

## Capitolo 1: Introduzione

### 1.1. Contesto: Il Paradigma del Mobile Crowdsensing

Negli ultimi anni, la diffusione capillare di dispositivi mobili dotati di una crescente varietà di sensori (GPS, accelerometri, microfoni) ha dato vita a un nuovo paradigma di raccolta dati su larga scala: il **Mobile Crowdsensing (MCS)**. A differenza dei tradizionali network di sensori statici, i sistemi MCS sfruttano la mobilità e l'ubiquità degli utenti comuni per acquisire informazioni geolocalizzate sull'ambiente circostante, con applicazioni che spaziano dal monitoraggio del traffico e dell'inquinamento acustico alla mappatura della qualità delle reti cellulari.

Il successo di un'iniziativa MCS dipende in modo critico dalla capacità di reclutare e mantenere un numero sufficiente di partecipanti disposti a contribuire con le risorse dei propri dispositivi (batteria, CPU, banda dati) e a sostenere i costi associati all'esecuzione dei "task" di sensing (e.g., deviazioni dal proprio percorso ottimale). Questo introduce una sfida fondamentale di natura economica e socio-tecnica: la progettazione di efficaci **meccanismi di incentivazione**.

### 1.2. Il Problema Centrale: Progettazione di Incentivi in Sistemi MCS

Un meccanismo di incentivazione in un sistema MCS deve risolvere un complesso problema di allocazione di risorse in condizioni di informazione asimmetrica. La piattaforma che richiede i dati non conosce a priori il costo privato che ogni potenziale partecipante deve sostenere per completare un task. Un meccanismo ideale dovrebbe, pertanto, essere in grado di:

1. **Stimolare la partecipazione:** Offrire una remunerazione sufficiente a coprire i costi degli utenti e a garantire loro un'utilità netta positiva (proprietà di *Individual Rationality*).
2. **Incentivare l'onestà:** Disincentivare comportamenti strategici, come la dichiarazione di costi gonfiati al fine di ottenere pagamenti maggiori (proprietà di *Truthfulness* o Incentivo-Compatibilità).
3. **Garantire l'efficienza:** Selezionare il sottoinsieme di utenti che massimizza il valore complessivo per la piattaforma, nel rispetto del vincolo di profittabilità (utilità non negativa) o di un budget predefinito.

Il paper "*Incentive Mechanisms for Mobile Crowdsensing*" di Yang et al. (2015) affronta questo problema in modo rigoroso, proponendo un meccanismo d'asta denominato **IMCU (Incentive Mechanism for Crowdsensing Users)** che garantisce teoricamente le proprietà sopra elencate.

### 1.3. Obiettivo e Scopo della Tesi (Fase 1)

L'obiettivo primario di questo lavoro è la **progettazione, l'implementazione e la validazione empirica di un sistema di Mobile Crowdsensing basato sui principi teorici del meccanismo IMCU**. Questa tesi si propone di andare oltre la mera descrizione teorica, costruendo un modello computazionale completo in grado di simulare il funzionamento del meccanismo in un contesto realistico e di verificarne la coerenza e la stabilità.

La **Fase 1**, oggetto della presente documentazione, si concentra sulla costruzione di una **baseline teorica pura**. L'obiettivo specifico di questa fase è dimostrare che un'implementazione fedele del modello IMCU, popolata da agenti con **razionalità perfetta**, si comporta esattamente come previsto dalla teoria. A tal fine, si assume che gli utenti, agendo come agenti economici perfettamente razionali, perseguano esclusivamente la massimizzazione della propria utilità personale e adottino sempre la strategia dominante offerta dal meccanismo: la dichiarazione veritiera del proprio costo privato.

Per ancorare la simulazione a un dominio applicativo realistico, il sistema viene calato nel contesto dei **tassisti operanti nella città di Roma**, utilizzando un dataset proprietario di tracciamenti GPS per modellare la domanda di sensing e la distribuzione spaziale dei partecipanti.

#### 1.4. Struttura della Documentazione

Il presente documento è strutturato per guidare il lettore attraverso la modellazione teorica, l'implementazione algoritmica e l'analisi dei risultati della Fase 1 del progetto.

- **Capitolo 2 - Modello Matematico del Sistema e degli Agenti:** Formalizza le entità fondamentali del sistema, il Task e l'Utente, definendone le proprietà matematiche e i modelli di costo e comportamento.
- **Capitolo 3 - Il Meccanismo d'Asta IMCU User-Centric:** Descrive in dettaglio gli algoritmi di selezione dei vincitori e di determinazione del pagamento critico che costituiscono il cuore del meccanismo IMCU.
- **Capitolo 4 - Metodologia Sperimentale e Preparazione dei Dati:** Illustra il processo ETL applicato al dataset grezzo e le metodologie per la discretizzazione dello spazio e la generazione delle popolazioni di task e utenti.
- **Capitolo 5 - Metriche di Valutazione e Metodologia di Analisi:** Definisce gli indicatori di performance (KPI) e gli strumenti statistici e di visualizzazione utilizzati per analizzare i risultati delle simulazioni.
- **Capitolo 6 - Protocollo Sperimentale e Esecuzione delle Simulazioni:** Descrive la pipeline di orchestrazione che integra tutti i moduli precedenti per eseguire la campagna di esperimenti in modo automatizzato e riproducibile.
- **Capitolo 7 - Risultati Sperimentali e Analisi Comparativa:** Presenta e interpreta i risultati ottenuti dallo scenario di riferimento e dall'analisi di sensitività su parametri chiave.
- **Capitolo 8 - Analisi Statistica Avanzata e Discussione:** Approfondisce i risultati tramite tecniche inferenziali e di clustering per scoprire relazioni latenti e regimi operativi del sistema.

Infine, le **Conclusioni** riassumono i contributi della Fase 1 e delineano le direzioni per i lavori futuri, incentrati sull'introduzione di modelli di agenti con razionalità limitata.

## Capitolo 2: Modello Matematico del Sistema e degli Agenti

### 2.1. Introduzione al Modello

Questo capitolo introduce le entità fondamentali che popolano l'ecosistema di Mobile Crowdsensing (MCS) e ne formalizza le proprietà matematiche e comportamentali. La modellazione si allinea ai principi dei meccanismi di incentivazione *user-centric*, come delineato da Yang et al. (2015), ponendo le basi per l'analisi di un'asta veritiera in condizioni di informazione privata e agenti economicamente razionali. Definiamo rigorosamente l'entità **Task**, che rappresenta la domanda di dati geolocalizzati, e l'entità **Utente**, che modella l'offerta di sensing da parte di un tassista operante nel contesto urbano di Roma.

### 2.2. L'Entità Task: Formalizzazione della Domanda di Sensing

Un task di sensing rappresenta un'unità atomica di richiesta di dati da parte della piattaforma. Ogni task è definito da una posizione geografica e da un valore intrinseco che la piattaforma associa alla sua esecuzione.

**Definizione 2.1 (Task di Sensing)** Un task di sensing  $\tau_j$  è una tupla  $\tau_j = \langle id_j, pos_j, \nu_j \rangle$ , dove:

- $id_j \in \mathbb{N}^+$  è un identificatore univoco.
- $pos_j \in \mathbb{R}^2$  è una coppia di coordinate geografiche (latitudine, longitudine) in gradi decimali (standard EPSG:4326), che identifica il luogo dove il dato deve essere raccolto.
- $\nu_j \in \mathbb{R}_0^+$  è il **valore** del task, che quantifica l'utilità marginale che la piattaforma ottiene dal completamento di  $\tau_j$ .

### Definizione 2.2 (Distribuzione del Valore dei Task - Baseline Calibrata)

Il valore  $\nu_j$  di un task generato casualmente è una variabile aleatoria  $V \sim U(\nu_{\min}, \nu_{\max})$  con distribuzione **uniforme** nel range  $[\nu_{\min}, \nu_{\max}]$ , la cui funzione di densità di probabilità (PDF) è data da:

$$f(\nu) = \begin{cases} \frac{1}{\nu_{\max} - \nu_{\min}} & \text{se } \nu \in [\nu_{\min}, \nu_{\max}] \\ 0 & \text{altrimenti} \end{cases}$$

A differenza dei parametri  $U[1.0, 5.0]$  utilizzati nel paper di riferimento, questa simulazione adotta un range **calibrato empiricamente** per bilanciare l'economia del sistema con i costi operativi realistici dei taxi:

- $\nu_{\min} = 1.8 \text{ €}$
- $\nu_{\max} = 15.0 \text{ €}$

Questa scelta di distribuzione uniforme è motivata da:

1. **Allineamento Metodologico:** L'uso di una distribuzione uniforme rispecchia l'approccio del paper IMCU, garantendo la comparabilità delle proprietà del meccanismo (pur con parametri diversi).
2. **Neutralità a priori:** In assenza di informazione specifica sulla struttura della domanda, la distribuzione uniforme rappresenta l'ipotesi di massima entropia.
3. **Equilibrio Economico:** Questo range produce un valore atteso  $\mathbb{E}[V] = (1.8 + 15.0)/2 = 8.4$  €, che è dimensionalmente coerente con i costi operativi calibrati degli utenti (Sezione 2.4.1), garantendo un'economia di simulazione sana.

**Nota per estensioni future (Fase 2):** Il framework implementato supporta anche distribuzioni alternative (e.g., log-normale con calibrazione empirica da dati di domanda urbana) per modellare scenari realistici di task non omogenei.

### 2.3. L'Entità Utente: Formalizzazione dell'Agente Razionale

Un utente rappresenta un agente autonomo e razionale (un tassista) che offre capacità di sensing. Ogni utente è caratterizzato da una posizione iniziale e da una funzione di costo privata, che ne determina il comportamento economico all'interno del meccanismo d'asta.

