRO reaches MSE, as demonstrated in Fig. 3. Specifically, we present the convergence analysis of **Alg. 2**, the convergence analysis of **Alg. 3**, and the analysis of the convergence rate of **Alg. 3** as follows.

*1) Convergence Analysis of **Alg. 2**:* Due to the fact that **Alg. 2** applies the ADMM framework to a specific domain, i.e, the update of video analytics configurations for crowdsourcing, the convergence property of ADMM [63] can still be maintained, which we show in **Proposition 3**. Since the convergence proof of ADMM itself is not our contribution, we only briefly summarize the convergence proof process of ADMM as found in [63] here.

Proposition 3. *ADMM-based Optimization for Frame Rate in **Alg. 2** can converge to the optimum for problem $\mathbb{P}_4^{\mathrm{w}}$.*

Algorithm 3: Incentive Mechanism MACRO for MSE

Input: $R_n, b_{n,m}, F_n, B_n, c_{n,m}, C_m, \forall n \in \boldsymbol{\mathcal{N}}, m \in \boldsymbol{\mathcal{M}}$

1 $t \leftarrow 0$, and Randomly Initialize $\boldsymbol{f}^0$, $\boldsymbol{\lambda}^0$ and $\boldsymbol{\mu}^0$;
2 **while** *Inequation (23) upon $\boldsymbol{f}^t$ is Not Satisfied* **do**
// Update Incentive Factors:
3 **for** *platform* $n = 1, 2, ..., N$ **do**
4 Update $\theta_{n,m}^{t+1}, \forall m \in \boldsymbol{\mathcal{M}}$ following Eq. (22);
// Update Frame Rate Decisions:
5 $t \leftarrow t + 1$;
6 Invoke **Alg. 2** with Input $(t, \boldsymbol{\theta}^t, \boldsymbol{f}^{t-1}, \boldsymbol{\lambda}^{t-1}, \boldsymbol{\mu}^{t-1})$, and Output $(\boldsymbol{f}^{t,\tau_{max}}, \boldsymbol{\lambda}^{t,\tau_{max}}, \boldsymbol{\mu}^{t,\tau_{max}})$;
7 $(\boldsymbol{f}^t, \boldsymbol{\lambda}^t, \boldsymbol{\mu}^t) \leftarrow (\boldsymbol{f}^{t,\tau_{max}}, \boldsymbol{\lambda}^{t,\tau_{max}}, \boldsymbol{\mu}^{t,\tau_{max}})$;

Output: $\boldsymbol{f}^t, \boldsymbol{\theta}^t$.

Proof. Following Theorem 3.1 in [63], through **Alg. 2**, the primal and dual optimality gaps [64] for $\mathbb{P}_4^{\mathrm{w}}$ can be proved to converge to 0 linearly, based on which the dual sequence is also linearly convergent to a dual optimal solution. It then follows that the output sequence of frame rate decisions can converge to a primal optimal solution. The complete proof is omitted, and see Theorem 3.1 in [63] for details. □

As **Proposition 3** shows **Alg. 2** is an ADMM process [65] for problem $\mathbb{P}_4^{\mathrm{w}}$, it can converge with KKT condition [66] met. When **Alg. 2** converages, e.g., in iterative round $t+1$, KKT condition indicates that $\forall n \in \boldsymbol{\mathcal{N}}, m \in \boldsymbol{\mathcal{M}}$,
$$\mathrm{d}\mathcal{L}_\rho(\boldsymbol{f}^{t+1}, \boldsymbol{\lambda}^{t+1}, \boldsymbol{\mu}^{t+1})/\mathrm{d}f_{n,m} = 0. \quad (24)$$
Intuitively, Eq. (24) means that **Alg. 2** obtains the optimal frame rate in the inner layer, based on which we will further analyze the convergence of **Alg. 3** later.

*2) Convergence Analysis of **Alg. 3**:* We then analyze the convergence of **Alg. 3** and present it in **Thm. 2**.

Theorem 2. *MACRO for MSE in **Alg. 3** can converge to MSE. Concretely, the social welfare for platforms always rises from iteration round t to $t+1$ before convergence.*

Proof. Expanding Eq. (24), we easily obtain
$$\begin{aligned} &\mathrm{d}(-I_m(\boldsymbol{\theta}_{\cdot,m}^{t+1}, \boldsymbol{f}_{\cdot,m}^{t+1}) - \sum\nolimits_{n=1}^{N} \lambda_n^{t+1}(R_n b_{n,m}/F_n) f_{n,m}^{t+1})/\mathrm{d}f_{n,m} \\ &+ \frac{\mathrm{d}(\frac{\rho}{2}\sum_{n=1}^{N}|\sum_{j=1}^{M}(R_n b_{n,j}/F_n) f_{n,j}^{t+1}) - B_n|^2)}{\mathrm{d}f_{n,m}} - \mu_m^{t+1} c_{n,m} = 0, \end{aligned} \quad (25)$$
$\forall n \in \boldsymbol{\mathcal{N}}, m \in \boldsymbol{\mathcal{M}}$. We then incorporate the incentive utility function (18) and the updated incentive factor (22) into (25), and multiply $(f_{n,m}^{t+1} - f_{n,m}^t)$ on the both sides. We obtain
$$\begin{aligned} &\frac{\mathrm{d}u_{n,m}^{\mathrm{p}}(f_{n,m}^{t+1})}{\mathrm{d}f_{n,m}}(f_{n,m}^{t+1} - f_{n,m}^t) + \lambda_n^{t+1}\frac{R_n b_{n,m}}{F_n}(f_{n,m}^{t+1} - f_{n,m}^t) \\ &- \rho|\sum_{j=1}^{M}(R_n b_{n,m}/F_n) f_{n,j}^{t+1} - B_n|(R_n b_{n,m}/F_n)(f_{n,m}^{t+1} - f_{n,m}^t) \\ &= (\frac{\mathrm{d}u_{n,m}^{\mathrm{w}}(f_{n,m}^t)}{\mathrm{d}f_{n,m}} - \frac{\mathrm{d}u_{n,m}^{\mathrm{w}}(f_{n,m}^{t+1})}{\mathrm{d}f_{n,m}} - \mu_m^{t+1} c_{n,m})(f_{n,m}^{t+1} - f_{n,m}^t). \end{aligned} \quad (26)$$
Besides, at each iteration round t, **Alg. 2** in inner layer can coverage, and the constraints (9) and (11) are satisfied as
$$\sum\nolimits_{m=1}^{M}(R_n b_{n,m}/F_n)(f_{n,m}^{t+1} - f_{n,m}^t) = B_n - B_n = 0, \quad (27)$$
$$\sum\nolimits_{n=1}^{N} c_{n,m}(f_{n,m}^{t+1} - f_{n,m}^t) = C_m - C_m = 0. \quad (28)$$
We combine equations (27) and (28) into (26) summed up for each n and m, and it can be obtained as

$$\sum_{n=1}^{N}\sum_{m=1}^{M}(\mathrm{d}u^{\mathrm{p}}_{n,m}(f^{t+1}_{n,m})/\mathrm{d}f_{n,m})(f^{t+1}_{n,m}-f^{t}_{n,m})$$
$$=\sum_{n=1}^{N}\sum_{m=1}^{M}(\frac{\mathrm{d}u^{\mathrm{w}}_{n,m}(f^{t}_{n,m})}{\mathrm{d}f_{n,m}}-\frac{\mathrm{d}u^{\mathrm{w}}_{n,m}(f^{t+1}_{n,m})}{\mathrm{d}f_{n,m}})(f^{t+1}_{n,m}-f^{t}_{n,m}). \quad (29)$$

