AoI-Guaranteed Incentive Mechanism for Mobile Crowdsensing With Freshness Concerns

Yin Xu, Mingjun Xiao, Yu Zhu, Jie Wu, Sheng Zhang, Jinrui Zhou

Published in: IEEE Transactions on Mobile Computing, 2024

Presented by: Erez Weintraub
Department of Electrical and Computer Engineering, Technion

Agenda

Mobile Crowdsensing (MCS)

- MCS: Platform stimulates some workers (a.k.a., mobile users) via social networks to periodically collect the desired data from a group of points-of-interest (PoI) to provide data services (e.g., traffic monitoring, environmental sensing, etc.) for requesters.

Mobile Crowdsensing (MCS)

- MCS: Platform stimulates some workers (a.k.a., mobile users) via social networks to periodically collect the desired data from a group of points-of-interest (PoI) to provide data services (e.g., traffic monitoring, environmental sensing, etc.) for requesters.

- Studies of MCS include for example: incentive mechanism design, privacy preserving approaches, task allocation schemes.

Mobile Crowdsensing (MCS)

- MCS: Platform stimulates some workers (a.k.a., mobile users) via social networks to periodically collect the desired data from a group of points-of-interest (PoI) to provide data services (e.g., traffic monitoring, environmental sensing, etc.) for requesters.

- Studies of MCS include for example: incentive mechanism design, privacy preserving approaches, task allocation schemes.

- This paper concentrates on the MCS incentive mechanism design, with concerns about the freshness of sensing data and workers' social benefits.

Mobile Crowdsensing (MCS)

- MCS: Platform stimulates some workers (a.k.a., mobile users) via social networks to periodically collect the desired data from a group of points-of-interest (PoI) to provide data services (e.g., traffic monitoring, environmental sensing, etc.) for requesters.

- Studies of MCS include for example: incentive mechanism design, privacy preserving approaches, task allocation schemes.

- This paper concentrates on the MCS incentive mechanism design, with concerns about the freshness of sensing data and workers' social benefits.
  - Age of Information (AoI) – the elapsed time of data from being collected by the worker to being received and processed by the platform.
    worker i $(i \in \mathcal{N})$, denoted by $\delta_i(t)$, is defined as: $\delta_i(t) = t - U_i(t)$.

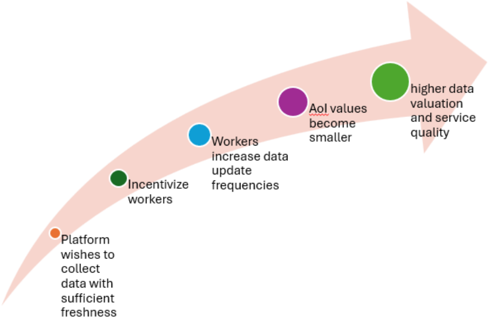
Mobile Crowdsensing (MCS)

- MCS: Platform stimulates some workers (a.k.a., mobile users) via social networks to periodically collect the desired data from a group of points-of-interest (PoI) to provide data services (e.g., traffic monitoring, environmental sensing, etc.) for requesters.

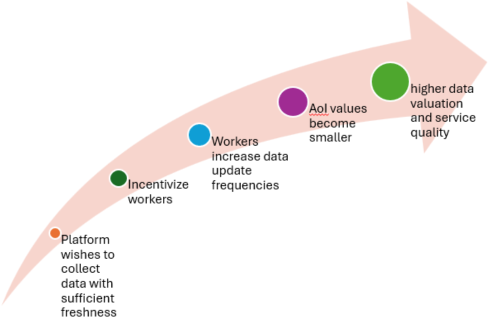
- Studies of MCS include for example: incentive mechanism design, privacy preserving approaches, task allocation schemes.

- This paper concentrates on the MCS incentive mechanism design, with concerns about the freshness of sensing data and workers' social benefits.
  - Age of Information (AoI) – the elapsed time of data from being collected by the worker to being received and processed by the platform.
    worker i ($i \in \mathcal{N}$), denoted by $\delta_i(t)$, is defined as: $\delta_i(t) = t - U_i(t)$.
  - Workers share their collected data with their social neighbors to obtain extra social benefits (i.e., additional utility from data sharing among workers).

Incentive Mechanism Challenges

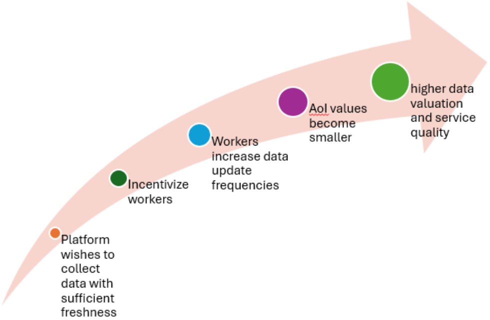


Incentive Mechanism Challenges



- Increasing data update frequency can result in higher worker costs and congestion on the receiving data platform queues.

Incentive Mechanism Challenges



- Increasing data update frequency can result in higher worker costs and congestion on the receiving data platform queues.
- Incomplete information – workers social relationships unknown or incomplete.

Major Contributions

- **Solving the problem of incentive mechanism design for MCS systems with data freshness concerns using novel incomplete information two-stage Stackelberg game with constraints, while considering workers' social benefits.**

Major Contributions

- Solving the problem of incentive mechanism design for MCS systems with data freshness concerns using novel incomplete information two-stage Stackelberg game with constraints, while considering workers' social benefits.

- **Utilize AoI metric to measure the freshness of data and derive the closed-form expression for the AoI of the data each worker uploads to the platform taking workers' social influences into account.**

Major Contributions

- Solving the problem of incentive mechanism design for MCS systems with data freshness concerns using novel incomplete information two-stage Stackelberg game with constraints, while considering workers' social benefits.

- Utilize AoI metric to measure the freshness of data and derive the closed-form expression for the AoI of the data each worker uploads to the platform taking workers' social influences into account.

- **Propose AIM, when all participants share the utility function parameters of the Stackelberg game. By deriving the optimal strategy for each participant, AIM can ensure that the platform and workers obtain their maximum utilities. Also theoretically prove that these optimal strategies constitute a unique Stackelberg equilibrium.**

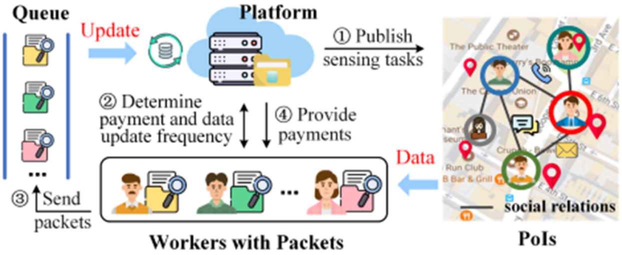
Major Contributions

- Solving the problem of incentive mechanism design for MCS systems with data freshness concerns using novel incomplete information two-stage Stackelberg game with constraints, while considering workers' social benefits.
- Utilize AoI metric to measure the freshness of data and derive the closed-form expression for the AoI of the data each worker uploads to the platform taking workers' social influences into account.
- Propose AIM, when all participants share the utility function parameters of the Stackelberg game. By deriving the optimal strategy for each participant, AIM can ensure that the platform and workers obtain their maximum utilities. Also theoretically prove that these optimal strategies constitute a unique Stackelberg equilibrium.
- **Propose DIM, when each participant has no prior knowledge of the game. Based on the DRL technique, DIM enables learning the optimal strategy directly from game experiences.**

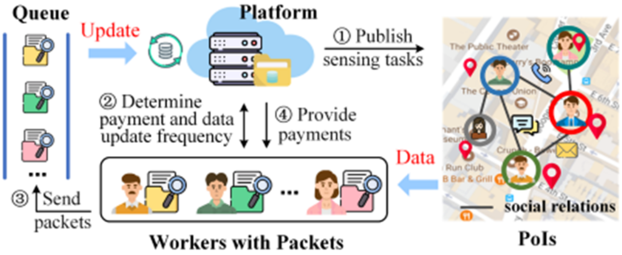
Major Contributions

- Solving the problem of incentive mechanism design for MCS systems with data freshness concerns using novel incomplete information two-stage Stackelberg game with constraints, while considering workers' social benefits.
- Utilize AoI metric to measure the freshness of data and derive the closed-form expression for the AoI of the data each worker uploads to the platform taking workers' social influences into account.
- Propose AIM, when all participants share the utility function parameters of the Stackelberg game. By deriving the optimal strategy for each participant, AIM can ensure that the platform and workers obtain their maximum utilities. Also theoretically prove that these optimal strategies constitute a unique Stackelberg equilibrium.
- Propose DIM, when each participant has no prior knowledge of the game. Based on the DRL technique, DIM enables learning the optimal strategy directly from game experiences.
- **Demonstrate effectiveness of the proposed AIM and DIM mechanisms' performance using real-world traces simulation.**

System Model

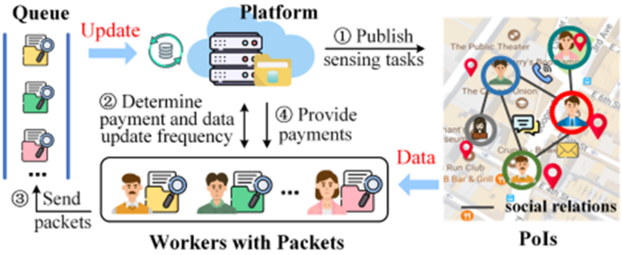


System Model



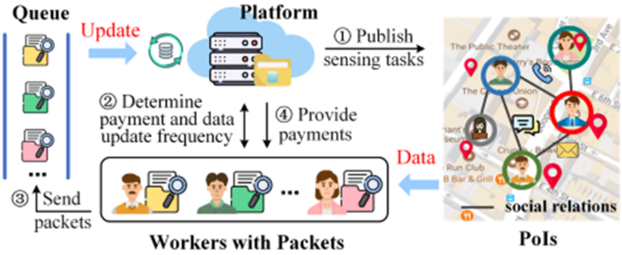
- Workers denoted by $\mathcal{N} \triangleq \{1, 2, \ldots, N\}$.

System Model



- Workers denoted by $N \triangleq \{1, 2, \ldots, N\}$.
- Data Update Frequency (p_i): Rate at which worker i uploads data.

System Model



- Workers denoted by $N \triangleq \{1, 2, \ldots, N\}$.
- Data Update Frequency (p_i): Rate at which worker i uploads data.
- Unit-Remuneration (R_i): Payment per update frequency to worker i.