**Definizione 2.3 (Utente)** Un utente  $U_i$  è una tupla  $U_i = \langle id_i, pos_i, \kappa_i \rangle$ , dove:

- $id_i \in \mathbb{N}^+$  è un identificatore univoco.
- $pos_i \in \mathbb{R}^2$  sono le coordinate geografiche della sua posizione iniziale.
- $\kappa_i \in \mathbb{R}^+$  è il **costo operativo per chilometro**, un parametro privato che rappresenta il costo marginale sostenuto dall'utente per ogni chilometro percorso.

Il parametro  $\kappa_i$  cattura l'eterogeneità della flotta di agenti, inglobando fattori quali il consumo di carburante del veicolo, i costi di manutenzione, l'ammortamento e gli oneri fiscali. Tale parametro è modellato come una variabile aleatoria estratta da una distribuzione uniforme  $\kappa_i \sim U[\kappa_{\min}, \kappa_{\max}]$ , con estremi calibrati su dati reali del settore taxi.

### 2.4. Modello di Costo, Bidding e Utilità

Il cuore del comportamento dell'agente risiede nella valutazione del costo per l'esecuzione di un insieme di task e nella conseguente formulazione di un'offerta (bid).

#### 2.4.1. Funzione di Costo Privato ( $c_i$ )

Il costo privato  $c_i$  per un utente  $i$  che si impegna a completare un insieme di task  $\Gamma_i$  è direttamente proporzionale alla distanza totale che deve percorrere.

**Definizione 2.4 (Costo Privato)** Il costo  $c_i$  per l'utente  $U_i$  per servire l'insieme di task  $\Gamma_i$  è definito come:

$$c_i(\Gamma_i) = \kappa_i \cdot D_i(\Gamma_i)$$

dove  $D_i(\Gamma_i)$  è la distanza totale di servizio stimata.

Per questa fase teorica, si adotta un modello di routing semplificato detto **Star Routing**, dove si assume che l'utente parta dalla sua posizione base  $\text{pos}_i$  per servire ogni task in  $\Gamma_i$  individualmente.

### Nota sulla Calibrazione del Costo (Divergenza dal Paper)

Il paper IMCU (Yang et al., 2015) modella il costo utente come:

$$c_i = \rho \cdot |\Gamma_i|$$

dove  $\rho \sim U[1, 10]$  è un **costo per task**, con un costo medio atteso  $\bar{c}_i^{\text{paper}} \approx 27.5 \text{ €}$  (assumendo 5 task).

Il nostro modello adatta questa formulazione al contesto taxi realistico usando **costo per chilometro**. A seguito di un'analisi rigorosa dei costi operativi (carburante, manutenzione, ammortamento) per i taxi a Roma, il nostro modello è calibrato con:

$$\kappa_i \sim U[0.45, 0.70] \text{ €/km}$$

Questo range, con un costo medio  $\bar{\kappa}_i = (0.45 + 0.70)/2 = 0.575 \text{ €/km}$ , è **scientificamente più realistico** del range  $U[0.5, 5.0]$  ipotizzato in bozze precedenti.

Verifica del costo atteso nel nostro modello (assumendo i medesimi 5 task a 2 km):

$$\bar{c}_i^{\text{nostro}} = \bar{\kappa}_i \cdot \bar{D}_i = 0.575 \cdot (5 \times 2) = 5.75 \text{ €}$$

**Conclusione sulla Calibrazione:** Il nostro modello *non* mira all'equivalenza numerica con i costi del paper (5.75 € vs 27.5 €), ma mira a un'**equivalenza metodologica** (uso di distribuzioni uniformi) applicata a parametri **empiricamente calibrati** sul dominio reale (taxi Roma), garantendo un'economia di simulazione valida dove  $v_i \sim c_i$ .

### Definizione 2.5 (Distanza di Servizio con Modello Star Routing)

La distanza totale  $D_i(\Gamma_i)$  è calcolata come la somma delle distanze geodetiche tra la posizione dell'utente e ciascun task:

$$D_i(\Gamma_i) = \lambda_{\text{urban}} \cdot \sum_{\tau_j \in \Gamma_i} d_H(\text{pos}_i, \text{pos}_j)$$

dove:

- $d_H(\cdot, \cdot)$  è la **distanza geodetica di Haversine**, che calcola la distanza ortodromica tra due punti su una superficie sferica (raggio Terra  $R = 6,371 \text{ km}$ ).
- $\lambda_{\text{urban}}$  è un **fattore di correzione urbano**.

La formula di Haversine per calcolare la distanza  $d$  tra due punti con latitudini  $\phi_1, \phi_2$  e longitudini  $\lambda_1, \lambda_2$  (in radianti) su una sfera di raggio  $R$  è:

$$a = \sin^2 \left( \frac{\Delta\phi}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left( \frac{\Delta\lambda}{2} \right)$$

$$d = 2R \cdot \text{atan2}(\sqrt{a}, \sqrt{1-a})$$

con  $\Delta\phi = \phi_2 - \phi_1$  e  $\Delta\lambda = \lambda_2 - \lambda_1$ .

### Calibrazione Fase 1 (Baseline Realistica):

A differenza dell'approccio teorico puro del paper (che usa distanza Euclidea,  $\lambda_{\text{urban}} = 1.0$ ), questa simulazione di Fase 1 adotta un **modello di costo più realistico** per il contesto urbano:

$$\lambda_{\text{urban}} = 1.30$$

Questo fattore, calibrato sulla rete stradale di Roma (Barthelemy, 2011), approssima la distanza stradale reale (tortuosità, sensi unici) come 1.3 volte la distanza aerea (Haversine). Tale scelta:

- Fornisce un costo  $c_i$  più accurato.
- Garantisce che la baseline teorica sia comunque fondata su un costo realistico.
- Permette una validazione delle proprietà del meccanismo in condizioni più stringenti.

#### 2.4.2. Strategia di Bidding in Regime di Razionalità Perfetta

Un presupposto fondamentale di questa fase è che gli agenti agiscano con **razionalità perfetta**. Essi conoscono le regole del meccanismo d'asta IMCU e agiscono per massimizzare la propria utilità. Il meccanismo IMCU è progettato per essere **Incentivo-Compatibile in Strategie Dominanti** (DSIC), o *truthful*.

**Teorema 2.1 (Veridicità del Meccanismo IMCU - da Yang et al., 2015)** *In un'asta IMCU user-centric, per qualsiasi utente  $i$  con costo privato  $c_i(\Gamma_i)$  per servire un insieme di task  $\Gamma_i$ , la strategia di sottomettere un'offerta (bid) veritiera  $b_i = c_i(\Gamma_i)$  è una **strategia dominante**. Ciò significa che tale strategia massimizza l'utilità dell'utente  $i$  indipendentemente dalle offerte sottomesse da tutti gli altri utenti.*

**Dimostrazione.** La dimostrazione formale si basa sulla struttura del pagamento critico del meccanismo (che sarà dettagliato nel Capitolo 3). Intuitivamente, un utente che offre  $b_i > c_i$  (overbidding) rischia di perdere l'asta in situazioni in cui avrebbe potuto vincere con un'utilità positiva. Un utente che offre  $b_i < c_i$  (underbidding) rischia di vincere l'asta ma di ricevere un pagamento  $p_i < c_i$ , risultando in un'utilità negativa. Pertanto, non esiste deviazione dalla strategia veritiera che possa migliorare l'esito per l'agente.

**Corollario 2.1 (Comportamento di Bidding Assunto)** Data l'assunzione di razionalità perfetta e il Teorema 2.1, il modello di comportamento per ogni utente  $U_i$  impone che l'offerta sottomessa alla piattaforma sia identica al suo costo privato calcolato.

$$b_i(\Gamma_i) = c_i(\Gamma_i)$$

Questo non è un vincolo meccanico, ma la simulazione dell'esito di equilibrio strategico in cui agenti razionali rispondono ottimalmente agli incentivi forniti dal sistema.

#### 2.4.3. Funzione di Utilità dell'Agente\*

L'utilità di un agente è il guadagno netto ottenuto dalla partecipazione all'asta.

**Definizione 2.6 (Utilità dell'Utente)** L'utilità  $u_i$  per l'utente  $U_i$  che ha sottomesso un'offerta per l'insieme  $\Gamma_i$  è definita a tratti:

$$u_i = \begin{cases} p_i - c_i(\Gamma_i) & \text{se } U_i \text{ è un vincitore} \\ 0 & \text{se } U_i \text{ non è un vincitore} \end{cases}$$

dove  $p_i$  è il pagamento ricevuto dalla piattaforma.

Il meccanismo d'asta deve inoltre garantire la **razionalità individuale**.

**Lemma 2.1 (Razionalità Individuale)** Un meccanismo d'asta è individualmente razionale se l'utilità di ogni partecipante è non negativa, ovvero  $u_i \geq 0$  per ogni utente  $i$ . Il meccanismo IMCU, grazie alla sua regola di pagamento, soddisfa questa proprietà.

Questa condizione assicura che un agente razionale abbia sempre un incentivo a partecipare all'asta, non potendo mai incorrere in una perdita netta.

### Capitolo 3: Il Meccanismo d'Asta IMCU User-Centric\*

#### 3.1. Introduzione al Meccanismo

Questo capitolo presenta il nucleo algoritmico del sistema: il meccanismo d'asta IMCU (Incentive Mechanism for Crowdsensing Users) nella sua formulazione *user-centric*. Tale meccanismo è progettato per operare in un contesto di informazione asimmetrica, dove i costi degli utenti sono privati. L'architettura dell'asta è non banale, in quanto deve simultaneamente selezionare un insieme di utenti "vincitori" e determinare per ciascuno un pagamento che incentivi un comportamento onesto.

