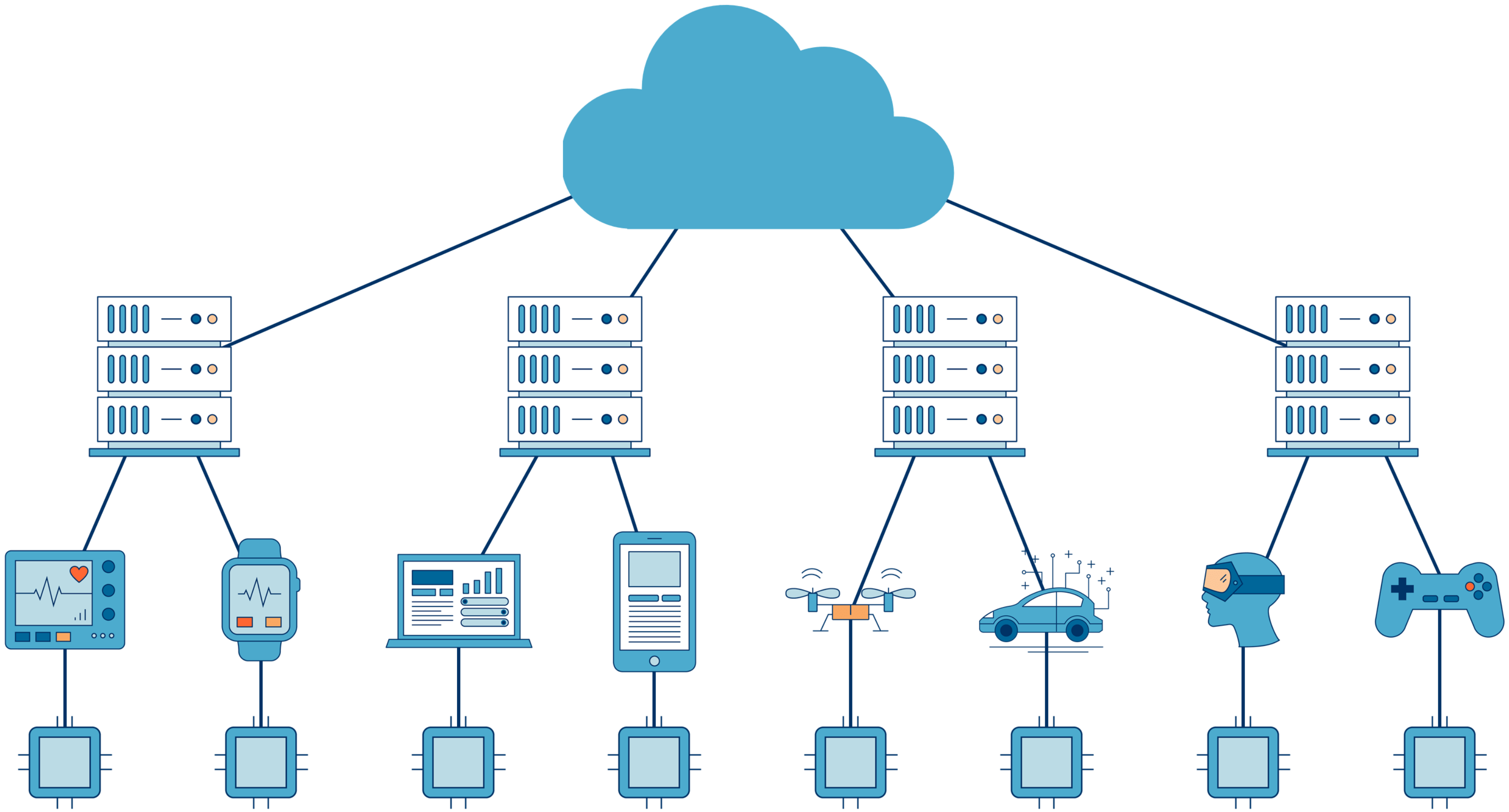
Modellazione e analisi di Task Scheduling in ambiente Green Edge Computing

Leonardo Menti

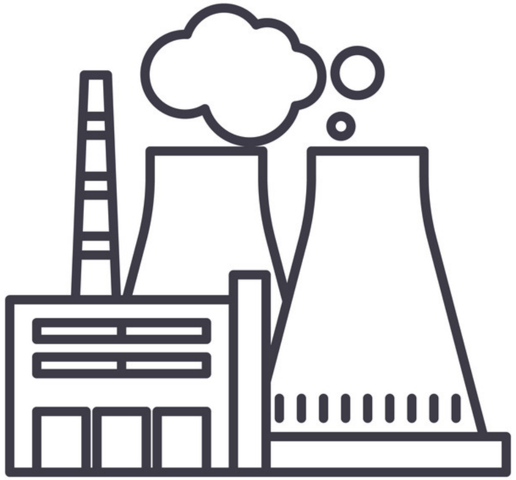
Relatore: Christian Quadri

Edge Computing



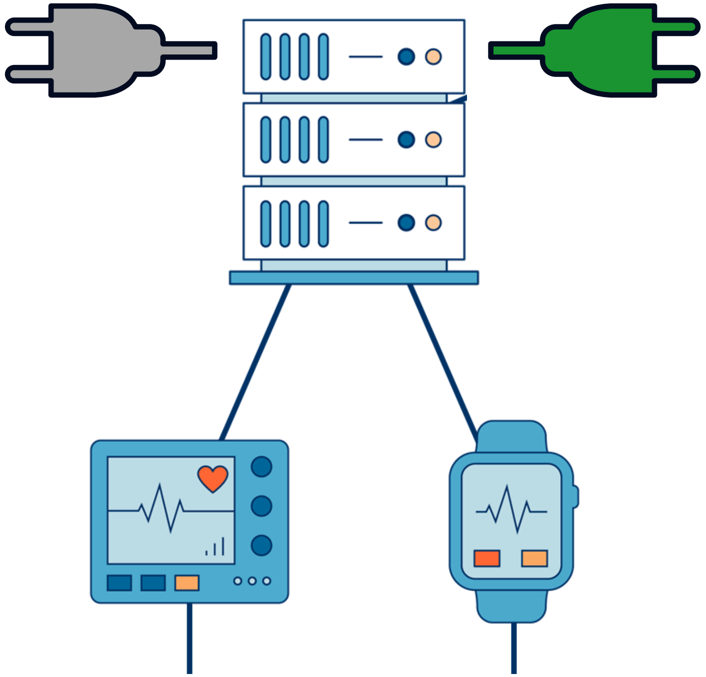
- Risorse di elaborazione vicino ai dispositivi
- Riduzione latenza e traffico di rete
- Garanzia di sicurezza e scalabilità
- Risorse limitate

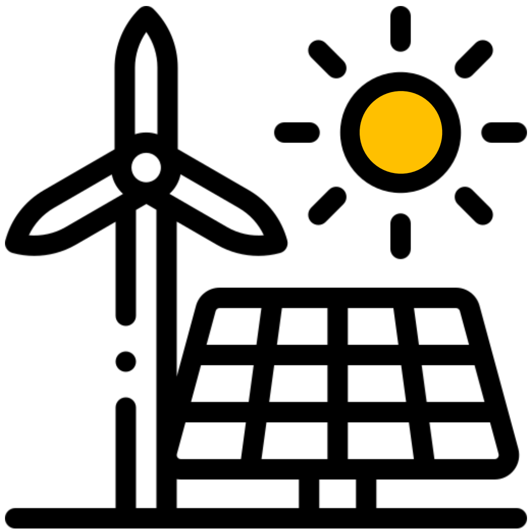
Green Edge Computing



Energia brown

$ $ $





Energia green

$

Sfide del green edge computing

- Distribuzione e utilizzo delle risorse computazionali limitate

Sfide del green edge computing

- Distribuzione e utilizzo delle risorse computazionali limitate

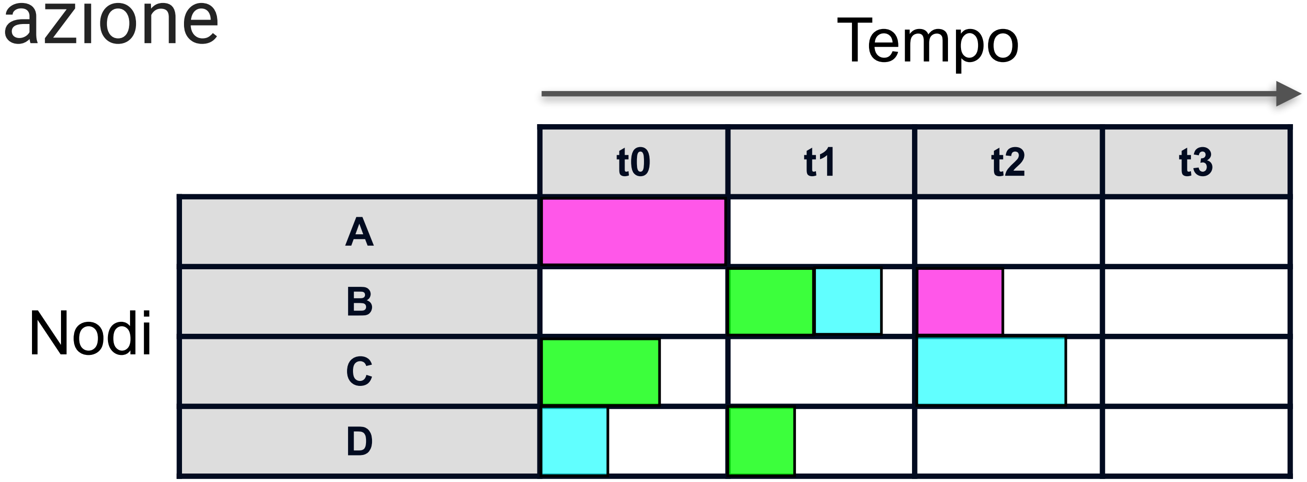
- Ottimizzazione del consumo energetico

Sfide del green edge computing

- Distribuzione e utilizzo delle risorse computazionali limitate
- Ottimizzazione del consumo energetico
- Massimizzazione del guadagno da parte del provider di servizi edge

Task Scheduling

- Computazione di task generati dai dispositivi mobili / IoT
- Frammentazione del task su più nodi
- Distribuzione geografica e temporale
- Costo energetico di computazione
- Deadline di completamento
- Reward



Obiettivi

- Formalizzare e implementare un modello di PLI per la risoluzione ottima del problema di task scheduling in ambiente green edge

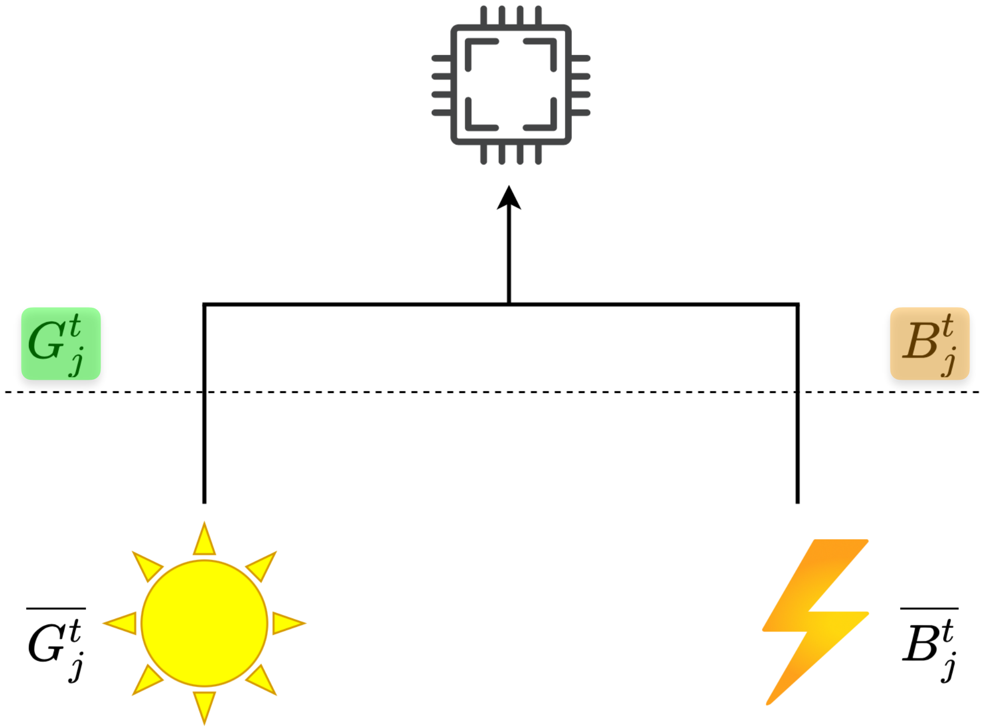
Obiettivi

- Formalizzare e implementare un modello di PLI per la risoluzione ottima del problema di task scheduling in ambiente green edge
- Progettare un algoritmo online per affrontare il problema

Obiettivi

- Formalizzare e implementare un modello di PLI per la risoluzione ottima del problema di task scheduling in ambiente green edge
- Progettare un algoritmo online per affrontare il problema
- Analisi e confronto delle performance