As $u^{\mathrm{p}}_{n,m}(f^{t}_{n,m})$ is concave, the right side of (29) ≥ 0, and

$$\sum_{n=1}^{N}\sum_{m=1}^{M}(\mathrm{d}u^{\mathrm{p}}_{n,m}(f^{t+1}_{n,m})/\mathrm{d}f_{n,m})(f^{t+1}_{n,m}-f^{t}_{n,m})\geq 0. \quad (30)$$

Based on the concavity of $u^{\mathrm{p}}_{n,m}(f^{t}_{n,m})$ and (30), we have

$$\sum_{n=1}^{N}\sum_{m=1}^{M}u^{\mathrm{p}}_{n,m}(f^{t+1}_{n,m})-\sum_{n=1}^{N}\sum_{m=1}^{M}u^{\mathrm{p}}_{n,m}(f^{t}_{n,m})\geq 0, \quad (31)$$

which shows that $\sum_{n=1}^{N}U^{\mathrm{p}}_{n}(\boldsymbol{f}^{t+1}_{n,\cdot})\geq\sum_{n=1}^{N}U^{\mathrm{p}}_{n}(\boldsymbol{f}^{t}_{n,\cdot})$, and it means that platforms' social welfare always rises from round t to $t+1$. To make $\leq$ in (31) be $=$, it is satisfied:

$$(\frac{\mathrm{d}u^{\mathrm{p}}_{n,m}(f^{t}_{n,m})}{\mathrm{d}f_{n,m}}-\frac{\mathrm{d}u^{\mathrm{p}}_{n,m}(f^{t+1}_{n,m})}{\mathrm{d}f_{n,m}})(f^{t+1}_{n,m}-f^{t}_{n,m})=0. \quad (32)$$

We finally combine equation (32) into (26) and obtain

$$\mathrm{d}u^{\mathrm{p}}_{n,m}(f^{t+1}_{n,m})/\mathrm{d}f_{n,m}+\lambda^{t+1}_{n}R_n b_{n,m}/F_n-\mu^{t+1}_{m}c_{n,m}$$
$$+\rho|\sum_{j=1}^{M}(R_n b_{n,m}/F_n)f^{t+1}_{n,j}-B_n|(R_n b_{n,m}/F_n)=0. \quad (33)$$

which indicates that $\boldsymbol{f}^{t+1}=\boldsymbol{f}^{t}=\boldsymbol{f}^{*}$, and platforms' social welfare will converge to reach the PE (i.e., condition (a) in **Definition 4**). As **Proposition 3** guarantees condition (b) in **Definition 4**, **Alg. 2** can finally converge to MSE. □

*3) Convergence Rate Analysis of **Alg. 3**:* Based on the convergence of **Alg. 3**, we further analyze its convergence rate and present it in **Thm. 3**.

Theorem 3. *The MACRO Mechanism for MSE in **Alg. 3** can converge linearly, and the convergence rate does not depend on the number of platforms N and workers M.*

Proof. Based on $\mathcal{L}(\boldsymbol{f},\boldsymbol{\lambda},\boldsymbol{\mu})$ in problem $\mathbb{P}^{\mathrm{p}}_{3}$, we define the augmented Lagrangian function $\tilde{\mathcal{L}}(\boldsymbol{f},\boldsymbol{\lambda},\boldsymbol{\mu})$ as follows:

$$\tilde{\mathcal{L}}(\boldsymbol{f},\boldsymbol{\lambda},\boldsymbol{\mu})=\mathcal{L}(\boldsymbol{f},\boldsymbol{\lambda},\boldsymbol{\mu})+\frac{\rho}{2}\sum_{n=1}^{N}|\sum_{m=1}^{M}\frac{R_n b_{n,m}}{F_n}f_{n,m}-B_n|^2. \quad (34)$$

We use $\boldsymbol{f}^{*}$, $\boldsymbol{\lambda}^{*}$ and $\boldsymbol{\mu}^{*}$ to represent the frame rate decision and corresponding Lagrangian multipliers when **Alg. 3** have reached MSE. Similar to (25), we get the followings:

$$\sum_{n=1}^{N}\lambda^{t}_{n}(\sum_{m=1}^{M}\frac{R_n b_{n,m}}{F_n}f^{t}_{n,m}-B_n)-\frac{\rho}{2}\sum_{n=1}^{N}|\sum_{m=1}^{M}\frac{R_n b_{n,m}}{F_n}f^{t}_{n,j}-B_n|^2$$
$$+\sum_{m=1}^{M}\mu^{t}_{m}(\sum_{n=1}^{N}f^{t}_{n,m}c_{n,m}-C_m)=0, \quad (35)$$

$$\sum_{n=1}^{N}\lambda^{*}_{n}(\sum_{m=1}^{M}\frac{R_n b_{n,m}}{F_n}f^{*}_{n,m}-B_n)-\frac{\rho}{2}\sum_{n=1}^{N}|\sum_{m=1}^{M}\frac{R_n b_{n,m}}{F_n}f^{*}_{n,j}-B_n|^2$$
$$+\sum_{m=1}^{M}\mu^{*}_{m}(\sum_{n=1}^{N}f^{*}_{n,m}c_{n,m}-C_m)=0, \quad (36)$$

$$\sum_{n=1}^{N}\lambda^{t}_{n}\sum_{m=1}^{M}\frac{R_n b_{n,m}}{F_n}(f^{*}_{n,m}-f^{t}_{n,m})+\sum_{m=1}^{M}\mu^{t}_{m}\sum_{n=1}^{N}c_{n,m}(f^{*}_{n,m}-f^{t}_{n,m})$$
$$-\rho\sum_{n=1}^{N}\sum_{m=1}^{M}(R_n b_{n,m}/F_n)((R_n b_{n,m}/F_n)f^{t}_{n,j}-B_n)=0. \quad (37)$$

Besides, because of the strong concavity [67] of the platform's utility function, there exists a $\delta>0$ such that

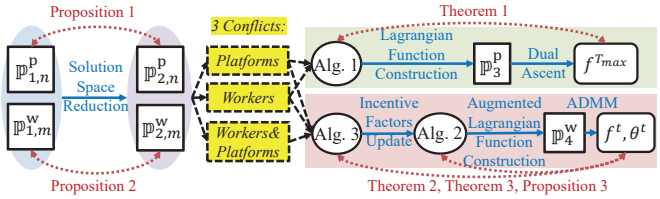


Fig. 4. Roadmap of Theoretical Analysis for MACRO

$$\sum_{n=1}^{N}\sum_{m=1}^{M}u^{\mathrm{p}}_{n,m}(f^{*}_{n,m})\leq\sum_{n=1}^{N}\sum_{m=1}^{M}u^{\mathrm{p}}_{n,m}(f^{t}_{n,m})+$$
$$\sum_{n=1}^{N}\sum_{m=1}^{M}\frac{\mathrm{d}u^{\mathrm{p}}_{n,m}(f^{t}_{n,m})}{\mathrm{d}f_{n,m}}(f^{*}_{n,m}-f^{t}_{n,m})-\sum_{n=1}^{N}\sum_{m=1}^{M}\frac{\delta}{2}|f^{*}_{n,m}-f^{t}_{n,m}|^2. \quad (38)$$

