NANCY

An Artificial Intelligent Aided Unified Network for Secure Beyond 5G Long Term Evolution [GA: 101096456]

Deliverable 4.5

Smart Pricing Policies

Programme: HORIZON-JU-SNS-2022-STREAM-A-01-06

Start Date: 01 January 2023

Duration: 36 Months

Co-funded by the European Union

6GSNS

NANCY project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101096456.

Document Control Page

Deliverable Name	Smart Pricing Policies
Deliverable Number	D4.5
Work Package	WP4 ‘Dynamic Resource Management and Smart Pricing’
Associated Task	T4.5 ‘Smart Pricing Policies’
Dissemination Level	Public
Due Date	31 March 2025 (M27)
Completion Date	31 March 2025
Submission Date	31 March 2025
Deliverable Lead Partner	8BELLS
Deliverable Author(s)	Ilias Theodoropoulos (8BELLS), Stratos Vamvourellis (8BELLS), Cristina Regueiro (TECN), Ramon Sanchez-Iborra (UMU), Gonzalo Alarcon-Hellin (UMU)
Version	1.0

Document History

Version	Date	Change History	Author(s)	Organisation
0.1	6 February 2025	Table of Contents definition	Ilias Theodoropoulos, Stratos Vamvourellis	8BELLS
0.2	12 February 2025	Marketplace related details (Section 4)	Cristina Regueiro	TECNALIA
0.3	17 February 2025	Section 6.1	Ramon Sanchez-Iborra, Gonzalo Alarcon-Hellin	UMU
0.4	20 February 2025	Sections 1 and 2	Ilias Theodoropoulos	8BELLS
0.5	26 February 2025	Sections 3.1-3.3	Ilias Theodoropoulos, Stratos Vamvourellis	8BELLS
0.6	7 March 2025	Section 6.2, Section 3.4, Section 5	Stratos Vamvourellis	8BELLS
0.7	14 March 2025	Bibliography, Section 7	Stratos Vamvourellis	8BELLS
0.8	20 March 2025	Final Version for Internal Review	Stratos Vamvourellis	8BELLS

0.9	26 March 2025	Addressing Reviewer Comments	Stratos Vamvourellis, Ilias Theodoropoulos	8BELLS
1.0	31 March 2025	Quality Revision	Dimitrios Pliatsios, Anna Triantafyllou	UOWM

Internal Review History

Name	Organisation	Date
Marisa Escalante	TECNALIA	25 March 2025
Maria Belesioti	OTE	26 March 2025

Quality Manager Revision

Name	Organisation	Date
Dimitrios Pliatsios, Anna Triantafyllou	UOWM	31 March 2025

Legal Notice

The information in this document is subject to change without notice.

The Members of the NANCY Consortium make no warranty of any kind about this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

The Members of the NANCY Consortium shall not be held liable for errors contained herein or direct, indirect, special, incidental, or consequential damages in connection with the furnishing, performance, or use of this material.

Co-funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or SNS JU. Neither the European Union nor the SNS JU can be held responsible for them.

Table of Contents

List of Figures

List of Tables

List of Acronyms

Acronym	Explanation
AEC	Agent Environment Cycle
AI	Artificial Intelligence
API	Application Interface
B-RAN	Blockchain Radio Access Network
B5G	Beyond 5G
EFGs	Extensive Form Games
FCN	Fully Connected Neural Network
IoT	Internet of Things
MARL	Multi-Agent Reinforcement Learning
MNO	Mobile Network Operator
MSE	Mean Squared Error
POSGs	Partially Observable Stochastic Games
PPO	Proximal Policy Optimization
QoS	Quality of Service
RL	Reinforcement Learning
SPM	Smart Pricing Module
TRPO	Trust Region Policy Optimization
VPG	Vanilla Policy Gradient

Executive Summary

This deliverable outlines the design, implementation, and evaluation of the Smart Pricing Module (SPM) within the NANCY framework under Task 4.5 “Smart Pricing Policies”, focusing on pricing and resource allocation in the Blockchain Radio Access Network (B-RAN) for Beyond 5G (B5G) ecosystems. The SPM employs AI-driven strategies, using Multi-Agent Reinforcement Learning (MARL) and auction theory, to facilitate a multi-round blind reverse auction, ensuring fair pricing and efficient resource sharing in a decentralized B5G environment. Testing confirms its effectiveness in achieving competitive pricing while preventing collusion.

1. Introduction

The evolution of wireless communication systems toward 6G demands innovative solutions to manage increasingly complex and dynamic network environments. B-RAN emerges as a transformative framework, leveraging blockchain technology and smart contracts to enable secure, decentralized, peer-to-peer connectivity and resource management [1]**.** Unlike traditional centralized approaches, B-RAN's distributed nature requires adaptive mechanisms to optimize resource allocation and pricing, which are key challenges in providing efficient and equitable access to network resources. Task 4.5 addresses these challenges by introducing the SPM, a sophisticated system designed to integrate Artificial Intelligence (AI), auction theory, and game theory into a dynamic pricing model tailored for B-RAN architectures.

The SPM redefines resource sharing by facilitating auction-based interactions that balance efficiency, fairness, and competitiveness, regardless of the resource provider. This deliverable documents the process of designing, implementing, and evaluating the SPM within the broader NANCY framework, offering a comprehensive exploration of its technical foundations and real-world implications. It details how AI-driven pricing strategies, supported by robust hosting infrastructure, deliver fair pricing and balanced customer distribution in a competitive B5G ecosystem [2]. Beyond its technical contributions, this document evaluates the module's performance through rigorous testing and benchmarking, while also considering its business impact—demonstrating how it fosters innovation and adaptability in a multi-stakeholder landscape. By blending cutting-edge technology with practical outcomes, this deliverable not only validates the SPM's role in shaping the future of B5G networks but also paves the way for further advancements and research.

1.1. Purpose of the Deliverable

This deliverable aims to provide a detailed record of the SPM within the NANCY framework, capturing its design, implementation, and evaluation processes. It seeks to present a structured analysis of the SPM's technical components, specifically its AI-driven pricing mechanisms and hosting infrastructure, demonstrating their functionality and performance in achieving fair pricing and balanced resource allocation. The document also clarifies the SPM's integration into the NANCY ecosystem, outlining its operational role and interactions with other system elements.

In addition, this deliverable assesses the SPM's effectiveness through empirical testing and performance metrics, offering evidence of its technical viability. It extends this analysis to explore business implications, highlighting how the module supports stakeholder collaboration and market adaptability. By documenting these aspects, the deliverable serves as both a validation of the SPM's current achievements and a foundation for identifying future improvements and research opportunities.

This deliverable is the result of Task 4.5 “Smart Pricing Policies”.

1.2. Structure of the Document

The deliverable is split into seven sections, each exploring a vital aspect of the module's design, implementation, and evaluation. This arrangement allows for a concise and logical display of the work done, leading the reader through the main ideas and outcomes.

- Section 2 covers the technical details of the system architecture and hosting factors that support the SPM. The module’s main components, which focus on achieving even customer distribution and fair pricing, are described in detail, along with the technologies used in the hosting infrastructure for reliable operation.
- Section 3 focuses on the AI-driven pricing model, explaining its core concepts and implementation. It begins with an introduction to MARL, showing how it enables AI agents to autonomously adapt pricing strategies in a competitive B-RAN environment. This section also explores the use of auction theory as the primary game theory approach, explaining how auction mechanisms model stakeholder interactions. The training process for the AI models, including reinforcement learning (RL) algorithms, is discussed, along with the effort to find the best reward system for the model’s needs. Finally, strategies for load balancing and customer distribution are discussed, in order to establish efficient operation and good user experience.
- The incorporation of the component within the wider NANCY framework is the subject of Section 4. This section explains how the SPM interacts with the NANCY Marketplace.
- Section 5 shows the outcomes of the performance review. The performance benchmarks used are defined, and the results gained from testing and simulations are shown. This section provides empirical evidence of the module's effectiveness, demonstrating that the proposed smart pricing methods are both practical and valuable.
- A business viewpoint is offered in Section 6, bridging the gap between the technical framework and its real-world impact on stakeholders. It explores how the multi-stakeholder model enhances flexibility, fosters innovation, and leverages the NANCY Marketplace and smart pricing techniques to create a more efficient and competitive B5G ecosystem.
- Finally, Section 7 concludes the document by summarizing the key successes of Task 4.5 and outlining the potential for the SPM’s future growth.

2. System Architecture & Hosting

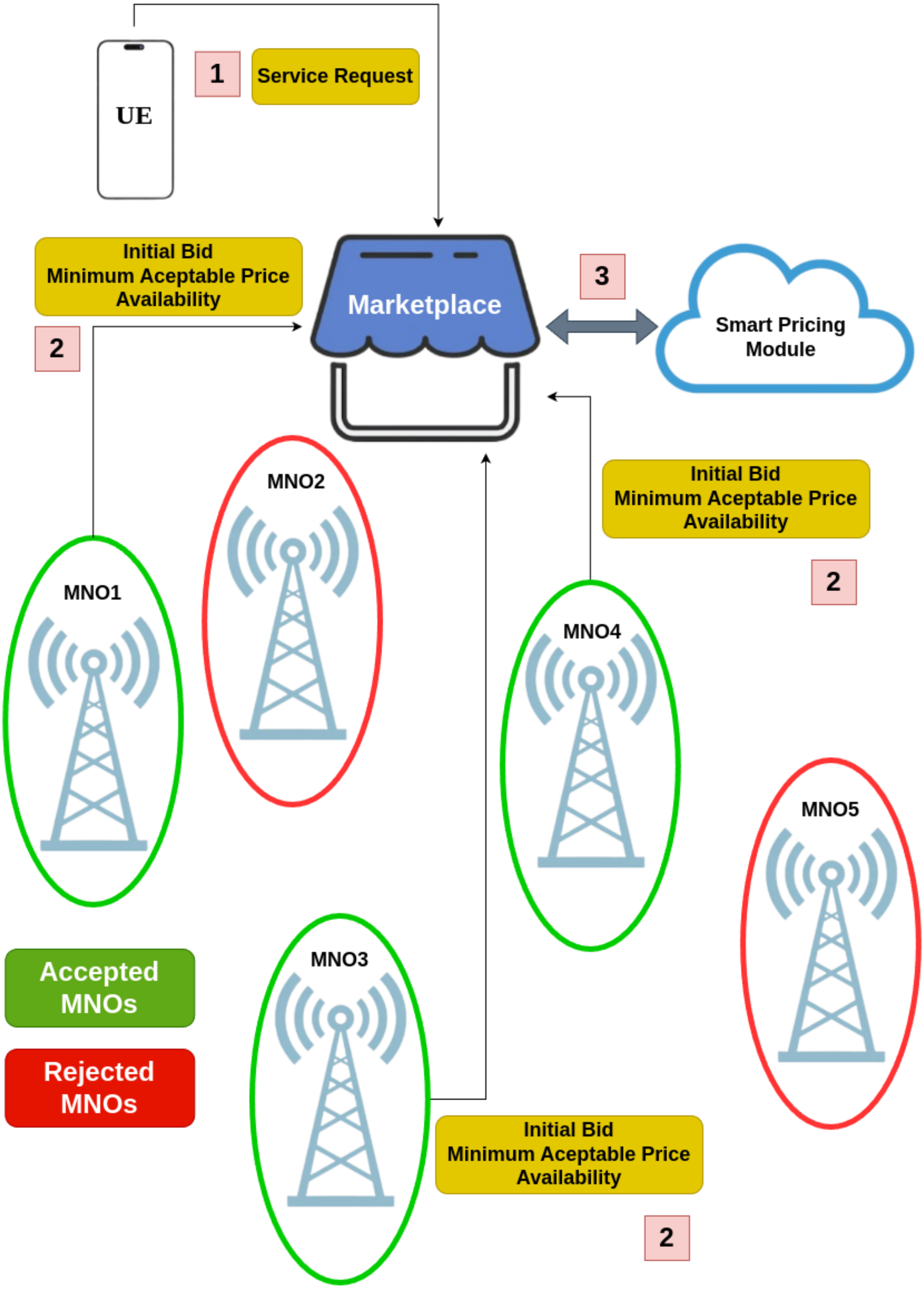

Figure 1: SPM Architecture (1)

Figure 1 illustrates the step-by-step process of the architecture. It begins when the UE sends a Service Request to the Marketplace, which acts as an intermediary between the UE and multiple MNOs.

The Marketplace first evaluates the MNOs, categorizing them as either "accepted" or "rejected" based on their ability to fulfil the UE's request. In the diagram, accepted MNOs (MNO1, MNO3, and MNO4) are shown in green, while rejected ones (MNO2 and MNO5) are in red.

Next, the Marketplace requests an initial bid, minimum acceptable price, and availability from the accepted MNOs. Their responses do not influence their prior acceptance or rejection.

The Marketplace then forwards the collected data to the SPM, which analyses it to select the MNO that will fulfil the UE's request. Once a decision is made, the SPM communicates it back to the Marketplace.

Finally, the Marketplace assigns the job to the chosen MNO, while rejected MNOs are no longer considered.