Come dimostrato nel lavoro di riferimento (Yang et al., 2015), il meccanismo IMCU garantisce un insieme di proprietà teoriche desiderabili:

1. **Veridicità (Truthfulness):** La strategia dominante per ogni agente razionale è rivelare il proprio costo veritiero.
2. **Razionalità Individuale (Individual Rationality):** La partecipazione all'asta non produce mai un'utilità negativa per alcun agente.
3. **Efficienza Computazionale (Computational Tractability):** L'algoritmo termina in tempo polinomiale rispetto al numero di utenti e task.
4. **Profittabilità (Profitability):** L'utilità netta della piattaforma è garantita essere non negativa.

Di seguito, formalizzo i due pilastri dell'algoritmo: la **selezione dei vincitori** e la **determinazione del pagamento critico**.

### 3.2. La Funzione di Valore e la Proprietà di Submodularità

Prima di descrivere l'algoritmo, è essenziale definire la funzione obiettivo della piattaforma e dimostrarne una proprietà matematica cruciale. La piattaforma valuta un insieme di utenti vincitori  $S \subseteq U$  attraverso una funzione di valore  $v : 2^U \rightarrow \mathbb{R}_0^+$ .

**Definizione 3.1 (Funzione di Valore della Piattaforma)** Dato un insieme di utenti vincitori  $S$ , il valore totale per la piattaforma,  $v(S)$ , è la somma dei valori  $\nu_j$  di tutti i task unici coperti dall'unione dei loro insiemi di task  $\Gamma_i$ .

$$v(S) = \sum_{\tau_j \in \bigcup_{i \in S} \Gamma_i} \nu_j$$

Questa funzione possiede la proprietà di **submodularità**, che è fondamentale per la validità degli algoritmi di approssimazione greedy.

**Definizione 3.2 (Submodularità)** Una funzione d'insieme  $f : 2^U \rightarrow \mathbb{R}$  è submodulare se per ogni coppia di insiemi  $A \subseteq B \subset U$  e per ogni elemento  $x \in U \setminus B$ , vale la seguente disuguaglianza:

$$f(A \cup \{x\}) - f(A) \geq f(B \cup \{x\}) - f(B)$$

Questa proprietà formalizza l'intuizione dei "rendimenti marginali decrescenti": l'aggiunta di un elemento a un insieme piccolo porta un beneficio maggiore o uguale rispetto all'aggiunta dello stesso elemento a un insieme più grande.

**Lemma 3.1 (Submodularità della Funzione di Valore IMCU)** La funzione di valore  $v(S)$  definita nella Definizione 3.1 è monotona non decrescente e submodulare.

**Corollario 3.1 (Garanzia Approssimazione Greedy)** Per funzioni submodulari monotone come  $v(S)$ , l'algoritmo greedy (Algoritmo 1) garantisce una soluzione  $S$  tale che:

$$v(S) \geq \left(1 - \frac{1}{e}\right) \cdot v(S^*)$$

dove  $S^*$  è la soluzione ottimale. Questo deriva dal classico risultato di Nemhauser et al. (1978) sull'ottimizzazione submodulare.

**Dimostrazione. Monotonicità:** Siano  $A \subseteq B \subseteq U$ . L'unione dei task coperti da  $A$  è un sottoinsieme dell'unione dei task coperti da  $B$ , ovvero  $\bigcup_{i \in A} \Gamma_i \subseteq \bigcup_{i \in B} \Gamma_i$ . Poiché  $\nu_j \geq 0$  per ogni task, segue che  $v(A) \leq v(B)$ .

**Submodularità:** Siano  $A \subseteq B \subset U$  e  $x \in U \setminus B$ . Il contributo marginale di  $x$  a un insieme  $S$  è  $v(S \cup \{x\}) - v(S) = \sum_{\tau_j \in \Gamma_x \setminus \bigcup_{i \in S} \Gamma_i} \nu_j$ . Poiché  $A \subseteq B$ , l'insieme dei task già coperti da  $A$  è un sottoinsieme di quelli coperti da  $B$ :  $\bigcup_{i \in A} \Gamma_i \subseteq \bigcup_{i \in B} \Gamma_i$ . Di



conseguenza, l'insieme dei *nuovi* task introdotti da  $x$  rispetto ad  $A$  è un superinsieme dei *nuovi* task introdotti rispetto a  $B$ :

$$\Gamma_x \setminus \bigcup_{i \in A} \Gamma_i \supseteq \Gamma_x \setminus \bigcup_{i \in B} \Gamma_i$$

Sommando i valori (non negativi) dei task in questi insiemi, si ottiene la disuguaglianza della submodularità:  $v(A \cup \{x\}) - v(A) \geq v(B \cup \{x\}) - v(B)$ .

### 3.3. Algoritmo 1: Selezione dei Vincitori (Winner Selection)

La fase di selezione costruisce iterativamente l'insieme dei vincitori  $S$  attraverso un approccio greedy che, a ogni passo, seleziona l'utente che offre il massimo "guadagno marginale" (marginal gain).

**Definizione 3.3 (Guadagno Marginale)** Dato un insieme di vincitori parziale  $S_k$  alla  $k$ -esima iterazione, il guadagno marginale di un utente candidato  $i \in U \setminus S_k$  è la differenza tra il suo contributo di valore marginale e la sua offerta:

$$\text{gain}(i, S_k) = [v(S_k \cup \{i\}) - v(S_k)] - b_i$$

L'algoritmo procede come segue:

1. **Inizializzazione:** Si parte con un insieme di vincitori vuoto,  $S_0 = \emptyset$ , e l'insieme di tutti gli utenti candidati  $C_0 = U$ .
2. **Iterazione  $k$  (per  $k = 0, 1, 2, \dots$ ):** a. Si calcola il guadagno marginale  $\text{gain}(i, S_k)$  per ogni utente  $i \in C_k$ . b. Si seleziona l'utente  $i^*$  che massimizza tale guadagno:

$$i^* = \arg \max_{i \in C_k} \{\text{gain}(i, S_k)\}$$

In caso di parità, la selezione è deterministica e risolta a favore dell'utente con l'identificatore  $id_i$  minore. c. **Condizione di Arresto:** Se il massimo guadagno è non positivo,  $\text{gain}(i^*, S_k) \leq 0$ , l'algoritmo termina. L'insieme finale dei vincitori è  $S = S_k$ . d. **Aggiornamento:** Altrimenti, si aggiorna l'insieme dei vincitori  $S_{k+1} = S_k \cup \{i^*\}$  e l'insieme dei candidati  $C_{k+1} = C_k \setminus \{i^*\}$ . Si procede all'iterazione  $k + 1$ .

Questo processo garantisce la **profittabilità** della piattaforma, poiché aggiunge utenti solo finché il loro contributo di valore supera il loro costo (espresso tramite il bid).

### 3.4. Algoritmo 2: Determinazione del Pagamento Critico

La determinazione del pagamento è la componente più sofisticata del meccanismo e quella che garantisce la veridicità. Per ogni vincitore  $i \in S$ , l'algoritmo calcola un **pagamento critico**  $p_i$ , ovvero il valore massimo che  $i$  avrebbe potuto offrire e vincere comunque l'asta, mantenendo fissi i bid degli altri.

Per un dato vincitore  $i \in S$ , il calcolo di  $p_i$  avviene simulando un'asta ausiliaria sull'insieme dei suoi "competitor",  $U \setminus \{i\}$ .

1. **Inizializzazione:** Per garantire la Razionalità Individuale, il pagamento  $p_i$  viene inizializzato al valore del bid dell'utente stesso:  $p_i \leftarrow b_i$ .
2. **Costruzione del Prefisso Greedy dei Competitor:** Si costruisce un insieme ordinato (un prefisso)  $T$  di competitor, selezionandoli iterativamente con la stessa logica greedy dell'Algoritmo 1. a. Sia  $T_0 = \emptyset$ . b. All'iterazione  $k$  (per  $k = 1, 2, \dots$ ), si seleziona il competitor  $j_k \in (U \setminus \{i\}) \setminus T_{k-1}$  che massimizza il guadagno marginale rispetto a  $T_{k-1}$ :

$$j_k = \arg \max_{j \in (U \setminus \{i\}) \setminus T_{k-1}} \{[v(T_{k-1} \cup \{j\}) - v(T_{k-1})] - b_j\}$$

c. Se il guadagno marginale di  $j_k$  è non positivo, la costruzione del prefisso si arresta. Sia  $K$  l'indice dell'ultimo competitor aggiunto con guadagno strettamente positivo. Il prefisso finale è  $T_K = \{j_1, \dots, j_K\}$ .

3. **Calcolo dei Pagamenti Candidati:** Per ogni competitor  $j_k$  nel prefisso (per  $k = 1, \dots, K$ ), si calcola una soglia di pagamento candidata. Questa soglia rappresenta il punto in cui il contributo marginale del vincitore  $i$  eguaglia il guadagno marginale del competitor  $j_k$ . **Definizione 3.4 (Valore Marginale Relativo)** Sia  $v_x(A) = v(A \cup \{x\}) - v(A)$  il contributo marginale di un utente  $x$  a un insieme  $A$ . Il pagamento candidato  $\text{cand}_{ik}$  generato dal competitor  $j_k$  (rispetto al prefisso  $T_{k-1}$ ) è:

$$\text{cand}_{ik} = v_i(T_{k-1}) - (v_{j_k}(T_{k-1}) - b_{j_k})$$

Per robustezza, si considera il minimo tra questo valore e il valore marginale stesso, per evitare pagamenti negativi in contesti teorici anomali:

$$\text{cand}_{ik} \leftarrow \min\{\text{cand}_{ik}, v_i(T_{k-1})\}$$

Il pagamento  $p_i$  viene aggiornato ad ogni passo:  $p_i \leftarrow \max(p_i, \text{cand}_{ik})$ .