We plus (36) on the left side of (38), and plus (35) and (37) on the right side, and we obtain that $\tilde{\mathcal{L}}(\boldsymbol{f}^{*},\boldsymbol{\lambda}^{*},\boldsymbol{\mu}^{*})$

$$\geq\tilde{\mathcal{L}}(\boldsymbol{f}^{t},\boldsymbol{\lambda}^{t},\boldsymbol{\mu}^{t})+(\delta/2)||\boldsymbol{f}^{*}-\boldsymbol{f}^{t}||_2^2+\mathbf{E}^{T}(\nabla_{\boldsymbol{f}}\tilde{\mathcal{L}}(\boldsymbol{f}^{t},\boldsymbol{\lambda}^{t},\boldsymbol{\mu}^{t}))^{T}(\boldsymbol{f}^{*}-\boldsymbol{f}^{t})\mathbf{E}$$
$$\geq\min_{\boldsymbol{f}'}\{\tilde{\mathcal{L}}(\boldsymbol{f}^{t},\boldsymbol{\lambda}^{t},\boldsymbol{\mu}^{t})+\frac{\delta}{2}||\boldsymbol{f}'-\boldsymbol{f}^{t}||_2^2+\mathbf{E}^{T}(\nabla_{\boldsymbol{f}}\tilde{\mathcal{L}}(\boldsymbol{f}^{t},\boldsymbol{\lambda}^{t},\boldsymbol{\mu}^{t}))^{T}(\boldsymbol{f}'-\boldsymbol{f}^{t})\mathbf{E}\}$$
$$=\tilde{\mathcal{L}}(\boldsymbol{f}^{t},\boldsymbol{\lambda}^{t},\boldsymbol{\mu}^{t})-(1/2\delta)||\nabla_{\boldsymbol{f}}\tilde{\mathcal{L}}(\boldsymbol{f}^{t},\boldsymbol{\lambda}^{t},\boldsymbol{\mu}^{t})||_2^2,$$

where $\boldsymbol{E}$ is a unit-column vector. Then we further have

$$||\nabla_{\boldsymbol{f}}\tilde{\mathcal{L}}(\boldsymbol{f}^{t},\boldsymbol{\lambda}^{t},\boldsymbol{\mu}^{t})||_2^2\geq 2\delta(\tilde{\mathcal{L}}(\boldsymbol{f}^{t},\boldsymbol{\lambda}^{t},\boldsymbol{\mu}^{t})-\tilde{\mathcal{L}}(\boldsymbol{f}^{*},\boldsymbol{\lambda}^{*},\boldsymbol{\mu}^{*})). \quad (39)$$

Upon the convexity of $\tilde{\mathcal{L}}(\boldsymbol{f},\boldsymbol{\lambda},\boldsymbol{\mu})$ and the Lipschitz condition [67] with parameter κ for $u^{\mathrm{w}}_{n,m}(f_{n,m})$, $\tilde{\mathcal{L}}(\boldsymbol{f}^{t},\boldsymbol{\lambda}^{t},\boldsymbol{\mu}^{t})$

$$\geq\tilde{\mathcal{L}}(\boldsymbol{f}^{t+1},\boldsymbol{\lambda}^{t+1},\boldsymbol{\mu}^{t+1})+\sum_{n=1}^{N}\sum_{m=1}^{M}((\frac{\partial\tilde{\mathcal{L}}(\boldsymbol{f}^{t+1},\boldsymbol{\lambda}^{t+1},\boldsymbol{\mu}^{t+1})}{\partial f_{n,m}})$$
$$\times(1/\kappa)(\mathrm{d}u^{\mathrm{w}}_{n,m}(f^{t+1}_{n,m})/\mathrm{d}f_{n,m}-\mathrm{d}u^{\mathrm{w}}_{n,m}(f^{t}_{n,m})/\mathrm{d}f_{n,m})). \quad (40)$$

Based on (25), incorporating (39) into (40), we finally get

$$\frac{\tilde{\mathcal{L}}(\boldsymbol{f}^{t+1},\boldsymbol{\lambda}^{t+1},\boldsymbol{\mu}^{t+1})-\tilde{\mathcal{L}}(\boldsymbol{f}^{*},\boldsymbol{\lambda}^{*},\boldsymbol{\mu}^{*})}{\tilde{\mathcal{L}}(\boldsymbol{f}^{t},\boldsymbol{\lambda}^{t},\boldsymbol{\mu}^{t})-\tilde{\mathcal{L}}(\boldsymbol{f}^{*},\boldsymbol{\lambda}^{*},\boldsymbol{\mu}^{*})}\leq 1-\frac{2\delta}{\kappa}, \quad (41)$$

which shows that the MACRO mechanism for MSE in **Alg. 3** converges linearly. To satisfy inequation (23) for the convergence of **Alg. 3**, we get the required iteration round

$$t=\frac{\log_2(\mathcal{L}(\boldsymbol{f}^{0},\boldsymbol{\lambda}^{0},\boldsymbol{\mu}^{0})-\mathcal{L}(\boldsymbol{f}^{*},\boldsymbol{\lambda}^{*},\boldsymbol{\mu}^{*}))-\log_2(\varepsilon\kappa/(2\delta))}{\log_2(1/(1-2\delta/\kappa))}, \quad (42)$$

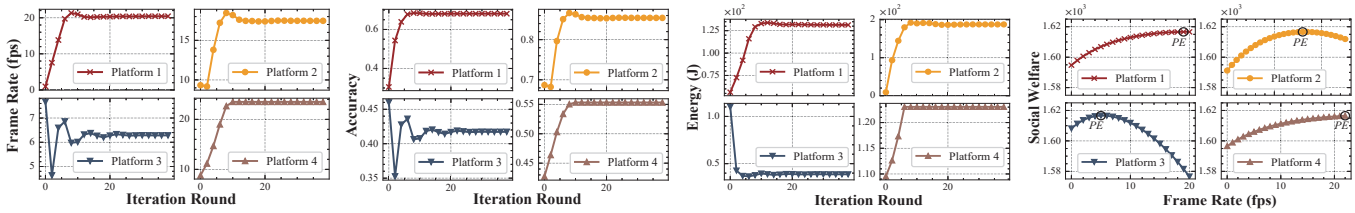
where $\boldsymbol{f}^{0}$, $\boldsymbol{\lambda}^{0}$ and $\boldsymbol{\mu}^{0}$ are the initial frame rate decision and Lagrangian multipliers, and the required iteration round for convergence does not depend on N and M. □

Conclusively, to enhance comprehension, a roadmap of the theoretical analysis for MACRO is shown in Fig. 4, elucidating the relationships among all problems, algorithms, propositions and theorems discussed in this work.