2.1. Overview of the Smart Pricing Module's Architecture

In modern digital marketplaces and networked environments, efficient resource allocation and competitive pricing are essential for optimal performance. Traditional pricing models often fail to adapt dynamically to fluctuating supply and demand. To address these challenges, the SPM leverages a reverse auction system with intelligent load balancing to ensure fair competition, price optimization, and efficient resource distribution (Figure 2). This section explores the architecture of the SPM, detailing its core components and benefits in dynamic pricing scenarios.

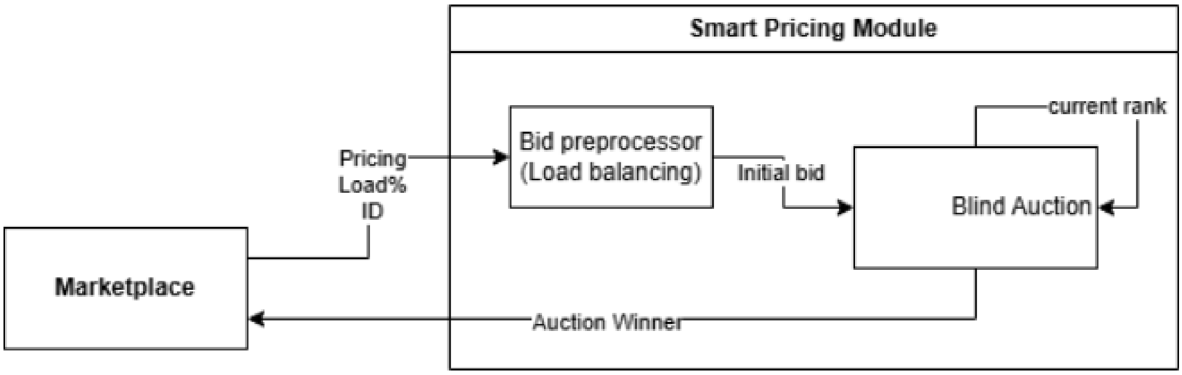

Figure 2: SPM Architecture (2)

Following load balancing, the reverse auction takes place, consisting of N iterative bidding rounds. Participants submit bids, each aiming to offer the lowest acceptable price for the services being auctioned. Throughout the process, the SPM dynamically ranks all participants based on predefined selection criteria, including current bids and their deviation from initial bids or minimum acceptable prices. However, to maintain fairness and strategic bidding, each provider is only aware of their ranking and not the bids of others. Based on their ranking, providers adjust their prices in subsequent rounds strategically to stay competitive. This ranking-based feedback mechanism promotes transparency while preserving competition. In the final round, the provider offering the lowest price is declared the winner. The SPM then sends the auction results, both the winner and the corresponding winning price, back to the Marketplace. This structured approach to pricing and resource allocation makes the SPM a critical enabler of dynamic resource sharing, particularly in future network architectures where resources may originate from diverse providers, including user equipment.

The reverse auction mechanism inherently promotes cost efficiency by driving competition and ultimately selecting the lowest acceptable price. The integrated pre-auction load balancing significantly enhances efficiency by optimizing resource distribution and preventing bottlenecks that could hinder providers and services. The multi-round nature of the auction allows for dynamic price discovery, empowering participants to react to market fluctuations and refine their bids accordingly. Additionally, the SPM offers fairness by ensuring that all participants have equal opportunities to adjust their bids based on market conditions. The ability to interact with the Marketplace and receive real-time data enables the SPM to dynamically adapt to changing economic conditions. Finally, the architecture is designed with scalability in mind, enabling it to manage a large number of participants.

2.2. Deployment

The SPM is deployed within a containerized environment on a dedicated server, leveraging technologies such as Docker to enhance scalability, isolation, and overall ease of management. Containerization certifies that the module runs in a consistent and portable runtime environment, eliminating discrepancies between different deployment stages, from development to production. This approach streamlines software updates, as new versions of the module can be deployed seamlessly without disrupting existing services. Additionally, the containerized setup simplifies dependency management by packaging all necessary libraries and configurations within the container, thereby minimizing compatibility issues and reducing deployment overhead. This guarantees a more stable and efficient deployment process, allowing the module to function reliably in diverse operational conditions. Docker is widely adopted in development and DevOps due to its ability to provide uniform environments across different stages of the software lifecycle, eliminating discrepancies caused by variations in package versions or dependencies that can otherwise lead to unexpected behaviour [3]. By running the SPM inside a container, it is certified that the module operates reliably across various infrastructure setups, reducing potential errors and making deployment more predictable.

Containerization enables streamlined updates and simplified dependency management, reducing manual intervention and improving operational efficiency [4]. Docker containers autonomously manage configurations and dependencies, allowing developers to use the same container across development and production environments. This capability is particularly beneficial for the SPM, as it allows seamless updates to pricing models and algorithms without disrupting service availability. Furthermore, by leveraging container orchestration tools such as Kubernetes, we can automate scaling, ensuring that the SPM can handle varying levels of demand efficiently while maintaining high performance.

To facilitate seamless interaction with the NANCY Marketplace, a dedicated API is hosted within the premises. This API serves as the central communication channel, retrieving crucial information about network providers while also communicating auction results. Designed for efficiency, the API is optimized to handle concurrent requests with minimal latency, ensuring that pricing updates and auction outcomes are relayed in real-time.

3. AI-Driven Pricing Model

3.1. Multi-Agent Reinforcement Learning Framework (MARL)

MARL, an extension of classical RL, enables multiple autonomous agents to interact within a shared, dynamic environment. Regarding the SPM, MARL provides a robust foundation for modelling competitive interactions among AI agents in a B-RAN ecosystem. These agents, representing service providers like MNOs, compete to maximize their profits. This section outlines the evolution of MARL frameworks, introduces the Agent Environment Cycle (AEC) model as the SPM's backbone, and describes its tailored PettingZoo-inspired API, laying the groundwork for the multi-round blind reverse auction and training processes detailed later.

Historically, single-agent RL benefited from standardized tools like OpenAI's Gym [5], which provided a unified API for environment interaction. MARL, however, faced challenges due to a lack of similar standardization, complicating scalable and reproducible multi-agent systems. Early MARL research, as noted by Terry et al. [6] in their PettingZoo paper, relied on diverse mathematical models. Partially Observable Stochastic Games (POSGs) allowed agents to act, observe, and receive rewards simultaneously but struggled in B-RAN settings where pricing decisions occur sequentially, and agents join or leave dynamically. Extensive Form Games (EFGs), with their tree-like action sequences, suited turn-based scenarios like auctions but lacked adaptability for the continuous, real-time demands of SPM's pricing and load balancing. These limitations drove the need for a more flexible framework tailored to the SPM's competitive, event-driven context.

To address this, the SPM adopts the AEC model from the PettingZoo library as its MARL foundation. In AEC, agents act sequentially, as shown in Figure 3, each observes the environment (e.g., current market state), selects an action (e.g., setting a bid in a reverse auction), and passes control to the next agent. Rewards, such as profit or ranking, are tied directly to individual actions. This sequential design offers key advantages for the SPM. It supports auction-based pricing (detailed in Section 3.2) where agents bid prices for offering their services in structured rounds. It also adapts to dynamic agent participation, maintaining stability as providers enter or exit B-RAN, or when users move from one geographical location to another.

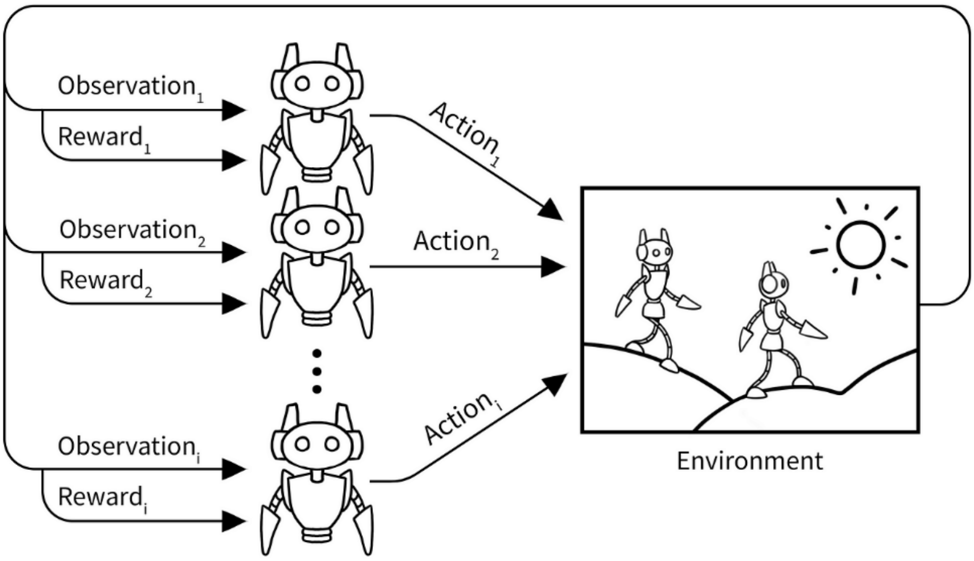

Figure 3: MARL [7]

Building on AEC, the SPM implements a PettingZoo-inspired API customized for its pricing needs. This API features an iterable agent sequence, cycling through providers in each auction round, and a state retrieval mechanism, delivering observations like bid rankings and remaining rounds, alongside rewards. It tracks dynamic agent sets via identifiers (e.g., provider_0), supporting B-RAN's fluctuating participants, and uses dictionary-based data access for agent-specific metrics (e.g., bid history, termination status). This bridges the simplicity of Gym's single-agent interface with MARL's complexity, making the SPM accessible yet scalable. For example, during a multi-round blind reverse auction (Section 3.2), agents adjust bids based on ranking feedback, with the API ensuring computational efficiency via tools like NumPy's vectorized operations.

The AEC model's flexibility enhances its fit for the SPM over earlier frameworks like POSGs, integrating seamlessly with the game-theoretic auction mechanisms and Proximal Policy Optimization (PPO) training approach explored later. Agents learn to balance competitive bidding with profitability, guided by a neural network (Section 3.3) that processes observations and enforces valid actions via action masking (e.g., restricting bids within provider-specific limits).

3.2. Game Theory Principles in Action: Multi-Round Blind Reverse Auction Environment

3.2.1.Auction Theory and Game-Theoretic Frameworks

The SPM anchors its pricing model in auction theory, specifically tailored to a multi-round blind reverse auction designed for the B-RAN ecosystem. In this strategic setup, MNOs or other resource providers compete over N rounds to offer the lowest price for requested network services, with the winner determined by the lowest final bid. Unlike forward auctions where buyers bid upward for goods, reverse auctions flip the dynamic (Figure 4): suppliers bid downward to secure contracts, prioritizing cost efficiency for the network while sustaining provider viability. This aligns with Jap (2002) [8], who frames online reverse auctions as an innovative procurement method where buyers solicit bids from multiple suppliers, driving competition to lower prices, a paradigm the SPM adapts for B5G resource allocation.

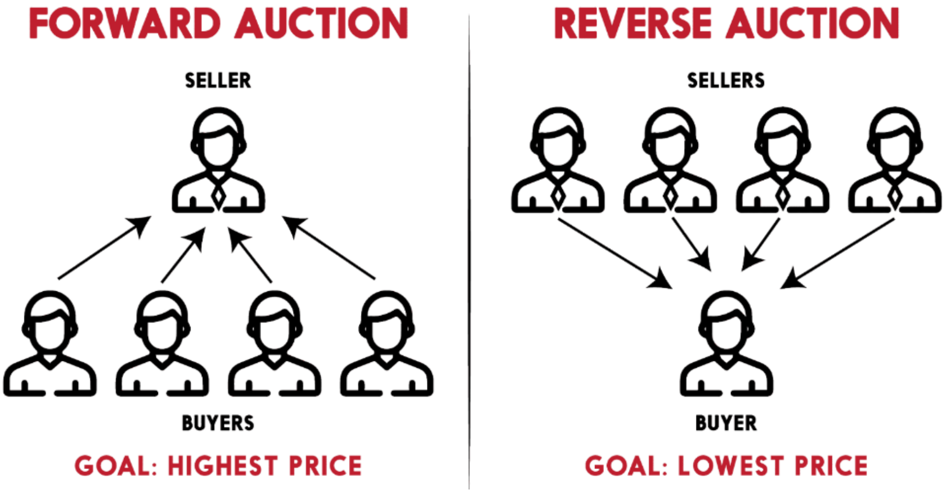

Figure 4: Forward vs Reverse Auctions [9]

Reverse auctions, as a broader procurement strategy, are characterized by several distinctive features that enhance their suitability for dynamic environments like B-RAN. They rely on real-time bidding, ensuring rapid price adjustments, which the SPM leverages through its iterative rounds and AI-driven tools. Outcomes are inherently price-driven, focusing on cost minimization, a principle central to the SPM's goal of fair and efficient resource allocation. Additionally, reverse auctions often incorporate transparent bid visibility and structured rules, though the SPM adapts this with its blind mechanism, assuring clarity and fairness, as seen in sectors like construction where buyers quickly secure competitive bids for materials and services. These attributes make reverse auctions a powerful tool for cost reduction, aligning with the SPM's objective to optimize pricing in a decentralized B5G context [10].