Utility Functions - Platform

Platform Utility Function:

$$\Phi(p_i, R_i; \eta, c, d) = \eta \sum_{i=1}^{N} \left(cp_i - dp_i^2 \right) - \sum_{i=1}^{N} R_i p_i$$

Utility Functions - Platform

Platform Utility Function:

$$\Phi(p_i, R_i; \eta, c, d) = \eta \sum_{i=1}^{N} (cp_i - dp_i^2) - \sum_{i=1}^{N} R_i p_i$$

Utility Functions - Platform

Platform Utility Function:

$$\Phi(p_i, R_i; \eta, c, d) = \boxed{\eta} \sum_{i=1}^{N} \boxed{(cp_i - dp_i^2)} - \sum_{i=1}^{N} \boxed{R_i p_i}$$

- η: weight for platform's revenue from data

Utility Functions - Platform

Platform Utility Function:

$$\Phi(p_i, R_i; \eta, c, d) = \eta \sum_{i=1}^{N} (cp_i - dp_i^2) - \sum_{i=1}^{N} R_i p_i$$

- η: weight for platform's revenue from data
- $(cp_i - dp_i^2)$: revenue from worker i's data updates

Utility Functions - Platform

Platform Utility Function:

$$\Phi(p_i, R_i; \eta, c, d) = \boxed{\eta} \sum_{i=1}^{N} (cp_i - dp_i^2) - \sum_{i=1}^{N} \boxed{R_i p_i}$$

- η: weight for platform's revenue from data
- $(cp_i - dp_i^2)$: revenue from worker i's data updates
- $R_i p_i$: cost of remuneration paid to worker i

Utility Functions - Worker

Worker Utility Function:

$$\Omega_i(p_i, P_{-i}; s_i, a_i, b_i) = R(p_i) + \Psi_i(p_i, P_{-i}) - \Theta_i(p_i; s_i, a_i, b_i) = R_i p_i + \sum_{j \in N_i} \nu_{ij} p_i p_j - s_i(a_i p_i^2 + b_i p_i)$$

Utility Functions - Worker

Worker Utility Function:

$$\Omega_i(p_i, P_{-i}; s_i, a_i, b_i) = \boxed{R(p_i)} + \Psi_i(p_i, P_{-i}) - \boxed{\Theta_i(p_i; s_i, a_i, b_i)} = \boxed{R_i p_i} + \sum_{j \in N_i} \nu_{ij} p_i p_j - s_i(a_i p_i^2 + b_i p_i$$

Utility Functions - Worker

Worker Utility Function:

$$\Omega_i(p_i, P_{-i}; s_i, a_i, b_i) = R(p_i) + \Psi_i(p_i, P_{-i}) - \Theta_i(p_i; s_i, a_i, b_i) = R_i p_i + \sum_{j \in N_i} \nu_{ij} p_i p_j - s_i(a_i p_i^2 + b_i p_i$$

- $R(p) = R_i p_i$ the remuneration platform pays to worker i.

Utility Functions - Worker

Worker Utility Function:

$$\Omega_i(p_i, P_{-i}; s_i, a_i, b_i) = \boxed{R(p_i)} + \boxed{\Psi_i(p_i, P_{-i})} - \boxed{\Theta_i(p_i; s_i, a_i, b_i)} = \boxed{R_i p_i} + \boxed{\sum_{j \in N_i} \nu_{ij} p_i p_j} - \boxed{s_i(a_i p_i^2 + b_i p_i}$$

- $R(p) = R_i p_i$ the remuneration platform pays to worker i.
- $\Psi_i(p_i, P_{-i}) = \sum_{j \in N_i} \nu_{ij} p_i p_j$ social benefits of worker i.

Utility Functions - Worker

Worker Utility Function:

$$\Omega_i(p_i, P_{-i}; s_i, a_i, b_i) = R(p_i) + \Psi_i(p_i, P_{-i}) - \Theta_i(p_i; s_i, a_i, b_i) = R_i p_i + \sum_{j \in N_i} \nu_{ij} p_i p_j - s_i(a_i p_i^2 + b_i p_i$$

- $R(p) = R_i p_i$ the remuneration platform pays to worker i.
- $\Psi_i(p_i, P_{-i}) = \sum_{j \in N_i} \nu_{ij} p_i p_j$ social benefits of worker i.
- $\Theta_i(p_i; s_i, a_i, b_i) = s_i(a_i p_i^2 + b_i p_i)$ cost function of worker i.

Two-stage Stackelberg game

- **Two-stage Stackelberg game,** $SG\left(p_i, R_i; \varphi\right)$**, where** $\varphi = \{\, s_i, a_i, b_i \mid \forall i \in \mathcal{N}\} \cup \{\eta, c, d\}$ **is the set of all participants' parameters, with public (SPP) and unknown parameters (SUP).**

Two-stage Stackelberg game

- Two-stage Stackelberg game, $SG\left(p_i, R_i; \varphi\right)$, where $\varphi = \{\, s_i, a_i, b_i \mid \forall i \in \mathcal{N}\} \cup \{\eta, c, d\}$ is the set of all participants' parameters, with public (SPP) and unknown parameters (SUP).

- **Stackelberg Equilibrium (SE) with AoI Constraints: An optimal incentive strategy $\langle p_i^*, R_i^* \rangle$ constitutes a SE iff the following set of inequalities is satisfied:**

Two-stage Stackelberg game

- Two-stage Stackelberg game, $SG\left(p_i, R_i; \varphi\right)$, where $\varphi = \{\, s_i, a_i, b_i \mid \forall i \in \mathcal{N}\} \cup \{\eta, c, d\}$ is the set of all participants' parameters, with public (SPP) and unknown parameters (SUP).

- Stackelberg Equilibrium (SE) with AoI Constraints: An optimal incentive strategy $\langle p_i^*, R_i^* \rangle$ constitutes a SE iff the following set of inequalities is satisfied:
  - **Leader: Platform (determines payment strategy)**

$$\Phi\left(p_i^*, R_i^*; \eta, c, d\right) \;\geq\; \Phi\left(p_i, R_i; \eta, c, d\right).$$

Two-stage Stackelberg game

- Two-stage Stackelberg game, $SG(p_i, R_i; \varphi)$, where $\varphi = \{ s_i, a_i, b_i \mid \forall i \in \mathcal{N}\} \cup \{\eta, c, d\}$ is the set of all participants' parameters, with public (SPP) and unknown parameters (SUP).

- Stackelberg Equilibrium (SE) with AoI Constraints: An optimal incentive strategy $\langle p_i^*, R_i^* \rangle$ constitutes a SE iff the following set of inequalities is satisfied:
  - Leader: Platform (determines payment strategy)

$$\Phi\left(p_i^*, R_i^*; \eta, c, d\right) \;\geq\; \Phi\left(p_i, R_i; \eta, c, d\right).$$

  - **Followers: Workers (determine data update frequencies)**

$$\Omega_i\left(p_i^*, R_i^*; s_i, a_i, b_i\right) \;\geq\; \Omega_i\left(p_i, R_i; s_i, a_i, b_i\right).$$

Two-stage Stackelberg game

- Two-stage Stackelberg game, $SG\left(p_i, R_i; \varphi\right)$, where $\varphi = \{\, s_i, a_i, b_i \mid \forall i \in \mathcal{N}\} \cup \{\eta, c, d\}$ is the set of all participants' parameters, with public (SPP) and unknown parameters (SUP).

- Stackelberg Equilibrium (SE) with AoI Constraints: An optimal incentive strategy $\langle p_i^*, R_i^* \rangle$ constitutes a SE iff the following set of inequalities is satisfied:
  - Leader: Platform (determines payment strategy)

$$\Phi\left(p_i^*, R_i^*; \eta, c, d\right) \ \geq \ \Phi\left(p_i, R_i; \eta, c, d\right).$$

  - Followers: Workers (determine data update frequencies)

$$\Omega_i\left(p_i^*, R_i^*; s_i, a_i, b_i\right) \ \geq \ \Omega_i\left(p_i, R_i; s_i, a_i, b_i\right).$$

  - **Subject to:**

$$\delta_i\left(p_i, P_{-i}\right) \ \leq \ \varepsilon, \quad \forall i \in N$$

$$\sum_{i=1}^{n} p_i \ \leq \ \hat{p}.$$

AoI of Data for a Single Worker

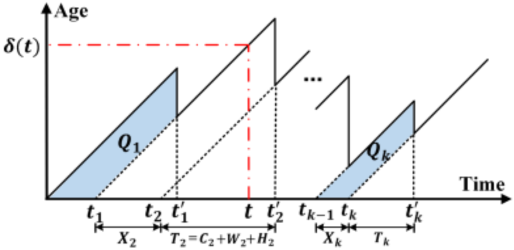
Average AoI $\overline{\delta}^T$ over the interval $[0, T]$:

$$\overline{\delta}^T = \frac{1}{T}\int_0^T \delta(t)\,dt = \frac{1}{T}\big(\text{area under the AoI curve}\big).$$

AoI of Data for a Single Worker

Average AoI $\overline{\delta}^T$ over the interval $[0, T]$:

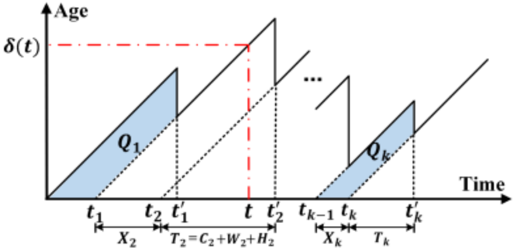
$$\overline{\delta}^T = \frac{1}{T}\int_0^T \delta(t)\,dt = \frac{1}{T}\big(\text{area under the AoI curve}\big).$$



AoI of Data for a Single Worker

Average AoI $\overline{\delta}^T$ over the interval $[0, T]$:

$$\overline{\delta}^T = \frac{1}{T}\int_0^T \delta(t)\,dt = \frac{1}{T}\big(\text{area under the AoI curve}\big).$$



$$\overline{\delta}^T = \frac{1}{T}\left(Q_1 + \sum_{k=2}^{I(T)} Q_k + \frac{T_{I(T)}^2}{2}\right)$$

AoI of Data for a Single Worker (continue)

$$\overline{\delta}^T = \frac{1}{T}\left(Q_1 + \sum_{k=2}^{I(T)} Q_k + \frac{T_{I(T)}^2}{2}\right) = \frac{Q_1 + \frac{T_{I(T)}^2}{2}}{T} + \frac{I(T)-1}{T} \cdot \frac{1}{I(T)-1}\sum_{k=2}^{I(T)} Q_k \tag{12}$$