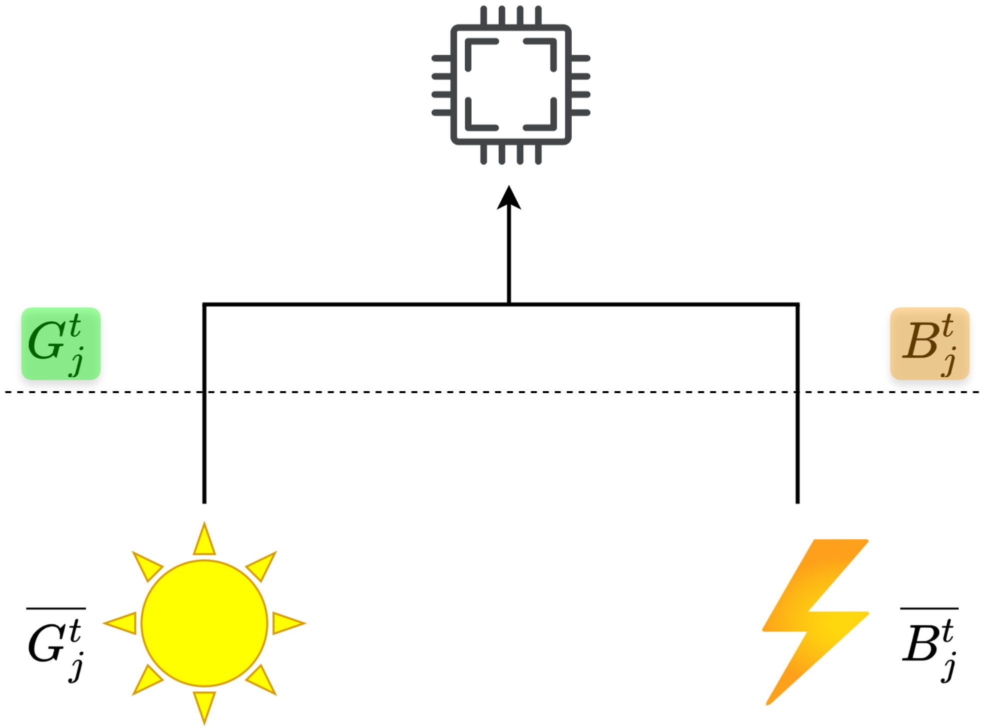
Modellazione standard



Ricavo Costo esecuzione

$$\max \sum_{i\in N} r_i \cdot Z_i - \sum_{t\in T}\sum_{j\in F}\left(w^t_{gj}\cdot G^t_j + w^t_{bj}\cdot B^t_j\right)$$

Modellazione standard

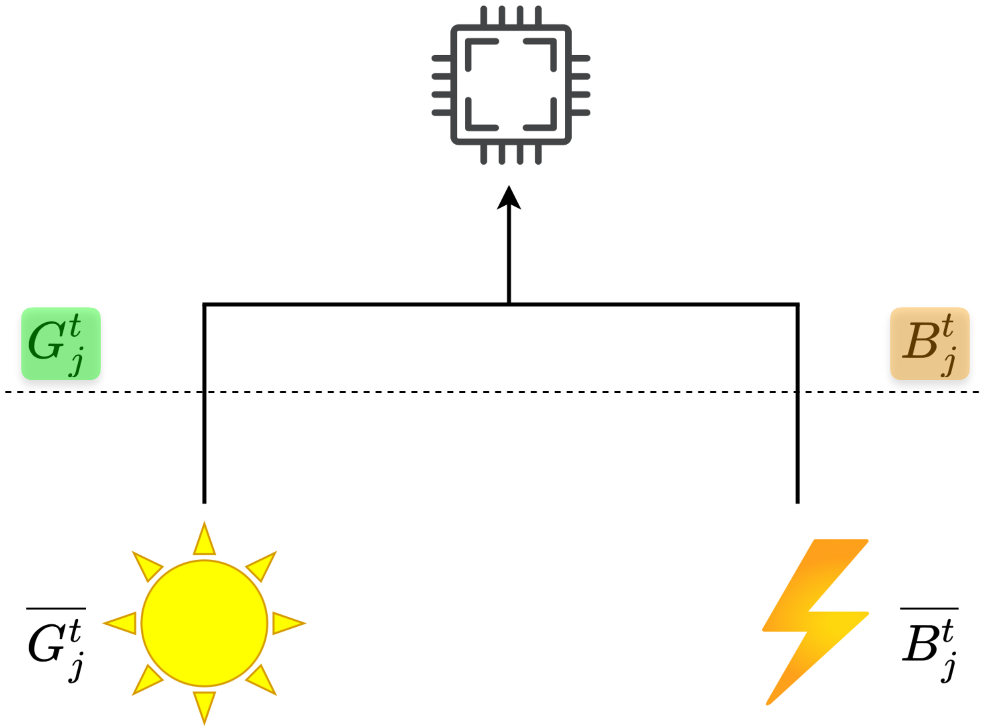


$$\max \sum_{i \in N} r_i \cdot Z_i - \sum_{t \in T} \sum_{j \in F} (w_{gj}^t \cdot G_j^t + w_{bj}^t \cdot B_j^t)$$

$$\text{s.t.} \sum_{j \in F} \sum_{t \geq a_i + \delta_{kj}}^{t \leq a_i + \tau_i - \delta'_{jk}} X_{ij}^t = Z_i \qquad \forall i \in N$$

- Accettazione task
- Processamento task
- Deadline task
- Vincoli topologici

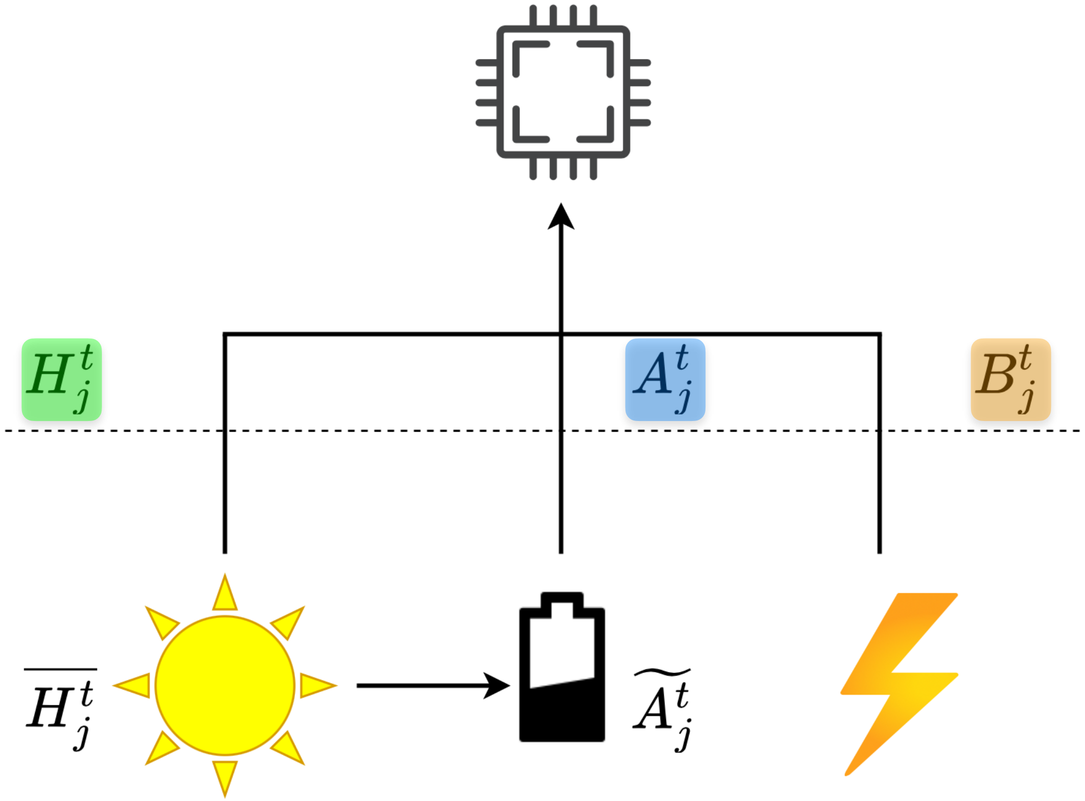
Modellazione standard



$$
\begin{aligned}
\max \quad & \sum_{i\in N} r_i \cdot Z_i - \sum_{t\in T}\sum_{j\in F}(w_{gj}^t \cdot G_j^t + w_{bj}^t \cdot B_j^t) & \\
\text{s.t.} \quad & \sum_{j\in F} \sum_{t\geq a_i+\delta_{kj}}^{t\leq a_i+\tau_i-\delta'_{jk}} X_{ij}^t = Z_i & \forall i \in N \\
& \sum_{i\in N} \mu_i X_{ij}^t d_i \leq G_j^t + B_j^t & \forall j \in F, \forall t \in T \\
& G_j^t \leq \overline{G_j^t} & \forall j \in F, \forall t \in T \\
& B_j^t \leq \overline{B_j^t} & \forall j \in F, \forall t \in T
\end{aligned}
$$

Vincoli di capacità energetica dei nodi

Modellazione con accumulatori



$$
\begin{aligned}
\max & \sum_{i \in N} r_i \cdot Z_i - \sum_{t \in T} \sum_{j \in F} (w^t_{hj} H^t_j + w^t_{aj} A^t_j + w^t_{bj} B^t_j) && \\
\text{s.t.} & \sum_{j \in F} \sum_{t \geq a_i + \delta_{kj}}^{t \leq a_i + \tau_i - \delta'_{jk}} X^t_{ij} = Z_i && \forall i \in N \\
& \sum_{i \in N} \mu_i X^t_{ij} d_i \leq H^t_j + A^t_j + B^t_j && \forall j \in F, \forall t \in T \\
& H^t_j + A^t_j + B^t_j \leq \overline{C_j} && \forall j \in F, \forall t \in T \\
& H^t_j \leq \overline{H^t_j} && \forall j \in F, \forall t \in T \\
& A^t_j \leq \widetilde{A^t_j} && \forall j \in F, \forall t \in T \\
& \widetilde{A^t_j} = \min\{\widetilde{A^{t-1}_j} - A^{t-1}_j + \overline{H^{t-1}_j} - H^{t-1}_j, \overline{A_j}\} && \forall j \in F, \forall t \in T \\
& \widetilde{A^0_j} = 0 && \forall j \in F
\end{aligned}
$$

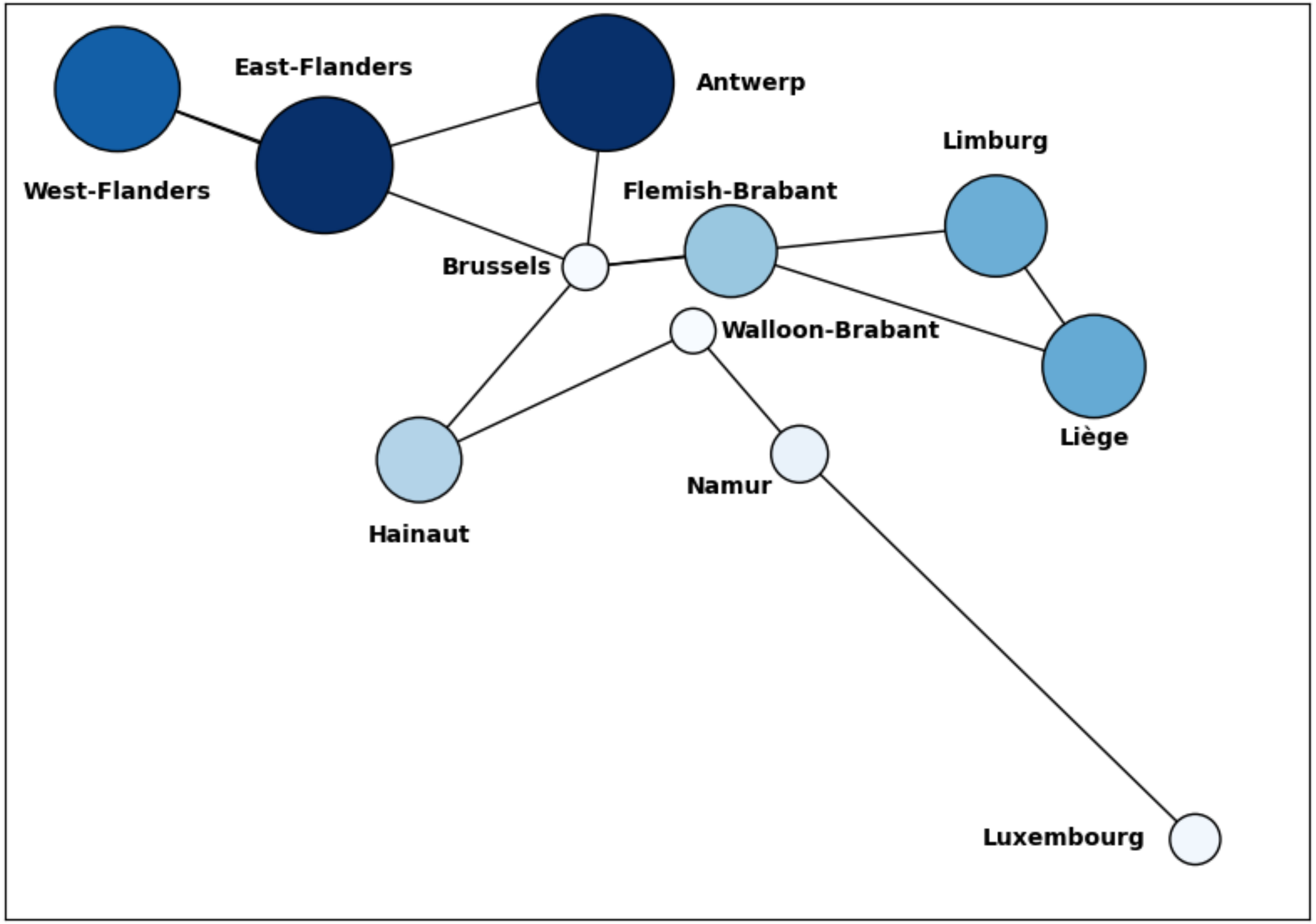
Modellazione con penalità

- Introduzione di una grossa penalità nella funzione obiettivo
- Forzare il modello ad accettare il maggior numero di task
- Ordine di grandezza della penalità è superiore rispetto ai parametri di configurazione dei task
- Garanzia di esecuzione dei task a costo minimo

$$\max \sum_{i \in N} r_i \cdot Z_i - \sum_{t \in T} \sum_{j \in F} (w_{gj}^t \cdot G_j^t + w_{bj}^t \cdot B_j^t) - \sum_{i \in N} \boxed{M} \cdot (1 - Z_i)$$