Jap emphasizes that reverse auctions diverge from traditional auction models due to their iterative nature and practical complexities, a point central to the SPM's multi-round design. Rather than a single bid determining the outcome, the SPM allows providers to refine offers across rounds, culminating in the lowest final price, a departure from static theoretical frameworks that Jap critiques as insufficient for capturing real-world dynamics. This iterative process enhances price discovery, leveraging competition to reflect true market value in B-RAN's decentralized context, as supported by Ling et al. [1], who highlight blockchain's role in enabling such distributed systems.

3.2.2. Multi-Round Bidding and Ranking Feedback

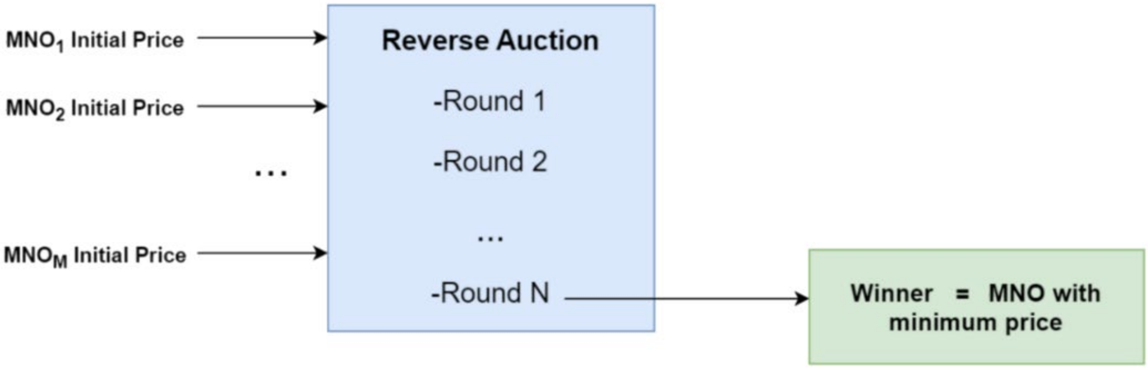

Figure 5: Multi-Round Reverse Auction

The multi-round structure of the SPM's reverse auction (Figure 5) introduces a dynamic, iterative dimension to the game, resembling a repeated game as conceptualized by Fudenberg and Tirole [11]. Across N bidding rounds, providers adjust their bids based on real-time ranking feedback, which serves as a proxy for market conditions and competitive positioning. For example, a provider ranked lower in one round might reduce their bid in the subsequent round to improve their standing, while a higher-ranked provider might maintain or even increase their bid, anticipating competitors' responses. This iterative process fosters strategic depth, as providers must balance the risk of bidding too aggressively (potentially sacrificing profit margins) against bidding too conservatively (risking loss to competitors).

The ranking-based feedback mechanism promotes transparency while preserving competition. Providers receive their position relative to others after each round but remain unaware of specific competitor bids, ensuring a level playing field. This approach aligns with Nash's concept of equilibrium in non-cooperative games, where no player can improve their outcome by unilaterally changing their strategy, given the strategies of others. Over time, the iterative bidding converges toward a Nash equilibrium, stabilizing pricing outcomes that reflect both market dynamics and provider strategies, as supported by simulations discussed in Section 5.

Furthermore, one of the critical advantages of the SPM's multi-round auction is its ability to mitigate the "winner's curse" [12], a common pitfall in auction mechanisms where the winning bidder overpays due to overestimation of value or underestimation of competition [13]. In traditional single-shot auctions, bidders may inflate their offers based on imperfect information, leading to inefficiencies in resource allocation. However, by distributing the bidding process across multiple rounds, the SPM allows providers to recalibrate their strategies based on ranking feedback, reducing the likelihood of overbidding [14].

3.2.3.Technical Implementation of the Reverse Auction Environment

The multi-round blind reverse auction is brought to life through a class implemented as a parallel MARL environment using the PettingZoo framework (see Figure 6). The bidding process, its core mechanism, is designed to model strategic decision-making in competitive pricing scenarios, enabling agents to iteratively adjust their bids over a fixed number of rounds to achieve the lowest rank, representing the most competitive bid, while adhering to personalized constraints defined by minimum and maximum bid limits. This process facilitates competition by allowing simultaneous bid adjustments in each round, optimizes decision-making by balancing competitiveness and profitability, and reflects real-world dynamics where the lowest bid wins. Key features include multi-agent parallel actions for real-time competition, dynamic ranking of bids in ascending order and historical bid tracking to inform future decisions, culminating in a final evaluation.

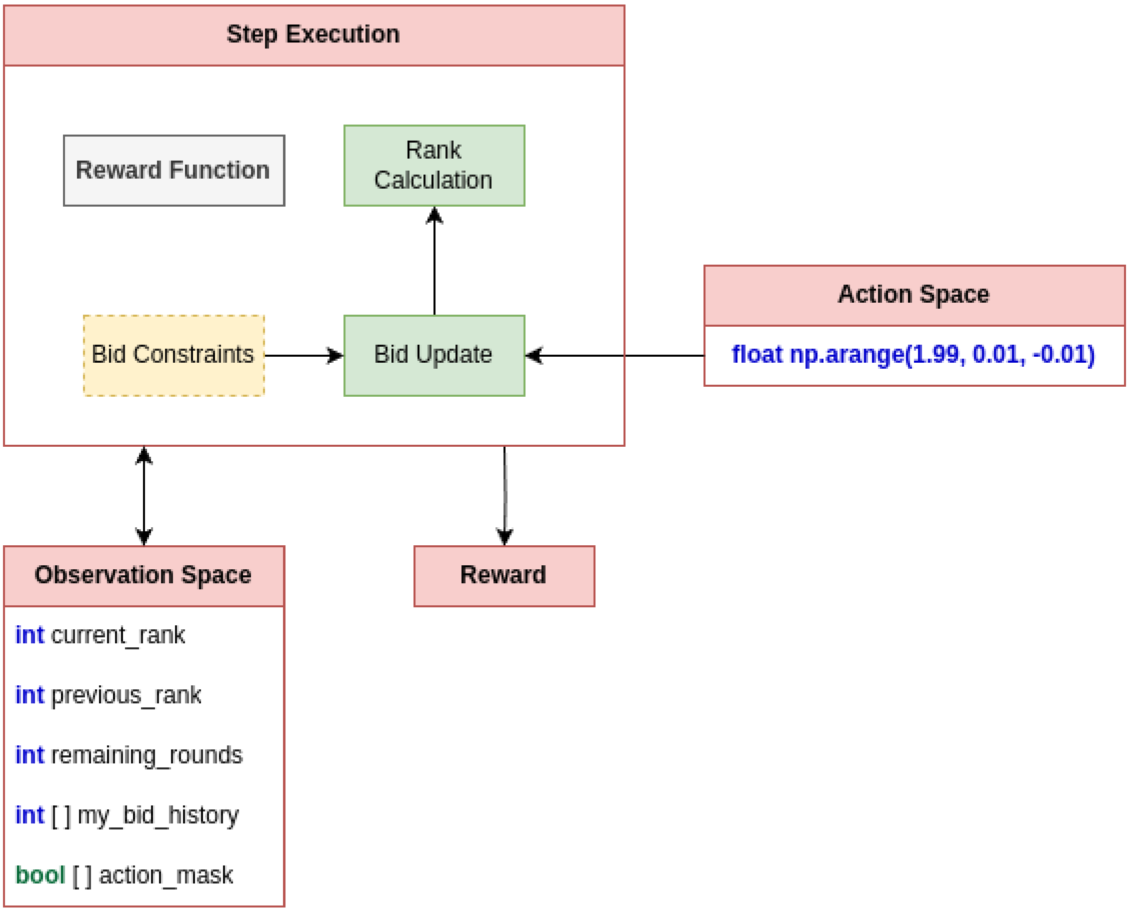

Figure 6: PettingZoo Environment

Action & Observation Spaces

The Action Space is structured as a discrete space representing bid adjustment options, defined as percentage reductions ranging from 1.99 to 0.01 in decrements of 0.01. Its purpose is to allow agents to incrementally lower their bids within constraints.

The Observation Space is structured as a dictionary, containing the current ranking, the previous ranking, the remaining auction rounds, each agent's bid history and the action mask. The first two are defined as Discrete values from 1 to the maximum number of agents. The remaining auction rounds are also indicated by a Discrete value, ranging from 0 to the maximum auction rounds. The bid history is defined as an array. Finally, the action mask is a vector indicating valid bid adjustments based on the current bid and constraints.

Bid Updates

Bid updates begin with an input dictionary of actions, such as {"provider_0": 0.50, "provider_1": 0.25}, where each agent specifies a bid adjustment. The process updates the current bid applying the chosen adjustment while respecting the minimum and maximum bid limits mentioned above. The updated bid is also recorded in the agent's bid history.

Rank Calculation & Reward Computation

Rank calculation occurs after all bids are updated. The environment sorts all current bids in ascending order, then assigns ranks where the lowest bid receives 1, the second-lowest 2, and so forth. The resulting rank array is stored, with current ranks updated and previous ranks retained for reward computation.

Reward Computation uses the reward function, which considers multiple inputs: the current ranking, the total number of agents, the current bid value relative to the bid constraints, the current round number and the total rounds. The objective is to encourage strategic bid lowering without excessive profit loss. The output is a scalar reward value per agent, included in the step output.

Step Execution

The step execution follows a structured workflow. It receives actions from all agents, updates bids and histories, calculates new ranks, computes rewards and observations, and increments the round counter until the final round. The output is a tuple containing observations and rewards per agent.

3.2.4.Strategic Blind Auction Design

The SPM’s blind auction mechanism establishes that resource providers in B-RAN iteratively refine their bids without direct access to competitors’ actual bid amounts. Instead, after each round of the multi-round process, participants receive only their relative ranking, preserving strategic uncertainty and creating a highly competitive pricing environment. This design, where the lowest final bid determines the winner, mirrors the principles of repeated games with imperfect public information, as articulated by Green and Porter [15]. In their theoretical models, players navigate environments where they must adjust strategies based on noisy or indirect signals, such as observed market prices or outputs, rather than having explicit knowledge of competitors’ actions. Similarly, in the SPM, bidders rely on ranking feedback as an imperfect, probabilistic indicator of market conditions, inferring competitive pressures to strategically lower their bids and drive prices downward over multiple rounds, ensuring dynamic and responsive pricing in a decentralized B5G network context.

By deliberately concealing bid details, the SPM effectively mitigates inefficiencies that plague traditional auction systems, particularly those stemming from collusion and price manipulation. In fully transparent bidding environments, participants can observe and potentially coordinate their strategies, risking tacit agreements that artificially inflate prices or distort market outcomes. The SPM’s blind structure counters these risks by forcing providers to make independent, strategic decisions based solely on their own bid and ranking, aligning with B-RAN’s blockchain-based framework. The design choices underpinning this mechanism, implemented in the reverse auction, reflect a deliberate effort to operationalize these theoretical advantages while meeting the SPM's goals of fairness, efficiency, and scalability.

3.3. Training Process

3.3.1.Policy

The agents are trained using PPO [16], which is an RL algorithm that balances exploration and exploitation while certifying stable policy updates. It builds on previous policy gradient methods such as Vanilla Policy Gradient (VPG) and Trust Region Policy Optimization (TRPO) [17], addressing their limitations.

VPG directly optimizes policy parameters by maximizing expected rewards. However, it suffers from high variance because it updates policy parameters using raw rewards without considering the relative importance of different actions. Additionally, since it lacks a mechanism to constrain policy updates, it may make overly aggressive changes that lead to instability in training.

To address these issues, TRPO introduces a trust region constraint, which limits the step size of policy updates to prevent drastic changes. This is achieved by enforcing the constraint: $KL\left(\pi_{\theta_{old}}\middle|\pi_{\theta}\right) \leq \delta$ where KL is the Kullback-Leibler divergence between the old and new policies, and δ is a small threshold that controls the allowed deviation. This ensures that updates are gradual and do not destabilize learning. However, TRPO requires solving a constrained optimization problem, which involves second-order derivatives and is computationally expensive.

PPO simplifies TRPO by replacing the complex constraint with a clipped surrogate objective, ensuring stable updates without requiring second-order derivatives. This prevents large policy shifts while maintaining sample efficiency, making PPO both powerful and easy to implement.

Mathematically, PPO optimizes the following objective function:

$$L(\theta) = E_t[\min(r_t(\theta)A_t, clip(r_t(\theta), 1-\varepsilon, 1+\varepsilon)A_t)]$$

where:

- t is the time step in a learning episode.
- $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$ is the probability ratio between the new and old policies.
- A_t is the advantage estimate, measuring how much better an action is compared to the expected return.
- ε is a hyperparameter that controls how much the policy is allowed to change per update.

By clipping the policy update, PPO prevents excessive divergence from the current policy, ensuring more stable and reliable learning.

The advantage function, denoted as A_t, or more accurately $A(s_t, a_t)$, is a crucial component in PPO, measuring how much better taking a specific action in a state is compared to the expected value of that state. It provides a way to determine whether an action was beneficial and should be reinforced or if it was suboptimal and should be discouraged. The advantage function is defined as $A(s_t, a_t) = Q(s_t, a_t) - V(s_t)$, where $Q(s_t, a_t)$ represents the expected cumulative reward of taking action a in state s at timestep t . By computing the advantage, PPO can decide whether a policy update should increase or decrease the probability of taking that action in the future.