V. Experiments and Result Analysis

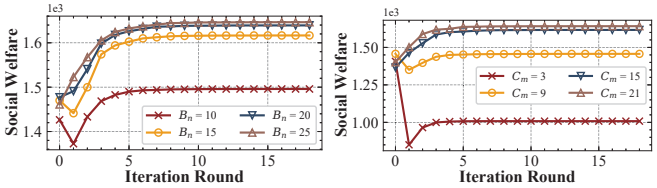
A. Experiment Settings

Our testbed-based experiments are conducted on Raspberry Pi and Jetson Xavier, acting as platforms and workers. Specifically, platforms implemented based on Raspberry Pi wirelessly transmit videos from datasets PANDA [43] and AICity [68] to workers, characterized as diverse analytics models in {yolov7, yolov7-x, yolov7-d6, yolov7-e6e} deployed on Jetson Xavier. Besides, platforms and workers wirelessly exchange parameters, including incentive factors and frame rate decisions. Measured by AITEK AWE2101 [53], the energy consumption for computation is uniformly set as 5 J per frame, and the transmission energy consumption is $\gamma^{\mathrm{d}}_{m}\sim N(5,0.5)\times 10^{-6}$ (J/bit).



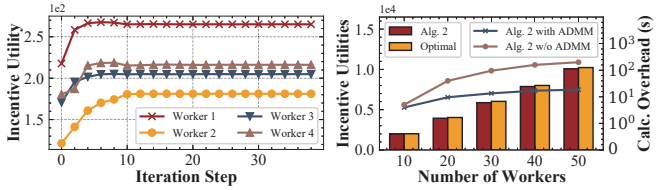
(a) Convergence of Frame Rate (b) Convergence of Accuracy (c) Convergence of Energy Consumption (d) Pareto Efficiency

Fig. 5. Convergence and Pareto Efficiency of **Alg. 1**: Dual Ascent for PE in Multi-platform Game



(a) Varying B_n (b) Varying C_m

Fig. 6. Varying B_n and C_m for **Alg. 1**



(a) Convergence of Incentive Utility (b) Effectivenes of ADMM

Fig. 7. Convergence and Calc. Overhead of ADMM-based **Alg. 2**

The revenues G_n and ω_m are generated based on dataset Sales Product [69]. According to [33, 55], the bandwidth budget B_n is set in $[10, 25]$ and computation capacity C_m is set in $[3, 21]$.

In our experiments, we concentrate primarily on the task of object detection [70], a prevalent video analytics task. For each object detection task dispatched from crowdsourcing platform n to worker m, we evaluate the accuracy as follows. Taking the maximum frame rate $F_n = 30$ as an example, when the frame rate decision $f_{n,m}$ is set to 15, it implies that the sampling rate is 0.5. Consequently, only the frames with even ids (starting from frame id 0) will be detected, while the frames with odd ids will adopt the detection results of the preceding even-numbered frame as their own. The detection results for each frame i is denoted as L_i^{Det}, a list of bounding boxes. In addition, we capture the detection results as ground truth L_i^{GT} by setting the frame sampling rate to 1. For each bounding box bbx^{Det} in L_i^{Det}, we judge whether there exists another bounding box in L_i^{GT} such that the 2 bounding boxes' Intersection Over Union (IOU) [71] exceeds a preset threshold (e.g., 80%). If such a bounding box exists, we classify bbx^{Det} as True Positive (TP) [72]. We use tp_i to denote the number of TP bounding boxes in frame i. Then the F1-score [47] based accuracy for the above object detection task is evaluated as

$$a_{n,m} = \mathbb{E}_i[\frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}] = \mathbb{E}_i[\frac{2 \cdot tp_i}{\text{len}(L_i^{\text{Det}}) + \text{len}(L_i^{\text{GT}})}], \quad (43)$$

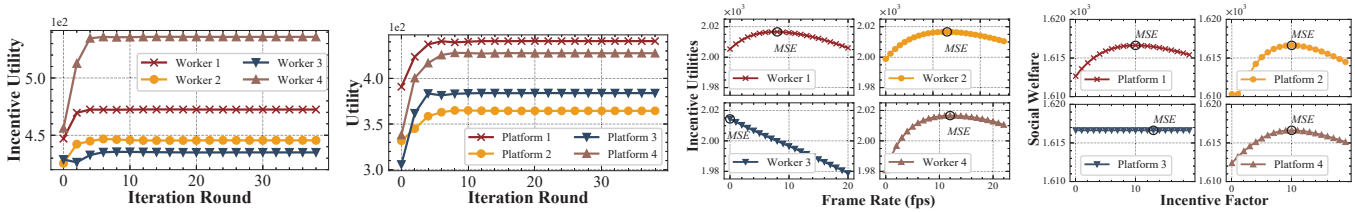
where $\text{len}(L)$ calculates the number of bounding boxes in L, and $\mathbb{E}[\cdot]_i$ means the average value for all frames.

We compare our mechanism with other methods for video analytics crowdsourcing as follows: *LOL* [25] only considers the conflicts among workers; *Crowd*^{2} [26] only considers the conflicts among multiple platforms and among multiple workers; *CSPW* [28] only considers the Conflicts between Single Platform and multiple Workers; *RAN* serves as a trivial baseline in a RANdomized way of determining configurations.

B. Experiment Results

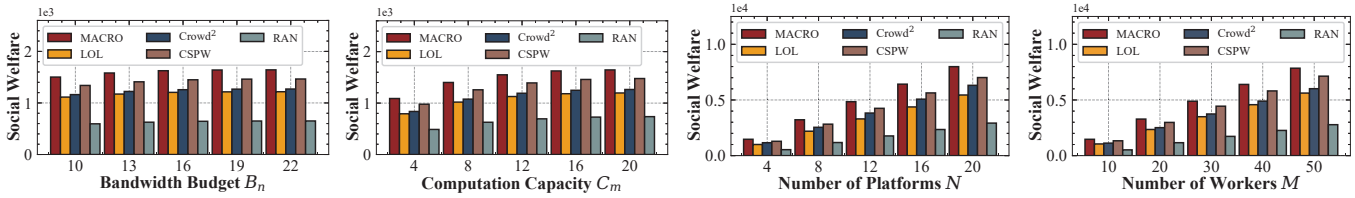
Convergence for Pareto Efficiency. We show the convergence of **Alg. 1** based on the dual ascent method in Fig. 5. Notably, at about the 15th iteration round, the frame rate decision, accuracy and energy for each platform all keep a steady state, which indicates the convergence of **Alg. 1**. Concretely, as demonstrated in Fig. 5(a), after **Alg. 1** converges, platform 1 selects the frame rate of 20 fps, and platform 2 selects 18 fps. However, the accuracy and energy consumption for platform 1 are lower than platform 2, as shown in Figs. 5(b) and 5(c), since platform 1 may choose a worker with the energy-saving mobile device running a low-resolution model, which verifies the heterogeneity of video analytics upon crowdsourcing. We then show the platforms' PE obtained from **Alg. 1** in Fig. 5(d). We notice that when each platform changes its frame rate to optimize its own utility, the social welfare will be decreased, which has a negative impact on others. Hence, each platform cannot improve its own utility unless others' utility is reduced, and it is consistent with the definition of PE. We further vary the values of B_n and C_m and observe their effects on the convergence of **Alg. 1** in Fig. 6. When B_n is increased from 10 to 25, the social welfare obtained by **Alg. 1** is improved as shown in Fig. 6(a). It is because each platform has more bandwidth budget to increase its frame rate decision for higher accuracy, thus resulting in higher profit. However, with the rise of B_n, we observe that the increase rate of social welfare gets slower due to the fact that when the bandwidth budget is high enough, the profits cannot rise any further. Similarly, when C_m varies from 3 to 21, the increase rate of social welfare slows down with sufficient computation resources, as shown in Fig. 6(b).