$$M \gg \max_{i \in N} r_i$$

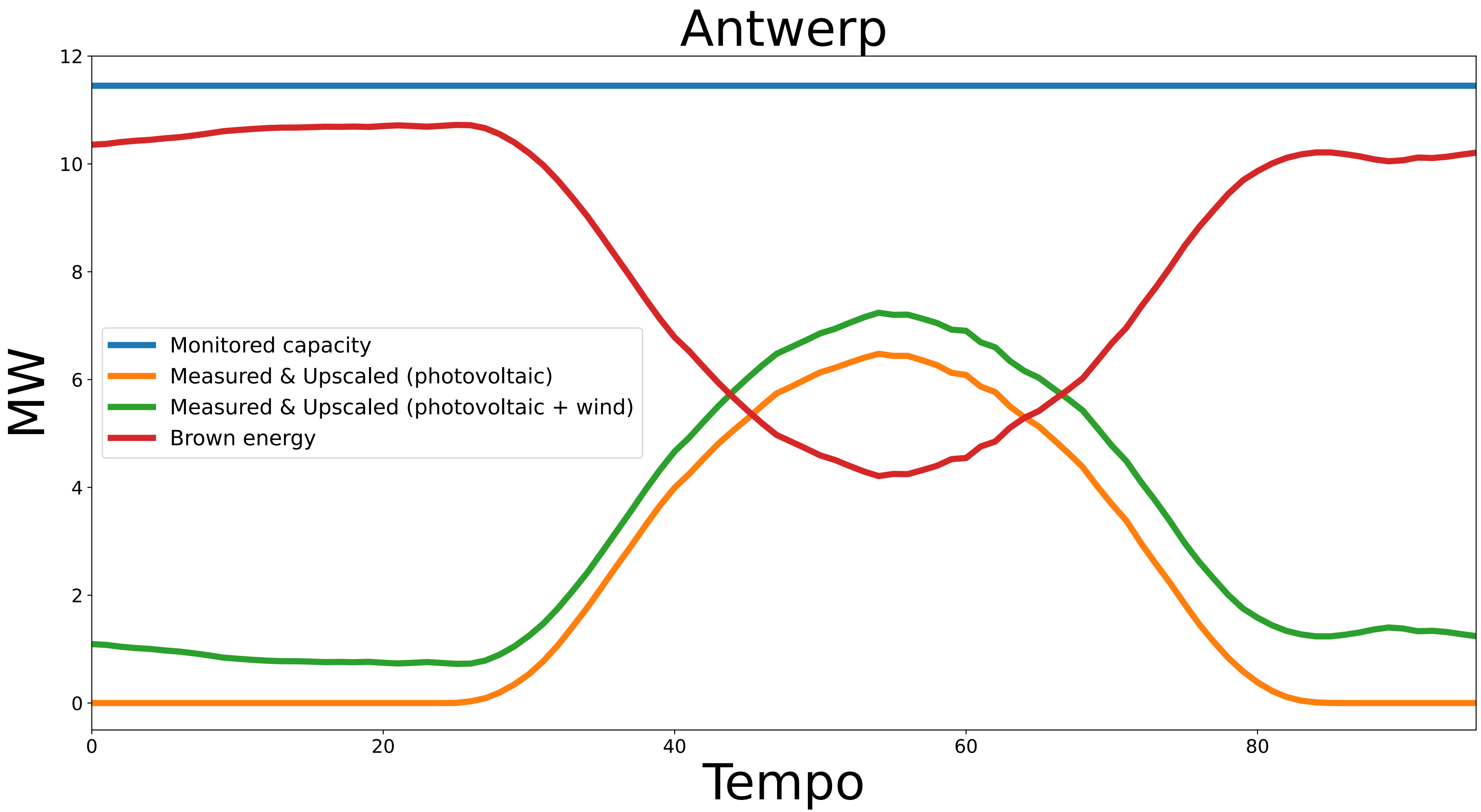
Infrastruttura di rete



Belnet

- Territorio Belgio
- 11 nodi
- Capacità computazionali differenti

Infrastruttura di rete

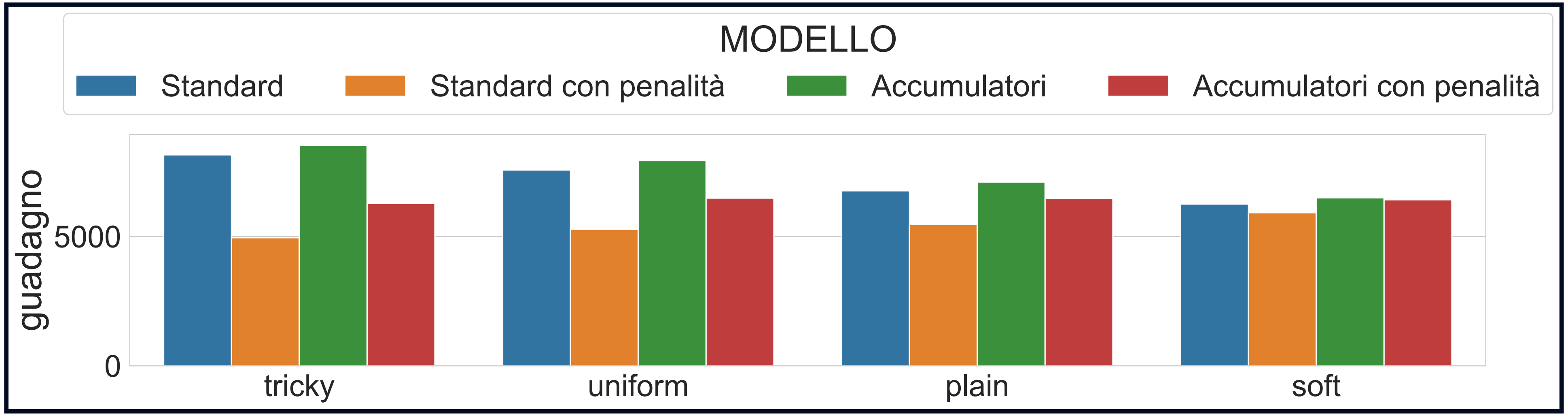


- Capacità energetica totale
- Energia green fotovoltaica + eolica
- Energia brown

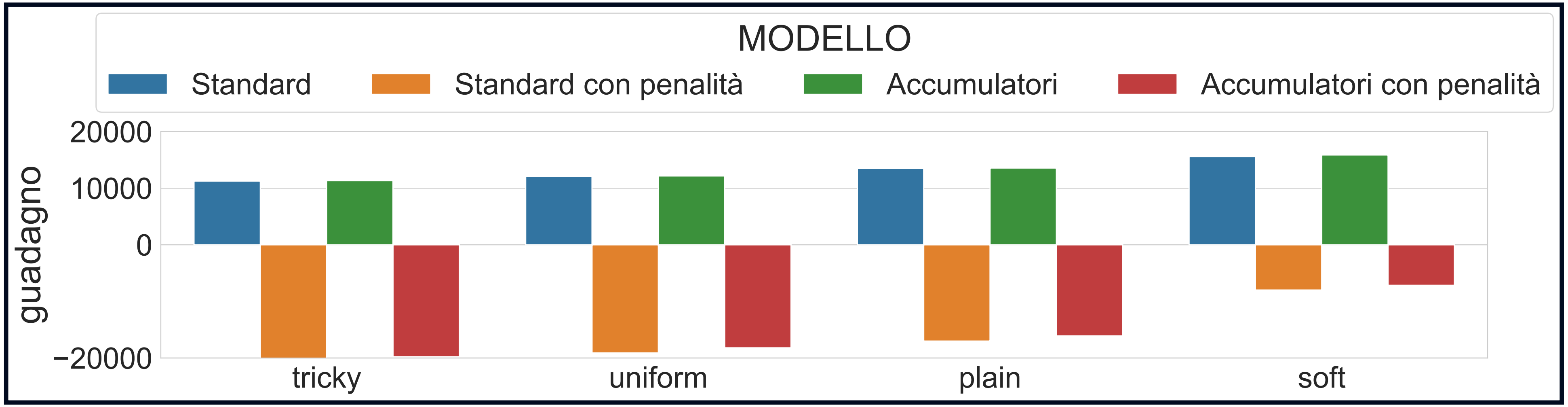
Parametri di configurazione

Numero di task	500, 1000, 2000, 5000	
Numero di nodi di generazione	1, 2, 3, tutti	
Numero di ripetizioni per scenario	50 seed	
Percentuale di tipi di task	Soft	TRIVIAL
	Plain	TRIVIAL, COMPETITIVE
	Uniform	
	Tricky	EXPENSIVE, EXTREME