4. **Pagamento Finale:** Dopo aver iterato su tutti i  $K$  competitor del prefisso, si calcola un'ultima soglia basata sul valore marginale del vincitore  $i$  rispetto all'intero prefisso  $T_K$ .

$$\text{after}_{\text{threshold}} = v_i(T_K)$$

Il pagamento critico finale per il vincitore  $i$  è il massimo tra tutti i valori accumulati:

$$p_i \leftarrow \max(p_i, \text{after}_{\text{threshold}})$$

Questo complesso processo assicura che il pagamento rifletta accuratamente la "pressione competitiva" esercitata dagli altri utenti.

### 3.5. Proprietà Teoriche Garantite

L'interazione tra l'Algoritmo 1 e l'Algoritmo 2 conferisce al meccanismo IMCU le sue proprietà formali.

**Teorema 3.1 (Razionalità Individuale)** Per ogni vincitore  $i \in S$ , il pagamento ricevuto  $p_i$  è maggiore o uguale alla sua offerta  $b_i$ . Di conseguenza, se l'offerta è veritiera ( $b_i = c_i$ ), l'utilità  $u_i = p_i - c_i \geq 0$ . **Dimostrazione.** La proprietà discende direttamente dal Passo 1 dell'Algoritmo 2, dove  $p_i$  è inizializzato a  $b_i$ . Poiché i passi successivi aggiornano  $p_i$  solo tramite l'operatore  $\max(\cdot)$ , il valore di  $p_i$  non può mai scendere al di sotto di  $b_i$ .

**Teorema 3.2 (Monotonicità)** Sia  $i$  un utente che vince l'asta con un'offerta  $b_i$ . Allora  $i$  vincerà anche l'asta con qualsiasi offerta  $b'_i < b_i$ , assumendo che le offerte degli altri utenti rimangano invariate. **Dimostrazione.** La condizione per cui un utente  $i$  viene selezionato dall'Algoritmo 1 in una certa iterazione  $k$  dipende dal suo guadagno marginale  $\text{gain}(i, S_k) = v_i(S_k) - b_i$ . Riducendo  $b_i$  a  $b'_i$ , il guadagno marginale non può che aumentare:  $\text{gain}'(i, S_k) > \text{gain}(i, S_k)$ . Questo aumenta (o lascia invariata) la sua priorità nella selezione greedy, garantendone la vittoria. La dimostrazione formale si trova in Yang et al. (2015), Teorema 3.

La monotonicità è una condizione necessaria per la veridicità, che rimane la proprietà più importante del meccanismo IMCU, come enunciato nel Teorema 2.1.

Assolutamente. Hai individuato il punto esatto in cui i parametri della simulazione devono essere allineati alla nostra "Ground Truth" validata. La documentazione deve riflettere ciò che il codice esegue.

## Capitolo 4: Metodologia Sperimentale e Preparazione dei Dati

### 4.1. Introduzione alla Metodologia

Per validare empiricamente il modello teorico IMCU, è indispensabile definire un protocollo sperimentale rigoroso. Questo capitolo descrive la metodologia adottata per (i) processare e strutturare il dataset grezzo, (ii) modellare un ambiente di sensing realistico attraverso la discretizzazione dello spazio geografico, e (iii) generare popolazioni sintetiche ma realistiche di task e utenti che agiranno come input per le simulazioni del meccanismo d'asta.

### 4.2. Fonte dei Dati e Processo ETL (Extract, Transform, Load)

L'analisi si fonda su un dataset proprietario di tracciamenti GPS della flotta di tassisti operanti nella città di Roma durante il mese di Febbraio 2014.

1. **Estrazione (Extract):** Il dataset grezzo, costituito da record testuali, viene acquisito. Ogni record contiene l'identificativo del tassista, un timestamp e la sua posizione geografica in formato WKT (Well-Known Text).
2. **Trasformazione (Transform):** Viene applicato un processo di pulizia e validazione. I record vengono filtrati per rientrare in una specifica **bounding box** geografica che circonda l'area metropolitana di Roma: latitudine  $\phi \in [41.78^\circ, 42.04^\circ]$  e longitudine  $\lambda \in [12.30^\circ, 12.72^\circ]$ . I record malformati o al di fuori di quest'area

vengono scartati. I timestamp vengono normalizzati e convertiti nel formato ISO 8601.

3. **Caricamento (Load):** I dati validati vengono strutturati in un formato ottimizzato per le query. Il corpus di dati viene partizionato in file CSV distinti su base temporale (giorno e ora), consentendo un accesso efficiente e a bassa latenza a specifiche finestre temporali, requisito fondamentale per le simulazioni orarie.

#### 4.3. Discretizzazione dello Spazio Geografico: Il Modello a Griglia

Per tradurre le posizioni continue del mondo reale in un formato computazionalmente trattabile, lo spazio geografico viene discretizzato in una griglia regolare.

**Definizione 4.1 (Griglia Spaziale)** L'area definita dalla bounding box viene suddivisa in una griglia di celle quadrate di lato  $d_{\text{cell}}$  (espresso in metri). L'origine della griglia  $(\phi_0, \lambda_0)$  corrisponde all'angolo sud-ovest della bounding box.

La proiezione da coordinate geografiche  $(\phi, \lambda)$  a indici di cella discreti  $(i_y, i_x)$  è data dalle seguenti formule:

$$i_y = \left\lfloor \frac{(\phi - \phi_0) \cdot M_{\text{LAT}}}{d_{\text{cell}}} \right\rfloor$$

$$i_x = \left\lfloor \frac{(\lambda - \lambda_0) \cdot M_{\text{LON}}(\bar{\phi})}{d_{\text{cell}}} \right\rfloor$$

dove:

- $M_{\text{LAT}} \approx 111,320 \text{ m/grado}$  è il fattore di conversione (costante) da gradi di latitudine a metri.
- $M_{\text{LON}}(\bar{\phi}) = M_{\text{LAT}} \cdot \cos(\bar{\phi})$  è il fattore di conversione da gradi di longitudine a metri, che dipende dalla latitudine media  $\bar{\phi} = (\phi + \phi_0)/2$  per tenere conto della curvatura terrestre.

L'operazione inversa, che mappa gli indici di una cella  $(i_y, i_x)$  al suo centroide geografico  $(\phi_c, \lambda_c)$ , è fondamentale per posizionare i task.

$$\phi_c = \phi_0 + \frac{(i_y + 0.5) \cdot d_{\text{cell}}}{M_{\text{LAT}}}$$

$$\lambda_c = \lambda_0 + \frac{(i_x + 0.5) \cdot d_{\text{cell}}}{M_{\text{LON}}(\phi_0)}$$

#### 4.4. Generazione dell'Insieme dei Task ( $\mathcal{T}$ )

L'insieme dei task di sensing  $\mathcal{T}$  viene generato dinamicamente per ogni finestra temporale della simulazione, basandosi sulla densità di osservazioni reali del dataset.

1. **Aggregazione:** Per una data finestra temporale (es. un'ora), tutti i punti GPS del dataset vengono proiettati sulla griglia. Per ogni cella  $(i_y, i_x)$ , si calcola il numero di osservazioni  $c_{iy,ix}$  che ricadono al suo interno. Questo conteggio funge da **proxy della domanda di servizio**.
2. **Creazione:** Per ogni cella con un conteggio non nullo ( $c_{iy,ix} > 0$ ), viene istanziato un singolo task  $\tau_j$ , posizionato nel centroide geografico della cella.

Per la baseline della Fase 1, si adotta il modello di distribuzione uniforme descritto di seguito, al fine di allinearsi metodologicamente al paper di riferimento.

#### Definizione 4.2 (Modello di Valore dei Task - Fase 1 Calibrata)

Per creare una baseline teorica e bilanciare l'economia della simulazione, il valore  $\nu_j$  di ogni task è generato da una **distribuzione uniforme**:

$$\nu_j \sim U(\nu_{\min}, \nu_{\max})$$

A differenza dei parametri  $U[1.0, 5.0]$  utilizzati nel paper di riferimento, questa simulazione adotta un range **calibrato empiricamente** per bilanciare l'economia del sistema con i costi operativi realistici dei taxi (Capitolo 2):

- $\nu_{\min} = 1.8 \text{ €}$
- $\nu_{\max} = 15.0 \text{ €}$

Questa scelta è motivata da:

1. **Allineamento Metodologico:** L'uso di una distribuzione uniforme rispecchia l'approccio del paper IMCU, garantendo la comparabilità delle proprietà del meccanismo (pur con parametri diversi).
2. **Neutralità a priori:** In assenza di informazione specifica sulla struttura della domanda, la distribuzione uniforme rappresenta l'ipotesi di massima entropia.
3. **Semplicità analitica:** Facilita il calcolo di metriche teoriche (es. approssimazione ratio  $1 - 1/e$ ) e la verifica delle proprietà del meccanismo.

#### Generazione Operativa:

Nella pipeline di simulazione (Capitolo 6), per ogni task  $\tau_j$  istanziato in una cella geografica con  $c > 0$  osservazioni GPS, il valore viene campionato come `nu_j = random.uniform(VALUE_MIN, VALUE_MAX)`. Questo garantisce che ogni task abbia valore atteso  $\mathbb{E}[\nu_j] = (1.8 + 15.0)/2 = 8.4 \text{ €}$  e varianza  $\text{Var}[\nu_j] = (15.0 - 1.8)^2/12 \approx 14.52 \text{ €}^2$ .

#### Estensione Fase 2 (Modello Realistico Demand-Based):

Nelle estensioni future, il framework supporta un modello **logaritmico dipendente dalla domanda** che lega il valore del task alla densità di osservazioni GPS nella cella:

$$\nu(c) = \nu_{\min} + (\nu_{\max} - \nu_{\min}) \cdot \min \left( 1, \frac{\ln(1 + c)}{\kappa} \right)$$

dove:

- $c$  è il numero di osservazioni GPS nella cella durante la finestra temporale.
- $\kappa$  è un fattore di scala che modula la saturazione (calibrato empiricamente:  $\kappa = 10$ ).