To compute $A(s_t, a_t)$, the agent must estimate $V(s_t)$, the value function, which predicts the cumulative future reward expected from a given state. The value function serves as a baseline for advantage computation, helping stabilize policy updates by reducing variance. Mathematically, the value function is updated using: $V(s\quad) \approx E[R_t | s_t = s]$, where R_t represents the discounted sum of future rewards. Discounting is used to prioritize immediate rewards over distant ones, ensuring stability in training. PPO estimates the value function during training by minimizing a loss function involving the predicted value and the actual return. This structured value function estimation enables PPO to balance exploration and exploitation effectively, leading to more stable RL outcomes.

It is important to note that PPO does not directly estimate $Q(s_t, a_t)$. Instead, it relies on the value function and the advantage function to guide policy updates. The advantage function indirectly captures information about the value of actions without needing a separate Q-function approximation.

By refining these estimates over time, the agent can improve the reliability of policy updates, reduce variance, and ultimately make more consistent decisions.

3.3.2.Model

The model is a neural network designed to process observations while ensuring only valid actions can be selected. It builds on Ray RLlib's [18] TorchModelV2 and consists of a fully connected neural network (FCN) with hidden layers. The model processes the state representation as input and outputs action logits, defining the probability distribution over available actions.

The neural network is responsible for estimating both the policy (action selection) and the value function (state evaluation), which work together for effective learning. The model consists of two components:

- **Policy Network (Actor):** Outputs action logits, which determine the probability of selecting each action.
- **Value Network (Critic):** Estimates, the expected return of a state, which helps calculate the advantage function.

Network Architecture and Components

- **Input Layer:** Receives the encoded state representation (observation space) from the environment.
- **Hidden Layers:** Fully connected layers with non-linear activations to extract relevant features.
- **Output Layers:**
  - **Action Logits:** Used to sample actions from a probability distribution.
  - **Value Function Estimate:** Used to compute the advantage function and guide policy updates.

Note that the output layers share input and hidden layer weights and are trained together.

Action Masking

A critical feature of this model is the action mask, which prevents the policy from selecting invalid actions. Before selecting an action, the logits corresponding to invalid actions are set to a very low value, effectively preventing the agent from choosing them. This ensures that only valid actions retain their computed logits, while invalid actions are forced to near-zero probability. This mechanism allows the policy network to learn only from valid action spaces, improving efficiency and preventing rule violations.

On a higher level, action masking is used to enforce the min/max values constrain. They restrict the agent's actions so that each bid falls into the allowed range.

Loss Functions

The model is trained using two primary loss functions:

- **Policy Loss (Actor’s Loss):** Based on the PPO-clipped surrogate objective, updating the policy network to maximize expected rewards while maintaining stable policy changes.
- **Value Loss (Critic’s Loss):** Using the Mean Squared Error (MSE) to optimize the value function’s accuracy, achieving precise state value estimates.

By continuously refining these estimates over time, the agent enhances the reliability of policy updates, reduces variance, and ultimately achieves more consistent decision-making. This structured value function estimation enables PPO to balance exploration and exploitation effectively, leading to more stable RL outcomes.

With this architecture, action masking, and well-defined loss functions, the model enables the agent to efficiently learn in complex environments while adhering to constraints imposed by the environment.

3.3.3.Reward Function

The reward function is a key component of the RL framework, responsible for guiding the agent's decision-making process throughout the auction. It provides numerical feedback based on each action taken, enabling the model to iteratively refine its bidding strategy. By systematically evaluating the outcomes of different bidding decisions, the agent learns to optimize its performance within the constraints of the auction environment [19].

An effective reward function must balance competitiveness and sustainability. If a function rewards only immediate auction wins, the agent may adopt overly aggressive bidding strategies, consistently lowering its bids to the minimum allowable value. This could lead to unsustainable market conditions where bids are too low to be practical. Conversely, if the function encourages overly conservative bidding, the agent may avoid competitive pricing altogether, resulting in fewer successful bids. A well-structured function must allow the agent to compete effectively while maintaining a viable pricing strategy, optimizing for long-term participation rather than isolated wins.

The reward function must take input from the observation space, which encapsulates the relevant auction environment data, including the agent's bid, the minimum and maximum bid constraints, the number of competing agents, and the agent's current ranking within the auction. These inputs allow the function to compute rewards dynamically, ensuring that feedback is shaped by the real-time state of the auction rather than static rules. By integrating multiple contextual factors, the function ensures that the agent's learning process is grounded in the actual auction mechanics, allowing it to adapt to different competitive scenarios over time.

In multi-agent environments, reward structures must account for both individual and relative performance. In an auction setting, simply rewarding an agent for placing a bid is insufficient; the function must also consider bid placement relative to competitors, strategic positioning over multiple rounds, and adaptability to changing conditions. The reward function must provide sufficient granularity, allowing the agent to distinguish between incremental optimizations rather than treating all outcomes as binary wins or losses. Without fine-grained distinctions, the agent might struggle to differentiate between a near-optimal bid that narrowly lost and a poorly placed bid that was never competitive.

Granularity is necessary for incremental improvements in decision-making. If the reward function provides only extreme signals, such as full reward for winning and none for losing, the agent may fail to recognize useful bidding behaviours that contribute to long-term success. A well-structured function provides graduated feedback, enabling the agent to adjust its bidding with precision rather than abrupt shifts [20] [21]. This facilitates smooth learning progression, enabling the agent to refine its bidding strategy across multiple auction rounds.

Moreover, a granular reward structure prevents exploitative behaviours that could emerge from a simplistic reward signal. For example, if an agent is rewarded solely based on winning without considering bid efficiency, it may adopt a reckless bidding strategy that disregards market stability. By incorporating detailed reward signals that reflect ranking, bid efficiency, and long-term adaptability, the system enables the agent to develop a more strategic and calculated approach to bidding.

The implemented reward function is designed to incentivize competitive yet structured bidding by evaluating multiple factors that influence auction outcomes. The reward is positively correlated with the agent’s rank in the auction, meaning that achieving a higher placement results in a better reward signal. This encourages the agent to optimize its bids over time, assuring that it remains competitive against other participants. However, simply winning an auction is not enough—the function also accounts for how well the agent performs relative to others, making it sensitive to fine-grained differences in bid positioning.

Additionally, the function penalizes bids that are placed too close to the lowest possible bid, discouraging excessive underbidding that could destabilize auction pricing. Instead, it encourages bidding strategies that balance competitiveness with profitability, guiding the agent toward more optimal decision-making across multiple auctions. The reward also scales dynamically over multiple rounds, meaning that an agent’s actions early in the auction process influence its long-term reward trajectory. This allows the model to avoid focusing solely on short-term success, instead learning to adjust its bidding behaviour strategically over extended training.

By structuring the reward function in this way, the system provides granular feedback that helps the agent refine its approach with increasing precision. Instead of rewarding only binary outcomes—such as whether the agent won or lost—the function considers rank positioning, bid placement, and long-term adaptability, allowing the model to develop a well-calibrated bidding strategy. This certifies that the agent can successfully navigate competitive auction environments while maintaining efficiency and sustainability.

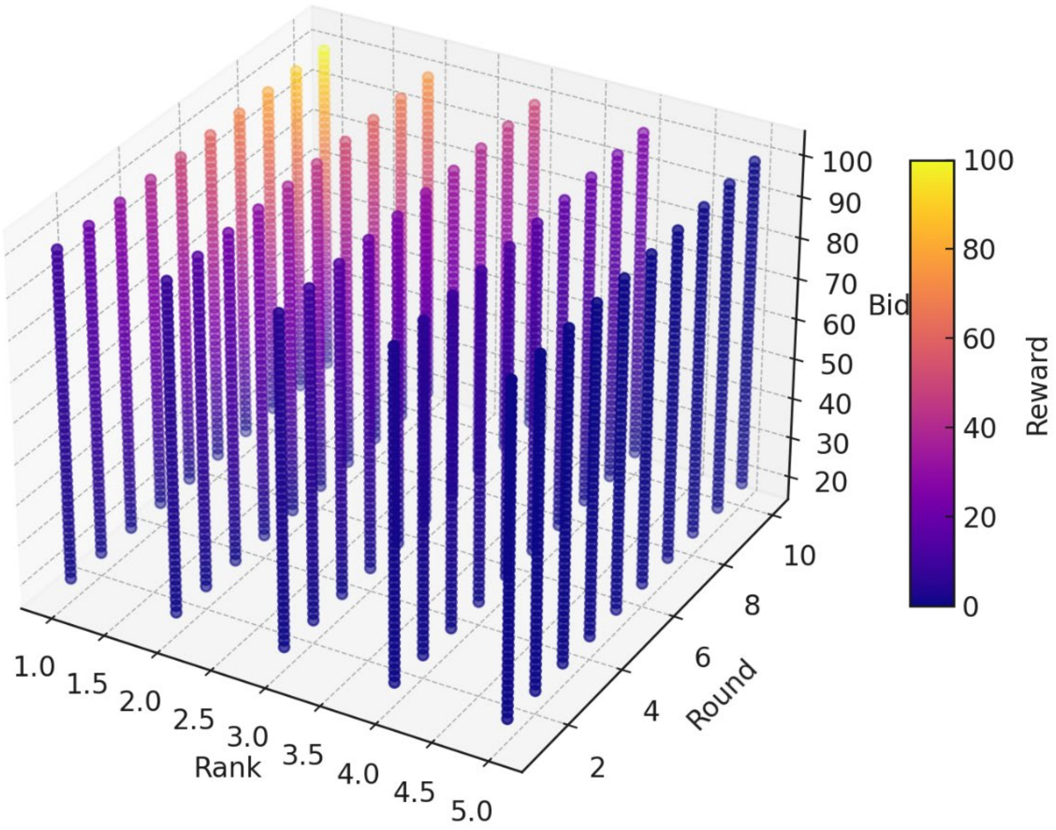

Figure 7: Reward Function

The 3D scatter plot (Figure 7) visualizes how the reward function behaves based on an agent's rank, round, and bid amount, with the colour representing the computed reward. This helps us understand how different variables interact with the reward function within the auction-based environment.

From the function, we see that reward is influenced by three primary factors:

1. **Rank in the auction:** Higher-ranked agents (lower numerical rank) receive a greater proportion of the reward. This is because the function scales rewards using a normalized rank factor, encouraging agents to compete for better placements.
2. **Bid amount relative to the maximum:** The function penalizes bids that are too close to the minimum allowed bid. Instead, rewards are positively correlated with bids closer to the maximum bid. This discourages overly conservative bidding and promotes competitive yet profitable strategies.
3. **Progression through rounds:** The function scales rewards based on the round number, meaning actions taken in later rounds yield higher rewards than those in earlier rounds. This encourages long-term strategic bidding rather than only focusing on immediate gains.

Based on the graph (Figure 7) we can say that:

- Higher-ranked agents (lower x-values) generally receive higher rewards if they bid competitively, as indicated by the colour gradient.

- The reward increases over rounds (y-axis), which aligns with the function's behaviour of scaling rewards dynamically based on the current round. Agents are incentivized to stay competitive throughout multiple rounds.
- Bids closer to the maximum (higher z-values) result in higher rewards, as the function explicitly rewards proximity to the maximum bid. This discourages low-ball bidding strategies.

The colour mapping enhances these insights, making it clear that the highest rewards occur for higher-ranked agents, in later rounds, who bid closer to the maximum bid. Conversely, agents that bid too low, especially in early rounds, receive minimal rewards. This structured approach allows the system to foster competitive and well-balanced bidding behaviour over time.

3.3.4.Training Environment Parameters

To make sure that the RL agent effectively adapts to a diverse set of auction scenarios, the training process was structured with a variety of carefully chosen environmental, training, and evaluation parameters. These parameters influence the agent’s ability to generalize, optimize bidding strategies, and compete effectively under different market conditions. Table 1 summarizes some of the key training environment parameters.

Table 1: Training Environment Parameters

Parameter	Description
Number of Providers	Randomly chosen between 2 and 10 (uniform distribution).
Minimum Bid Limit	Each provider's minimum bid is randomly selected between 20 and 40.
Maximum Bid Limit	Each provider's maximum bid is randomly selected between 80 and 100.
Auction Rounds	Each auction runs for 10 rounds, allowing for strategic bidding adjustments.

These parameters establish the constraints within which the agent must operate. By randomizing the number of providers and their respective bid limits, the training environment introduces variability, ensuring that the agent does not overfit to a static market. Instead, it must learn to adapt to different auction conditions, optimizing its bids relative to competitors with diverse constraints. Table 2 summarizes some of the key training configuration parameters.

Table 2: Training Configuration

Parameter	Value	Purpose
Batch Size	512	Controls how many experiences are used per training step, affecting learning stability.
Entropy Coefficient	0.9 (decaying to 0.001)	Encourages exploration early in training, shifting to exploitation over time.

Learning Rate	0.001	Determines how quickly the agent updates its policy. A lower value stabilizes learning.
Training Steps	500	Number of iterations the agent undergoes to refine its strategy.

A key aspect of the training setup is the entropy coefficient, which plays a crucial role in balancing exploration and exploitation when sampling actions. In RL, an agent selects actions based on a probability distribution over possible choices. Higher entropy means the probabilities are more evenly spread, making the agent more likely to sample a wider range of actions. Lower entropy means the probabilities are more concentrated, making the agent more likely to select the highest-rated action with little randomness.