Guadagno



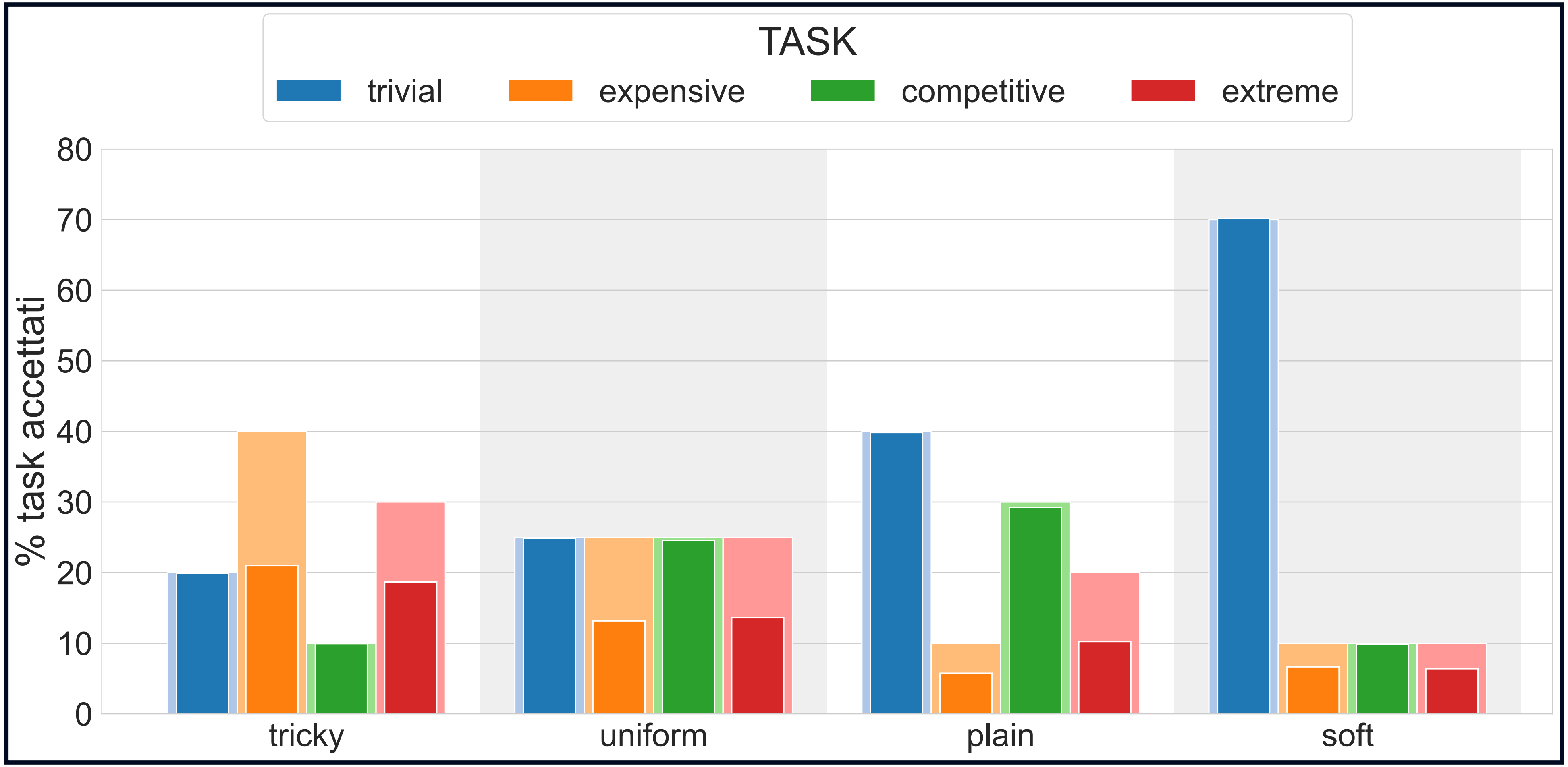
1000 task



5000 task

valori medi su 50 seed

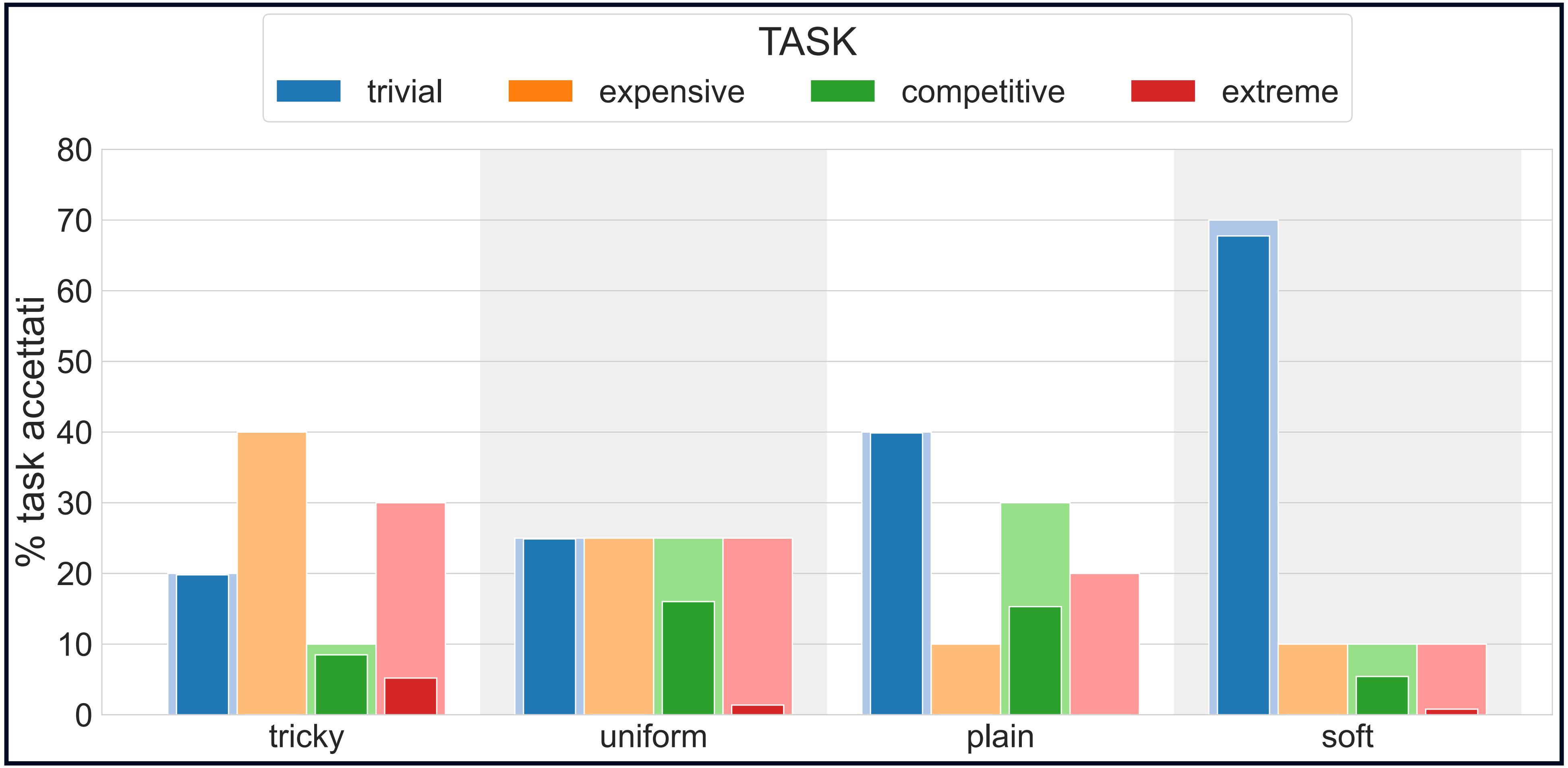
Task accettati



Modello standard senza penalità - 1000 task

valori medi su 50 seed

Task accettati



Modello standard senza penalità - 5000 task

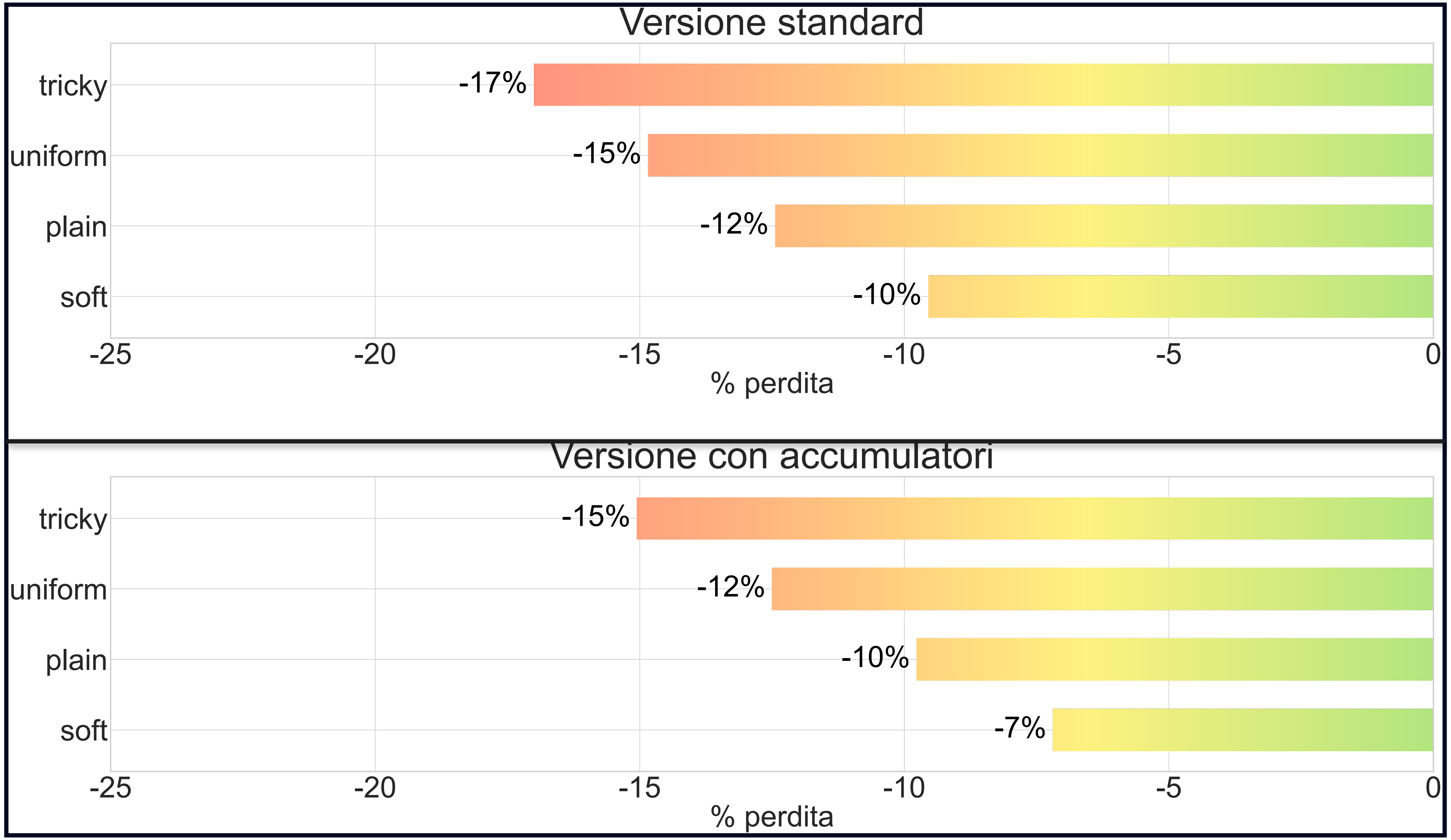
valori medi su 50 seed

Euristica

- Algoritmo centralizzato
- Gestione FIFO task
- Modello PL per singolo task
- Minimizzazione del costo di processamento task
- Condizioni di accettazione task:
  - risorse computazionali disponibili
  - guadagno positivo

$$
\begin{aligned}
\min \sum_{j \in F} \sum_{t \geq a}^{t \leq a+\tau} & \left(w_{gj}^t \cdot G_j^t + w_{bj}^t \cdot B_j^t\right) & \\
\text{s.t.} \sum_{j \in F} \sum_{t \geq a+\delta_{kj}}^{t \leq a+\tau-\delta'_{jk}} & X_j^t = 1 & \\
\mu X_j^t d & \leq G_j^t + B_j^t & \forall j \in F, \forall t \in \{a, a+d\} \\
G_j^t & \leq \overline{G_j^t} & \forall j \in F, \forall t \in \{a, a+d\} \\
B_j^t & \leq \overline{B_j^t} & \forall j \in F, \forall t \in \{a, a+d\}
\end{aligned}
$$

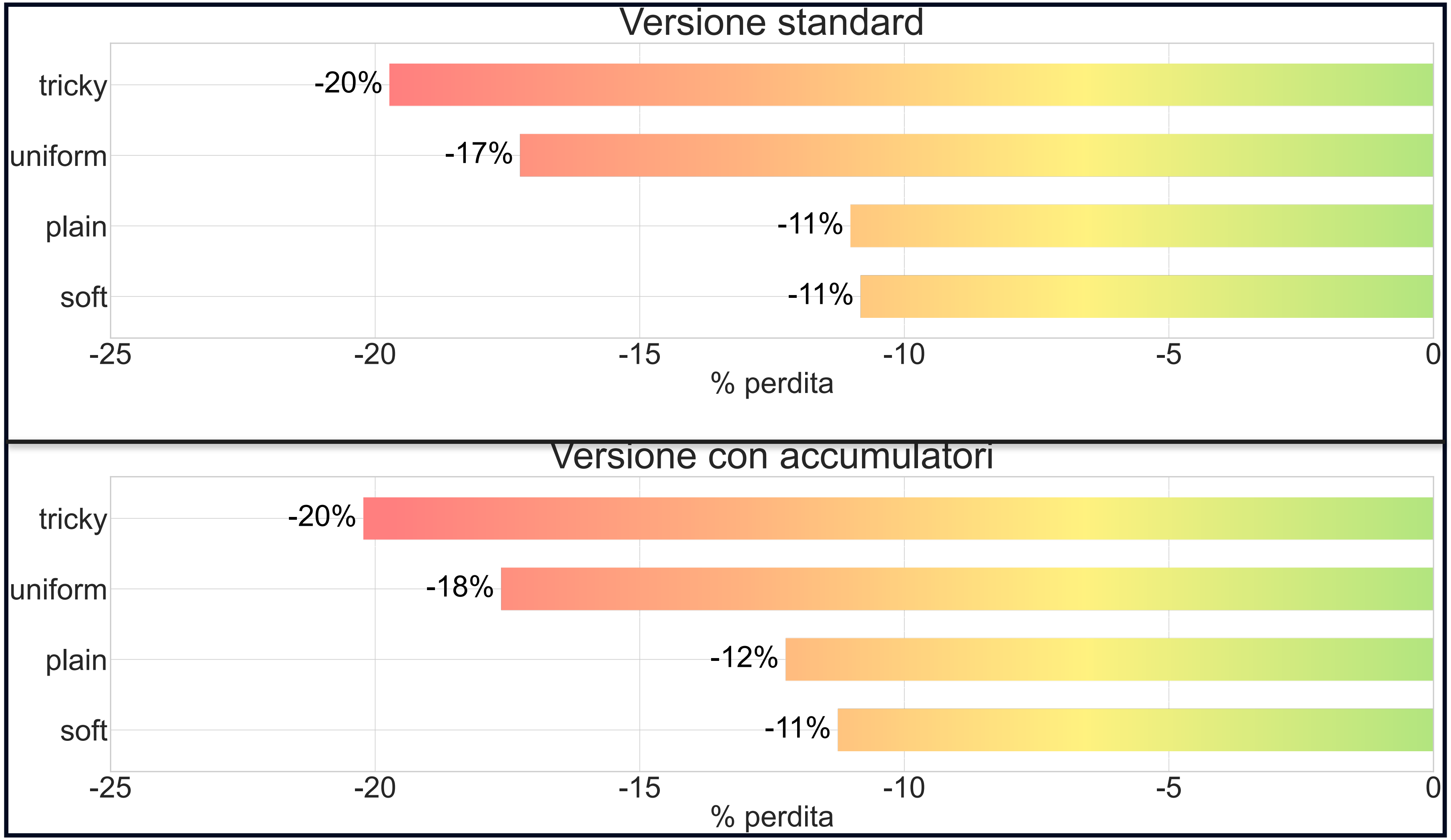
Guadagno - perdita percentuale



1000 task

valori medi su 50 seed

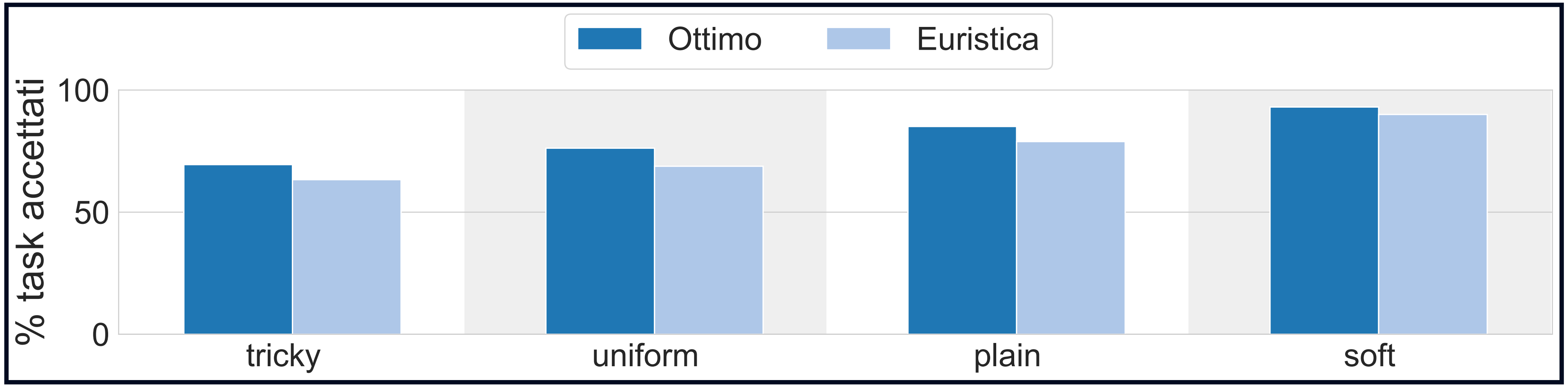
Guadagno - perdita percentuale



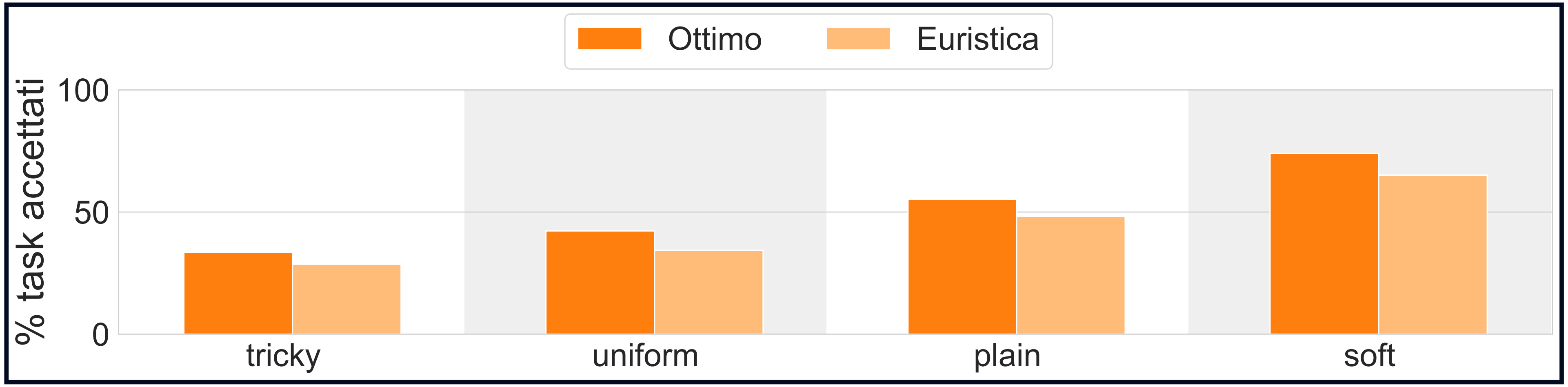
5000 task

valori medi su 50 seed

Task accettati



1000 task



5000 task

Modello standard

valori medi su 50 seed

Conclusioni

- Modello PLI per task scheduling in ambiente green edge computing
- Versione standard, con accumulatori, con penalità
- Algoritmo con modello PL per singolo task
- Creazione infrastruttura di rete secondo diversi scenari di configurazione
- Campagna sperimentale e confronti