Questo modello, basato su rendimenti marginali decrescenti, riflette l'ipotesi che aree già densamente monitorate abbiano valore incrementale ridotto, coerente con la teoria economica dei beni pubblici. La funzione è **concava** ( $\partial^2 \nu / \partial c^2 < 0$ ) e **monotona crescente** ( $\partial \nu / \partial c > 0$ ).

**Nota Implementativa:** Nella codebase, il parametro `value_mode` controlla quale modello usare:

- `value_mode='uniform'` → Fase 1 baseline (default)
- `value_mode='demand_log'` → Fase 2 estensione

#### 4.5. Generazione della Popolazione di Utenti ( $\mathcal{U}$ )

La popolazione di agenti (utenti) per ogni simulazione viene generata campionando le posizioni reali dei tassisti.

1. **Identificazione:** Per la medesima finestra temporale usata per generare i task, si identificano tutti i tassisti unici attivi.
2. **Posizionamento:** Per ogni tassista unico, si estrae la sua ultima posizione nota all'interno della finestra. Questa posizione diventa la posizione iniziale  $\text{pos}_i$  dell'agente  $U_i$ .
3. **Campionamento:** Se il numero di tassisti unici supera un massimo predefinito, viene applicata una strategia di **campionamento stratificato spaziale**. L'area geografica viene suddivisa in strati e da ciascuno viene estratto un numero di utenti proporzionale alla sua densità, garantendo che la distribuzione spaziale della popolazione sintetica sia rappresentativa di quella reale.
4. **Parametrizzazione:** A ogni utente  $U_i$  viene assegnato un costo operativo per chilometro  $\kappa_i$ , campionato da una distribuzione uniforme  $\kappa_i \sim U[\kappa_{\min}, \kappa_{\max}]$ . Questo modella l'eterogeneità intrinseca della flotta (e.g., veicoli diversi, stili di guida diversi).

##### 4.5.1. Assegnazione Task agli Utenti (Task Assignment)

Dopo la generazione delle popolazioni di task e utenti, è necessario definire quale sottoinsieme di task  $\Gamma_i \subseteq \mathcal{T}$  ciascun utente  $U_i$  è disposto a considerare per l'esecuzione. Questo processo modella le **preferenze geografiche** degli agenti.

**Motivazione del Modello di Prossimità:** Nel contesto taxi, un tassista razionale preferisce task vicini alla sua posizione iniziale, per minimizzare costi di trasferimento e massimizzare guadagni orari. Si assume che un utente consideri solo task entro un **raggio di servizio massimo**  $r_{\max}$  dalla sua posizione.

### Definizione 4.3 (Task Assignment via Radius)

Dato un utente  $U_i$  con posizione  $\text{pos}_i$  e un raggio massimo  $r_{\max}$  (espresso in metri), l'insieme di task candidati  $\Gamma_i$  è definito come:

$$\Gamma_i = \{\tau_j \in \mathcal{T} \mid d_H(\text{pos}_i, \text{pos}_j) \leq r_{\max}\}$$

dove  $d_H(\cdot, \cdot)$  è la distanza geodetica di Haversine.

#### Calibrazione del Radius:

Il parametro  $r_{\max}$  influenza il trade-off tra:

- **Copertura task:** Radius grande → più task accessibili, maggiore valore potenziale
- **Competizione:** Radius grande → sovrapposizione tra utenti, pagamenti ridotti
- **Efficienza operativa:** Radius piccolo → costi di trasferimento ridotti, ma possibile sottoutilizzo

Per gli esperimenti Fase 1, si testano 3 configurazioni:

Radius ( $r_{\max}$ )	Tempo Guida	Scenario
1500 m	~3 minuti	Conservativo (alta efficienza, bassa copertura)
2500 m	~5 minuti	<b>Bilanciato</b> (trade-off ottimale)
4000 m	~8 minuti	Aggressivo (alta copertura, bassa efficienza)

#### Nota sul Paper IMCU:

Il paper di riferimento (Yang et al., 2015) specifica solo che "each user selects a subset of tasks based on its preference," senza dettagliare il meccanismo di selezione. Il modello di prossimità geografica qui adottato è una scelta implementativa ragionevole per il contesto taxi, coerente con l'assunzione di razionalità economica (minimizzazione costi di spostamento).

#### Filtering Post-Assignment:

Dopo l'assegnazione, utenti con  $|\Gamma_i| = 0$  (nessun task nel radius) o con  $|\Gamma_i| < \text{threshold}_{\min}$  vengono esclusi dalla partecipazione all'asta, poiché non possono contribuire utilità positiva alla piattaforma.

## Capitolo 5: Metriche di Valutazione e Metodologia di Analisi

### 5.1. Introduzione

Per valutare le performance e validare le proprietà del meccanismo IMCU, è necessario definire un insieme di metriche quantitative e strumenti di visualizzazione. Questo capitolo illustra gli indicatori di performance chiave (KPI), le tecniche di analisi distributiva e le visualizzazioni spaziali che verranno impiegate per interpretare i risultati delle simulazioni.

## 5.2. Indicatori di Performance Chiave (KPI)

Le seguenti metriche verranno monitorate su base oraria per analizzare l'evoluzione dinamica del sistema.

- **Valore Totale della Piattaforma ( $v(S)$ ):** Misura il valore lordo generato, calcolato come  $\sum_{\tau_j \in \bigcup_{i \in S} \Gamma_i} \nu_j$ . Rappresenta il beneficio totale ottenuto dal completamento dei task da parte dei vincitori  $S$ .
- **Costo Totale dei Pagamenti ( $\sum_{i \in S} p_i$ ):** Rappresenta la spesa totale sostenuta dalla piattaforma per remunerare i vincitori.
- **Utilità Netta della Piattaforma ( $u_0$ ):** È l'indicatore primario di sostenibilità economica, definito come  $u_0 = v(S) - \sum_{i \in S} p_i$ .
- **Efficienza Economica ( $\eta$ ):** Una metrica normalizzata che misura la frazione del valore totale trattenuta come utilità, definita come  $\eta = u_0/v(S)$  per  $v(S) > 0$ .

## 5.3. Analisi della Distribuzione dei Pagamenti

Per comprendere l'equità e la concentrazione degli incentivi, si analizza la distribuzione statistica dei pagamenti  $p_i$  ai vincitori.

- **Analisi per Quartili:** Verranno utilizzati diagrammi a scatola e baffi (boxplot) per visualizzare la distribuzione dei pagamenti su base oraria, evidenziando mediana, quartili e la presenza di valori anomali (outlier).
- **Funzione di Distribuzione Cumulativa (CDF):** La CDF dei pagamenti,  $F(p) = P(P \leq p)$ , mostrerà la probabilità che un pagamento scelto a caso sia inferiore o uguale a un certo valore, fornendo una visione completa della distribuzione.

### 5.3.1. Analisi della Disuguaglianza: Curva di Lorenz e Indice di Gini

Per quantificare rigorosamente la disuguaglianza nella distribuzione dei pagamenti, si ricorre a strumenti classici della statistica economica.

**Definizione 5.1 (Curva di Lorenz)** La Curva di Lorenz  $L(x)$  rappresenta la frazione cumulativa del totale dei pagamenti ricevuta dalla frazione cumulativa  $x$  dei vincitori, quando questi sono ordinati per pagamento crescente. Una curva che si discosta significativamente dalla linea di perfetta uguaglianza ( $L(x) = x$ ) indica una forte disuguaglianza.

**Definizione 5.2 (Indice di Gini)** L'indice di Gini,  $G$ , è una misura sintetica della disuguaglianza derivata dalla Curva di Lorenz. È definito come il rapporto tra l'area compresa tra la linea di perfetta uguaglianza e la Curva di Lorenz ( $A$ ) e l'area totale sotto la linea di perfetta uguaglianza ( $A+B$ ).

$$G = \frac{A}{A+B} = 1 - 2 \int_0^1 L(x) dx$$



L'indice varia da  $G = 0$  (perfetta uguaglianza, tutti ricevono lo stesso pagamento) a  $G = 1$  (perfetta disuguaglianza, un singolo vincitore riceve il totale dei pagamenti).

## 5.4. Analisi Spaziale

Le visualizzazioni spaziali sono fondamentali per comprendere dove la domanda di sensing si concentra e con quale efficacia il meccanismo IMCU riesce a soddisfarla.

- **Mappe di Calore (Heatmap) della Domanda:** Verranno generate mappe di calore basate sulla densità di osservazioni del dataset per ogni finestra temporale, visualizzando geograficamente la distribuzione della domanda di task.
- **Mappe di Calore della Copertura:** Similmente, verranno generate mappe di calore che mostrano quali celle geografiche sono state effettivamente coperte dai vincitori selezionati dal meccanismo.
- **Analisi Comparativa e dei Gap:** Confrontando fianco a fianco le mappe di domanda e copertura, è possibile identificare visivamente le aree di "gap", ovvero zone ad alta domanda che ricevono una bassa copertura. Questa analisi è cruciale per valutare l'efficienza spaziale del meccanismo e per informare future strategie di allocazione dei task.

## Capitolo 6: Protocollo Sperimentale e Esecuzione delle Simulazioni

### 6.1. Panoramica del Protocollo

Questo capitolo descrive il protocollo sperimentale completo adottato per la Fase 1 del progetto. L'obiettivo è orchestrare una simulazione su larga scala del meccanismo d'asta IMCU, utilizzando i dati reali della flotta di taxi a Roma come base per la generazione di un ambiente di sensing realistico. Il protocollo è progettato per garantire **riproducibilità**, **tracciabilità** e **robustezza** scientifica, definendo una pipeline sequenziale che va dalla configurazione dei parametri all'archiviazione dei risultati finali.