Early in training, the agent is encouraged to explore a diverse set of bidding strategies, preventing it from prematurely converging to a suboptimal policy. This is achieved by setting a high entropy coefficient (0.9), which increases the randomness of action selection, allowing the agent to test different bid placements. This phase is critical in helping the model discover effective bidding patterns that might not be immediately obvious.

As training progresses, the entropy coefficient gradually decreases, reducing randomness and shifting the focus toward the exploitation of learned strategies [22]. By the final stages of training, entropy is minimal (0.001), ensuring that the agent consistently follows the best bidding strategies it has developed over time rather than randomly selecting suboptimal actions. This decay schedule prevents the agent from becoming stuck in local optima too early, while also ensuring that it ultimately converges to a stable and effective policy.

By carefully structuring these training parameters, the model is able to learn effective bidding strategies that generalize across different auction conditions. These settings ensure that the agent not only learns how to win auctions efficiently but also how to balance competition and profitability, leading to a robust and adaptable bidding strategy.

3.3.5.Self-Play

Self-Play is a fundamental mechanism used to train the RL agents in this auction environment [23]. By competing against themselves in multiple simulated auctions, agents iteratively refine their strategies without requiring external expert demonstrations. This approach is particularly useful in strategic games and economic simulations, where an optimal strategy emerges through repeated interactions between agents [24].

In self-play, multiple agents participate in training rounds where each agent's behaviour influences the learning of others. This creates an evolving competition, ensuring that strategies remain adaptive rather than static. The multi-round blind reverse auction format benefits from self-play as agents must continuously adjust their bidding strategies based on observed competition dynamics rather than predefined rules.

Through repeated self-play iterations, the agents explore different bidding behaviours, learning which strategies yield the highest probability of winning while maintaining efficiency. Over time, this self-play approach allows the agents to develop sophisticated bidding strategies, balancing competitive pricing with strategic adaptation to market conditions.

Ultimately, self-play enables agents to improve autonomously, refining their competitive behaviour through repeated exposure to the auction dynamics. This process ensures that the model remains adaptive, responding effectively to a wide range of auction conditions without requiring manually crafted strategies.

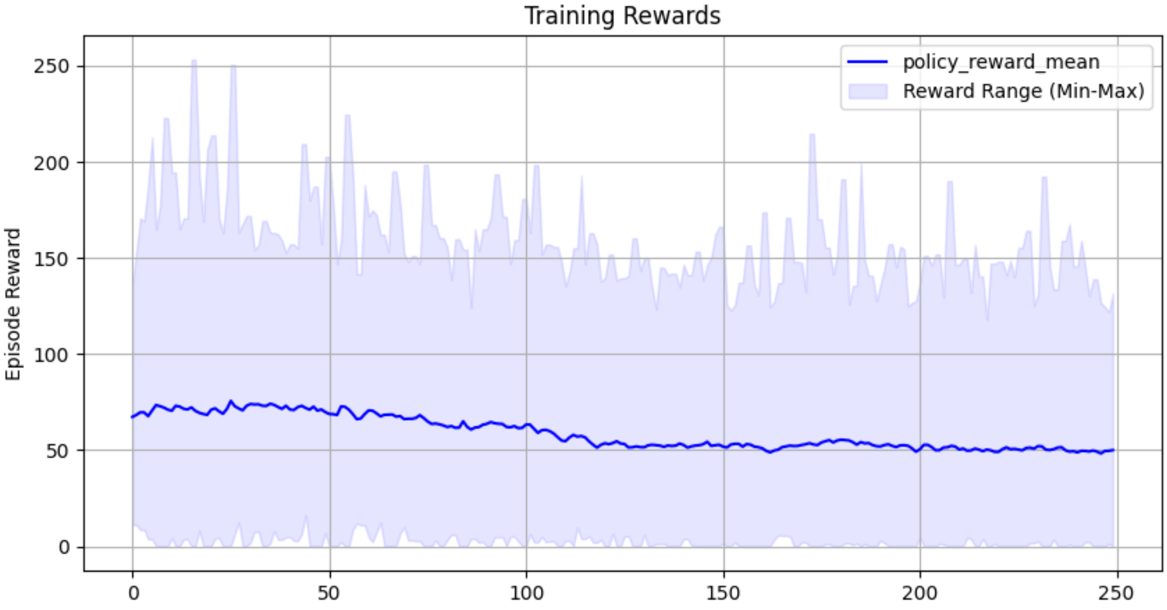

Figure 8: Training Rewards

Figure 8 visualizes the episode rewards during training. The x-axis represents the training steps, while the y-axis shows the episode reward. The blue line represents the average reward. The shaded regions represent the maximum and minimum episode rewards for this training step.

The average episode reward function during training, instead of increasing over time, shows a gradual decline, which might initially seem counterintuitive. Given that the reward function ranges from 0 to 1000 per episode, one might expect the agent to optimize its policy to achieve higher rewards.

However, in the self-play training framework, the decreasing reward trend can be attributed to the evolving competition between agents [25].

At the beginning of training, agents are still exploring the environment, their actions are largely unoptimized. This results in higher variance in rewards, as some actions may lead to significantly better outcomes than others. During this phase, some agents may achieve high episodic rewards due to random favourable conditions rather than strategic superiority.

As training progresses, agents improve their strategies and the overall competition intensifies, leading to tighter bid margins and lower overall rewards [26]. In a reverse auction, where the goal is to place the lowest possible bid while still winning, agents gradually learn that bidding lower can secure a win but also reduces the total reward received [27]. This aligns with the observed decline in the average reward, as the model is shifting toward a more competitive environment where optimal bids approach the minimum necessary to win rather than maximizing individual episodic rewards.

The maximum rewards (shaded region's upper bound) remain highly volatile throughout training. This suggests that some episodes still produce significantly higher rewards, possibly due to occasional exploration behaviours, fluctuations in opponent strategies, or temporary instability in policy updates. However, as self-play continues, the frequency of such high-reward episodes decreases, reflecting that agents are settling into equilibrium strategies where excessive overbidding or underbidding becomes rare.

Finally, the training process converges as agents stabilize their bidding strategies. The decline in reward values suggests that the model is not focused on maximizing individual rewards but rather on optimizing its bidding to win the auction efficiently. This behaviour is a direct result of self-play dynamics, where agents continuously adjust their strategies against evolving opponents, leading to progressively refined decision-making and a stable competitive environment.

3.4. Load Balancing & Customer Distribution

In competitive reverse auctions, maintaining a balance between fair pricing and efficient resource allocation is crucial. Without proper regulation, providers with lower capacity usage might struggle to secure clients, while those with high-capacity utilization may continue to win auctions despite being heavily loaded. This imbalance can lead to service degradation, price inflation, or inefficiencies in the bidding process. To address this challenge, a dynamic load balancing mechanism is implemented to adjust provider participation based on real-time availability, ensuring that the auction system remains fair, efficient, and beneficial to both providers and customers.

The load balancing mechanism is designed to dynamically adjust providers' pricing thresholds based on their current capacity without ever lowering the minimum bid they have set. Instead, the function increases the minimum bid by varying amounts according to the provider's capacity usage. This ensures that each provider's baseline pricing is preserved while still creating a penalty for those who are more occupied. In this way, the system guarantees that no provider is forced below their own acceptable minimum, maintaining fairness and respecting each provider's original pricing strategy.

By selectively raising the minimum bid for providers with higher resource usage, the system indirectly discourages them from winning too many auctions when they are already under heavy load. Providers with low-capacity utilization are not penalized, allowing them to remain competitive in the bidding process. This selective adjustment is crucial because it protects the customer by steering the auction toward providers who have the availability to deliver quality service without compromising their performance due to overcommitment.

In addition to adjusting the minimum bid based on provider availability, the mechanism also modifies the initial bid to further enhance the chances of providers with greater available resources. Just as increasing the minimum bid makes highly occupied providers less competitive, the initial bid adjustment ensures that providers with more available capacity start the auction at a more favourable position. This increases their likelihood of winning while maintaining fair pricing dynamics. By doing so, the system naturally steers the auction toward providers who can accommodate more clients, ensuring that demand is matched with the providers best suited to fulfil it efficiently.

A key benefit of this approach is that it prevents the artificial inflation of market prices while maintaining competitive bidding dynamics. Without this mechanism, providers with limited availability could still win bids at low prices, despite not being in a position to effectively serve new clients. By ensuring that underutilized providers remain competitive, the system keeps prices fair for customers while reducing the risk of service bottlenecks caused by overburdened providers winning too many auctions.

Overall, the strategy behind this load balancing and customer distribution approach is to align provider participation with their actual resource availability without reducing their baseline bids. The careful increase in minimum bids for overloaded providers guides the auction toward a fairer and more efficient allocation of work. This ultimately benefits customers by increasing the likelihood that they will be matched with providers who can meet their service needs promptly and effectively, preserving service quality and market stability.

4. Integration into the NANCY Framework

The SPM is considered in the inter-operator domain flow, which is explained in detail in deliverable D5.2 “Security and Privacy Distributed Blockchain-based Mechanisms”, in order to identify the most suitable service to be offered to a client from those registered in the common marketplace. This flow is executed when an operator needs a new service (from another operator) to attend to their clients' demands.

This flow is managed by the blockchain-based marketplace, which receives the search request of a new service with specific specifications (quality level, minimum resources, location etc.) from the original operator. The marketplace selects the most suitable services according to the defined specifications and calls the SPM to identify the most suitable one, along with the most suitable price.

This communication between the blockchain-based marketplace and the non-blockchain-based SPM happens through the Smart Pricing Oracle, which handles the communication between the marketplace and the SPM, as shown in Figure 9.

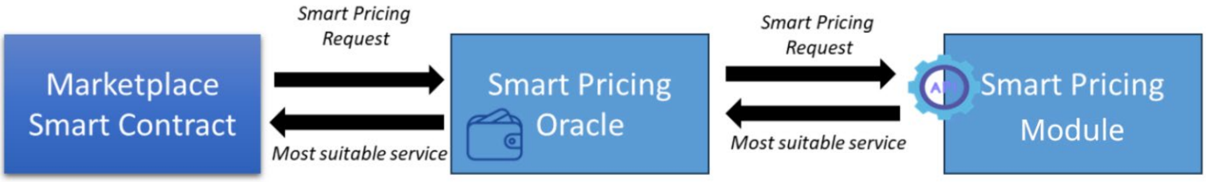

Figure 9: Marketplace and SPM Communication via an Oracle

The oracle generates an HTTP request to the *price_calculation* endpoint of the SPM, indicating in “data” the list of services suitable according to the specifications. In the current implementation, the information related to each suitable service the marketplace shares is limited to:

- *Service_id*: it refers to the service identifier.
- *Provider_id*: ir refers to the operator identifier offering the service.
- *Minprice*: it refers to the minimum acceptable price for the service (to be considered as reference by the SPM).
- *Maxprice*: it refers to the maximum acceptable price for the service (to be considered as reference by the SPM).
- *Availability*: it refers to the percentage of available resources of the associated provider.

This information could be updated if needed for future more complex SPM implementations, as long as the required information by the SPM is available in the marketplace.

Taking this into consideration, the request from the Smart Pricing Oracle of the marketplace to the SPM, considering three suitable services according to the specifications, is as follows:

```
curl          --request          POST          --url          https://sp-mock-
nancy.cybersec.digital.tecnalia.dev/price_calculation header 'Content-Type:
application/json'                           --data                          '{
                                 "services":                                  [
                                                                              {
       "provider_id":"637223108d14e08b3386afdddbdc2f2bff22041c1ac9c6942364c
3d6a22e9915",
                                  "minprice":                               30,
                                 "maxprice":                               120,
       "service_id":"0355534650b3d3b5fe8d35fcb4a91bf175fab6a505347667381674
25294bf5f3",
      "availability":               30                                     },
                                                                              {
       "provider_id":"637223108d14e08b3386afdddbdc2f2bff22041c1ac9c6942364c
3d6a22e9915",
                                  "minprice":                               35,
                                 "maxprice":                               115,
       "service_id":"212dee4384ec40d35c6af8adec5c7dab40cd481f206777c156b14c
103f0cf707",
      "availability":               30                                     },
                                                                              {
       "provider_id":"637223108d14e08b3386afdddbdc2f2bff22041c1ac9c6942364c
3d6a22e9915",
                                  "minprice":                               40,
                                 "maxprice":                               112,
       "service_id":"218dc57aa7a52edef9510f46840f8fa6e485bb8ffad53665674c90
64b28b1d74",
      "availability": 30     }
]
}'
```

The Smart Pricing operation happens as explained in Section 3, currently generating an HTTP response indicating in “services” the most suitable service from those provided by the marketplace in the request. In the current implementation, the provided information is limited to:

- *Service_id*: it refers to the service identifier.
- *Provider_id*: ir refers to the operator identifier offering the service.
- *price*: it refers to the most suitable price for the service according to the existing competence.

Taking this into consideration, the response of the SPM to the Smart Pricing Oracle is as follows:

```
{
  "services":                                                               {

"service_id":"2cfe938a2b83f0c8b8178e1e399ce771b03854bc0a68744c359b48102e97
086d",
"provider_id":"317573acebab0c914649af2c4002e2ea6a6cd7587cb165d34f494111cfe
717ce",
    "price":                          74.12                              }
}
```

The marketplace receives this information, processes it and considers the selected service as the “winning” one. The marketplace can now send all the registered information about the winning service to the Digital Agreement Creator component for the new SLA generation to be signed by the new operator identified as “provider_id”.