Effectiveness of ADMM. We demonstrate the convergence of ADMM-based **Alg. 2** in Fig. 7(a). When **Alg. 2** is running, given the incentive factors $\boldsymbol{\theta}$ from platforms, the workers sequentially update their frame rates to optimize a series of subproblems in Eq. (19), which are decoupled from problem $\mathbb{P}_4^{\text{pw}}$ by ADMM. Notably, at about the 10th iteration step, **Alg. 2**, serving as the inner layer of MACRO, converges to the maximum of all workers' incentive utilities. Moreover, we compare the incentive utilities obtained from **Alg. 2** with the optimum, as shown in Fig. 7(b). Since **Alg. 2** based on ADMM decouples the complex problem of maximizing the incentive utilities of all workers into multiple subproblems, it can lead to a de-



(a) Incentive Utility for Workers (b) Utility for Platforms (c) MSE for Workers (d) MSE for Platforms

Fig. 8. Convergence and Multi-leader Stackelberg Equilibrium of **Alg. 3**: Incentive Mechanism MACRO for MSE



(a) Varying Bandwidth Budget (b) Varying Computation Capacity (c) Varying Number of Platforms (d) Varying Number of Workers

Fig. 9. Scalability of MACRO on Bandwidth Budget, Computation Capacity, Number of Platforms and Number of Workers Compared with Other Methods

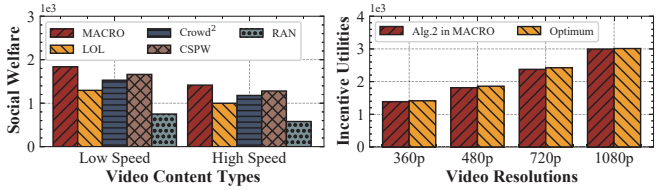
crease in incentive utilities. However, from Fig. 7(b), we see that ADMM-based **Alg. 2** can obtain almost as much incentive utilities as the optimum. In addition, compared to the modified **Alg. 2** without ADMM, which causes a computing overhead of around 200s when the number of workers is 50, the overhead of ADMM-based **Alg. 2** keeps stable, as shown in Fig. 7(b), highlighting the effectiveness and efficiency of ADMM.

Convergence for MSE. We further show that the utility for platforms and incentive utility for workers can both coverage to MSE in Fig. 8. As shown in Figs. 8(a) and 8(b), the workers' incentive utilities and platforms' utilities all converge at about the 10th iteration round and then remain stable, which verifies the convergence of MACRO described in **Theorem 2**. When **Alg. 3** has converged, we further change each worker's frame rate decision and observe its effect on the total incentive utilities. We find that each worker cannot adjust its frame rate to improve its incentive utility without reducing others', as demonstrated in Fig. 8(c). Moreover, Fig. 8(d) shows that each platform cannot change its incentive factor to raise its utility.

Scalability of MACRO. We finally illustrate the scalability of MACRO with respect to different bandwidth budgets, computation capacities, numbers of platforms and numbers of workers compared with other approaches in Fig. 9. Similar to the results in Figs. 6(a) and 6(b), when the values of B_n and C_m rise, platforms' social welfare also increase, but the rate of increase slows down, as shown in Figs. 9(a) and 9(b). Moreover, we change the number of platforms and workers and compare MACRO with other methods, and the results are illustrated in Figs. 9(c) and 9(d). Since LOL and Crowd^2 only take the conflicts among workers and among platforms into consideration, their social welfare is lower than CSPW when there exist the conflicts between platform and workers. Nevertheless, the disadvantage of CSPW is that it cannot tackle the conflicts between multiple platforms and workers, which are well solved by MACRO. We observe that when the number of platforms varies from 4 to 20 and the number of platforms varies from 10 to 50, MACRO always outperforms the other methods since it simultaneously considers the conflicts among workers, among platforms, and between workers and platforms.

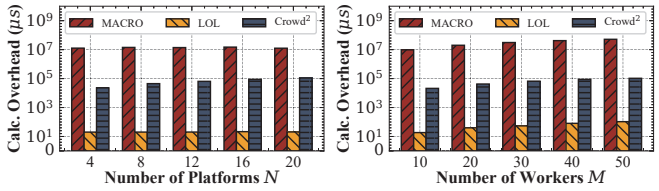
Furthermore, we test MACRO across various video content types and resolutions, as illustrated in Fig. 10. Concretely, in Fig. 10(a), we focus on low-speed and high-speed vehicle movements. We observe that the social welfare achieved by all algorithms, including MACRO, is higher for low-speed videos compared to the high-speed. This is because high-speed videos require higher configurations to achieve comparable analytical accuracy-based benefits. However, due to the bandwidth constraints on platforms, it is infeasible to infinitely increase the frame rates, which leads to lower social welfare obtained from high-speed videos. In Fig. 10(b), we vary the video resolutions and compare **Alg. 2** in MACRO with the optimum (as defined in Fig. 7(b)). As the resolution increases from 360p to 1080p, we find that the incentive utilities for workers also increase. The reason is that the compensations to workers are related to the energy consumption on their mobile devices, which is affected by the amount of video data. Thus, videos with higher resolutions result in higher incentive utilities. To summarize, MACRO improves the social welfare by 26% on average.

Overhead of MACRO. We further investigate the computational cost of MACRO compared with LOL and Crowd^2, which are mainly designed for video analytics crowdsourcing. We focus on the time cost associated with decisions on worker selection and video analytics configuration, excluding the time taken to run video analytics models. As shown in Fig. 11(a), the time cost of MACRO remains relatively stable, when we vary the number of platforms. The reason is that the bulk of MACRO's computations (as shown in Fig. 7(b)) are distributed across multiple platforms and workers via ADMM, with computations on platforms being executed in parallel, thus saving time. Besides, as showcased in Fig. 11(b), when the number of workers increases, there is an increase in MACRO's time cost, which is related to the serial computations performed on the workers in ADMM framework. However, we believe that the advantage of ADMM lies in its ability to distribute the high computational overhead of a complex problem. Additionally, as the number of platforms increases, we observe that the computation time for LOL remains stable. This is because it uses a single-agent multi-armed bandit method that can be executed



(a) Various Video Content Types (b) Various Video Resolutions

Fig. 10. MACRO Tested on Various Video Content Types and Resolutions



(a) Various Numbers of Platforms (b) Various Numbers of Workers

Fig. 11. Overhead of MACRO on Various Numbers of Platforms and Workers

on multiple platform in parallel. When the number of workers increases, the time for LOL to explore more workers also rises. For Crowd^2, which models worker selection as a multi-knapsack problem, the computational cost increases continuously with both the number of platforms and workers. In summary, we believe that although MACRO has a higher computational cost compared to other methods, it only needs to be run once at the beginning of the video analytics crowdsourcing scenario to decide the worker selection and video analytics configuration. Therefore, a decision time of around 10 seconds is considered acceptable in exchange for the benefits in social welfare it improves, as shown in Fig. 9, compared with other methods.