AoI of Data for a Single Worker (continue)

$$\overline{\delta}^T = \frac{1}{T}\left(Q_1 + \sum_{k=2}^{I(T)} Q_k + \frac{T_{I(T)}^2}{2}\right) = \frac{Q_1 + \frac{T_{I(T)}^2}{2}}{T} + \frac{I(T)-1}{T}\cdot\frac{1}{I(T)-1}\sum_{k=2}^{I(T)} Q_k \tag{12}$$

$$Q_k = \frac{1}{2}(T_k + X_k)^2 - \frac{1}{2}T_k^2 = X_k T_k + \frac{X_k^2}{2} \tag{13}$$

$$\overline{\delta}^T = \frac{Q_1 + \frac{T_{I(T)}^2}{2}}{T} + \frac{I(T)-1}{T}\cdot\frac{1}{I(T)-1}\sum_{k=2}^{I(T)}\left(X_k T_k + \frac{X_k^2}{2}\right)$$

AoI of Data for a Single Worker (continue)

$$\overline{\delta}^T = \frac{1}{T}\left(Q_1 + \sum_{k=2}^{I(T)} Q_k + \frac{T^2_{I(T)}}{2}\right) = \frac{Q_1 + \frac{T^2_{I(T)}}{2}}{T} + \frac{I(T)-1}{T}\cdot\frac{1}{I(T)-1}\sum_{k=2}^{I(T)} Q_k \tag{12}$$

$$Q_k = \frac{1}{2}(T_k + X_k)^2 - \frac{1}{2}T_k^2 = X_k T_k + \frac{X_k^2}{2} \tag{13}$$

$$\overline{\delta}^T = \frac{Q_1 + \frac{T^2_{I(T)}}{2}}{T} + \frac{I(T)-1}{T}\cdot\frac{1}{I(T)-1}\sum_{k=2}^{I(T)}\left(X_k T_k + \frac{X_k^2}{2}\right)$$

$p = \lim_{T\to\infty}\frac{I(T)}{T}$, indicate the data update frequency in the steady state.

$$\overline{\delta} = \lim_{T\to\infty}\overline{\delta}^T = p\big(\mathbb{E}[XT] + \mathbb{E}[X^2/2]\big). \tag{14}$$

AoI of Data for a Single Worker (Plugging M/M/1)

For an M/M/1, FCFS queue with arrival rate p and service rate μ, plus an added constant "collection time" β each time a service occurs:

- Effective service-time distribution has mean $\beta + 1/\mu$.
- Offered load: $\rho = p\big(\beta + 1/\mu\big)$.
- Mean system time (waiting + service): $\mathbb{E}[T] = \dfrac{\beta + 1/\mu}{1-\rho}$.
- Each update arrives after an exponentially distributed interarrival time X, $\mathbb{E}[X] = \dfrac{1}{p}, \quad \mathbb{E}[X^2] = \dfrac{2}{p^2}$

$$\bar{\delta} = \frac{(\rho - 1)\big(\rho^2 \; - \; \mu\,\rho\,\beta\big) + 1}{\rho\,\mu\,(1-\rho)}.$$

AoI of Data for Multiple Workers

Theorem 1: (AoI for Multiple Workers)

N workers compete for the data update through an M/M/1 FCFS queue, in which each worker i's data update frequency, collection time, serving rate, and offered loads are $p_i, \beta_i, \mu,$ and $\rho_i,$ respectively. Then, the average AoI $\bar{\delta}_i$ of worker i's data satisfies

where $\rho = \sum_{i=1}^{N} \rho_i, \quad \rho_i = \frac{p_i}{\mu}, \quad$ and $\rho_{-i} = \sum_{j \neq i} \rho_j.$

AoI of Data for Multiple Workers

Theorem 1: (AoI for Multiple Workers)

N workers compete for the data update through an M/M/1 FCFS queue, in which each worker i's data update frequency, collection time, serving rate, and offered loads are $p_i, \beta_i, \mu,$ and $\rho_i,$ respectively. Then, the average AoI $\bar{\delta}_i$ of worker i's data satisfies

$$\bar{\delta}_i = \frac{\alpha\beta_i}{\sum_{j\in\mathcal{N}_i} v_{i,j}} + \frac{p_i}{\mu^2(1-\rho_{-i})}\left[\frac{\rho_i\rho_{-i}}{(1-\rho_{-i})^2} + \frac{\rho_i/(1-\rho)}{1-\rho_{-i}} + \frac{\rho_{-i}}{\rho_i}\right] + \frac{1}{\mu} + \frac{1}{p_i}$$

where $\rho = \sum_{i=1}^{N} \rho_i, \quad \rho_i = \frac{p_i}{\mu}, \quad$ and $\rho_{-i} = \sum_{j\neq i} \rho_j.$

AoI of Data for Multiple Workers

Theorem 1: (AoI for Multiple Workers)

N workers compete for the data update through an M/M/1 FCFS queue, in which each worker i's data update frequency, collection time, serving rate, and offered loads are $p_i, \beta_i, \mu,$ and ρ_i, respectively. Then, the average AoI $\bar{\delta}_i$ of worker i's data satisfies

$$\bar{\delta}_i = \boxed{\frac{\alpha\beta_i}{\sum_{j\in\mathcal{N}_i} v_{i,j}}} + \frac{p_i}{\mu^2(1-\rho_{-i})}\left[\frac{\rho_i\rho_{-i}}{(1-\rho_{-i})^2} + \frac{\rho_i/(1-\rho)}{1-\rho_{-i}} + \frac{\rho_{-i}}{\rho_i}\right] + \boxed{\frac{1}{\mu} + \frac{1}{p_i}}$$

$$\text{where } \rho = \sum_{i=1}^{N}\rho_i, \quad \rho_i = \frac{p_i}{\mu}, \quad \text{and } \rho_{-i} = \sum_{j\neq i}\rho_j.$$

AoI of Data for Multiple Workers

Theorem 1: (AoI for Multiple Workers)

N workers compete for the data update through an M/M/1 FCFS queue, in which each worker i's data update frequency, collection time, serving rate, and offered loads are p_i, β_i, μ, and ρ_i, respectively. Then, the average AoI $\bar{\delta}_i$ of worker i's data satisfies

$$\bar{\delta}_i = \frac{\alpha\beta_i}{\sum_{j\in\mathcal{N}_i} v_{i,j}} + \frac{p_i}{\mu^2(1-\rho_{-i})}\left[\frac{\rho_i\rho_{-i}}{(1-\rho_{-i})^2} + \frac{\rho_i/(1-\rho)}{1-\rho_{-i}} + \frac{\rho_{-i}}{\rho_i}\right] + \frac{1}{\mu} + \frac{1}{p_i}$$

- C_k: collection time.

$$\text{where } \rho = \sum_{i=1}^{N}\rho_i, \quad \rho_i = \frac{p_i}{\mu}, \quad \text{and } \rho_{-i} = \sum_{j\neq i}\rho_j.$$

AoI of Data for Multiple Workers

Theorem 1: (AoI for Multiple Workers)

N workers compete for the data update through an M/M/1 FCFS queue, in which each worker i's data update frequency, collection time, serving rate, and offered loads are $p_i, \beta_i, \mu,$ and ρ_i, respectively. Then, the average AoI $\bar{\delta}_i$ of worker i's data satisfies

$$\bar{\delta}_i = \frac{\alpha\beta_i}{\sum_{j\in\mathcal{N}_i} v_{i,j}} + \frac{p_i}{\mu^2(1-\rho_{-i})}\left[\frac{\rho_i\rho_{-i}}{(1-\rho_{-i})^2} + \frac{\rho_i/(1-\rho)}{1-\rho_{-i}} + \frac{\rho_{-i}}{\rho_i}\right] + \frac{1}{\mu} + \frac{1}{p_i}$$

- C_k: collection time.
- W_k: wait time.

$$\text{where } \rho = \sum_{i=1}^{N}\rho_i, \quad \rho_i = \frac{p_i}{\mu}, \quad \text{and } \rho_{-i} = \sum_{j\neq i}\rho_j.$$

AoI of Data for Multiple Workers

Theorem 1: (AoI for Multiple Workers)

N workers compete for the data update through an M/M/1 FCFS queue, in which each worker i's data update frequency, collection time, serving rate, and offered loads are $p_i, \beta_i, \mu,$ and ρ_i, respectively. Then, the average AoI $\bar{\delta}_i$ of worker i's data satisfies

$$\bar{\delta}_i = \boxed{\frac{\alpha\beta_i}{\sum_{j\in\mathcal{N}_i} v_{i,j}}} + \frac{p_i}{\mu^2(1-\rho_{-i})}\left[\frac{\rho_i\rho_{-i}}{(1-\rho_{-i})^2} + \frac{\rho_i/(1-\rho)}{1-\rho_{-i}} + \frac{\rho_{-i}}{\rho_i}\right] + \boxed{\frac{1}{\mu} + \frac{1}{p_i}}$$

- C_k: collection time.
- W_k: wait time.
- H_k: handling time.

where $\rho = \sum_{i=1}^{N} \rho_i, \quad \rho_i = \frac{p_i}{\mu}, \quad$ and $\rho_{-i} = \sum_{j\neq i} \rho_j$.

Bayesian Game

- **A Bayesian game, or Bayesian-Nash game, is a type of game in game theory where players have incomplete information about some aspects of the game.**

Bayesian Game

- A *Bayesian game*, or *Bayesian-Nash game*, is a type of game in game theory where players have incomplete information about some aspects of the game.
- **Each player is associated with a type, which determines their preferences, payoffs, or available strategies.**

Bayesian Game

- A *Bayesian game*, or *Bayesian-Nash game*, is a type of game in game theory where players have incomplete information about some aspects of the game.
- Each player is associated with a *type*, which determines their preferences, payoffs, or available strategies.
- **There is a commonly known prior probability distribution over the types of players.**

Bayesian Game

- A *Bayesian game*, or *Bayesian-Nash game*, is a type of game in game theory where players have incomplete information about some aspects of the game.
- Each player is associated with a *type*, which determines their preferences, payoffs, or available strategies.
- There is a commonly known prior probability distribution over the types of players.
- **A strategy in a Bayesian game specifies what a player will do for every possible type they might have.**

Bayesian Game

- A *Bayesian game*, or *Bayesian-Nash game*, is a type of game in game theory where players have incomplete information about some aspects of the game.
- Each player is associated with a *type*, which determines their preferences, payoffs, or available strategies.
- There is a commonly known prior probability distribution over the types of players.
- A strategy in a Bayesian game specifies what a player will do for every possible type they might have.
- **The payoff for each player depends on their own type, their chosen action, and the types and actions of other players.**

Bayesian Game

- A *Bayesian game*, or *Bayesian-Nash game*, is a type of game in game theory where players have incomplete information about some aspects of the game.
- Each player is associated with a *type*, which determines their preferences, payoffs, or available strategies.
- There is a commonly known prior probability distribution over the types of players.
- A strategy in a Bayesian game specifies what a player will do for every possible type they might have.
- The payoff for each player depends on their own type, their chosen action, and the types and actions of other players.
- **A Bayesian Nash equilibrium is a strategy profile where each player's strategy maximizes their expected utility, given their beliefs about the other players' types and strategies.**

AoI guaranteed Incentive Mechanism (AIM)

Propose AoI guaranteed Incentive Mechanism (AIM) by leveraging the backward deduction approach and the KKT conditions.