More details about the inter-domain flow can be read in D5.2 “Security and Privacy Distributed Blockchain-based Mechanisms”.

5. Performance Evaluation

In this test, each player's maximum and minimum allowed bid is randomly assigned, making the winner of each auction unpredictable. However, the frequency with which each player participates in an auction varies based on the total number of bidders. Players 1 and 2 always participate because the number of bidders ranges from 2 to 10. Player 3 appears slightly less often, and this trend continues, with each subsequent player participating in fewer auctions. Player 10, for instance, only joins when the number of bidders is exactly 10, which occurs in only 1 out of 9 auctions. Furthermore, even when Player 10 does join, they have just a 1 in 9 chance of winning, making their overall likelihood of winning an auction quite low. This pattern follows a truncated harmonic series distribution, where players appearing in fewer auctions have proportionally fewer chances to win.

Figure 10 represents the distribution of auction wins among different providers in the multi-round blind reverse auction. The x-axis lists the providers (agents), while the y-axis represents the number of wins each provider secured in the simulations. The blue bars indicate the actual number of wins for each provider based on the simulation results. The red line represents the theoretical distribution, which serves as an expected reference for comparison. This theoretical distribution resembles a truncated harmonic series, as explained above. As expected, the two distributions follow a similar trend.

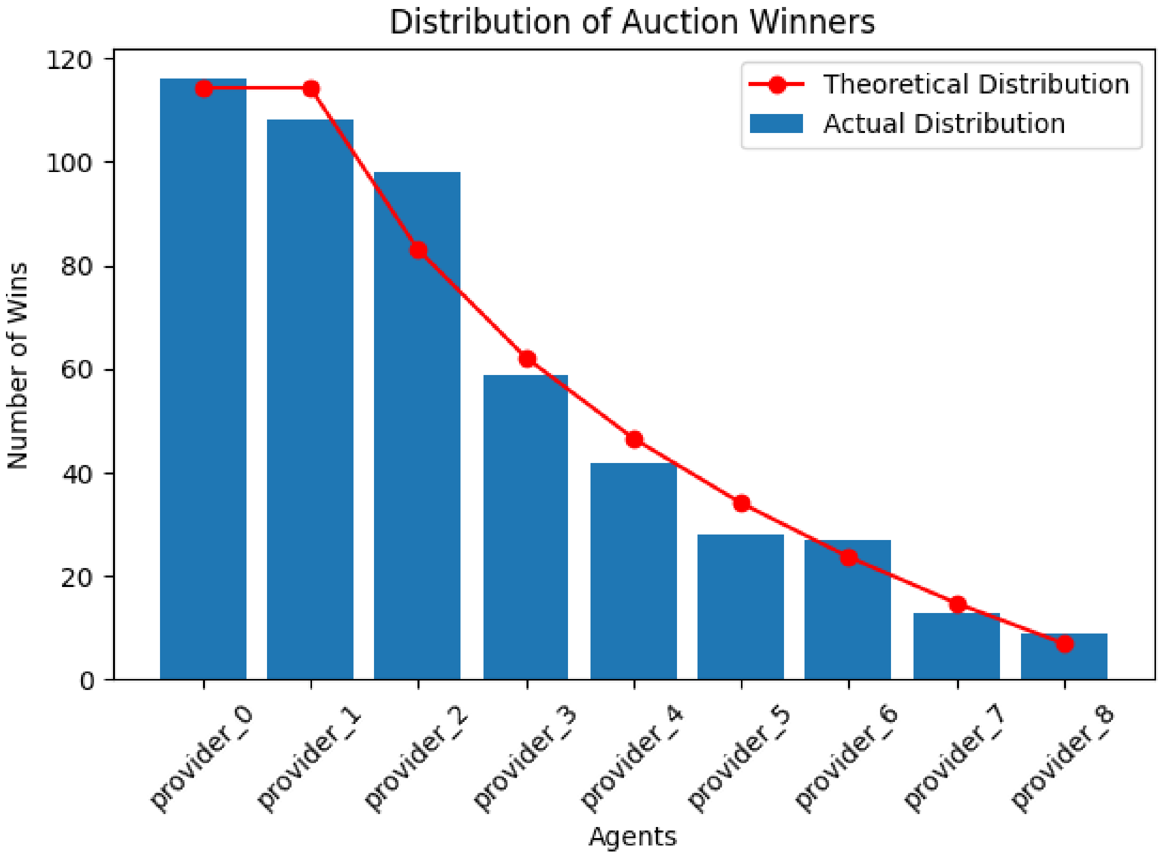

Figure 10: Auction Winner Distribution in Tests

5.1. Results from Testing and Simulations

5.1.1. Test Parameters

To evaluate the performance of the auction system, a series of simulations were conducted using randomized parameters to emulate every bidding environment (Table 3). The random parameters were used to ensure a known distribution of different inputs, allowing for a comprehensive analysis of bidding behavior across varied conditions. The results from these simulations provide insights into the behavior of the RL agent, helping to evaluate its effectiveness, strategic decision-making, and adaptability in a competitive environment.

Table 3: Test Parameters

Parameter	Description
Number of Providers	A randomly chosen integer between 2 and 10 (uniform distribution).
Minimum Bid Limit	Each provider was assigned a minimum bid limit randomly selected between 20 and 40 (uniform distribution).
Maximum Bid Limit	Each provider was assigned a maximum bid limit randomly selected between 80 and 100 (uniform distribution).
Current Bids	The initial bid for each agent was randomly set within their defined Minimum and Maximum Bid Limits using a uniform distribution.
Rounds	The auction process had 10 rounds.

5.1.2. Results and Analysis

The tests generated results across 500 auctions, leading to the key findings of Table 4.

Table 4: Simulation Statistics

Metric	Result	Explanation
Average Winning Bid – Lowest Possible Difference	11.02	This metric represents the average difference between the winning bid and the lowest possible bid allowed for each winning agent. A difference of 11.02 suggests that the final prices in the auction were generally close to the Minimum Bid Limit, indicating competitive but not excessively aggressive bidding.
Average Rank 1 vs. Rank 2 Bid Difference	3.61	This represents the average gap between the lowest (winning) and the second-lowest bid. A relatively small difference of 3.61 indicates that the bidding process was competitive, with the top two agents often placing very similar final bids.
Percentage of Lowest Starting Price Wins	32.20%	This metric highlights how often the agent with the lowest starting bid ultimately won the auction. A success rate of

		32.20% suggests that while an initial low bid provides an advantage, other strategic factors (such as incremental bidding behaviour) play a more crucial role in determining the winner.
Percentage of Lowest Minimum Bid Limit Wins	69.20%	This represents how often the agent with the lowest Minimum Bid Limit won the auction. A success rate of 69.20% indicates that having a lower bidding range significantly improves the likelihood of winning. However, since this is a multi-round blind reverse auction, having the lowest Minimum Bid Limit does not always guarantee a win. The result reflects a balanced system where strategic bidding plays a crucial role.
Percentage of Providers Bidding at Minimum Bid Limit	0.00%	No providers consistently bid at their Minimum Bid Limit (or very close to it). This suggests that bidders are generally cautious and strategic, avoiding bids that might be too low to secure a win.
Percentage of Providers Bidding at Maximum Bid Limit	0.00%	Similarly, no providers bid at their Maximum Bid Limit (or very close to it). This indicates that participants do not blindly bid at their upper limit but rather adjust their bids dynamically to remain competitive.

The results reinforce that while having a lower Minimum Bid Limit increases the likelihood of winning, it does not guarantee success due to the blind nature of the auction. This ensures that the auction mechanism remains fair and competitive.

Winning Price Distribution Analysis

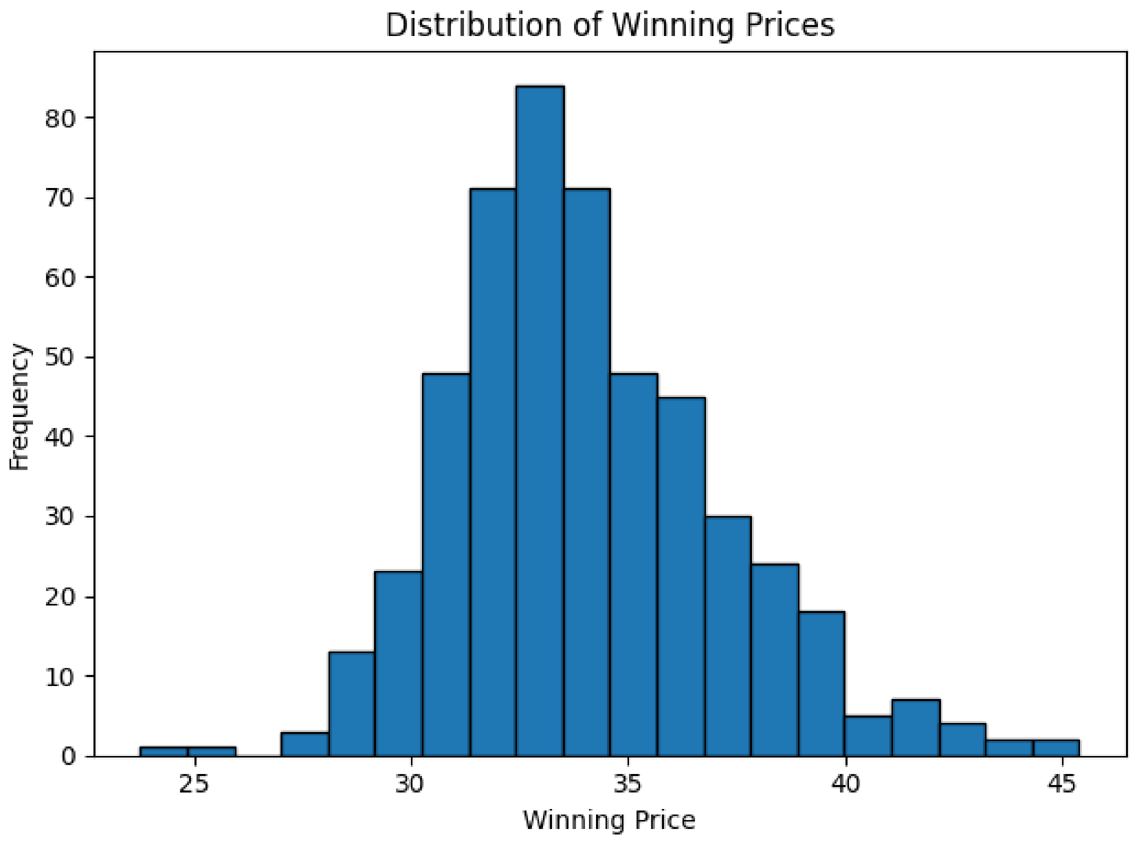

Figure 11: Winning Prices Distribution

The histogram (Figure 11), illustrating the distribution of winning prices, shows that the auction system is functioning effectively. The distribution follows a bell-shaped curve, centering around a mean winning price in the low-to-mid 30s.

The distribution follows a normal-like pattern where most winning prices cluster around an average value of 32-35. This indicates a competitive and stable market, and the absence of extreme outliers suggests rational and strategic bidding. The spread of winning prices demonstrates that providers engage in strategic bidding rather than consistently bidding at extreme values, highlighting a dynamic and adaptive auction process.

Very few winning prices fall below 25 or above 45, showing that providers respect their bidding constraints and avoid unrealistic pricing. This ensures a well-balanced competition and prevents auction manipulation. Additionally, the distribution suggests efficient price discovery, with winning bids naturally stabilizing around competitive values. If the distribution were heavily skewed, it could indicate inefficiencies or weaknesses in agent strategies.

The results indicate that the auction mechanism fosters fair and competitive market dynamics where no single strategy dominates. Providers adjust their bids dynamically rather than defaulting to fixed rules, improving the realism of auction behavior. Additionally, the fair spread of bids suggests that providers are competing without artificially inflating or depressing prices.