VI. Related Work

Worker Recruitment in Crowdsourcing. During the past decade, a substantial body of research has concentrated on crowdsourcing [14, 15, 25, 26, 38–40, 73–76]. Among them, some studies have addressed the problems of worker recruitment. For example, Song *et al.* [38] selected the optimal combination of workers minimizing the cumulative empirical entropy via combinatorial multi-armed bandit. Wang *et al.* [40] proposed a privacy-preserving framework for online task assignment and worker recruitment to minimize total travel distances. In summary, the aforementioned studies primarily focus on designing methods for recruiting crowdsourcing workers for generalized tasks. In contrast to them, our MACRO and other studies [25, 26] have considered the approaches to selecting workers for specific video analytics tasks. Different from LOL [25], which formulates a mixed integer program to select the most suitable workers for maximum profit via volatile multi-armed bandit-based Lyapunov optimization method, MACRO considers workers recruited from multiple platforms. Though $Crowd^2$ [26] tackles the conflicts between platforms for worker recruitment, MACRO simultaneously considers the conflicts among workers, among platforms, and between platforms and workers, filling an existing gap in research.

Incentive Mechanism with Game Theory. Some related works have explored incentive mechanisms with game theory [28, 36, 41, 42]. For example, Xu *et al.* [36] presented a novel incentive mechanism with multi-armed bandit and three-stage Stackelberg game to achieve a multi-win situation. However, the related studies [28, 36] have only considered the competitive dynamics between a single platform and multiple workers, employing Stackelberg game theory and incentive strategies to maximize their respective benefits. Different from them, our MACRO further considers the conflicts between multiple platforms using multi-leader game. Additionally, some of related studies [41, 42] have delved into the competitive relationships between numerous users and multiple task initiators or service providers. For example, Zhan *et al.* [41] investigated the DRL-based approach to assigning profitable amount of incentives to multiple task initiators and mobile users. However, their methodologies cannot be directly applied to the context of crowdsourcing for video analytics tasks, where we have investigated the conflicting interests between workers and platforms when selecting the video analytics configurations. Furthermore, the aforementioned studies do not account for the scenarios with a large number of workers, for which MACRO introduces an ADMM-based method to reduce computational complexity.

Configuration Selection for Video Analytics. Numerous existing studies [22–24, 48] focus on optimizing the completion of video analytics tasks by adjusting the configurations. For example, Zhang *et al.* [22] studied the impact of batch size on video transmission and analysis, based on which they proposed a DRL-based method for the batch size adaptation. To adaptively adjust the configurations (e.g., resolution), ASVA [23] was proposed as a streaming framework designed for live video analytics. To summarize, prior works have studied the adjustment of various configurations, including batch size of video data [22], video resolution [23], video frame rate [24], and bandwidth consumption [48]. However, there exists only one decision-maker to determine the configuration in each of the above studies. In contrast, MACRO addresses a scenario where both crowdsourcing platforms and workers possess the capability to select the optimal video analytics configuration for their own utilities, which is challenging. Besides, the optimization objectives for adjusting configurations in the above studies are also different, such as video analytics accuracy [22, 24], timeliness [23], resource consumption [48], and so on. Different from them, MACRO needs to simultaneously consider the platforms' accuracy-based utility and workers' energy-related utility, which are in conflict with each other.

VII. Conclusion

To simultaneously solve inter-platform and platform-worker conflicts for heterogeneous video analytics tasks upon crowdsourcing, we design an incentive mechanism MACRO based on multi-leader game. A dual ascent-based method is proposed to determine proper video analytics configurations for multi-platform game and reach the Pareto efficiency. Furthermore, for multi-leader game involving platform-worker conflicts, we design the incentive function and its incentive factor updating strategy, and present an ADMM-based incentive maximization method. Rigorous proofs show MACRO's linear convergence to the multi-leader Stackelberg equilibrium, and trace-driven experiments testify its great performance compared to others.

REFERENCES

[1] S. Peng, B. Zhang, Y. Yan, and C. Li, "A multi-platform cooperation based task assignment mechanism for mobile crowdsensing," *IEEE IoTJ*, 2023.

[2] W. Liu, Y. Yang, E. Wang, and J. Wu, "Dynamic user recruitment with truthful pricing for mobile crowdsensing," in *IEEE INFOCOM*, 2020, pp. 1113–1122.

[3] W. Liu, E. Wang, Y. Yang, and J. Wu, "Worker selection towards data completion for online sparse crowdsensing," in *IEEE INFOCOM*, vol. 33, 2022.

[4] C. Van Pelt and A. Sorokin, "Designing a scalable crowdsourcing platform," in *ACM SIGMOD*, 2012, pp. 765–766.

[5] G. Paolacci, J. Chandler, and P. G. Ipeirotis, "Running experiments on amazon mechanical turk," *Judgment and Decision making*, vol. 5, no. 5, pp. 411–419, 2010.

[6] H. Wang, C. H. Liu, Z. Dai, J. Tang, and G. Wang, "Energy-efficient 3d vehicular crowdsourcing for disaster response by distributed deep reinforcement learning," in *ACM SIGKDD*, 2021, pp. 3679–3687.

[7] K. Ali, D. Al-Yaseen, A. Ejaz, T. Javed, and H. S. Hassanein, "Crowdits: Crowdsourcing in intelligent transportation systems," in *IEEE WCNC*, 2012, pp. 3307–3311.

[8] K. Hara, J. Sun, R. Moore, D. Jacobs, and J. Froehlich, "Tohme: detecting curb ramps in google street view using crowdsourcing, computer vision, and machine learning," in *ACM UIST*, 2014, pp. 189–204.

[9] P. Bellavista, A. Corradi, L. Foschini, and R. Ianniello, "Scalable and cost-effective assignment of mobile crowdsensing tasks based on profiling trends and prediction: The participact living lab experience," *Sensors*, vol. 15, no. 8, pp. 18 613–18 640, 2015.

[10] J. Chang, P. Gebhard, A. Haeberlen, Z. Ives, I. Lee, O. Sokolsky, and K. K. Venkatasubramanian, "Trustforge: Flexible access control for collaborative crowd-sourced environment," in *2013 Eleventh Annual Conference on Privacy, Security and Trust*. IEEE, 2013, pp. 291–300.

[11] X. Zhang, Z. Yang, Z. Zhou, H. Cai, L. Chen, and X. Li, "Free market of crowdsourcing: Incentive mechanism design for mobile sensing," *IEEE transactions on parallel and distributed systems*, vol. 25, no. 12, pp. 3190–3200, 2014.