1. **Solve the worker, follower (stage-2), game.**

AoI guaranteed Incentive Mechanism (AIM)

Propose AoI guaranteed Incentive Mechanism (AIM) by leveraging the backward deduction approach and the KKT conditions.

1. Solve the worker, follower (stage-2), game.
    1.1 **Employ Bayesian strategy to handle incomplete information, uncertainty associated with social network effects and the strategies of other workers, by exploiting worker's degree in the social network to derive its utility.**

AoI guaranteed Incentive Mechanism (AIM)

Propose AoI guaranteed Incentive Mechanism (AIM) by leveraging the backward deduction approach and the KKT conditions.

1. Solve the worker, follower (stage-2), game.
   1.1 Employ Bayesian strategy to handle incomplete information, uncertainty associated with social network effects and the strategies of other workers, by exploiting worker's degree in the social network to derive its utility.
   1.2 **Determine worker's optimal data update frequency p_i^* under a given unit-remuneration R_i.**

AoI guaranteed Incentive Mechanism (AIM)

Propose AoI guaranteed Incentive Mechanism (AIM) by leveraging the backward deduction approach and the KKT conditions.

1. Solve the worker, follower (stage-2), game.
    1.1 Employ Bayesian strategy to handle incomplete information, uncertainty associated with social network effects and the strategies of other workers, by exploiting worker's degree in the social network to derive its utility.
    1.2 Determine worker's optimal data update frequency p_i^* under a given unit-remuneration R_i.
2. **Solve the platform, leader (stage-1) game, to derive the optimal unit-remuneration R_i^* paid by the platform.**

Solving the Bayesian Sub-Game

$$\Omega_i(p_i, P_{-i}; s_i, a_i, b_i) = R(p_i) + \Psi_i(p_i, P_{-i}) - \Theta_i(p_i; s_i, a_i, b_i) = R_i p_i + \sum_{j \in N_i} \nu_{ij} p_i p_j - s_i \left(a_i p_i^2 + b_i p_i \right) \tag{3}$$

- **Set social network influence $\nu_{ij} = \nu$ for all i, j ($i \neq j$).**

Solving the Bayesian Sub-Game

(3) $$\Omega_i(p_i, P_{-i}; s_i, a_i, b_i) = R(p_i) + \Psi_i(p_i, P_{-i}) - \Theta_i(p_i; s_i, a_i, b_i) = R_i p_i + \sum_{j \in N_i} \nu_{ij} p_i p_j - s_i \left(a_i p_i^2 + b_i p_i \right)$$

- Set social network influence $\nu_{ij} = \nu$ for all i, j $(i \neq j)$.
- **harness worker i degree, $f \in G$, to describe the type of each worker.**

Solving the Bayesian Sub-Game

$$\Omega_i(p_i, P_{-i}; s_i, a_i, b_i) = R(p_i) + \Psi_i(p_i, P_{-i}) - \Theta_i(p_i; s_i, a_i, b_i) = R_i p_i + \sum_{j \in N_i} \nu_{ij} p_i p_j - s_i \left(a_i p_i^2 + b_i p_i \right) \tag{3}$$

- Set social network influence $\nu_{ij} = \nu$ for all i, j $(i \neq j)$.
- harness worker i degree, $f \in G$, to describe the type of each worker.
- $\mathbb{E}[\sum_{j \in N_i} p_j] = f \times \overline{P_{-i}}$, **where $\overline{P_{-i}}$ is the average data update frequency of worker i's neighbors.**

Solving the Bayesian Sub-Game

(3) $$\Omega_i(p_i, P_{-i}; s_i, a_i, b_i) = R(p_i) + \Psi_i(p_i, P_{-i}) - \Theta_i(p_i; s_i, a_i, b_i) = R_i p_i + \sum_{j \in N_i} \nu_{ij} p_i p_j - s_i \left(a_i p_i^2 + b_i p_i \right)$$

- Set social network influence $\nu_{ij} = \nu$ for all i, j $(i \neq j)$.
- harness worker i degree, $f \in G$, to describe the type of each worker.
- $\mathbb{E}[\sum_{j \in N_i} p_j] = f \times \overline{P_{-i}}$, where $\overline{P_{-i}}$ is the average data update frequency of worker i's neighbors.
- **Use symmetric type space Bayesian game (i.e., workers with the same type f will choose the same data update frequency $p(f)$ and will be awarded the same remuneration $R(f)$.**

Solving the Bayesian Sub-Game

$$(3) \qquad \Omega_i(p_i, P_{-i}; s_i, a_i, b_i) = R(p_i) + \Psi_i(p_i, P_{-i}) - \Theta_i(p_i; s_i, a_i, b_i) = R_i p_i + \sum_{j \in N_i} \nu_{ij} p_i p_j - s_i \left(a_i p_i^2 + b_i p_i \right)$$

- Set social network influence $\nu_{ij} = \nu$ for all i, j $(i \neq j)$.
- harness worker i degree, $f \in G$, to describe the type of each worker.
- $\mathbb{E}[\sum_{j \in N_i} p_j] = f \times \overline{P_{-i}}$, where $\overline{P_{-i}}$ is the average data update frequency of worker i's neighbors.
- Use symmetric type space Bayesian game (i.e., workers with the same type f will choose the same data update frequency $p(f)$ and will be awarded the same remuneration $R(f)$.
- **Using the "Configuration Model", randomly chosen social network neighbor of worker i has the degree distribution $\overline{F}(f) = \dfrac{F(f)f}{(\sum_{f' \in G} F(f')f')} = \dfrac{F(f)f}{\underline{f}}$, where $\overline{P_{-f}} = \sum_{f \in G} \overline{F}(f) p(f)$.**

Solving the Bayesian Sub-Game

$$(3) \qquad \Omega_i(p_i, P_{-i}; s_i, a_i, b_i) = R(p_i) + \Psi_i(p_i, P_{-i}) - \Theta_i(p_i; s_i, a_i, b_i) = R_i p_i + \sum_{j \in N_i} \nu_{ij} p_i p_j - s_i \left(a_i p_i^2 + b_i p_i \right)$$

- Set social network influence $\nu_{ij} = \nu$ for all i, j $(i \neq j)$.
- harness worker i degree, $f \in G$, to describe the type of each worker.
- $\mathbb{E}[\sum_{j \in N_i} p_j] = f \times \overline{P_{-i}}$, where $\overline{P_{-i}}$ is the average data update frequency of worker i's neighbors.
- Use symmetric type space Bayesian game (i.e., workers with the same type f will choose the same data update frequency $p(f)$ and will be awarded the same remuneration $R(f)$.
- Using the "Configuration Model", randomly chosen social network neighbor of worker i has the degree distribution $\overline{F}(f) = \dfrac{F(f)f}{(\sum_{f' \in G} F(f')f')} = \dfrac{F(f)f}{\underline{f}}$, where $\overline{P_{-f}} = \sum_{f \in G} \overline{F}(f) p(f)$.
- **Thus utility of the worker with degree f can be represented as follows:**

$$\overline{\Omega}_f(p(f), P_{-f}) = R(f)p(f) + \upsilon p(f) f \overline{P_{-f}} - \left(ap^2(f) + bp(f) \right) s.$$

Worker's (Follower) Optimal Strategy

Definition (Bayesian Nash Equilibrium (BNE))

A BNE is defined as a strategy profile that maximizes the expected payoff of each player for the given types and strategies performed by other players.
A strategy vector $\Gamma = (\Gamma_1(\psi_1), \Gamma_2(\psi_2), \ldots, \Gamma_N(\psi_N))$ is a Bayesian Nash Equilibrium if and only if the following condition is satisfied for each player i:

$$\Gamma_i(\psi_i) \in \arg\max_{p_i \in P_i} \Omega_i(p_i, \Gamma_{-i}, \psi_i, \psi_{-i}).$$

Worker's (Follower) Optimal Strategy

Definition (Bayesian Nash Equilibrium (BNE))

A BNE is defined as a strategy profile that maximizes the expected payoff of each player for the given types and strategies performed by other players.
A strategy vector $\Gamma = (\Gamma_1(\psi_1), \Gamma_2(\psi_2), \dots, \Gamma_N(\psi_N))$ is a Bayesian Nash Equilibrium if and only if the following condition is satisfied for each player i:

$$\Gamma_i(\psi_i) \in \arg\max_{p_i \in P_i} \Omega_i(p_i, \Gamma_{-i}, \psi_i, \psi_{-i}).$$

Theorem (Follower's Optimal Strategy)

Given any unit remuneration $R(f)$, the closed-form expression of the action (i.e., data update frequency) of the follower game is:

$$p(f) = \frac{1}{2as} R(f) - \frac{b}{2a} + \frac{\upsilon f(\overline{R} - bs)}{2as(2as - \upsilon \overline{f})},$$

where $\overline{R} = \sum_{f \in G} \overline{F}(f) R(f)$ *and* $\overline{f} = \sum_{f \in G} \overline{F}(f) f$.