La simulazione analizza una giornata specifica scomponendola in una sequenza di aste orarie indipendenti. I risultati di ciascuna asta vengono registrati e successivamente aggregati per consentire un'analisi sia a livello micro (orario) sia a livello macro (giornaliero).

### 6.2. Parametrizzazione degli Esperimenti

Ogni esecuzione sperimentale è definita da un insieme di parametri che ne governano il comportamento. La loro esplicitazione è fondamentale per la riproducibilità.

- **Parametri Temporal:**
  - $T_{\text{day}}$ : Il giorno specifico della simulazione (formato "AAAA-MM-GG").
  - $T_{\text{start}}, T_{\text{end}}$ : L'intervallo orario della simulazione,  $[T_{\text{start}}, T_{\text{end}})$ .
- **Parametri Spaziali:**

- $d_{\text{cell}}$ : La dimensione (in metri) del lato di una cella della griglia geografica utilizzata per la discretizzazione dello spazio.
  - $R_{\text{task}}$ : Il **raggio massimo di assegnazione** (in metri). Un parametro cruciale che definisce la distanza massima entro la quale un task può essere considerato rilevante per un utente. Se non specificato, si assume un raggio infinito.
- **Parametri del Meccanismo d'Asta:**
    - $N_{\text{max\_users}}$ : Il numero massimo di utenti (tassisti) da campionare e includere in ogni asta oraria.
    - $[\kappa_{\text{min}}, \kappa_{\text{max}}]$ : L'intervallo di valori per il campionamento del costo operativo per chilometro  $\kappa_i$ . Per la Fase 1, questo è calibrato empiricamente (vedi Sezione 2.4.1) a  $[0.45, 0.70]$  €/km.
    - $M_{\text{value}}$ : Il modello utilizzato per la generazione del valore  $\nu_j$  dei task (per la Fase 1 è 'uniform', come definito nella Sezione 4.4).
  - **Parametri di Riproducibilità:**
    - $S_{\text{seed}}$ : Il seme (seed) intero per l'inizializzazione del generatore di numeri casuali (RNG), che garantisce il determinismo nel campionamento degli utenti e nell'assegnazione dei parametri stocastici.

### 6.3. Fasi di Esecuzione della Simulazione

L'esperimento si articola in una pipeline automatizzata che esegue le seguenti fasi in sequenza.

#### Fase 1: Inizializzazione e Setup dell'Ambiente

Prima dell'avvio della simulazione, l'ambiente viene configurato per garantire la tracciabilità. Viene generato un identificatore univoco per l'esperimento e inizializzato un sistema di logging strutturato che archivia tutti gli eventi, gli avvisi e gli errori su file dedicati. Il generatore di numeri casuali viene inizializzato con il seme  $S_{\text{seed}}$  per assicurare che ogni esecuzione, a parità di parametri, produca risultati identici.

#### Fase 2: Esecuzione dell'asta

Il cuore del protocollo è un ciclo che itera su ogni ora della finestra temporale definita. Per ciascuna ora  $h$ , viene eseguita un'asta IMCU completa e indipendente, seguendo i passi sottoelencati.

##### 1. Istanziamento dell'Ambiente di Sensing:

- **Generazione dei Task ( $\mathcal{T}_h$ ):** Utilizzando la metodologia descritta nella Sezione 4.4, viene generato l'insieme di task  $\mathcal{T}_h$  basandosi sulla densità di osservazioni del dataset per l'ora  $h$ .

- **Generazione degli Utenti ( $\mathcal{U}_h$ ):** Similmente, viene generata la popolazione di utenti  $\mathcal{U}_h$  campionando le posizioni dei tassisti attivi nell'ora  $h$ , come descritto nella Sezione 4.5.

2. **Assegnazione Task-Utente (Filtraggio Geografico):** A differenza di un modello teorico puro in cui ogni utente potrebbe fare un'offerta per qualsiasi task, si introduce un vincolo di prossimità geografica, modellato dal raggio  $R_{\text{task}}$ . Per ogni utente  $U_i \in \mathcal{U}_h$ , l'insieme di task  $\Gamma_i$  per cui può sottoporre un'offerta è limitato a quelli all'interno del suo raggio. **Definizione 6.1 (Insieme di Task Ammissibili)** Per un utente  $U_i$  posizionato in  $\text{pos}_i$ , l'insieme dei task ammissibili  $\Gamma_i$  è un sottoinsieme di  $\mathcal{T}_h$  tale che:

$$\Gamma_i = \{\tau_j \in \mathcal{T}_h \mid d(\text{pos}_i, \text{pos}_j) \leq R_{\text{task}}\}$$

dove  $d(\cdot, \cdot)$  è la distanza geodetica. Questo vincolo modella realisticamente la disponibilità di un utente a deviare dal proprio percorso.

3. **Calcolo dei Costi e delle Offerte Veritiere:** Per ogni utente  $U_i$  con un insieme non vuoto di task ammissibili  $\Gamma_i \neq \emptyset$ , viene calcolato il costo privato  $c_i(\Gamma_i)$  secondo il modello definito nella Sezione 2.4.1. Coerentemente con l'assunzione di razionalità perfetta della Fase 1 e con il Teorema 2.1, l'offerta sottomessa dall'agente viene impostata come perfettamente veritiere:  $b_i = c_i(\Gamma_i)$ .
4. **Esecuzione del Meccanismo d'Asta IMCU:** L'insieme di coppie  $(\Gamma_i, b_i)$  per tutti gli utenti  $U_i$  con  $\Gamma_i \neq \emptyset$  costituisce l'input per il meccanismo d'asta IMCU, come descritto nel Capitolo 3. L'algoritmo viene eseguito per:
- Determinare l'insieme dei vincitori  $S_h \subseteq \mathcal{U}_h$ .
  - Calcolare il pagamento critico  $p_i$  per ogni vincitore  $i \in S_h$ .
5. **Raccolta e Archiviazione dei Risultati Orari:** Tutti gli output dell'asta per l'ora  $h$  vengono registrati: l'insieme dei vincitori, i pagamenti, e tutti i KPI e le metriche diagnostiche definite nel Capitolo 5 (e.g.,  $v(S_h)$ ,  $\sum p_i$ ,  $u_{0,h}$ ).

### Fase 3: Aggregazione dei Risultati e Analisi

Al termine del ciclo orario, i risultati raccolti vengono aggregati per consentire un'analisi a un livello di granularità superiore (per blocchi di ore e per l'intera giornata).

1. **Consolidamento dei KPI:** Le serie temporali dei KPI orari vengono aggregate per calcolare le somme totali e le medie giornaliere.
2. **Aggregazione Spaziale:** I dati sulla densità della domanda e sulla copertura effettiva dei task vengono sommati su tutte le ore per generare mappe di calore aggregate. Questo permette di effettuare l'analisi comparativa e dei gap (Sezione 5.4) sull'intera giornata.
3. **Analisi Distributiva:** Tutti i pagamenti effettuati durante la giornata vengono raccolti in un unico insieme per calcolare la CDF, la Curva di Lorenz e l'indice di Gini

complessivi (Sezione 5.3).

#### Fase 4: Visualizzazione e Reporting

L'ultima fase del protocollo è dedicata alla presentazione dei risultati.

1. **Generazione delle Figure:** Utilizzando le metodologie di visualizzazione definite nel Capitolo 5, vengono generate tutte le figure scientifiche (serie temporali, grafici di distribuzione), formattate secondo standard di pubblicazione.
2. **Creazione di Report Sintetici:** Vengono prodotti report testuali e file CSV che riassumono i principali risultati aggregati, i KPI e le statistiche distributive, pronti per essere inclusi nell'analisi della tesi.
3. **Archiviazione dei Metadati:** Tutte le informazioni relative all'esperimento, inclusi i parametri di configurazione, gli hash dei dati di input, i timestamp di esecuzione e le versioni del software, vengono salvati in un file di metadati. Questo garantisce la completa tracciabilità e la potenziale riproducibilità dell'intero esperimento da parte di terzi.

### Capitolo 7: Risultati Sperimentali e Analisi Comparativa

#### 7.1. Introduzione

Questo capitolo presenta i risultati ottenuti dall'esecuzione del protocollo sperimentale descritto nel Capitolo 6. L'obiettivo è duplice: in primo luogo, analizzare in dettaglio le performance del meccanismo IMCU in uno scenario di riferimento (baseline); in secondo luogo, condurre un'analisi di sensitività su un parametro chiave—il raggio di assegnazione dei task  $R_{\text{task}}$ —per comprendere il suo impatto sull'efficienza e sulla dinamica del sistema.

#### 7.2. Analisi dello Scenario di Riferimento (Baseline)

Si definisce come scenario di riferimento (o baseline) la simulazione condotta sulla giornata del 2014-02-01 (ore 8-19) con i parametri calibrati e ritenuti più rappresentativi, tra cui un raggio di assegnazione  $R_{\text{task}} = 2500$  metri.

Prima di analizzare le performance economiche, è cruciale verificare che l'implementazione rispetti le **proprietà formali** garantite dal paper IMCU (Yang et al., 2015).

#### Risultati Test Automatizzati (12 ore, 175 vincitori totali):

Proprietà	Test Eseguiti	Violazioni	Status
Individual Rationality (IR)	175 vincitori	0	Verificata
Profitability (P)	12 ore	0 (min $u_0 = 1,359.55$ €)	Verificata

<b>Truthfulness (T)</b>	17,500 campioni (100×175)	0	<b>Verificata</b>
<b>Monotonicity (M)</b>	12 ore	0	<b>Verificata</b>
<b>Critical Payment (C)</b>	175 vincitori	0	<b>Verificata</b>
<b>Submodularity (S)</b>	120,000 tentativi (10.000×12)	0	<b>Verificata</b>

**Conclusione:** L'implementazione è **formalmente corretta** e replicabile come baseline teorico puro. Tutte le proprietà del paper sono rispettate al 100% in ogni singola asta oraria, garantendo la piena validità scientifica dei risultati.