Sviluppi futuri

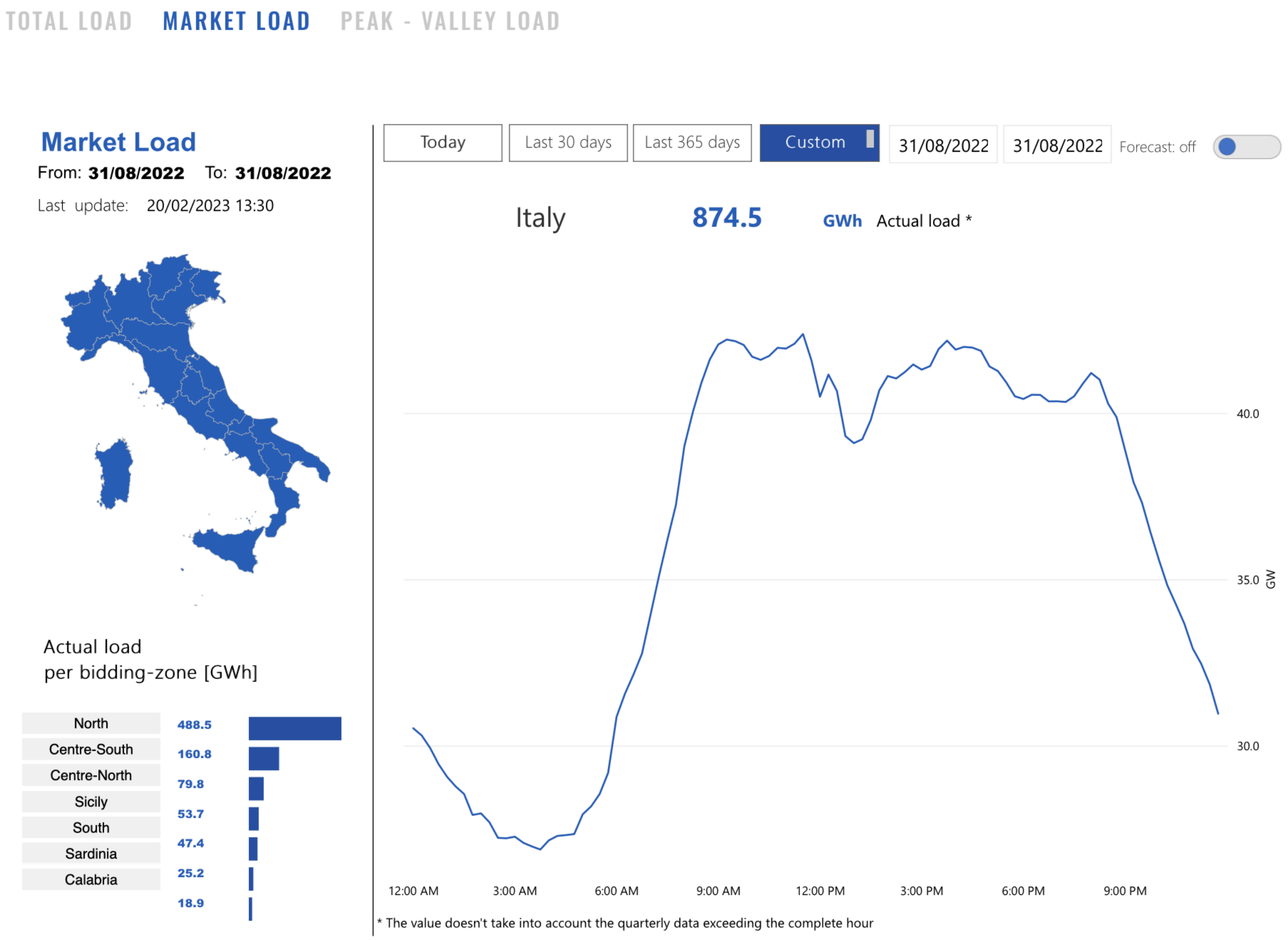
- Insieme più ampio di scenari di valutazione:
  - Diverse topologie di rete
  - Diversa dinamica temporale delle risorse green