Explanation of Average Percentage Price Difference Per Round

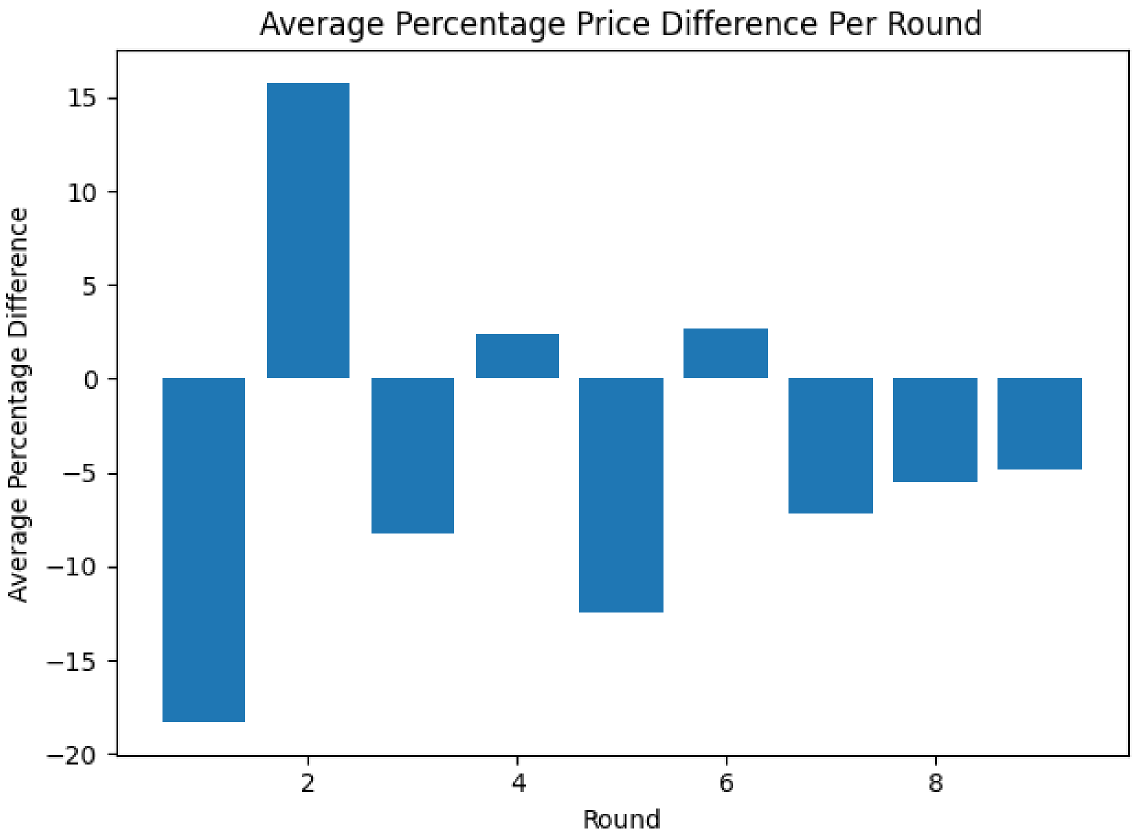

Figure 12: Average Agent Behavior

Figure 12 represents the average percentage price difference per round, indicating the average action taken by agents in the auction process. It effectively showcases how RL agents behave in a multi-round blind reverse auction, making strategic bid adjustments to maximize their chances of winning while adapting to competition. The x-axis represents the rounds in the auction, while the y-axis shows the average percentage difference in price adjustments made by the agents.

In the early rounds, there are significant fluctuations, with an initial sharp drop in bids followed by a large positive adjustment in round two. This suggests that agents initially reduce their bids aggressively but then make upward corrections, likely to test competitive behavior or respond to constraints. As the auction progresses, bid adjustments become more refined, with smaller alternating increases and decreases in the middle rounds, reflecting strategic fine-tuning. A notable drop in round five indicates a moment where agents may have collectively reduced bids more aggressively. In the later rounds, we see a consistent negative trend, meaning that agents are progressively lowering their bids to remain competitive, aligning with the expected strategy in a reverse auction. This pattern demonstrates that agents are learning and adapting rather than bidding randomly, ensuring competitive bidding dynamics while preventing static bidding or collusion. Ultimately, this behavior leads to an efficient auction process, where agents gradually reach their optimal bid while remaining responsive to market pressures.

5.1.3.Example Auction

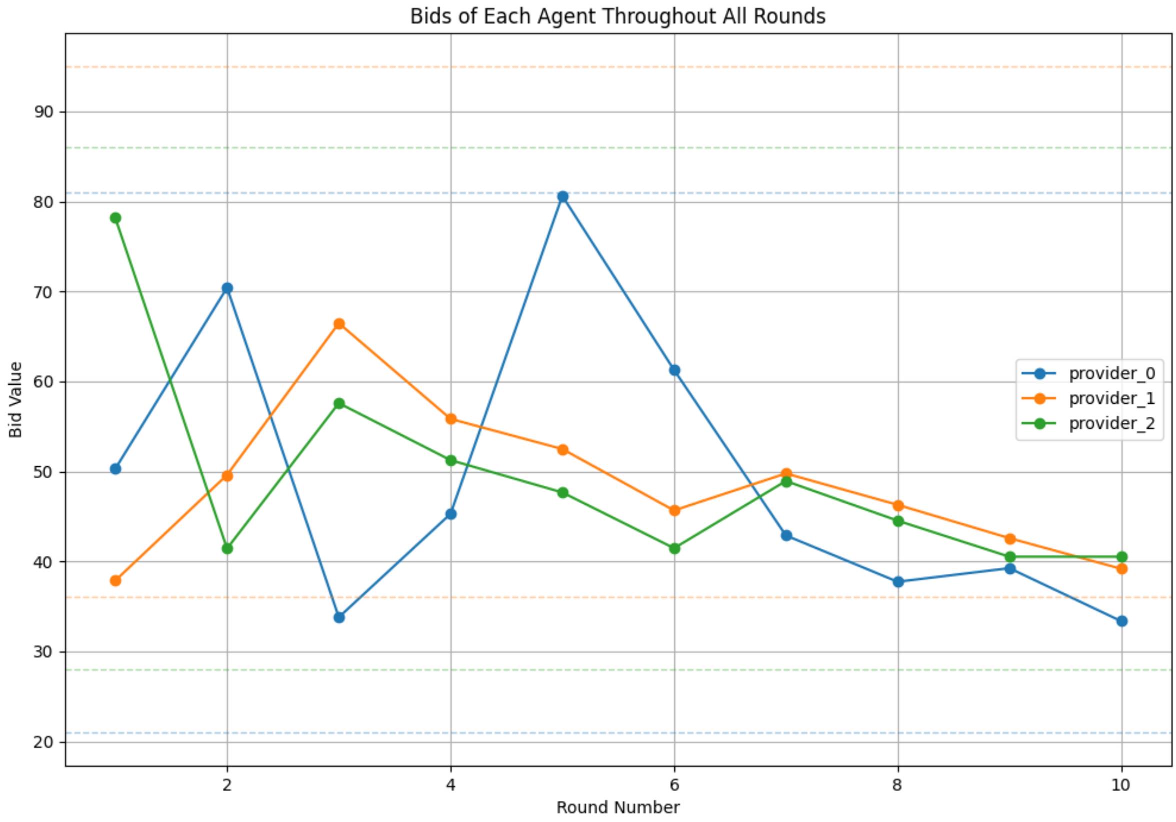

Figure 13: Example Auction

Figure 13 visualizes an example auction where multiple agents (providers) participate across 10 rounds, with the x-axis representing the round number and the y-axis showing the bid value submitted by each agent per round. The dotted lines at the top and bottom represent the maximum and minimum allowed bid constraints for each agent. The initial bid is randomly selected within these constraints. The blue line represents provider_0, following its trajectory can offer insights into possible bidding strategies, even though the exact reasoning behind its actions remains unknown due to the black-box nature of the neural network controlling the agent.

At the start of the auction, provider_0 places a moderate bid, followed by a sharp increase in round 2, which could suggest an attempt to test competition or a misjudgment of the bidding landscape. However, in round 3, provider_0 drastically lowers its bid, possibly reacting to market pressure or recognizing that its previous bid was too high. The following rounds exhibit a series of erratic shifts, with another bid increase in round 5, before the agent transitions into a more consistent downward trend from round 6 onward.

The later rounds suggest that provider_0 gradually adopts a more competitive bidding approach, where bids continuously decrease in an incremental and controlled manner. While we cannot determine with certainty why the agent made these choices, it appears to be engaging in progressive price-cutting, likely as a response to competition and the nature of the reverse auction format. The earlier, more erratic bidding could indicate an exploratory phase, where the agent was adjusting to the auction environment, while the later rounds suggest more refined decision-making, possibly aiming to secure a win with the lowest viable bid.

It is worth noting that although provider_0 had the lowest minimum allowed bid, it did not win the auction by simply bidding the absolute lowest price. Instead, its bidding strategy appears to have balanced competitiveness with profitability, choosing to stay within a competitive range rather than immediately lowering its bid to the minimum possible value. This suggests that the agent factored in the need to win the auction while still securing a reasonable profit margin. Even though it had the flexibility to bid much lower, it did not adopt an overly aggressive underbidding strategy, indicating a more nuanced decision-making process rather than a purely price-minimizing approach.

Since the agent operates as a black-box neural network, we can only speculate on the reasoning behind its bidding behavior. However, the pattern observed—initial exploration, abrupt corrections, and eventual stabilization into strategic, controlled bidding—is consistent with what we might expect from an RL model that is optimizing towards a winning strategy. This gradual adaptation highlights the agent’s potential ability to learn from prior rounds, adjust dynamically, and improve its competitiveness over time, even if the specific decision-making process remains opaque.

5.2. Performance Benchmarks

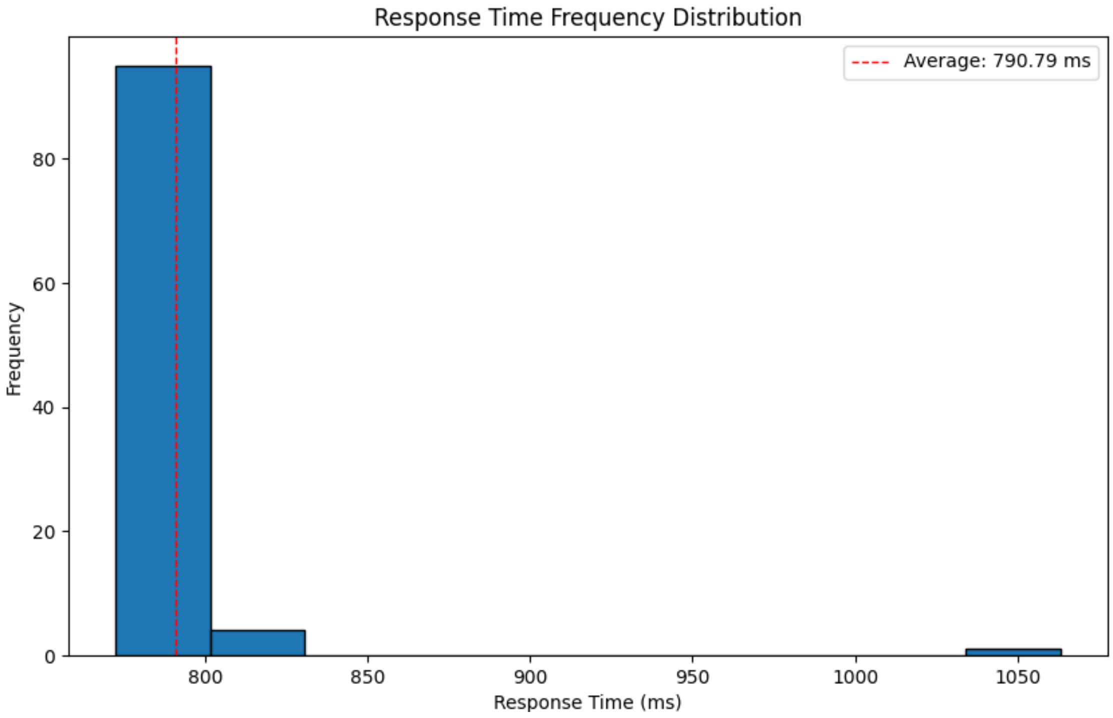

Figure 14: SPM Response Time

This test involved 100 sequential requests to the API. There were no failures, each request completed with 100% success rate. Figure 14 indicates that the response times are consistently good for NANCY research project, with most responses falling within the 700–800 ms. range and an overall average of about 790.79 ms. This reliability and performance, especially with a 100% success rate, strongly support the API’s effectiveness for the project’s needs.

6. Business Perspective

6.1. 6G Networks: A Multi-Stakeholder Ecosystem

The evolution towards 6G networks represents a paradigm change in the structure and functioning of the telecommunications ecosystem, as it emerges as a highly collaborative, decentralised and flexible environment. In this context, a multi-stakeholder scenario emerges as a key model, in which any stakeholder can assume the role of provider or consumer of services or resources, depending on each situation within this flexible and dynamic landscape. This approach paves new economic opportunities, fosters innovation, and redefines the way networks are conceived, deployed and operated.

From the providers' perspective, the multi-stakeholder model introduces several strategic advantages that optimise services supply and business profitability:

- **Greater flexibility and agility**: This model allows service providers to adapt swiftly to evolving market demand without relying solely on traditional network operators. By leveraging a more decentralised approach, providers can dynamically scale services, respond to demand fluctuations in real time and adapt their offerings to meet specific customer needs, thereby improving the user experience and optimising resource allocation, ensuring more efficient service delivery.
- **Cost reduction and resource optimisation**: By facilitating the sharing of infrastructure and network capabilities among multiple stakeholders, this model significantly reduces operational expenses. This cost-effectiveness reduces barriers to entry for new players in the 6G ecosystem, fostering a more diverse and competitive market, and the reduction of redundant network deployments contributes to a more sustainable and eco-friendly approach to telecommunications.
- **New sources of income**: The multi-stakeholder framework paves the way for more entities (individual entrepreneurs and organisations) to become service providers, driving the development of innovative business models and sources of income. Leveraging the 6G infrastructure, stakeholders can introduce cutting-edge services such as immersive virtual and augmented reality experiences, advanced Internet of Things (IoT) applications and highly personalised communication solutions, all of them considered in NANCY.
- **Fostering innovation**: Enabling the participation of diverse actors in service delivery fosters a dynamic and highly competitive environment, as the contribution of new ideas and diverse approaches drives the development of more sophisticated, creative and efficient solutions.