[12] G. Zhuo, Q. Jia, L. Guo, M. Li, and P. Li, "Privacy-preserving verifiable data aggregation and analysis for cloud-assisted mobile crowdsourcing," in *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*. IEEE, 2016, pp. 1–9.

[13] H. Jin, L. Su, H. Xiao, and K. Nahrstedt, "Inception: Incentivizing privacy-preserving data aggregation for mobile crowd sensing systems," in *Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2016, pp. 341–350.

[14] T. Ren, X. Zhou, K. Li, Y. Gao, J. Zhang, and K. Li, "Efficient cross dynamic task assignment in spatial crowdsourcing," in *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 2023, pp. 1420–1432.

[15] X. Chen, Y. Zhao, K. Zheng, B. Yang, and C. S. Jensen, "Influence-aware task assignment in spatial crowdsourcing," in *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 2022, pp. 2141–2153.

[16] Y. Kang, X. Miao, K. Liu, L. Chen, and Y. Liu, "Quality-aware online task assignment in mobile crowdsourcing," in *2015 IEEE 12th International Conference on Mobile Ad Hoc and Sensor Systems*. IEEE, 2015, pp. 127–135.

[17] X. Li, Y. Wang, B. Xie, L. Meng, Z. Liu, X. Tong, A. Zhou, and Z. Cai, "Planning-based mobile crowdsourcing bidirectional multi-stage online task assignment," *Computer Networks*, vol. 233, p. 109879, 2023.

[18] S. Huber, L. Demetz, and M. Felderer, "Analysing the performance of mobile cross-platform development approaches using ui interaction scenarios," in *Software Technologies: 14th International Conference, ICSOFT 2019, Prague, Czech Republic, July 26–28, 2019, Revised Selected Papers 14*. Springer, 2020, pp. 40–57.

[19] D. Carvalho, L. Rodrigues, P. T. Endo, S. Kosta, and F. A. Silva, "Mobile edge computing performance evaluation using stochastic petri nets," in *2020 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2020, pp. 1–6.

[20] M.-H. Guo, T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, T.-J. Mu, S.-H. Zhang, R. R. Martin, M.-M. Cheng, and S.-M. Hu, "Attention mechanisms in computer vision: A survey," *Computational visual media*, vol. 8, no. 3, pp. 331–368, 2022.

[21] S. Xu, J. Wang, W. Shou, T. Ngo, A.-M. Sadick, and X. Wang, "Computer vision techniques in construction: a critical review," *Archives of Computational Methods in Engineering*, vol. 28, pp. 3383–3397, 2021.

[22] L. Zhang, Y. Zhang, X. Wu, F. Wang, L. Cui, Z. Wang, and J. Liu, "Batch adaptative streaming for video analytics," in *IEEE INFOCOM*, 2022, pp. 2158–2167.

[23] M. Zhang, F. Wang, and J. Liu, "Casva: Configuration-adaptive streaming for live video analytics," in *IEEE INFOCOM*, 2022, pp. 2168–2177.

[24] Y. Li, A. Padmanabhan, P. Zhao, Y. Wang, G. H. Xu, and R. Netravali, "Reducto: On-camera filtering for resource-efficient real-time video analytics," in *ACM SIGCOMM*, 2020, pp. 359–376.

[25] Y. Chen, S. Zhang, Y. Jin, Z. Qian, M. Xiao, N. Chen, and Z. Ma, "Learning for crowdsourcing: Online dispatch for video analytics with guarantee," in *IEEE INFOCOM*, 2022, pp. 1908–1917.

[26] Y. Chen, S. Zhang, Y. Yan, Y. Jin, N. Chen, M. Ji, and M. Xiao, "Crowd^2: Multi-agent bandit-based dispatch for video analytics upon crowdsourcing," in *IEEE INFOCOM*, 2023.

[27] J. Jiang, G. Ananthanarayanan, P. Bodik, S. Sen, and I. Stoica, "Chameleon: scalable adaptation of video analytics," in *ACM SIGCOMM*, 2018, pp. 253–266.

[28] M. Xiao, Y. Xu, J. Zhou, J. Wu, S. Zhang, and J. Zheng, "Aoi-aware incentive mechanism for mobile crowdsensing using stackelberg game," in *IEEE INFOCOM*, 2023.

[29] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using yolo: Challenges, architectural successors, datasets and applications," *multimedia Tools and Applications*, vol. 82, no. 6, pp. 9243–9275, 2023.

[30] S. Zhai, D. Shang, S. Wang, and S. Dong, "Df-ssd: An improved ssd object detection algorithm based on densenet and feature fusion," *IEEE access*, vol. 8, pp. 24 344–24 357, 2020.

[31] D. Wu, J. Wang, R. Q. Hu, Y. Cai, and L. Zhou, "Energy-efficient resource sharing for mobile device-to-device multimedia communications," *IEEE Trans. Veh. Technol.*, vol. 63, no. 5, pp. 2093–2103, 2014.

[32] M. A. Hoque, M. Siekkinen, and J. K. Nurminen, "Energy efficient multimedia streaming to mobile devices—a survey," *IEEE Commun. Surv. Tutor.*, vol. 16, no. 1, pp. 579–597, 2012.

[33] T. Adiono, S. Fuada, and S. Harimurti, "Bandwidth budget analysis for visible light communication systems utilizing commercially available components," in *IEEE ICEEE*, 2017, pp. 1375–1380.

[34] Y. Wu, Y. Wang, and G. Cao, "Photo crowdsourcing for area coverage in resource constrained environments," in *IEEE INFOCOM*, 2017, pp. 1–9.

[35] Z. Xiong, J. Kang, D. Niyato, P. Wang, and H. V. Poor, "Cloud/edge computing service management in blockchain networks: Multi-leader multi-follower game-based admm for pricing," *IEEE Transactions on Services Computing*, vol. 13, no. 2, pp. 356–367, 2019.

[36] Y. Xu, M. Xiao, J. Wu, S. Zhang, and G. Gao, "Incentive mechanism for spatial crowdsourcing with unknown social-aware workers: A three-stage stackelberg game approach," *IEEE TMC*, 2022.

[37] J. Zhang and Q. Zhang, "Stackelberg game for utility-based cooperative cognitiveradio networks," in *ACM MobiHoc*, 2009, pp. 23–32.

[38] Y. Song and H. Jin, "Minimizing entropy for crowdsourcing with combinatorial multi-armed bandit," in *IEEE INFOCOM*, 2021, pp. 1–10.

[39] Z. Shi, S. Jiang, L. Zhang, Y. Du, and X.-Y. Li, "Crowdsourcing system for numerical tasks based on latent topic aware worker reliability," in *IEEE INFOCOM*, 2021, pp. 1–10.

[40] H. Wang, E. Wang, Y. Yang, J. Wu, and F. Dressler, "Privacy-preserving online task assignment in spatial crowdsourcing: A graph-based approach," in *IEEE INFOCOM*, 2022, pp. 570–579.

[41] Y. Zhan, C. H. Liu, Y. Zhao, J. Zhang, and J. Tang, "Free market of multi-leader multi-follower mobile crowdsensing: An incentive mechanism design by deep reinforcement learning," *IEEE TMC*, vol. 19, no. 10, pp. 2316–2329, 2019.