- Soluzioni online basate su Deep Reinforcement Learning

- Approccio distribuito al problema di task scheduling in ambito green edge

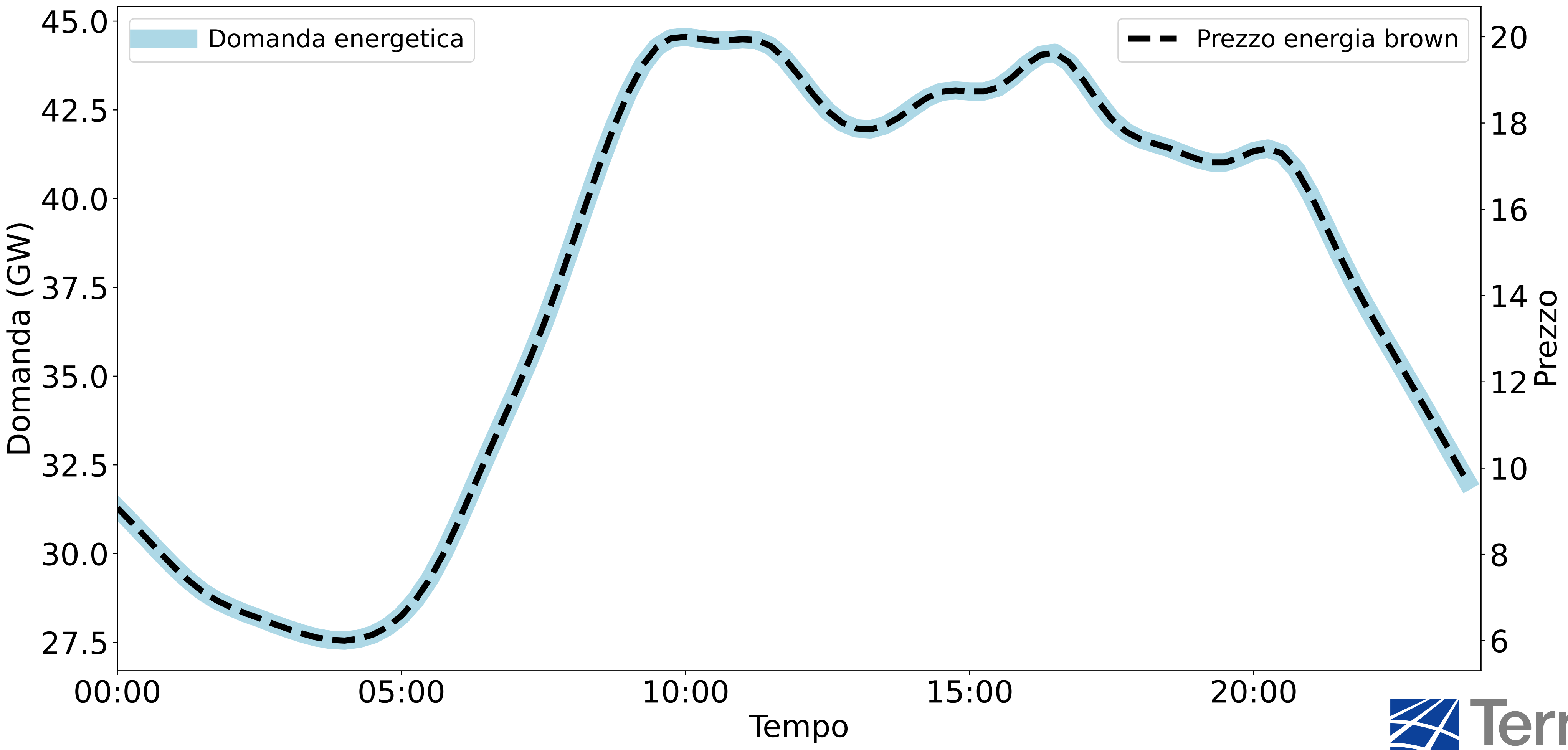
GRAZIE

Infrastruttura di rete



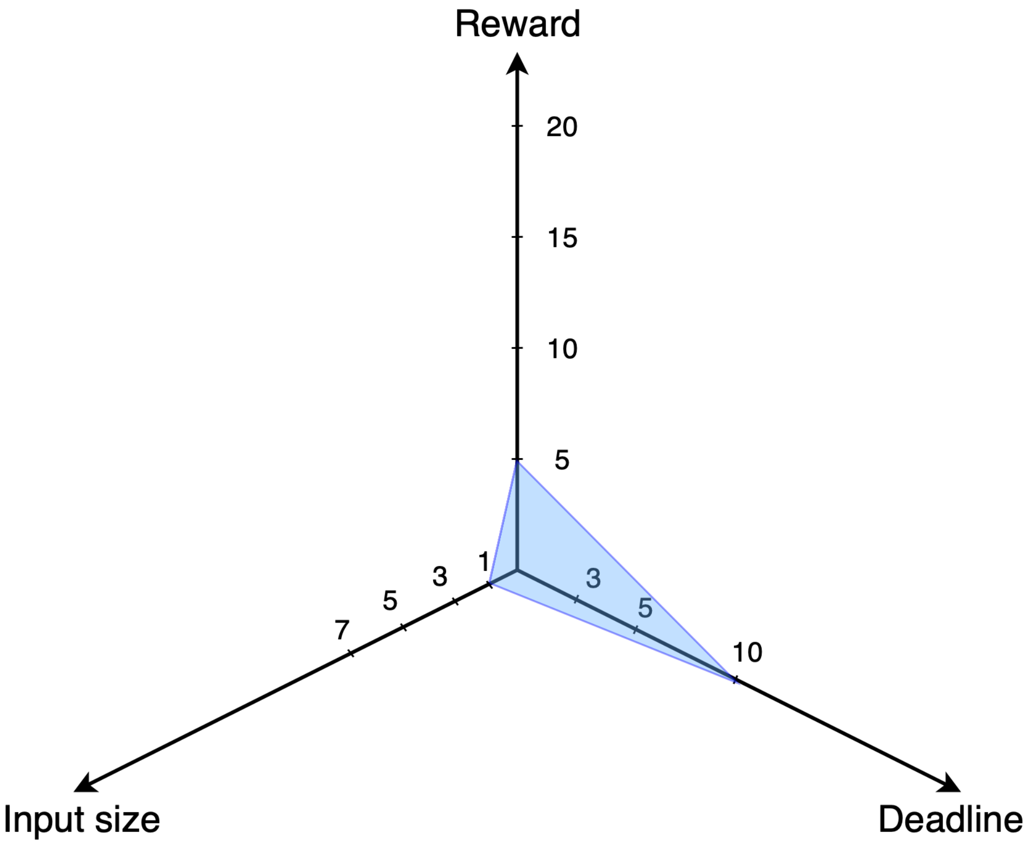
Terna
Driving Energy

Infrastruttura di rete

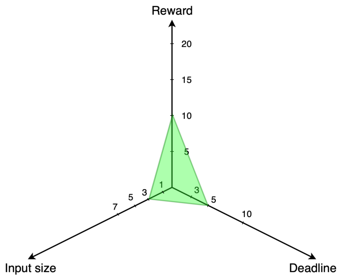


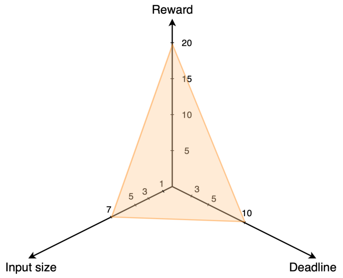
Terna
Driving Energy

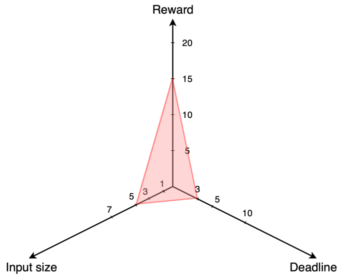
Tipologie di task



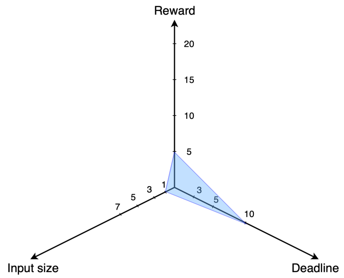
TRIVIAL

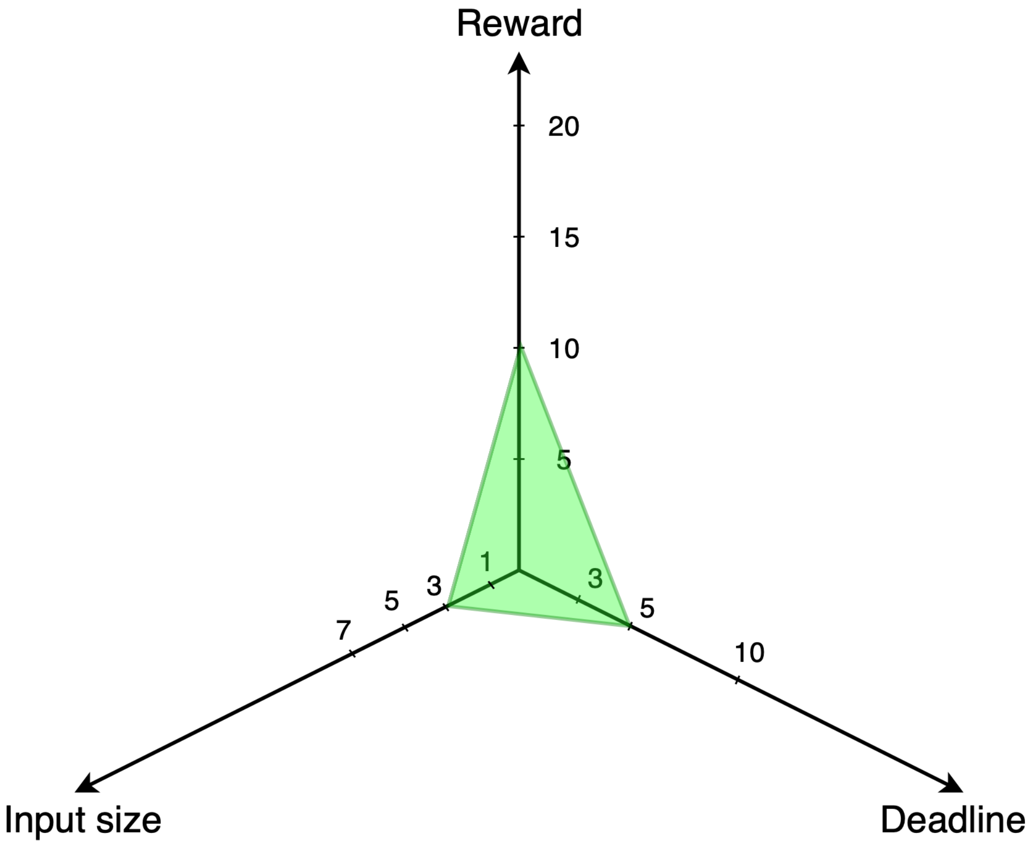




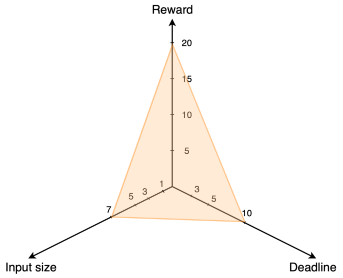


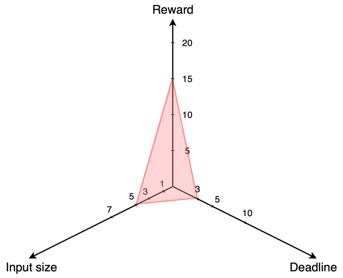
Tipologie di task



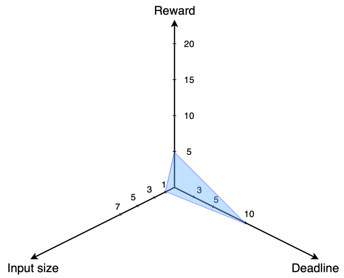


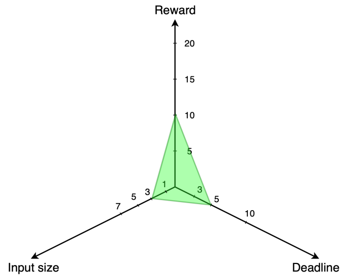
COMPETITIVE

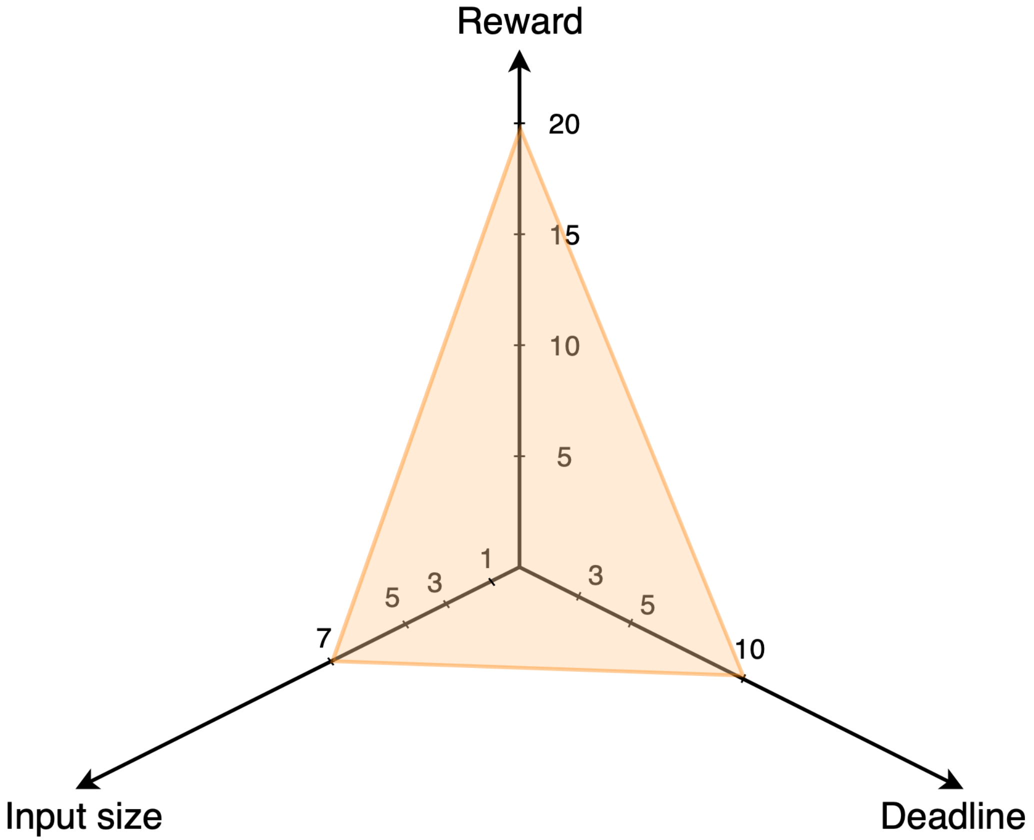




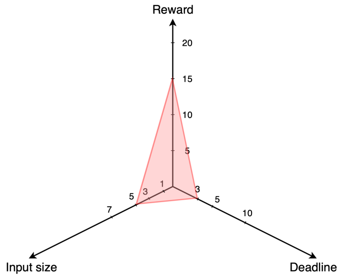
Tipologie di task



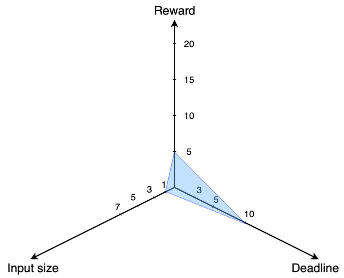


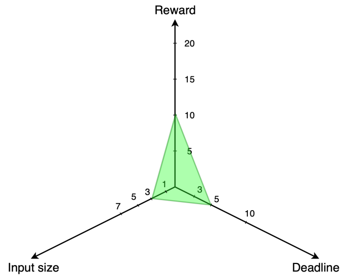


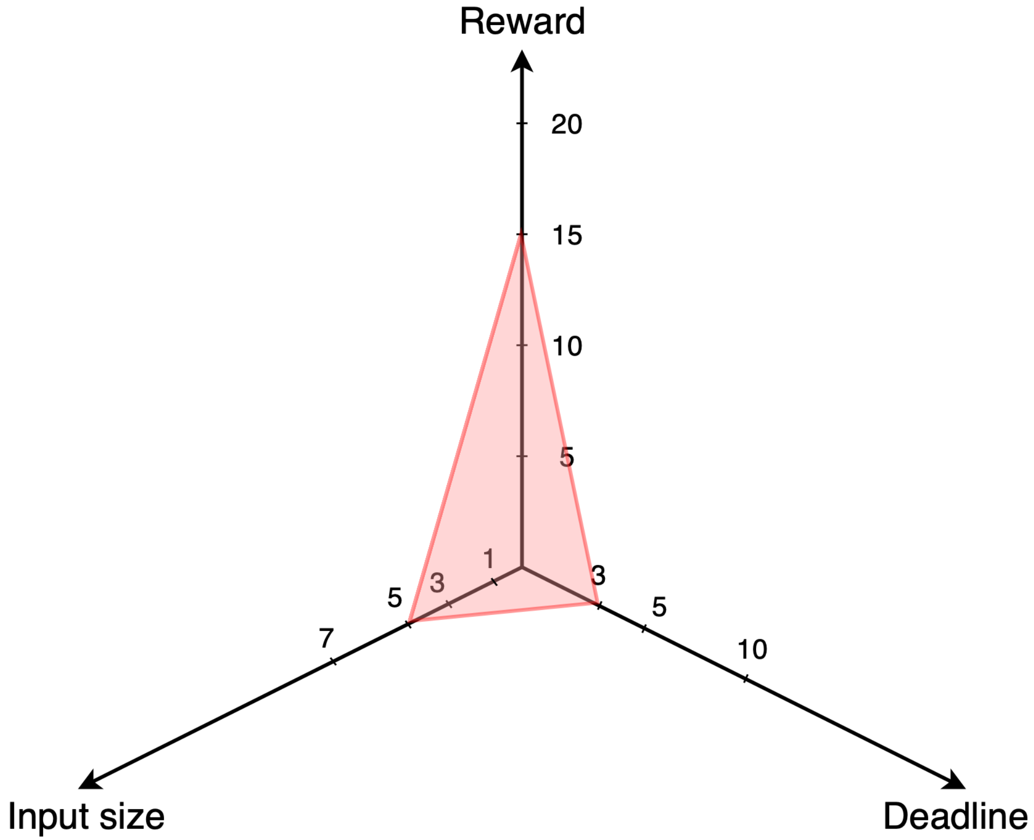
EXPENSIVE



Tipologie di task

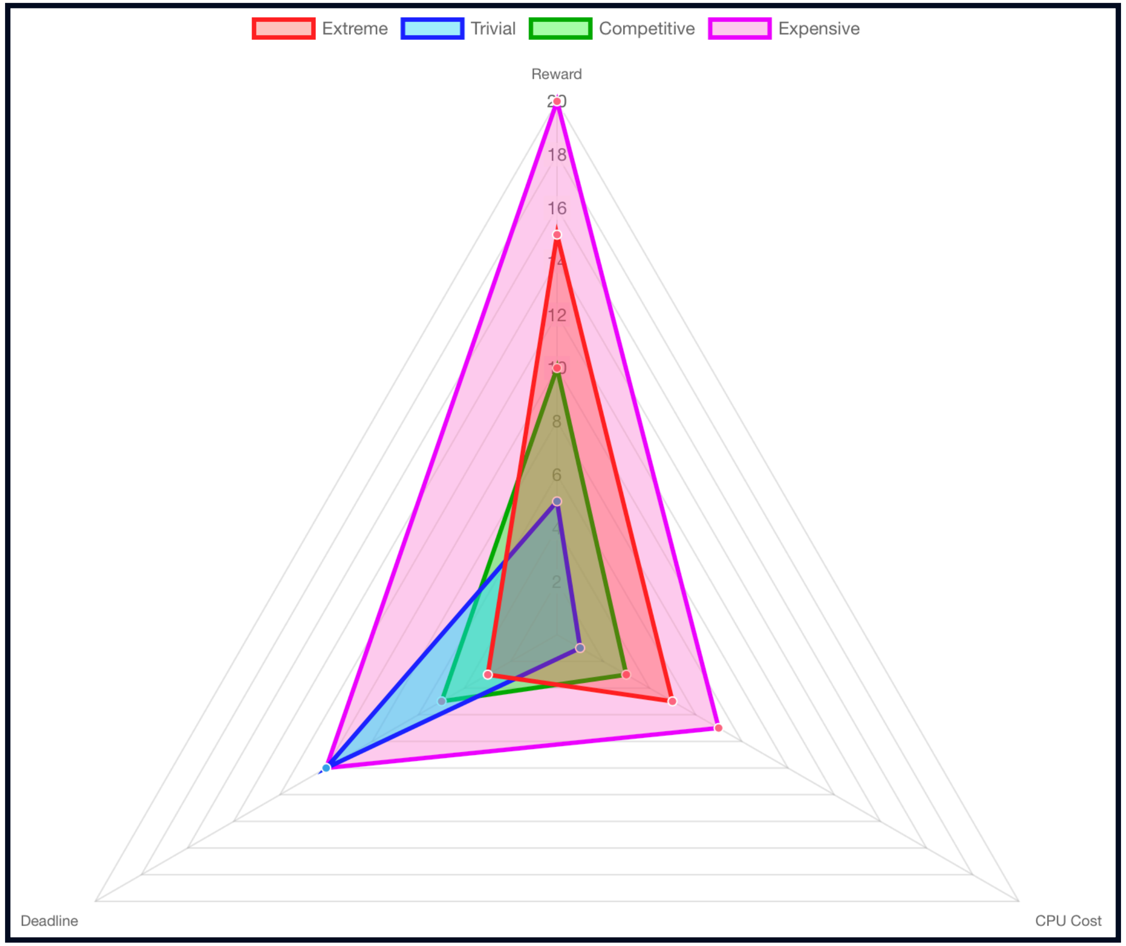






EXTREME

Tipologie di task

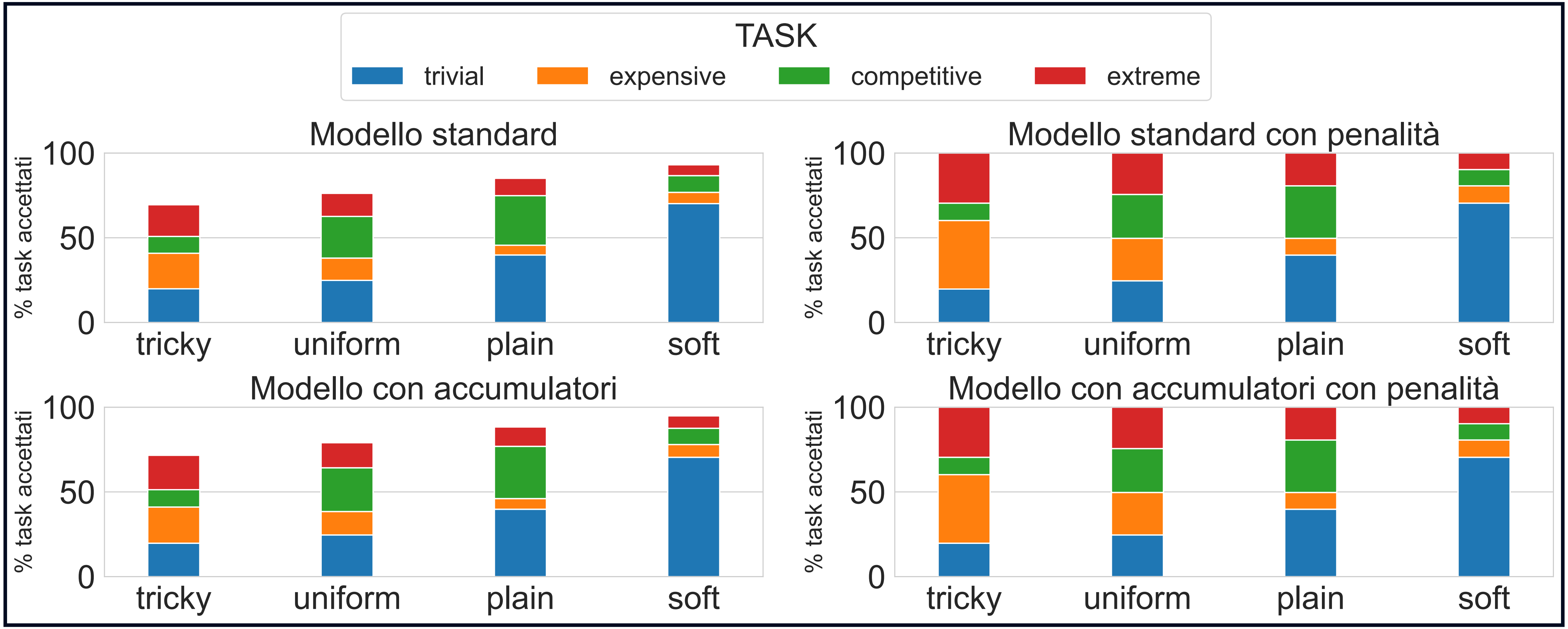


Tipologie di task

```
1: {'#': 1,
    'I': 2000,
    'O': 100,
    'a': 76,
    'd': 5,
    'k': 0,
    'r': 10,
    'type': 'competitive',
    'u': 3},
```