**Nota Tecnica:** I test sono eseguiti automaticamente dal modulo `imcu.py` (funzione `_check_properties()`) e loggati per ogni iterazione oraria. I log di esecuzione (es. [INF0] Proprietà: IR=True, Profit=True, Monot=True, Crit=True, Truth=True, SubmViol=0) confermano zero violazioni.

#### 7.2.1. Andamento Temporale degli Indicatori di Performance (KPI)

L'analisi dell'evoluzione oraria dei KPI (vedi *Figura 2014-02-01\_kpi\_timeseries.png*) rivela i cicli di attività tipici di un contesto urbano. Si osserva un chiaro picco di attività alle 13:00 (ora di pranzo) sia nel **valore totale della piattaforma** ( $v(S)$ ) che nell'**utilità della piattaforma** ( $u_0$ ). Questo conferma che il modello di generazione dei task cattura efficacemente la domanda reale.

L'utilità  $u_0$  segue fedelmente l'andamento di  $v(S)$ , mantenendo un'efficienza (come si vedrà nella Sezione 7.3) notevolmente stabile, con un minimo di  $\eta = 0.5267$  e un massimo di  $\eta = 0.6824$ . Il sistema non solo è sempre profittevole ( $\min u_0 = 1,359.55$  €), ma i suoi margini sono robusti e prevedibili.

#### 7.2.2. Analisi della Distribuzione dei Pagamenti

La distribuzione dei pagamenti ai 175 vincitori della giornata di riferimento mostra una **leggera asimmetria destra** (skew positivo), come indicato dalla **media superiore alla mediana** (78.70 € vs 72.94 €). Questo pattern è tipico delle distribuzioni di reddito e suggerisce che, sebbene la maggior parte degli utenti riceva pagamenti consistenti, esiste una "coda destra" di pagamenti più elevati per utenti che hanno completato task di maggior valore o in aree a più alta competizione.

#### Distribuzione dettagliata (Baseline 2500m):

Statistica	Valore	Interpretazione
Pagamento minimo	3.93 €	Vincitore con costi molto bassi (task vicini)

Pagamento massimo	222.19 €	Vincitore con molti task o alta competizione locale
<b>Pagamento mediano</b>	<b>72.94 €</b>	50% utenti riceve $\geq 72.94$ €
Pagamento medio	78.70 €	Media aritmetica, influenzata dalla coda destra
Deviazione standard	47.50 €	Variabilità moderata (CV = 60.4%)
Rapporto Max/Mediana	3.05×	Presenza di outlier moderati

### Analisi Asimmetria:

Il fatto che **Media > Mediana** ( $78.70 > 72.94$ , differenza di +7.9%) indica:

1. La distribuzione ha una **leggera coda destra** (valori alti che alzano la media).
2. La **massa centrale** della distribuzione è concentrata attorno ai 73 €.
3. Il rapporto Max/Mediana di 3.05× conferma che esistono pagamenti significativamente alti, ma non in modo estremo.

### Indice di Gini: **G = 0.3304**

L'**indice di Gini**, calcolato su tutti i 175 pagamenti, si attesta a **G = 0.3304**, come visualizzato nella *Figura 2014-02-01\_payments\_lorenz.png*. Questo valore indica una distribuzione dei pagamenti **notevolmente equa**.

1. **Il meccanismo IMCU distribuisce i pagamenti in modo equo:** Un valore di Gini vicino a 0.33 è considerato basso e indica che una piccola frazione di utenti non sta monopolizzando il budget.
2. **Presenza di eterogeneità controllata:** La leggera asimmetria destra è bilanciata, e il meccanismo di pagamento critico previene che pochi utenti catturino un surplus sproporzionato.
3. **Equità confermata:** La Curva di Lorenz si discosta moderatamente dalla linea di perfetta uguaglianza, confermando visivamente l'assenza di una forte concentrazione di ricchezza.

### Confronto con letteratura MCS:

Il valore Gini osservato è **inferiore** a quello riportato in altri studi empirici su piattaforme crowdsensing reali (es. Waze, OpenSignal), dove si osservano Gini nel range 0.4-0.6 a causa di effetti network e winner-takes-all. Questo conferma la **proprietà di fairness** del meccanismo IMCU rispetto a meccanismi di mercato non regolati.

#### 7.2.3. Analisi Spaziale: Domanda vs. Copertura

(Questa sezione è qualitativa e rimane valida) Il confronto tra la mappa di calore della densità delle osservazioni (proxy della domanda) e quella della copertura effettiva dei task (luogo dei task completati dai vincitori) evidenzia l'efficacia spaziale del meccanismo. Si

nota una forte correlazione positiva: le aree a più alta domanda (centro storico, stazioni, aeroporti) sono anche quelle con la maggiore copertura. Tuttavia, emergono anche dei **gap di copertura**: aree periferiche con una domanda non trascurabile che non vengono servite, probabilmente a causa di costi di deviazione troppo elevati per i tassisti di passaggio.

7.3. Analisi di Sensitività sul Raggio di Assegnazione ( $R_{task}$ )

Per investigare l'impatto del vincolo di prossimità geografica, sono state eseguite tre campagne di simulazione complete, variando unicamente il parametro  $R_{task}$  sui valori {1500m, 2500m, 4000m}.

7.3.1. Effetto sui KPI Aggregati

L'analisi comparativa dei risultati aggregati giornalieri rivela **trade-off chiari** tra le diverse metriche di performance, come mostrato nei grafici *radius\_comparison\_aggregates.png* e *radius\_comparison\_kpi\_timeseries.png*.

Tabella 7.1: Confronto KPI Aggregati per Configurazione Radius (Dati Validati)

Metrica	1500m	2500m	4000m	Trend Osservato
Valore piattaforma $v(S)$	27,374 €	34,253 €	<b>38,638</b> €	<b>Monotono crescente</b>
Pagamenti totali $\Sigma p_i$	12,312 €	13,772 €	<b>15,029</b> €	Monotono crescente
Utilità piattaforma $u_0$	15,062 €	20,481 €	<b>23,609</b> €	<b>Monotono crescente</b>
Efficienza $\eta$ (Media Oraria)	54.8%	59.6%	<b>61.4%</b>	<b>Monotono crescente</b>
Vincitori totali	<b>280</b>	175	100	Decrescente
Stabilità (CV Efficienza)	0.098	0.078	<b>0.076</b>	<b>Monotono crescente (CV decresce)</b>
Gini	<b>0.2792</b>	0.3304	0.3679	Crescente (disuguaglianza aumenta)

**Nota:** Grassetto indica valori massimi (o migliori, come per Gini e CV).

Analisi Dettagliata per Metrica:

- 1. Valore ( $v(S)$ ), Utilità ( $u_0$ ) ed Efficienza ( $\eta$ ): Crescono Monotonicamente

Si osserva una **correlazione positiva forte e monotona** tra il raggio di assegnazione e tutte le metriche di profitto della piattaforma:

- Aumentando il raggio, gli utenti hanno accesso a un insieme  $\Gamma_i$  più ricco di task.
- Questo permette al meccanismo greedy (Algoritmo 1) di trovare "vincitori" con un guadagno marginale  $\text{gain}(i, S_k)$  significativamente più alto.
- **1500m  $\rightarrow$  4000m**: Il valore totale generato ( $v(S)$ ) cresce del **+41%** e l'utilità netta ( $u_0$ ) cresce del **+57%**.

#### Interpretazione:

- A **1500m**, il radius è **troppo restrittivo**. Gli utenti sono "miopi" e non possono raggruppare task in modo efficiente. Il sistema è sub-ottimale e lascia molto valore "sul tavolo".
- Aumentando il raggio, si aumenta la liquidità del mercato e la capacità combinatoria dell'asta, portando a selezioni globalmente migliori.
- L'efficienza media oraria ( $\eta$ ) cresce dal 54.8% al 61.4%, e la sua stabilità (misurata dal Coefficiente di Variazione CV) **migliora**, passando da 0.098 a 0.076. Un raggio più ampio rende il sistema non solo più profittevole, ma anche più stabile.

---

## 2. Numero di Vincitori e Disuguaglianza (Gini): Il Trade-off Sociale

L'aumento dell'efficienza economica ha un costo sociale misurabile:

- **Numero di Vincitori**: Crolla drasticamente da 280 (a 1500m) a soli 100 (a 4000m).
- **Indice di Gini**: Aumenta da 0.2792 (molto equo) a 0.3679 (disuguaglianza moderata).

#### Interpretazione:

- Con un raggio ampio (4000m), pochi utenti "super-efficienti" (con basso costo  $\kappa_i$  e posizione centrale) possono costruire insiemi  $\Gamma_i$  molto ampi e profittevoli.
- Questi utenti dominano l'asta, "rubando" task ai competitor più piccoli o periferici.
- Il risultato è un sistema **più efficiente dal punto di vista economico** (massimizza  $u_0$ ), ma **meno equo** (Gini più alto) e **meno partecipativo** (meno vincitori).