[42] J. Nie, J. Luo, Z. Xiong, D. Niyato, P. Wang, and H. V. Poor, "A multi-leader multi-follower game-based analysis for incentive mechanisms in socially-aware mobile crowdsensing," *IEEE TWC*, vol. 20, no. 3, pp. 1457–1471, 2020.

[43] X. Wang, X. Zhang, Y. Zhu, Y. Guo, X. Yuan, L. Xiang, Z. Wang, G. Ding, D. J. Brady, Q. Dai, and L. Fang, "Panda: A gigapixel-level human-centric video dataset," in *IEEE CVPR*, 2020.

[44] D. A. Iancu and N. Trichakis, "Pareto efficiency in robust optimization," *INFORMS Management Science*, vol. 60, no. 1, pp. 130–147, 2014.

[45] G. Raghavan, A. Salomaki, and R. Lencevicius, "Model based estimation and verification of mobile device performance," in *ACM EMSOFT*, 2004, pp. 34–43.

[46] J. Dong and Y. Ye, "Adaptive downsampling for high-definition video coding," *IEEE TCSVT*, vol. 24, no. 3, pp. 480–488, 2014.

[47] C. Goutte and E. Gaussier, "A probabilistic interpretation of precision, recall and f-score, with implication for evaluation," in *European conference on information retrieval*. Springer, 2005, pp. 345–359.

[48] C. Wang, S. Zhang, Y. Chen, Z. Qian, J. Wu, and M. Xiao, "Joint configuration adaptation and bandwidth allocation for edge-based real-time video analytics," in *IEEE INFOCOM*, 2020, pp. 257–266.

[49] E. C. Norton, B. E. Dowd, and M. L. Maciejewski, "Marginal effects—quantifying the effect of changes in risk factors in logistic regression models," *Jama*, vol. 321, no. 13, pp. 1304–1305, 2019.

[50] Y. Ma, G. Zhou, and S. Wang, "Wifi sensing with channel state information: A survey," *ACM CSUR*, vol. 52, no. 3, pp. 1–36, 2019.

[51] E. Dahlman, S. Parkvall, and J. Skold, *4G: LTE/LTE-advanced for mobile broadband*. Academic press, 2013.

[52] M. Shafi, A. F. Molisch, P. J. Smith, T. Haustein, P. Zhu, P. De Silva, F. Tufvesson, A. Benjebbour, and G. Wunder, "5g: A tutorial overview of standards, trials, challenges, deployment, and practice," *IEEE JSAC*, vol. 35, no. 6, pp. 1201–1221, 2017.

[53] X. Cheng, J. Yu, M. Zhang, F. Xia, Z. Zhang, and M. Ji, "An energy efficient task scheduling for a cloud-edge system," in *Journal of Physics: Conference Series*, vol. 2425, no. 1. IOP Publishing, 2023, p. 012028.

[54] C. Zhou, C.-K. Tham, and M. Motani, "Online auction for scheduling concurrent delay tolerant tasks in crowdsourcing systems," *Elsevier CN*, vol. 169, pp. 107 045–107 061, 2020.

[55] X. Wang, J. Ye, and J. C. Lui, "Decentralized task offloading in edge computing: A multi-user multi-armed bandit approach," in *IEEE INFOCOM*, 2022, pp. 1199–1208.

[56] Z. Luo and P. Tseng, "On the convergence rate of dual ascent methods for linearly constrained convex minimization," *INFORMS Mathematics of Operations Research*, vol. 18, no. 4, pp. 846–867, 1993.

[57] M. Fortin and R. Glowinski, *Augmented Lagrangian methods: applications to the numerical solution of boundary-value problems*. Elsevier, 2000.

[58] K. Dalkir, *Knowledge management in theory and practice*. routledge, 2013.

[59] D. M. Kreps, "Nash equilibrium," in *Game Theory*. Springer, 1989, pp. 167–177.

[60] Z. Zheng, L. Song, Z. Han, G. Y. Li, and H. V. Poor, "Game theory for big data processing: Multileader multifollower game-based admm," *IEEE Transactions on Signal Processing*, vol. 66, no. 15, pp. 3933–3945, 2018.

[61] J.-J. Laffont and J. Tirole, *A theory of incentives in procurement and regulation*. MIT press, 1993.

[62] D. Han and X. Yuan, "A note on the alternating direction method of multipliers," *Springer Journal of Optimization Theory and Applications*, vol. 155, pp. 227–238, 2012.

[63] T.-Y. Lin, S.-Q. Ma, and S.-Z. Zhang, "On the sublinear convergence rate of multi-block admm," *Springer ORSC*, vol. 3, pp. 251–274, 2015.

[64] D. Ding, K. Zhang, T. Basar, and M. Jovanovic, "Natural policy gradient primal-dual method for constrained markov decision processes," *NeurIPS*, vol. 33, pp. 8378–8390, 2020.

[65] M. Hong and Z.-Q. Luo, "On the linear convergence of the alternating direction method of multipliers," *Springer Mathematical Programming*, vol. 162, no. 1-2, pp. 165–199, 2017.

[66] G. Gordon and R. Tibshirani, "Karush-kuhn-tucker conditions," *Optimization*, vol. 10, no. 725/36, p. 725, 2012.

[67] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[68] M. Naphade, Z. Tang, M.-C. Chang, D. C. Anastasiu, A. Sharma, R. Chellappa, S. Wang, P. Chakraborty, T. Huang, J.-N. Hwang, and S. Lyu, "The 2019 ai city challenge," in *IEEE CVPR Workshops*, 2019, p. 452–460.

[69] "Sales Product Data," https://www.kaggle.com/datasets/knightbearr/sales-product-data, 2019.

[70] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proceedings of the IEEE*, 2023.

[71] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *IEEE/CVF CVPR*, 2019, pp. 658–666.

[72] E. Lepeschkin and B. Surawicz, "Characteristics of true-positive and false-positive results of electrocardiographs master two-step exercise tests," *New England Journal of Medicine*, vol. 258, no. 11, pp. 511–520, 1958.

[73] B. Li, Y. Cheng, Y. Yuan, Y. Yang, Q. Jin, and G. Wang, "Acta: Autonomy and coordination task assignment in spatial crowdsourcing platforms," *Proceedings of the VLDB Endowment*, vol. 16, no. 5, pp. 1073–1085, 2023.

[74] Y. Zhao, K. Zheng, Z. Wang, L. Deng, B. Yang, T. B. Pedersen, C. S. Jensen, and X. Zhou, "Coalition-based task assignment with priority-aware fairness in spatial crowdsourcing," *The VLDB Journal*, pp. 1–22, 2023.

[75] A. Drutsa, V. Fedorova, D. Ustalov, O. Megorskaya, E. Zerminova, and D. Baidakova, "Crowdsourcing practice for efficient data labeling: Aggregation, incremental relabeling, and pricing," in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 2020, pp. 2623–2627.

[76] R. Shraga, C. Scharf, R. Ackerman, and A. Gal, "Incognitomatch: Cognitive-aware matching via crowdsourcing," in *Proceedings of the 2020 ACM SIGMOD international conference on management of data*, 2020, pp. 2753–2756.