Platform (Leader) Game With Constraints

- **Platform utility function when applying the configuration model:**

$$\overline{\Phi} = \mathbb{E}[\Phi] = \mathbb{E}\left[\eta \sum_{i=1}^{N} \left(cp_i - dp_i^2\right) - \sum_{i=1}^{N} R_i p_i\right] = N \sum_{f \in G} F(f) \left[(\eta c - R(f))p(f) - \eta d p^2(f)\right],$$

Platform (Leader) Game With Constraints

- Platform utility function when applying the configuration model:

$$\overline{\Phi} = \mathbb{E}[\Phi] = \mathbb{E}\left[\eta \sum_{i=1}^{N} \left(cp_i - dp_i^2\right) - \sum_{i=1}^{N} R_i p_i\right] = N \sum_{f \in G} F(f) \left[(\eta c - R(f))p(f) - \eta dp^2(f)\right],$$

- **where optimizing $\overline{\Phi}(R(f))$ with the AoI and the total update frequency constraints:** $g(R(f)) = \delta_f(R(f)) - \epsilon \leq 0, \qquad g'(R(f)) = N \sum_{f} F(f)p(f) - \hat{p} \leq 0.$

Platform (Leader) Game With Constraints

- Platform utility function when applying the configuration model:

$$\overline{\Phi} = \mathbb{E}[\Phi] = \mathbb{E}\left[\eta \sum_{i=1}^{N} \left(cp_i - dp_i^2\right) - \sum_{i=1}^{N} R_i p_i\right] = N \sum_{f \in G} F(f) \left[(\eta c - R(f))p(f) - \eta d p^2(f)\right],$$

- where optimizing $\overline{\Phi}(R(f))$ with the AoI and the total update frequency constraints:
  $g(R(f)) = \delta_f(R(f)) - \epsilon \leq 0, \qquad g'(R(f)) = N \sum_{f} F(f)p(f) - \hat{p} \leq 0.$
- **Finding optimal solution, using Lagrangian function:**
  $\mathcal{L}(R(f), \zeta) = \overline{\Phi}(R(f)) + \zeta_1 g(R(f)) + \zeta_2 \tilde{g}'(R(f))$, **where ζ_1 and ζ_2 the Lagrangian multipliers.**

Platform (Leader) Game With Constraints

- Platform utility function when applying the configuration model:

$$\overline{\Phi} = \mathbb{E}[\Phi] = \mathbb{E}\left[\eta \sum_{i=1}^{N}\left(cp_i - dp_i^2\right) - \sum_{i=1}^{N} R_i p_i\right] = N \sum_{f\in G} F(f)\left[(\eta c - R(f))p(f) - \eta dp^2(f)\right],$$

- where optimizing $\overline{\Phi}(R(f))$ with the AoI and the total update frequency constraints:
  $g(R(f)) = \delta_f(R(f)) - \epsilon \leq 0, \qquad g'(R(f)) = N\sum_{f} F(f)p(f) - \hat{p} \leq 0.$
- Finding optimal solution, using Lagrangian function:
  $\mathcal{L}(R(f), \zeta) = \overline{\Phi}(R(f)) + \zeta_1 g(R(f)) + \zeta_2 \tilde{g}'(R(f))$, where ζ_1 and ζ_2 the Lagrangian multipliers.

- **Since it is a convex optimization problem, the optimal solution must satisfy the Karush-Kuhn-Tucker (KKT) optimality conditions:**

$$\left.\frac{\partial \mathcal{L}}{\partial R(f)}\right|_{R(f)=R^*(f)} = 0, \quad \zeta_1 g(R(f)) = 0, \quad \zeta_2 \tilde{g}'(R(f)) = 0,$$

$$g(R(f)) \leq 0, \quad g'(R(f)) \leq 0, \quad \zeta_1 \geq 0, \quad \zeta_2 \geq 0.$$

Platform (Leader) Game With Constraints (continue)

Case 1: $\zeta_1 = 0, \zeta_2 = 0$, both constraints are inactive. Plus the sum over all $l \neq f$: the first-order derivative of $\overline{\Phi}(R(f))$ be equal to zero:

$$\frac{\partial \overline{R}}{\partial R(f)} = \overline{F}(f), \quad \frac{\partial p(l)}{\partial R(f)} = \frac{\nu_l \overline{F}(f)}{2as(2as - \nu f)} = \Delta \overline{F}(f) l \quad (l \neq f),$$

$$\frac{\partial p(f)}{\partial R(f)} = \frac{1}{2as} + \frac{\nu f F(f)}{2as(2as - \nu f)} = \frac{1}{2as} + \Delta F(f) f \quad (l = f),$$

Platform (Leader) Game With Constraints (continue)

Case 1: $\zeta_1 = 0, \zeta_2 = 0$, both constraints are inactive. Plus the sum over all $l \neq f$: the first-order derivative of $\overline{\Phi}(R(f))$ be equal to zero:

$$\frac{\partial \overline{R}}{\partial R(f)} = \overline{F}(f), \quad \frac{\partial p(l)}{\partial R(f)} = \frac{\nu_l \overline{F}(f)}{2as(2as-\nu f)} = \Delta\overline{F}(f)l \quad (l \neq f),$$

$$\frac{\partial p(f)}{\partial R(f)} = \frac{1}{2as} + \frac{\nu f F(f)}{2as(2as-\nu f)} = \frac{1}{2as} + \Delta F(f)f \quad (l = f),$$

$$\frac{\partial \bar{\Phi}}{\partial R(f)} = NF(f)\left[-p(f) + (\eta c - R(f) - 2\eta dp(f))\left(\frac{1}{2as} + \Delta\overline{F}(f)f\right)\right]$$
$$+N\sum_{l \neq f} F(l)\left[(\eta c - R(l) - 2\eta dp(l))\Delta\overline{F}(f)f\right] = 0.$$

Maximum Platform and Worker Utilities

Considering social network is seen as a configuration model with large numbers of workers, optimal unit remuneration:

$$R^*(f) = \frac{2a^2\tilde{s}^2}{(2as\Delta\overline{F}(f)f+1)(as+\eta d)+as}\left[\frac{b}{2a} - \Delta f(\overline{R}^* - bs)\right] + \left(\Delta\overline{F}(f)f + \frac{1}{2as}\right)$$

$$\left(\eta c + \frac{b\eta d}{a} - 2\Delta f\eta d(\overline{R}^* - bs)\right) + \Delta f\left[-\left(1+\frac{\eta d}{as}\right)\frac{\Lambda^*}{f} + \frac{\eta(ac+bd)}{a} - 2\Delta\eta d(\overline{R}^* - bs)\overline{f}\right].$$

Maximum Platform and Worker Utilities

Considering social network is seen as a configuration model with large numbers of workers, optimal unit remuneration:

$$R^*(f) = \frac{2a^2\tilde{s}^2}{(2as\Delta\overline{F}(f)f+1)(as+\eta d)+as}\left[\frac{b}{2a}-\Delta f(\overline{R}^*-bs)\right]+\left(\Delta\overline{F}(f)f+\frac{1}{2as}\right)$$
$$\left(\eta c+\frac{b\eta d}{a}-2\Delta f\eta d(\overline{R}^*-bs)\right)+\Delta f\left[-\left(1+\frac{\eta d}{as}\right)\frac{\Lambda^*}{f}+\frac{\eta(ac+bd)}{a}-2\Delta\eta d(\overline{R}^*-bs)\overline{f}\right].$$

The optimal data update frequency of worker i:

$$p_i^*(f) = \frac{1}{2as}R^*(f)-\frac{b}{2a}+\frac{\nu f(\overline{R}^*-bs)}{2as(2as-\nu\overline{f})}$$

Maximum Platform and Worker Utilities

Considering social network is seen as a configuration model with large numbers of workers, optimal unit remuneration:

$$R^*(f) = \frac{2a^2\tilde{s}^2}{(2as\Delta\overline{F}(f)f+1)(as+\eta d)+as}\left[\frac{b}{2a} - \Delta f(\overline{R}^* - bs)\right] + \left(\Delta\overline{F}(f)f + \frac{1}{2as}\right)$$
$$\left(\eta c + \frac{b\eta d}{a} - 2\Delta f\eta d(\overline{R}^* - bs)\right) + \Delta f\left[-\left(1+\frac{\eta d}{as}\right)\frac{\Lambda^*}{f} + \frac{\eta(ac+bd)}{a} - 2\Delta\eta d(\overline{R}^* - bs)\overline{f}\right].$$

The optimal data update frequency of worker i:

$$p_i^*(f) = \frac{1}{2as}R^*(f) - \frac{b}{2a} + \frac{\nu f(\overline{R}^* - bs)}{2as(2as - \nu\overline{f})}$$

Maximum Platform and Worker Utilities:

$$\Phi^* = N\sum_{f\in G} F(f)\left[(\eta c - R^*(f))p^*(f) - \eta d(p^*(f))^2\right].$$

$$\Omega_i^* = R^*(f)p^*(f) + \nu p^*(f)fP_{-f} - \left(a(p^*(f))^2 + bp^*(f)\right)s.$$

AoL Constraint Active

Case 2: $\zeta_1 \neq 0, \zeta_2 = 0$, AoI constraint $g(R(f)) = \delta_f(R(f)) - \epsilon \leq 0$ is active.

$$g(R(f)) = \frac{\alpha\beta}{f\nu} + \frac{p^2(f)}{\mu^2\breve{\rho}}\left(\frac{\rho_{-i}(f)}{\mu(\breve{\rho})^2} + \frac{1}{\breve{\rho}\mu(1-\rho(f))} + \frac{\rho_{-i}(f)\mu}{p^2(f)}\right) + \frac{1}{\mu} + \frac{1}{p(f)} - \epsilon = 0,$$

where:

- $\breve{\rho} = 1 - \rho_{-i}(f)$ represents the remaining service capacity.
- $\rho(f) = \dfrac{\sum_f p(f)}{\mu}$ is the total offered load.
- $\rho_{-i}(f) = \dfrac{\sum_f p(f) - p(f)}{\mu}$ represents the load from other workers.

Since $g(R(f))$ depends on $p(f)$, and $p(f)$ is determined by $R(f)$, combine the derivative of the utility function with the derivative of the AoI constraint to ensure both are met:

$$\frac{\partial\bar{\Phi}}{\partial R(f)} + \frac{\partial g}{\partial R(f)} = 0.$$

Total Data Update Frequency Constraint Active

Case 3: $\zeta_1 = 0, \zeta_2 \neq 0$, total data update frequency constraint $g'(R(f)) = \sum_f F(f)p(f) - \hat{p} \leq 0$ is active, and the system needs to adjust $R(f)$ and $p(f)$ to bring it back within bounds.
$L(R(f), \zeta_2) = \bar{\Phi}(R(f)) + \zeta_2 g'(R(f))$, where $g'(R(f)) = \sum_f F(f)p(f) - \hat{p}$.

For optimal $R^*(f)$ derivative the Lagrangian with respect to $R(f)$ and set it to zero:

$$\frac{\partial \bar{\Phi}}{\partial R(f)} + \zeta_2 \frac{\partial g'}{\partial R(f)} = 0.$$

Total Data Update Frequency Constraint Active

Case 3: $\zeta_1 = 0, \zeta_2 \neq 0$, total data update frequency constraint $g'(R(f)) = \sum_f F(f)p(f) - \hat{p} \leq 0$ is active, and the system needs to adjust $R(f)$ and $p(f)$ to bring it back within bounds.
$L(R(f), \zeta_2) = \bar{\Phi}(R(f)) + \zeta_2 g'(R(f))$, where $g'(R(f)) = \sum_f F(f)p(f) - \hat{p}$.