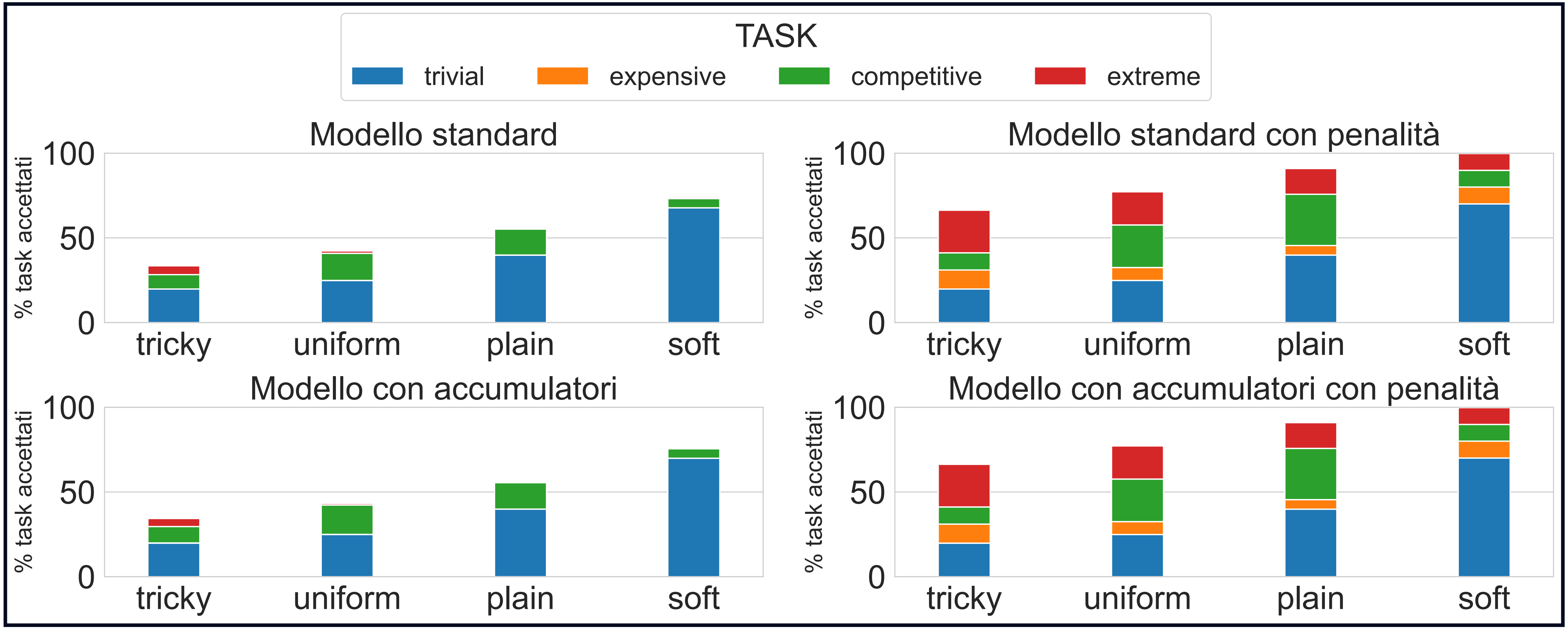
Esempio di task

Task accettati



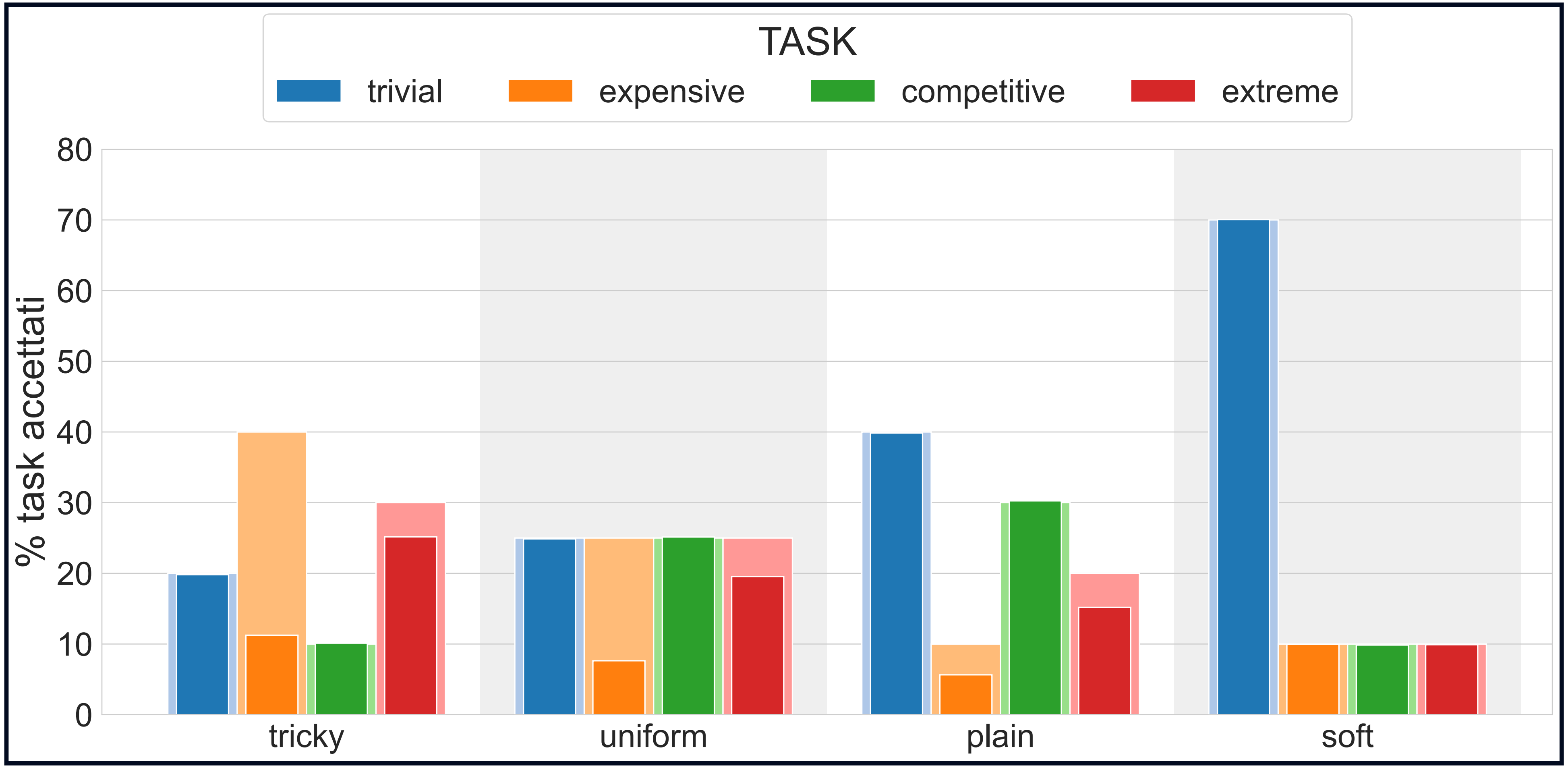
1000 task

Task accettati



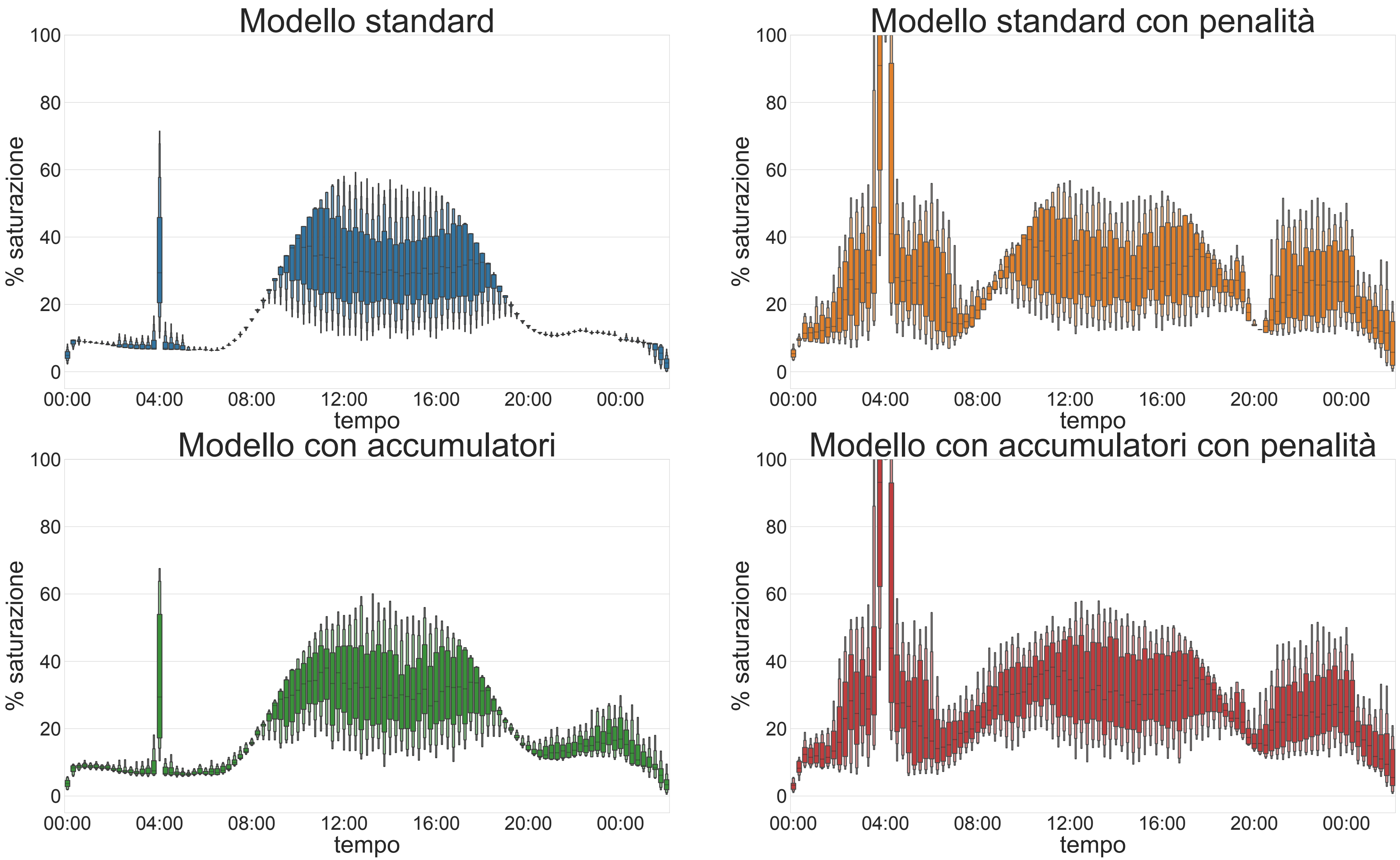
5000 task

Task accettati



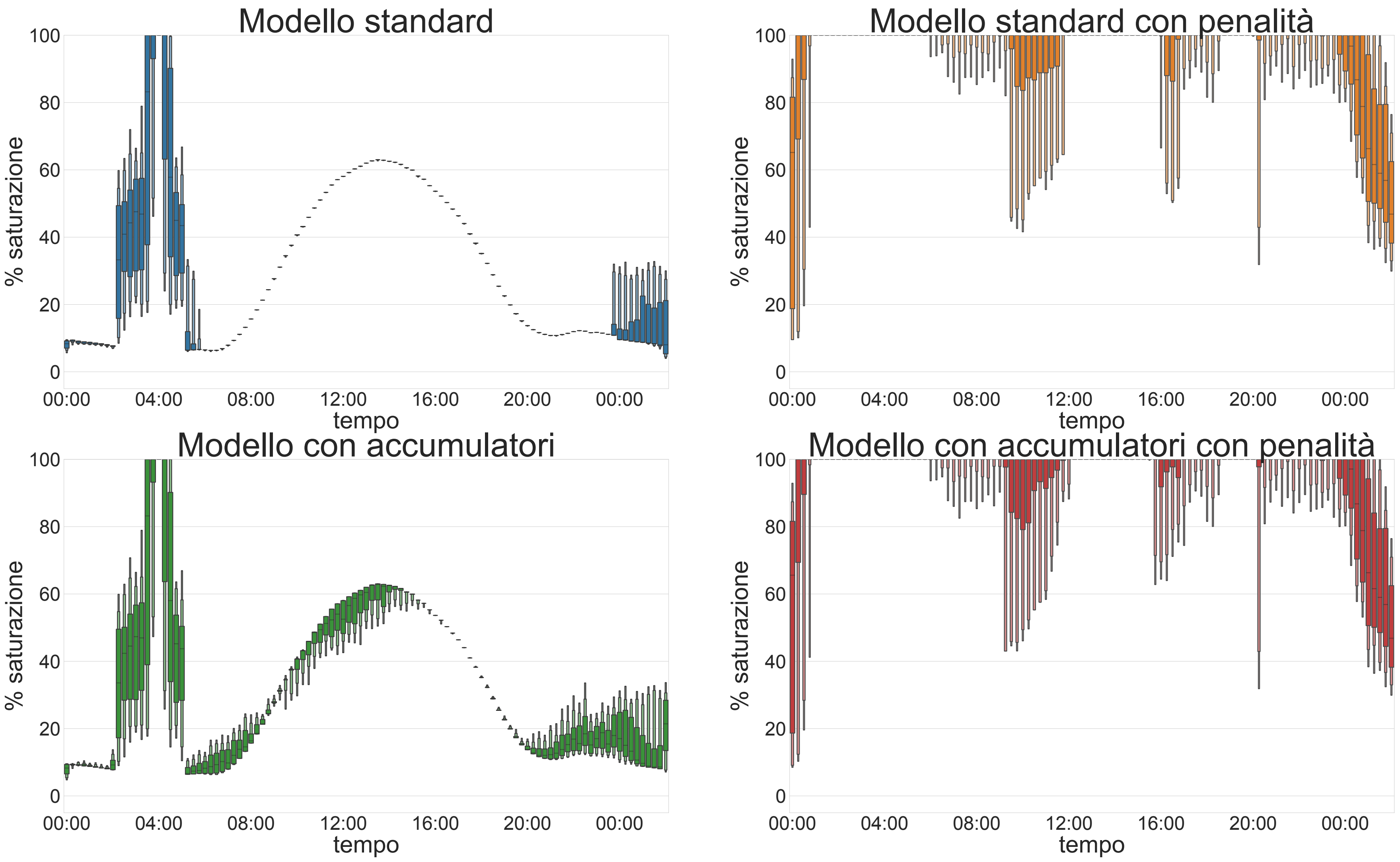
Modello standard con penalità - 5000 task