This increased competition, in addition to accelerating technological advances, also results in a richer and more diversified range of services, thereby benefiting end-users.

On the other hand, from a consumer perspective, the open and flexible nature of the 6G ecosystem offers a more personalised experience, with a wider variety of options and better Quality of Service (QoS):

- **Greater customization and choice**: With a more dynamic and competitive ecosystem, consumers have access to a much more varied offer of services, thus driving the creation of solutions adapted to different user profiles, allowing them to choose services that specifically fit their needs, preferences and budgets.
- **Better quality and user experience**: Continuous service improvement is also enhanced by this competition between providers, in this way, consumers can expect faster and more efficient networks, lower latency, greater connection stability, and faster and more effective customer support, and the ability to choose between different offerings forces providers to focus on excellence and innovation to differentiate themselves in the market.
- **Greater control and autonomy**: The multi-stakeholder model gives consumers a more active role in managing their communication services, so they can freely select their providers, customise the packages they purchase and decide how to manage their data and privacy, allowing them to optimise their experience and have more precise control over their information and security.
- **Access to innovative services**: The diversity of actors involved in the 6G ecosystem drives the development and availability of advanced technological solutions, so consumers can benefit from these innovations such as augmented and virtual reality applications, personalised AI services, home automation through the IoT and new forms of immersive communication.

Considering these advancements, the multi-stakeholder scenario envisioned for 6G networks in NANCY not only unlocks new business opportunities but also enhances flexibility, agility, and cost efficiency, all while fostering continuous innovation. By embracing a more collaborative and decentralized approach, this model has the potential to redefine the telecommunications landscape, paving the way for a more inclusive, dynamic, and user-centric digital future.

Furthermore, within this evolving ecosystem, as stated above, marketplaces and smart pricing solutions play a crucial role in optimizing both the provision and consumption of services. On the one hand, marketplaces simplify the discovery, contracting, and management of services, making them more accessible to a broader range of participants; while on the other hand, smart pricing mechanisms dynamically adjust costs based on resource availability, ensuring a fair and efficient distribution of value. Together, these elements contribute to creating a more efficient, transparent, competitive, and innovative 6G services market.

6.2. Balancing Competitiveness & Fair Pricing

In the evolving 6G landscape, the SPM serves a critical dual purpose: driving competitiveness among resource providers while ensuring fair pricing that benefits both these providers and end users in the B-RAN. It strikes a balance that fosters a dynamic marketplace, aligning with our vision of a scalable, equitable, and user-centric 6G ecosystem. Having already discussed the module’s technical aspects, the focus is on the strategic implications and business value of the SPM’s approach, highlighting its role in creating a sustainable and inclusive pricing model.

Competitiveness is at the heart of the SPM, encouraging MNOs to vie for contracts by offering the lowest bids through a multi-round auction process. This rivalry ensures efficient price discovery, allowing B-RAN to allocate resources at market-driven rates. However, unchecked competition could push prices too low, jeopardizing provider profitability or leading to higher costs passed onto users—outcomes that undermine long-term market health. The SPM addresses this by embedding fairness into its design, ensuring that providers can sustain operations while users gain affordable access to B5G services. For stakeholders, this balance translates to a robust ecosystem where competition fuels innovation without sacrificing stability or accessibility.

Fair pricing for providers is achieved by setting boundaries that prevent bids from dropping below viable levels, preserving their ability to operate and invest in network infrastructure. Simultaneously, fair pricing for users ensures that the cost efficiencies gained from competitive bidding translate into affordable service rates, avoiding scenarios where providers offset low bids with steep user fees. The SPM’s blind auction mechanism supports this by giving providers ranking feedback after each round, encouraging strategic adjustments without revealing competitors’ offers. This opacity prevents collusion, fostering genuine competition that benefits users with lower, yet sustainable, prices.

The SPM’s value proposition lies in its ability to cater to diverse stakeholders. Providers benefit from a competitive yet fair playing field that rewards efficiency without punishing profitability, encouraging participation from both large MNOs and smaller players. Users, meanwhile, enjoy affordable access to high-quality B5G services, supporting the project’s goal of equitable connectivity. The system’s adaptability—tested through varied scenarios and validated by simulations—ensures it can scale to meet growing demand, while its blockchain integration enhances trust and transparency, key selling points for a decentralized market. Outputs from the SPM, such as auction analytics, provide stakeholders with clear evidence of this balance, reinforcing confidence in B-RAN’s commercial viability.

In conclusion, the SPM delivers a pricing strategy that harmonizes competitiveness with fairness for providers and users alike. By fostering rivalry that drives efficiency, safeguarding provider sustainability, and ensuring affordable user rates, it positions B-RAN as a forward-thinking solution for 6G networks.

7. Conclusion & Way Forward

7.1. Summary of Key Achievements

The Smart Pricing Module within the NANCY framework has successfully demonstrated an AI-driven approach to dynamic pricing and resource allocation in the Blockchain Radio Access Network for Beyond 5G ecosystems. By integrating Multi-Agent Reinforcement Learning and auction theory, the Smart Pricing Module ensures competitive, fair, and efficient resource sharing in a decentralized environment while providing monetary incentives for users. The key achievements of Task 4.5 are as follows:

1. **Development and Deployment of the Smart Pricing Module**
    a. Designed and implemented a multi-round blind reverse auction mechanism tailored for Blockchain Radio Access Network.
    b. Ensured cost-efficient and competitive pricing.
    c. Deployed the module in a containerized environment for scalability and reliability.
2. **AI-Driven Pricing Model**
    a. Utilized Multi-Agent Reinforcement Learning to optimize pricing decisions.
    b. Employed Proximal Policy Optimization reinforcement learning to optimize bidding behaviour for strategic bid adjustments and competitive balance.
3. **Auction-Based and Game-Theoretic Approaches**
    a. Designed a multi-round blind reverse auction, inspired by game-theoretic auction mechanisms, to maximize revenue and ensure fair competition.
    b. Implemented ranking-based feedback to optimize bid adjustments while preserving privacy.
4. **Integration with the NANCY Ecosystem**
    a. Established a seamless interface with the blockchain-based NANCY Marketplace via a dedicated API.
    b. Enabled real-time price discovery and resource allocation in a decentralized ecosystem based on market demand and provider availability.
5. **Performance Evaluation and Business Impact**
    a. Conducted extensive simulations and benchmarking, validating the Smart Pricing Module’s effectiveness.
    b. Demonstrated efficient customer distribution, preventing price manipulation and ensuring sustainability.

c. Showcased a practical pricing model that maintains Mobile Network Operator's profitability while incentivizing users.

Through these advancements, the Smart Pricing Module has set a foundation for intelligent, fair, and scalable pricing mechanisms in future Beyond 5G and 6G networks. This achievement contributes directly to the NANCY project's broader objectives, leveraging AI and blockchain to enable secure and intelligent resource management, flexible networking, and orchestration.

7.2. Opportunities for Expansion

The SPM could expand its multi-round blind reverse auction framework within the B-RAN by integrating QoS as a key parameter in its pricing model, ensuring that price determination reflects not only cost but also the service quality each MNO can deliver. Currently, the SPM focuses on achieving the lowest final bid for network services, but incorporating more focused QoS metrics such as latency, bandwidth, or resolution support, instead of just the resource availability of each provider, could enhance its utility in 6G networks where diverse user demands require tailored service levels. For instance, an MNO offering higher QoS, such as ultra-low latency for autonomous vehicle communication, could justify a higher bid compared to another providing basic connectivity for IoT sensors, allowing the SPM to compute prices that balance cost with performance. By leveraging its AI-driven capabilities, including MARL, the SPM could dynamically assess and weigh QoS factors alongside bids, ensuring optimal resource allocation that meets user-specific needs while maintaining competitive pricing.

Lastly, utilizing greater vectorization and parallelism in the environment can accelerate inference by reducing redundant computations and improving hardware efficiency. Techniques such as batched inference, parallel action sampling, and optimized tensor operations may further enhance performance. Regarding the neural network model, pruning, quantization, and distillation could reduce computational overhead. Combining some of these methods should help minimize latency and improve the real-time responsiveness of the SPM.

References

[1] X. Ling, J. Wang, T. Bouchoucha, B. C. Levy and Z. Ding, "Blockchain Radio Access Network (B-RAN): Towards Decentralized Secure Radio Access Paradigm," *IEEE Journals & Magazine,* 2019.

[2] H. Yang, A. Alphones, Z. Xiong, D. Niyato, J. Zhao and K.-Y. Wu, "Artificial-Intelligence-Enabled Intelligent 6G Networks," *IEEE Network,* vol. 34, p. 272–280, 2020.

[3] V. Singh, A. Singh , A. Aggarwal and S. Aggarwal, "Advantages of using Containerization Approach for Advanced Version Control System," in *IEEE Conference Publication*, 2022.

[4] Docker, "Docker Documentation," 2025.

[5] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang and W. Zaremba, "OpenAI Gym," *arXiv.org,* June 2016.

[6] J. K. Terry, B. Black, N. Grammel, M. Jayakumar, A. Hari, R. Sullivan, L. Santos, S. Labs, N. L. Williams, Y. Lokesh, P. Ravi and S. Labs, "PettingZoo: A Standard API for Multi-Agent Reinforcement Learning," in *NeurIPS*, 2021.

[7] J. K. Terry, "Multi-Agent Deep Reinforcement Learning in 13 Lines of Code Using PettingZoo," *Medium,* January 2023.

[8] S. D. Jap, "Online Reverse Auctions: Issues, Themes, and Prospects for the Future," *Journal of the Academy of Marketing Science,* vol. 30, p. 506–525, 2002.

[9] L. Alfred, "Reverse Auction: What It Is & How to Crush It in Sales," [Online]. Available: https://blog.hubspot.com/sales/reverse-auction.

[10] N. Motaiah, "Reverse Auction Strategy Guide: Tips for Buyers and Suppliers," 2025.

[11] D. Fudenberg and J. Tirole, Game Theory, Ane Books, 2005.

[12] J. H. Kagel and D. Levin, "The Winner’s Curse and Public Information in Common Value Auctions," *The American Economic Review,* vol. 76, p. 894–920, 1986.

[13] E. Y. L. Chou, L. Lee and G. Ho, "The Winner's Curse? Overbidding Behavior in Auctions," *Social Cognitive and Affective Neuroscience,* vol. 9, p. 1545–1551, 2014.

[14] C. Frydman and C. F. Camerer, "Using the Neural Circuitry of Reward to Design Economic Auctions," *Nature Neuroscience,* vol. 14, p. 129–134, 2011.

[15] E. J. Green and R. H. Porter, "Noncooperative Collusion under Imperfect Price Information," *Econometrica,* vol. 52, p. 87–100, 1984.

[16] J. Schulman, F. Wolski, P. Dhariwal, A. Radford and O. Klimov, "Proximal Policy Optimization Algorithms," *arXiv.org,* July 2017.

[17] J. Schulman, S. Levine, P. Moritz, M. Jordan and P. Abbeel, "Trust Region Policy Optimization," in *Proceedings of the 32nd International Conference on Machine Learning*, 2015.

[18] E. Liang, R. Liaw, P. Moritz, R. Nishihara, R. Fox, K. Goldberg, J. E. Gonzalez, M. I. Jordan and I. Stoica, "RLLIB: Abstractions for Distributed Reinforcement Learning," *arXiv.org,* December 2017.

[19] K. V. C. Chow, C. O'Leary, F. Paxton-Hall, D. Lambie and K. O'Byrne, "A Self-Rewarding Mechanism in Deep Reinforcement Learning for Financial Trading Strategies," *Mathematics,* vol. 12, p. 4020, 2020.

[20] K. V. C. Chow, C. O'Leary, F. Paxton-Hall, D. Lambie and K. O'Byrne, "Reward Shaping for Improved Learning in Real-time Strategy Game," *arXiv preprint,* 2024.

[21] K. V. C. Chow, C. O'Leary, F. Paxton-Hall, D. Lambie and K. O'Byrne, "A Reward Shaping Approach for Reserve Price Optimization using Deep Reinforcement Learning," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, 2021.

[22] Z. Ahmed, N. Le Roux, M. Norouzi and D. Schuurmans, "Understanding the Impact of Entropy on Policy Optimization," *arXiv preprint,* 2018.

[23] P. Rawat, "Approximating Auction Equilibria with Reinforcement Learning," *arXiv preprint,* 2024.

[24] A. DiGiovanni and E. C. Zell, "Survey of Self-Play in Reinforcement Learning," *arXiv.org,* July 2021.

[25] K. Zhang, Z. Yang and T. Ba?ar, "Multi-agent Reinforcement Learning: A Selective Overview of Theories and Algorithms," *arXiv preprint,* 2019.

[26] V. Thoma, M. Curry, N. He and S. Seuken, "Learning Best Response Policies in Dynamic Auctions via Deep Reinforcement Learning," *arXiv preprint,* 2023.

[27] I. Gemp, T. Anthony, J. Kramar and others, "Designing All-Pay Auctions Using Deep Learning and Multi-Agent Simulation," *Scientific Reports,* vol. 12, p. 16937, 2022.