For optimal $R^*(f)$ derivative the Lagrangian with respect to $R(f)$ and set it to zero:

$$\frac{\partial \bar{\Phi}}{\partial R(f)} + \zeta_2 \frac{\partial g'}{\partial R(f)} = 0.$$

$$R^*(f) = \frac{as(\eta c - bs + \zeta_2)}{2as + \eta d}\left(1 - \frac{f}{\underline{f}}\right) + \frac{2asf\hat{p}}{N\underline{f}} + bs - 2as\Delta f(\overline{R} - bs),$$

where:

- $\bar{f} = \sum_f F(f)f$ is the average degree,
- $\hat{p}$ is the threshold for total update frequency,
- ζ_2 is determined by solving $g'(R(f)) = 0$.

Activating Both Constraints

Case 4: $\zeta_1 \neq 0, \zeta_2 \neq 0$, AoL and total data update frequency constraints are active.

$$\frac{\partial \bar{\Phi}}{\partial R(f)} + \zeta_1 \frac{\partial g}{\partial R(f)} + \zeta_2 \frac{\partial g'}{\partial R(f)} = 0,$$

along with: $g(R(f)) = 0, \quad g'(R(f)) = 0.$

- Solving this system is complex because of nonlinearity of functions and the interdependencies betrween constraints and the utility function.
- Numerical methods (e.g., Bisection, Newton's, etc.) are employed to approximate.
- Iterative Process.

The AoI-Guaranteed Incentive Mechanism (AIM)

Algorithm 1: The AoI-Guaranteed Incentive Mechanism.

Input : degree distribution $F(f)$, worker i's degree f, and some public parameters a, b, c, d, η, s;

Output: $R^*(f)$, $p^*(f)$, Φ^*, and Ω_i^*;

Platform: Determine its tentative optimal strategy (i.e., the unit-remuneration $R^*(f)$) according to Eq. (37);

for *each worker* i, $i \in \mathcal{N}$ **do**

  Determine its tentative optimal strategy (i.e., the data update frequency $p_i^*(f)$) based on $R^*(f)$ and Eq. (38);

if $\delta_i(p_i, P_{-i}) \leq \varepsilon$ *for* $\forall i$ **then**

  if $\sum_{i=1}^{N} p_i \leq \hat{p}$ **then**

    Platform: Obtain Φ^* according to Eq. (39);

    Worker i: Obtain Ω_i^* according to Eq. (40);

  else Solving Eq. (42) and $g'(R(f))=0 \Rightarrow R^*(f)$;

    Platform: Update its strategy as $R^*(f)$;

    Worker i: Update $p_i^*(f)$ based on $R^*(f)$;

    Calculate Φ^* and Ω_i^* based on Eqs. (39) and (40);

else

  if $\sum_{i=1}^{N} p_i \leq \hat{p}$ **then**

    Solving Eq. (41) and $\partial\mathcal{L}/\partial R(f)=0 \Rightarrow R^*(f)$;

  else Solving Eq. (43)$\Rightarrow R^*(f)$;

  Platform and Workers: Update $p_i^*(f), R^*(f), \Phi^*, \Omega_i^*$;

The BNE Analysis

Lemma 1: The follower game has at least one pure Bayesian Nash Equilibrium (BNE).

Proof: if the Bayesian sub-game satisfies the (Milgrom-Shannon) Single Crossing Property of Incremental Returns (SCP-IR), the Bayesian subgame has at least one pure BNE.

- The partial derivative of the follower's payoff function with respect to their strategy p_i and others' strategies P_{-i} is:

$$\frac{\partial^2 \bar{\Omega}_i(p_i, P_{-i}, R)}{\partial p_i \partial P_{-i}} = \nu f > 0.$$

  This indicates that the payoff increases with others' strategies.

- The second-order partial derivative with respect to p_i^2 is:

$$\frac{\partial^2 \bar{\Omega}_i(p_i, P_{-i}, R)}{\partial p_i^2} = -2as < 0,$$

  confirming concavity of the payoff with respect to p_i, which satisfies the SCP-IR.

The BNE Analysis (continue)

Lemma 2: Uniqueness of the Bayesian Nash Equilibrium

The Bayesian sub-game has at most one BNE if the following condition is satisfied:

$$\left| \left(\frac{\partial^2 \overline{\Omega}_i (p_i, P_{-i}, \mathcal{R})}{\partial p_i \, \partial P_{-i}} \right) \Big/ \left(\frac{\partial^2 \overline{\Omega}_i (p_i, P_{-i}, \mathcal{R})}{\partial p_i^2} \right) \right| < 1, \quad \forall\, i \in \mathcal{N}.$$

The left-hand side represents the interaction between the strategies of player i and others.

If the interaction is sufficiently weak, the uniqueness of the equilibrium is guaranteed.

Result: When the condition $\nu f_{\max} - 2as < 0$ is met, the uniqueness of the BNE is ensured.

The BNE Analysis (continue)

Lemma 2: Uniqueness of the Bayesian Nash Equilibrium

The Bayesian sub-game has at most one BNE if the following condition is satisfied:

$$\left| \left(\frac{\partial^2 \overline{\Omega}_i(p_i, P_{-i}, \mathcal{R})}{\partial p_i \, \partial P_{-i}} \right) \Big/ \left(\frac{\partial^2 \overline{\Omega}_i(p_i, P_{-i}, \mathcal{R})}{\partial p_i^2} \right) \right| < 1, \quad \forall\, i \in \mathcal{N}.$$

The left-hand side represents the interaction between the strategies of player i and others.

If the interaction is sufficiently weak, the uniqueness of the equilibrium is guaranteed.

Result: When the condition $\nu f_{\max} - 2as < 0$ is met, the uniqueness of the BNE is ensured.

Stackelberg Game Equilibrium: The leader in the Stackelberg game can optimize their strategy with the certainty that the followers' response is unique and predictable (as derived from Lemmas 1 and 2).

Unique Stackelberg Equilibrium

Theorem (Unique Stackelberg Equilibrium)

The optimal incentive strategy $\langle p^*(f), R^*(f)\rangle$ determined by AIM constitutes the unique Stackelberg equilibrium while satisfying AoI constraints.

- **Uniqueness of Equilibrium**
  - In Stage I, $R^*(f)$ is derived as the unique optimal solution for the platform's utility maximization.
  - In Stage II, $p^*(f)$ forms a unique Bayesian Nash Equilibrium (BNE), as shown in the equilibrium analysis.

  Together, these ensure that the entire Stackelberg game has a unique equilibrium.
- **AoI Constraints**
  The AoI constraint, $g(R(f)) = \delta_f(R(f)) - \epsilon \leq 0$, ensures that the age of information for workers' data is below a specified threshold. This constraint is satisfied as part of the incentive mechanism design, ensuring that all workers' data meet the freshness requirement.

DRL-based Incentive Mechanism (DIM)

- **DRL-based Incentive Mechanism (DIM) with AoI guarantees to address the SUP problem, where the platform and workers have no prior knowledge of the Stackelberg game.**

DRL-based Incentive Mechanism (DIM)

- DRL-based Incentive Mechanism (DIM) with AoI guarantees to address the SUP problem, where the platform and workers have no prior knowledge of the Stackelberg game.

- **Platform conducts interactions with workers to learn the optimal strategies from game experiences.**

DRL-based Incentive Mechanism (DIM)

- DRL-based Incentive Mechanism (DIM) with AoI guarantees to address the SUP problem, where the platform and workers have no prior knowledge of the Stackelberg game.
- Platform conducts interactions with workers to learn the optimal strategies from game experiences.
- **Learning Framework: Twin Delayed Deep Deterministic Policy Gradient (TD3).**

DRL-based Incentive Mechanism (DIM)

- DRL-based Incentive Mechanism (DIM) with AoI guarantees to address the SUP problem, where the platform and workers have no prior knowledge of the Stackelberg game.
- Platform conducts interactions with workers to learn the optimal strategies from game experiences.
- Learning Framework: Twin Delayed Deep Deterministic Policy Gradient (TD3).
  - **TD3 contains two types of networks, actor network and two critic networks.**

DRL-based Incentive Mechanism (DIM)

- DRL-based Incentive Mechanism (DIM) with AoI guarantees to address the SUP problem, where the platform and workers have no prior knowledge of the Stackelberg game.
- Platform conducts interactions with workers to learn the optimal strategies from game experiences.
- Learning Framework: Twin Delayed Deep Deterministic Policy Gradient (TD3).
  - TD3 contains two types of networks, actor network and two critic networks.
  - **Actor network learns the policy $\pi_\theta(a \mid s)$, and decides what action to take in a given state.**

DRL-based Incentive Mechanism (DIM)

- DRL-based Incentive Mechanism (DIM) with AoI guarantees to address the SUP problem, where the platform and workers have no prior knowledge of the Stackelberg game.
- Platform conducts interactions with workers to learn the optimal strategies from game experiences.
- Learning Framework: Twin Delayed Deep Deterministic Policy Gradient (TD3).
  - TD3 contains two types of networks, actor network and two critic networks.
  - Actor network learns the policy $\pi_\theta(a \mid s)$, and decides what action to take in a given state.
  - **Critic network evaluates the quality, $Q(s, a)$, of the action and guide actor to improve decisions.**

DRL-based Incentive Mechanism (DIM)

- DRL-based Incentive Mechanism (DIM) with AoI guarantees to address the SUP problem, where the platform and workers have no prior knowledge of the Stackelberg game.
- Platform conducts interactions with workers to learn the optimal strategies from game experiences.
- Learning Framework: Twin Delayed Deep Deterministic Policy Gradient (TD3).
    - TD3 contains two types of networks, actor network and two critic networks.
    - Actor network learns the policy $\pi_\theta(a \mid s)$, and decides what action to take in a given state.
    - Critic network evaluates the quality, $Q(s, a)$, of the action and guide actor to improve decisions.
    - **Each actor (or critic) network is composed of two sub-networks: online network and target network.**

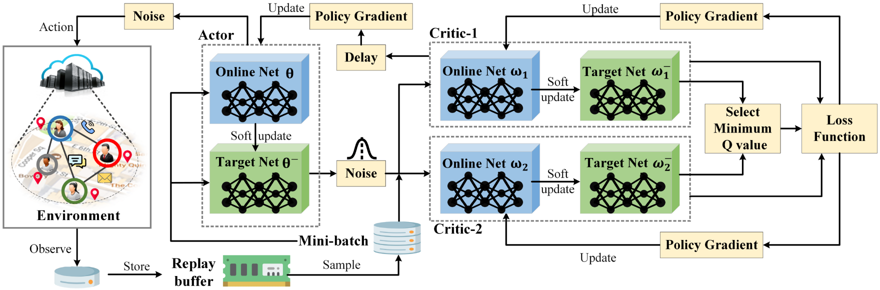
DRL-based Incentive Mechanism (DIM)

- DRL-based Incentive Mechanism (DIM) with AoI guarantees to address the SUP problem, where the platform and workers have no prior knowledge of the Stackelberg game.