Saturazione



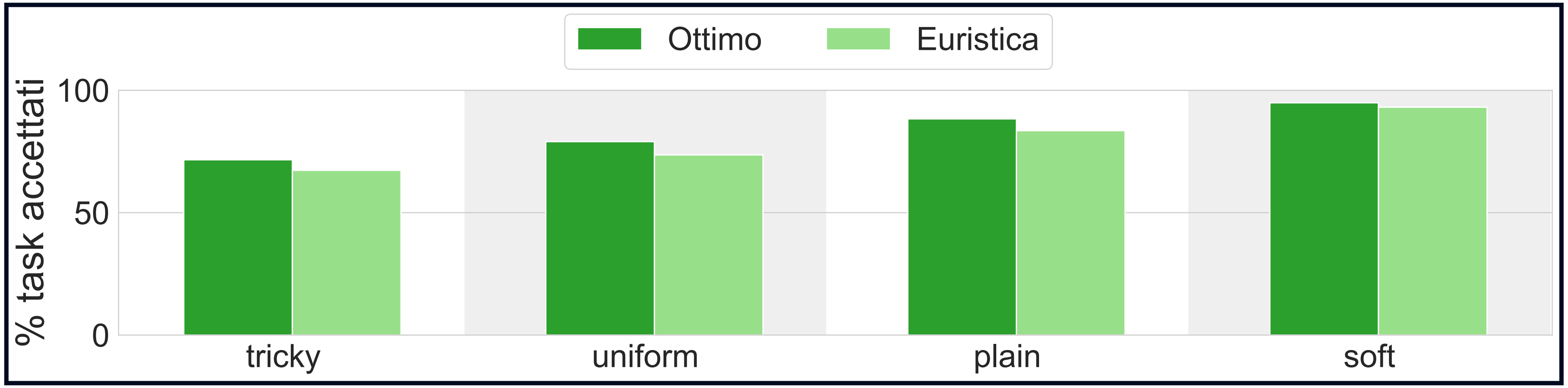
1000 task

Saturazione

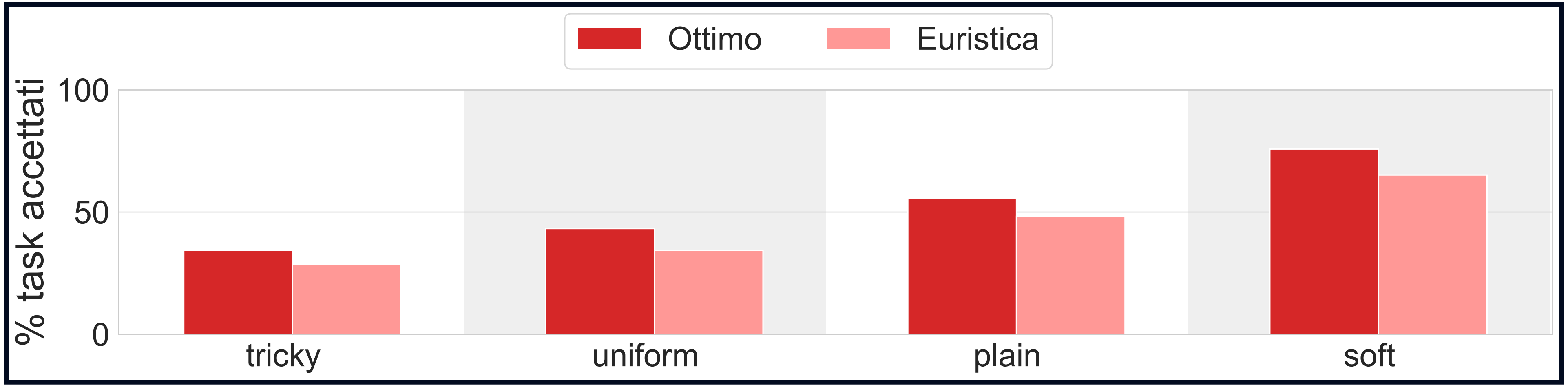


5000 task

Task accettati



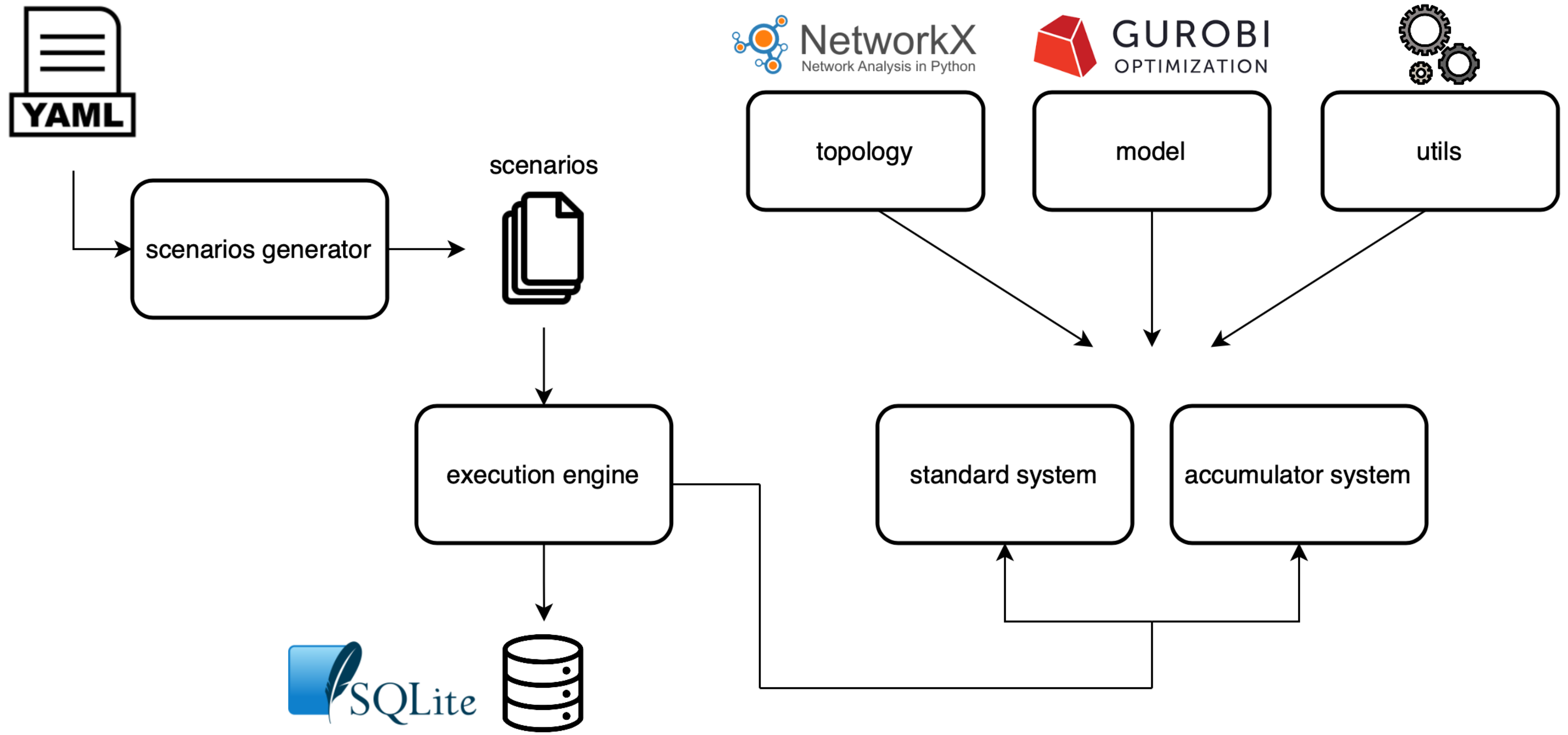
1000 task



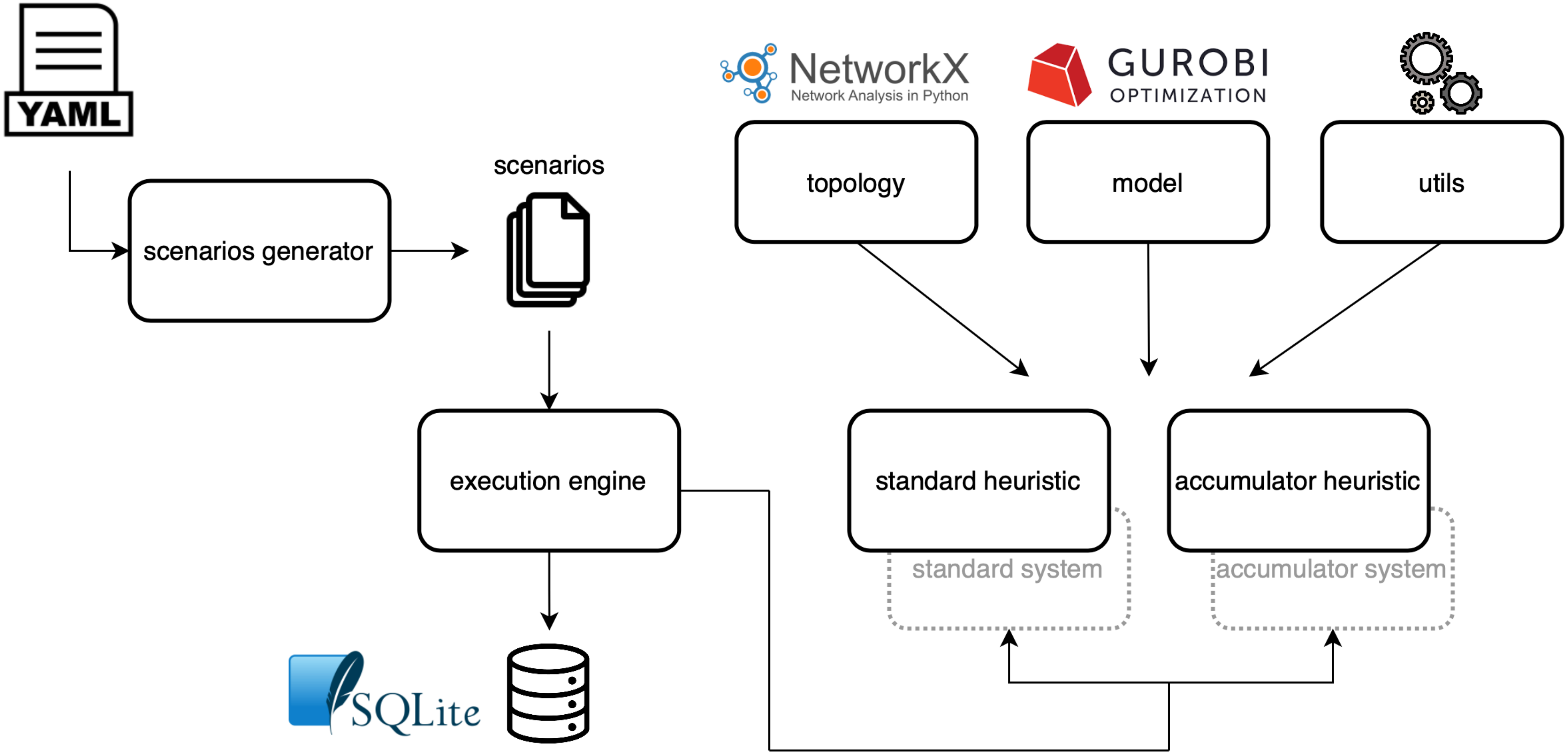
5000 task

Modello con accumulatori

Implementazione



Implementazione



Scenari di configurazione


```yaml
model_type: ['standard', 'accumulator', 'standard_forced', 'accumulator_forced']
timeslots: 96
number_of_tasks: [500, 1000, 2000, 5000]
task_prob:
  trivial: [0.4, 0.7, 0.2, 0.25]
  expensive: [0.1, 0.1, 0.4, 0.25]
  competitive: [0.3, 0.1, 0.1, 0.25]
  extreme: [0.2, 0.1, 0.3, 0.25]

arrival_time:
  distributions:
    - uniform:
        min_value: 0
        max_value: timeslots

input_nodes: [[0, 10], [1, 7, 9], [6], [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]]
output_size: 100

energy_consumption: 0.015
green_cost: 0.5
battery_cost: 0.8
min_brown_cost: 6
max_brown_cost: 20.0
```


File di configurazione


```
{'arrival_time': 'uniform(0,timeslots)',
 'battery_cost': 0.8,
 'energy_consumption': 0.015,
 'green_cost': 0.5,
 'input_nodes': [0, 10],
 'max_brown_cost': 20.0,
 'min_brown_cost': 6,
 'model_type': 'standard',
 'number_of_tasks': 500,
 'output_size': 100,
 'seed': 1652921195,
 'task_names': ['trivial', 'expensive', 'competitive', 'extreme'],
 'task_prob': [0.4, 0.1, 0.3, 0.2],
 'timeslots': 96}
```


Scenario