---

### Conclusione Complessiva:

Non esiste una configurazione di  $r_{\max}$  **universalmente ottimale**. La scelta dipende dall'obiettivo della piattaforma, rivelando un trade-off fondamentale tra efficienza



economica e impatto sociale:

Obiettivo	Radius Ottimale	Motivazione
Massimizzare Profitto e Stabilità	4000m	Max $u_0$ (23,6k €), Max $\eta$ (61.4%), Max Stabilità (CV=0.076)
Massimizzare Partecipazione ed Equità	1500m	Max Vincitori (280), Min Gini (0.279)
Bilanciare le Metriche	2500m	Trade-off di riferimento (Baseline)

Per la Fase 1 baseline, si conferma  $r_{\max} = 2500 \text{ m}$  come configurazione di riferimento. Sebbene 4000m sia economicamente superiore, 2500m rappresenta un **punto di equilibrio scientificamente valido** che:

1. Genera un'eccellente efficienza (59.6%).
2. Mantiene un'ottima equità (Gini = 0.33).
3. Garantisce una partecipazione ampia (175 vincitori) senza la frammentazione eccessiva vista a 1500m.

7.3.2. Effetto sulla Disuguaglianza dei Pagamenti

L'indice di Gini mostra una correlazione positiva con  $R_{\text{task}}$ . Aumentando il raggio, la competizione per i task più preziosi si intensifica, e gli utenti con costi marginali più bassi (basso  $\kappa_i$ ) riescono a capitalizzare maggiormente la loro posizione dominante, vincendo più aste e accentuando la disuguaglianza nella distribuzione dei pagamenti.

Capitolo 8: Analisi Statistica Avanzata e Discussione

8.1. Introduzione

Oltre all'analisi descrittiva, sono state applicate tecniche di statistica inferenziale e di apprendimento non supervisionato per scoprire relazioni strutturali e pattern latenti nei dati generati dalla simulazione baseline ( $R_{\text{task}} = 2500\text{m}$ ).

8.2. Analisi Correlazionale

Per investigare le relazioni tra le variabili di sistema, sono stati calcolati i coefficienti di correlazione di Pearson (per relazioni lineari) e di Spearman (per relazioni monotone), basandosi sui dati orari della simulazione baseline.

Risultati dell'Analisi di Correlazione (Spearman):

- Correlazione vincitori ↔ efficienza:  $\rho = +0.213$  ( $p = 0.5067$ )
- Correlazione  $v(S)$  ↔ efficienza:  $\rho = -0.119$  ( $p = 0.7129$ )
- Correlazione  $\Sigma p_i$  ↔ efficienza:  $\rho = -0.510$  ( $p = 0.0899$ )

**Interpretazione:** Il risultato più significativo di questa analisi è la **mancanza di correlazioni statisticamente significative** (tutti i p-value sono  $> 0.05$ ). L'efficienza del sistema ( $\eta$ ) non è una semplice funzione monotona né del valore totale generato ( $v(S)$ ) né del numero di vincitori.

Questo è un risultato cruciale: dimostra che l'efficienza è una **proprietà emergente e complessa** del meccanismo d'asta. Non è sufficiente avere molti vincitori o un alto valore lordo per garantire un'alta efficienza netta. Questa conclusione è ulteriormente rafforzata da un'analisi di regressione lineare esplorativa (che modella  $\eta$  in funzione di  $v(S)$  e *vincitori*), la quale produce un punteggio  $R^2 = 0.0987$ , indicando che un modello lineare non è in grado di spiegare la variabilità dell'efficienza.

**8.3. Identificazione di Regimi Operativi tramite Clustering**

Per determinare se le 12 ore della giornata di simulazione (dalle 8:00 alle 19:00) possano essere raggruppate in "regimi operativi" omogenei, è stato applicato l'algoritmo di clustering **K-Means**.

**Metodologia:** Le ore sono state rappresentate come vettori nello spazio delle feature  $\{v(S), \eta, \text{numero di vincitori}\}$ . Per garantire che nessuna variabile dominasse il calcolo della distanza euclidea a causa della sua scala, le feature sono state preventivamente **standardizzate** tramite trasformazione Z-score:

$$Z = \frac{X - \mu}{\sigma}$$

dove  $\mu$  è la media e  $\sigma$  la deviazione standard della feature  $X$ .

**Risultati:** Il Metodo Elbow (*Figura advanced\_analysis\_elbow\_method.png*) ha confermato che  $k = 3$  è il numero ottimale di cluster. L'algoritmo ha identificato tre cluster distinti e interpretabili, come mostrato in *Figura advanced\_analysis\_summary.jpg*.

**Tabella 8.1: Caratteristiche Centroids dei 3 Cluster Identificati (Dati Validati)**

Cluster	Ore Tipiche	$v(S)$ Medio	$\eta$ Media	N_winners Medio	Interpretazione
0 - Efficienza Standard	10, 11, 15, 16, 17	2,828 €	55.3%	14.2	Alta domanda, efficienza base
1 - Picco di Valore	13, 14	3,817 €	64.0%	18.0	Picco di valore e efficienza
2 - Alta Efficienza	8, 9, 12, 18, 19	2,496 €	62.1%	13.6	Domanda moderata, alta efficienza

**Osservazioni:** L'analisi dei cluster rivela una dinamica non banale:

- **Cluster 1 (Picco di Valore):** Corrisponde alle ore di picco (13:00, 14:00), genera il massimo valore  $v(S)$ , impiega il maggior numero di vincitori E ottiene l'efficienza  $\eta$  più alta (64.0%). Questo contrasta con l'ipotesi che una competizione intensa possa erodere i margini; in questo caso, un mercato "liquido" permette all'asta di trovare soluzioni eccellenti.
- **Cluster 2 (Alta Efficienza):** Rappresenta le ore di "spalla" (mattina presto, sera presto, ora di pranzo 12:00). Ha il  $v(S)$  più basso ma un'efficienza  $\eta$  quasi pari a quella del picco (62.1%).
- **Cluster 0 (Efficienza Standard):** Raggruppa le ore centrali del mattino e del pomeriggio. Nonostante un  $v(S)$  robusto (2.828 €), questo cluster mostra l'efficienza  $\eta$  più bassa (55.3%).

**Implicazione:** L'efficienza del sistema non è inversamente correlata alla domanda. Il sistema IMCU performa al meglio (sia in termini di valore lordo  $v(S)$  che di efficienza netta  $\eta$ ) durante le ore di picco assoluto (Cluster 1).

#### 8.4. Discussione Complessiva

I risultati della Fase 1 confermano che il meccanismo IMCU, quando applicato a un modello di dati semi-realistico e calibrato, si comporta in modo coerente con le sue proprietà teoriche (IR, Profitability, Truthfulness). L'analisi ha tuttavia rivelato complessità emergenti:

- Esiste un **trade-off fondamentale tra efficienza economica ed equità sociale** nel parametro  $R_{task}$  (Capitolo 7). Un raggio ampio (4000m) massimizza il profitto ( $\eta = 61.4\%$ ) ma minimizza la partecipazione (100 vincitori), mentre un raggio ristretto (1500m) massimizza la partecipazione (280 vincitori) e l'equità (Gini=0.279) a scapito del profitto ( $\eta = 54.8\%$ ).
- Il sistema esibisce **regimi operativi eterogenei** (Cluster K-Means) durante la giornata. Contrariamente a un'intuizione semplicistica, le ore di picco della domanda (Cluster 1) sono risultate essere anche le più efficienti.

#### Conclusioni e Lavori Futuri

La Fase 1 di questo progetto ha raggiunto con successo il suo obiettivo: la progettazione, implementazione e validazione empirica di un sistema MCS basato sui principi teorici del meccanismo d'asta IMCU. È stato sviluppato un protocollo sperimentale completo che, partendo da dati grezzi, ha permesso di simulare il comportamento del sistema in un contesto realistico, verificando la coerenza delle sue proprietà fondamentali come la **Razionalità Individuale** e la **Profittabilità**.

Le analisi quantitative, incluse quelle di sensitività e quelle statistiche avanzate, hanno fornito intuizioni cruciali sul funzionamento del meccanismo. In particolare, è stato dimostrato il trade-off tra efficienza economica ed equità sociale legato al raggio  $R_{task}$ , e sono stati identificati distinti regimi operativi (cluster) durante la giornata.

Il principale limite di questa fase risiede nella sua assunzione fondante: la **razionalità perfetta** degli agenti. Concretamente, questo si traduce nell'assunzione che:

1. **Perfetta conoscenza dei costi:** Ogni utente conosce esattamente il proprio costo privato  $c_i(\Gamma_i)$ , senza errori di stima o incertezza.
2. **Strategie dominanti:** Ogni utente riconosce che  $b_i = c_i$  è strategia dominante e la adotta sistematicamente, senza deviazioni euristiche.
3. **Nessun comportamento strategico complesso:** Gli utenti non colludono, non tentano underbidding speculativo, né adottano strategie adattive basate su apprendimento.

Sebbene questa assunzione sia **indispensabile per creare una baseline teorica controllata**, essa si discosta significativamente dal comportamento umano reale osservato in sistemi MCS operativi (Guo et al., 2018; Zhang et al., 2020). Studi empirici mostrano che:

- **30-40% degli utenti** sottomettono bid non-veritieri per massimizzare guadagni percepiti (anche quando subottimale).
- **Bias cognitivi** (anchoring, loss aversion) influenzano formazione delle offerte.
- **Apprendimento imperfetto:** Utenti novizi impiegano 5-10 iterazioni prima di convergere a strategie near-ottimali.

Pertanto, la **Fase 2** del progetto si concentrerà sul superamento di questo limite. Verranno introdotti modelli di agenti con **razionalità limitata (bounded rationality)**, capaci di adottare strategie di offerta sub-ottimali, euristiche o opportunistiche (e.g., offerte non veritiere). L'obiettivo sarà investigare la robustezza del meccanismo IMCU in presenza di tali comportamenti e, successivamente, progettare nella **Fase 3** meccanismi adattivi in grado di apprendere e contrastare strategie non veritiere, avvicinando ulteriormente il modello alla complessità di un'applicazione reale.