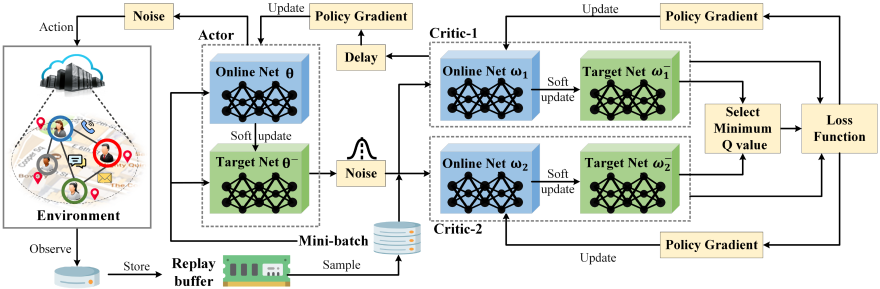
- Platform conducts interactions with workers to learn the optimal strategies from game experiences.

- Learning Framework: Twin Delayed Deep Deterministic Policy Gradient (TD3).
  - TD3 contains two types of networks, actor network and two critic networks.
  - Actor network learns the policy $\pi_\theta(a \mid s)$, and decides what action to take in a given state.
  - Critic network evaluates the quality, $Q(s, a)$, of the action and guide actor to improve decisions.
  - Each actor (or critic) network is composed of two sub-networks: online network and target network.
  - **The target networks used to compute the target actions and the target Q-values of the next state.**

DIM Learning Framework (continue 1)

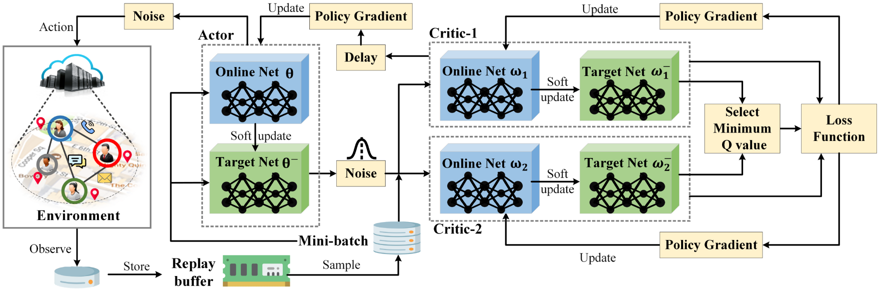


DIM Learning Framework (continue 1)



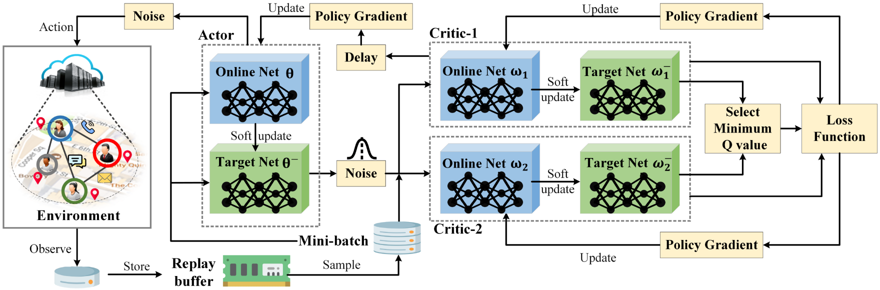
- By selecting the minimum Q-value of the two critics target networks, TD3 can achieve more accurate value and the overestimation bias is significantly reduced.

DIM Learning Framework (continue 1)



- By selecting the minimum Q-value of the two critics target networks, TD3 can achieve more accurate value and the overestimation bias is significantly reduced.
- Actor network update frequency is less than the critic networks to mitigate policy oscillations, reduce overfitting risks, and stabilize the training process.

DIM Learning Framework (continue 1)



- By selecting the minimum Q-value of the two critics target networks, TD3 can achieve more accurate value and the overestimation bias is significantly reduced.
- Actor network update frequency is less than the critic networks to mitigate policy oscillations, reduce overfitting risks, and stabilize the training process.
- Random noise promotes more diverse exploration and reduces the tendency to exploit a single action.

DIM Learning Framework (continue 2)

- **At each time slot t, the platform plays the role of the game leader, announcing the payment to each worker.**
  Subsequently, workers act as game followers and determine their respective data update frequencies.

DIM Learning Framework (continue 2)

- At each time slot t, the platform plays the role of the game leader, announcing the payment to each worker.
  Subsequently, workers act as game followers and determine their respective data update frequencies.
- **based on the current state observed from the environment, the platform or each worker can map the state to an appropriate action and will execute the action. Upon observing the impact of the action on the environment, the platform or each worker will receive an immediate reward, and the current state is transited to the next state.**

DIM Learning Framework (continue 2)

- At each time slot t, the platform plays the role of the game leader, announcing the payment to each worker.
  Subsequently, workers act as game followers and determine their respective data update frequencies.

- based on the current state observed from the environment, the platform or each worker can map the state to an appropriate action and will execute the action. Upon observing the impact of the action on the environment, the platform or each worker will receive an immediate reward, and the current state is transited to the next state.

- **The experience (i.e., current state, action, reward, and next state) is saved in the finite-size replay buffer, which used for the network update.**

Payment Strategy for the Platform

- Payment decision in the two-stage Stackelberg game is modeled as a Markov Decision Process (MDP).

Payment Strategy for the Platform

- Payment decision in the two-stage Stackelberg game is modeled as a Markov Decision Process (MDP).

- MDP Freamwork:

Payment Strategy for the Platform

- Payment decision in the two-stage Stackelberg game is modeled as a Markov Decision Process (MDP).

- MDP Freamwork:
  - State Space: includes vectors of historical payment strategies and corresponding data update frequencies over a defined number of past τ time slots:

$$s^t = \{R^{t-\tau}, P^{t-\tau}, R^{t-\tau+1}, P^{t-\tau+1}, \ldots, R^{t-1}, P^{t-1}\}.$$

Payment Strategy for the Platform

- Payment decision in the two-stage Stackelberg game is modeled as a Markov Decision Process (MDP).

- MDP Freamwork:
    - State Space: includes vectors of historical payment strategies and corresponding data update frequencies over a defined number of past τ time slots:
    $$s^t = \{R^{t-\tau}, P^{t-\tau}, R^{t-\tau+1}, P^{t-\tau+1}, \dots, R^{t-1}, P^{t-1}\}.$$

    - Action Space: platform's payment strategy at the current time slot t according to the actor network with random noise: $R^t = \Pi_\theta(s^t) + \xi$.

Payment Strategy for the Platform

- Payment decision in the two-stage Stackelberg game is modeled as a Markov Decision Process (MDP).
- MDP Freamwork:
  - State Space: includes vectors of historical payment strategies and corresponding data update frequencies over a defined number of past τ time slots:

$$s^t = \{R^{t-\tau}, P^{t-\tau}, R^{t-\tau+1}, P^{t-\tau+1}, \ldots, R^{t-1}, P^{t-1}\}.$$

  - Action Space: platform's payment strategy at the current time slot t according to the actor network with random noise: $R^t = \Pi_\theta(s^t) + \xi$.
  - Reward Function:

$$\Upsilon(\mathbf{s}^t, \mathcal{R}^t) = \varrho_1 \Phi(P^t, \mathcal{R}^t) - \varrho_2 \left[\frac{\sum_{i=1}^{N} p_i^t - \hat{p}}{\hat{p}} \right]^+ - \varrho_3 \sum_{i=1}^{N} \left[\frac{\delta_i^t - \epsilon}{\epsilon} \right]^+,$$

Network Training

1. **Platform: picks action $R_t \rightarrow$ Environment (workers in parallel).**

Network Training

1. Platform: picks action $R_t \rightarrow$ Environment (workers in parallel).

2. **Environment: aggregates the workers' responses $\rightarrow$ returns** $(s_{t+1},$ **reward**$)$.

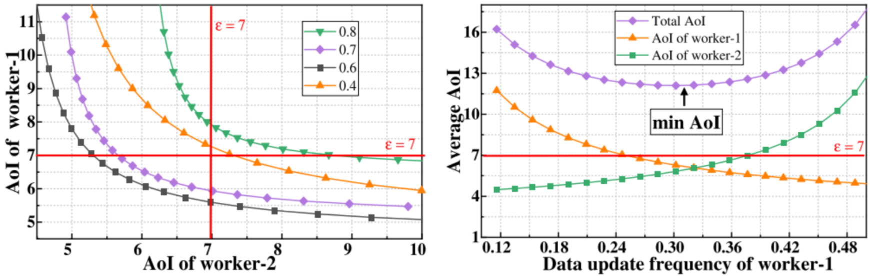
Network Training

1. Platform: picks action $R_t \rightarrow$ Environment (workers in parallel).
2. Environment: aggregates the workers' responses $\rightarrow$ returns (s_{t+1}, reward).
3. **Platform: stores $\langle s_t, R_t, \textbf{reward}, s_{t+1} \rangle$ in replay buffer.**

Network Training

1. Platform: picks action $R_t \rightarrow$ Environment (workers in parallel).
2. Environment: aggregates the workers' responses $\rightarrow$ returns (s_{t+1}, reward).
3. Platform: stores $\langle s_t, R_t, \text{reward}, s_{t+1} \rangle$ in replay buffer.
4. **Platform: (possibly) samples mini-batches from replay buffer $\rightarrow$ updates neural networks.**

Evaluation of AoI



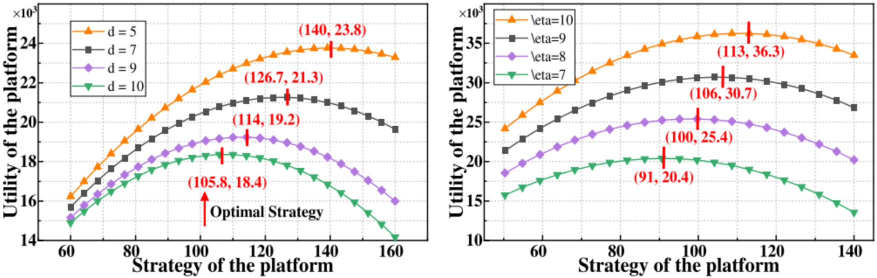
(a) worker-1's AoI vs. worker-2's AoI (b) total AoI and each worker's AoI

Evaluating AoI: two workers, fixed total load $\rho_1 + \rho_2 = \hat{\rho}$.

(a) AoI value of worker-1 decreases with the increase of worker-2's AoI value.

(b) AoI optimization problem depends on both the total load $\hat{\rho}$ and the allocation of data update frequency among workers.

Evaluation of AIM - Platform

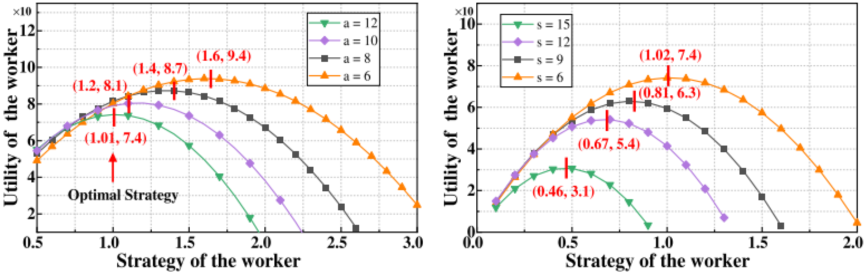


(a) quadratic parameter d (b) conversion parameter η

Evaluating AIM: Platform Strategy Vs. Utility with different parameters.

- PU will always find a maximum point.
- A small d or a larger η will result in the growth of the optimal *PS* and the optimal *PU*.

Evaluation of AIM - Worker

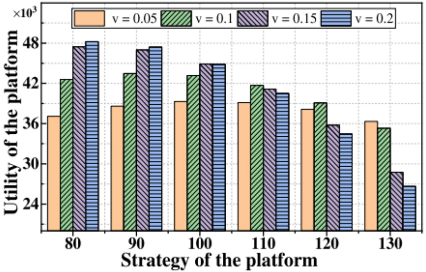


(a) quadratic parameter a (b) conversion parameter s

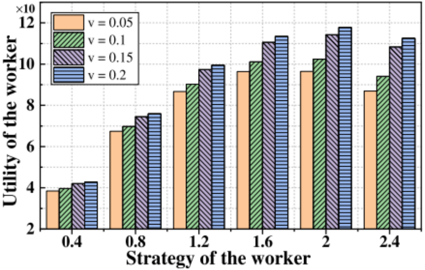
Evaluating AIM: Worker Strategy Vs. Utility with different parameters.

- WU find a maximum point.
- worker's utility increase when applying a smaller a or s since the cost of the worker becomes smaller.

Evaluation of Social Network Effect



(a) utility of the platform



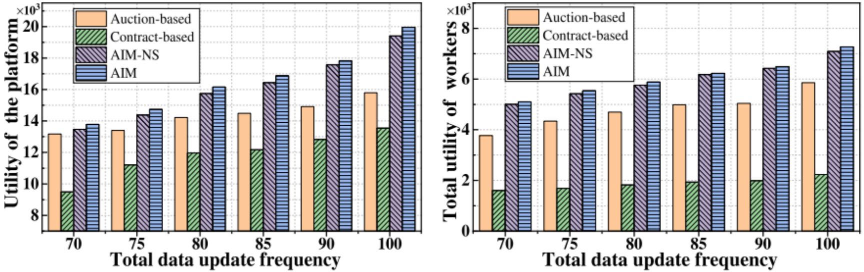
(b) utility of the worker

Evaluating Influence of social network effects:

- Enlarging social network effect coefficient υ from 0.05 to 0.2, enables both the worker and platform to achieve higher utility.

Evaluation of Different Incentive Mechanisms

Evaluation of Different Incentive Mechanisms:

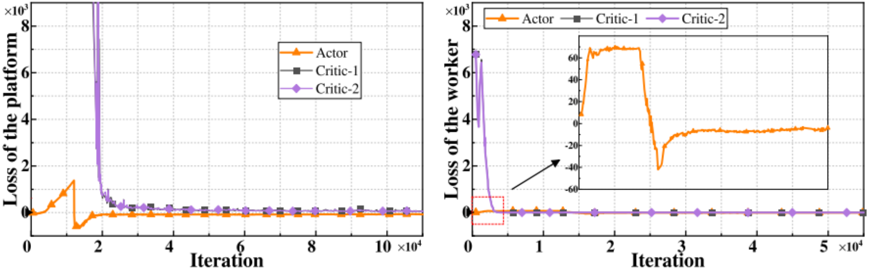


(a) PU vs. $\hat{p}$ (b) total WU vs. $\hat{p}$

- When $\hat{p} = 100$, the achieved PU of AIM is about 47.3% and 26.4% higher than those of the contract-based and auction-based algorithms on average, respectively.

Evaluation of DIM Conversion

Evaluation of DIM Conversion via training loss:



(a) loss of the platform (b) loss of the worker

- Both the actor network and two critic networks of the platform tend to a stable state as the training time increases.

Conclusion

Modeling Approach: The problem is framed as a two-stage Stackelberg game with an embedded Bayesian sub-game to account for incomplete information.

Conclusion

Modeling Approach: The problem is framed as a two-stage Stackelberg game with an embedded Bayesian sub-game to account for incomplete information.

Proposed Mechanisms:

Conclusion

Modeling Approach: The problem is framed as a two-stage Stackelberg game with an embedded Bayesian sub-game to account for incomplete information.

Proposed Mechanisms:

- **AIM (AoI-guaranteed Incentive Mechanism):**

Conclusion

Modeling Approach: The problem is framed as a two-stage Stackelberg game with an embedded Bayesian sub-game to account for incomplete information.

Proposed Mechanisms:

- **AIM (AoI-guaranteed Incentive Mechanism):**
    - Designed for scenarios where participants share utility function parameters.
    - Ensures unique Stackelberg equilibrium and maximizes the utilities of both the platform and workers.

Conclusion

Modeling Approach: The problem is framed as a two-stage Stackelberg game with an embedded Bayesian sub-game to account for incomplete information.

Proposed Mechanisms:

- **AIM (AoI-guaranteed Incentive Mechanism):**
  - Designed for scenarios where participants share utility function parameters.
  - Ensures unique Stackelberg equilibrium and maximizes the utilities of both the platform and workers.
- **DIM (DRL-based Incentive Mechanism):**

Conclusion

Modeling Approach: The problem is framed as a two-stage Stackelberg game with an embedded Bayesian sub-game to account for incomplete information.

Proposed Mechanisms:

- **AIM (AoI-guaranteed Incentive Mechanism):**
  - Designed for scenarios where participants share utility function parameters.
  - Ensures unique Stackelberg equilibrium and maximizes the utilities of both the platform and workers.
- **DIM (DRL-based Incentive Mechanism):**
  - Extended for scenarios with no prior knowledge of utility parameters.
  - Utilizes Deep Reinforcement Learning (DRL) to enable participants to learn optimal strategies through experience.
  - Ensures AoI values remain within a specified threshold.

Conclusion

Modeling Approach: The problem is framed as a two-stage Stackelberg game with an embedded Bayesian sub-game to account for incomplete information.

Proposed Mechanisms:

- **AIM (AoI-guaranteed Incentive Mechanism):**
  - Designed for scenarios where participants share utility function parameters.
  - Ensures unique Stackelberg equilibrium and maximizes the utilities of both the platform and workers.
- **DIM (DRL-based Incentive Mechanism):**
  - Extended for scenarios with no prior knowledge of utility parameters.
  - Utilizes Deep Reinforcement Learning (DRL) to enable participants to learn optimal strategies through experience.
  - Ensures AoI values remain within a specified threshold.

Performance Evaluation:

Conclusion

Modeling Approach: The problem is framed as a two-stage Stackelberg game with an embedded Bayesian sub-game to account for incomplete information.

Proposed Mechanisms:

- **AIM (AoI-guaranteed Incentive Mechanism):**
  - Designed for scenarios where participants share utility function parameters.
  - Ensures unique Stackelberg equilibrium and maximizes the utilities of both the platform and workers.
- **DIM (DRL-based Incentive Mechanism):**
  - Extended for scenarios with no prior knowledge of utility parameters.
  - Utilizes Deep Reinforcement Learning (DRL) to enable participants to learn optimal strategies through experience.
  - Ensures AoI values remain within a specified threshold.

Performance Evaluation:

- Experiments with real-world data validate the efficacy of AIM and DIM.
- Both mechanisms outperform baseline methods in utility optimization and AoI guarantee.

My Contribution

Areas for improvements:

My Contribution

Areas for improvements:

- Practical Applicability: The paper focuses on theoretical formulations and simulations but does not sufficiently address the challenges of implementing AIM or DIM in real-world MCS systems.

My Contribution

Areas for improvements:

- Practical Applicability: The paper focuses on theoretical formulations and simulations but does not sufficiently address the challenges of implementing AIM or DIM in real-world MCS systems.
- Social Benefit Modeling: The paper assumes stable social relationships and fixed system parameters, which may not reflect the dynamic nature of real-world systems.

My Contribution

Areas for improvements:

- Practical Applicability: The paper focuses on theoretical formulations and simulations but does not sufficiently address the challenges of implementing AIM or DIM in real-world MCS systems.
- Social Benefit Modeling: The paper assumes stable social relationships and fixed system parameters, which may not reflect the dynamic nature of real-world systems.

Future Directions:

My Contribution

Areas for improvements:

- Practical Applicability: The paper focuses on theoretical formulations and simulations but does not sufficiently address the challenges of implementing AIM or DIM in real-world MCS systems.
- Social Benefit Modeling: The paper assumes stable social relationships and fixed system parameters, which may not reflect the dynamic nature of real-world systems.

Future Directions:

- Explore social interaction models, such as time-varying influence networks or dynamic feedback from workers.

My Contribution

Areas for improvements:

- Practical Applicability: The paper focuses on theoretical formulations and simulations but does not sufficiently address the challenges of implementing AIM or DIM in real-world MCS systems.
- Social Benefit Modeling: The paper assumes stable social relationships and fixed system parameters, which may not reflect the dynamic nature of real-world systems.

Future Directions:

- Explore social interaction models, such as time-varying influence networks or dynamic feedback from workers.
- Propose mechanisms to integrate other metrics beyond AoI (e.g., data quality or energy efficiency).

My Contribution

Areas for improvements:

- Practical Applicability: The paper focuses on theoretical formulations and simulations but does not sufficiently address the challenges of implementing AIM or DIM in real-world MCS systems.
- Social Benefit Modeling: The paper assumes stable social relationships and fixed system parameters, which may not reflect the dynamic nature of real-world systems.

Future Directions:

- Explore social interaction models, such as time-varying influence networks or dynamic feedback from workers.
- Propose mechanisms to integrate other metrics beyond AoI (e.g., data quality or energy efficiency).
- Investigate the impact of collaborative behavior among workers.