Learning Menu-Based Mechanisms for Truthful Budget-Feasible Procurement

Peng Chen[†]
Southeast University
pengchen@seu.edu.cn

Xiang Liu[†]
Southeast University
The Chinese University of Hong Kong
xiangliu@seu.edu.cn

Hau Chan
University of Nebraska-Lincoln
hchan3@unl.edu

Yan Lyu
Southeast University
lvyanly@seu.edu.cn

Xueyong Xu[*]
North Information Control Research Academy Group Co., Ltd.
xxyyeah@163.com

Weiwei Wu[*]
Southeast University
weiweiwu@seu.edu.cn

Abstract

Budget-feasible mechanisms (BFMs) are a fundamental tool for budget-constrained procurement, but designing strong BFMs remains challenging, particularly for general valuation classes. Classical approaches rely on problem-specific analysis and worst-case approximations, often yielding conservative mechanisms in practice. This motivates data-driven methods for discovering mechanisms with stronger empirical performance. However, strict budget constraints expose a key limitation of existing neural automated mechanism design (AMD): they either fail to provide exact economic guarantees or suffer from severe scalability bottlenecks. We propose BFMNet, a scalable menu-based neural framework for learning truthful budget-feasible procurement mechanisms. BFMNet first learns *self-bid-independent* menus from market context, ensuring dominant-strategy incentive compatibility (DSIC) and individual rationality (IR) by construction, and then applies an ex-post, instance-level menu transformation to enforce budget feasibility. We prove that the resulting mechanism satisfies exact DSIC, IR, and budget feasibility. By avoiding intractable global discretization grids and restrictive Lipschitz constraints, BFMNet remains both expressive and scalable. Extensive experiments show that BFMNet improves utility over classical baselines by up to 34.1%, remains competitive with state-of-the-art neural baselines that provide only approximate economic guarantees, and scales robustly to larger auction instances.

1 Introduction

Budget-constrained procurement is a central paradigm for resource allocation in the digital economy, with broad applications in social networks, federated learning, and crowdsourcing markets [1–6]. In these settings, a budget-limited buyer procures data, services, or computational resources from strategic sellers who incur private costs and may misreport them to maximize individual utility. A

[†]These authors contributed equally to this work.
[*]Corresponding authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

desirable procurement mechanism should achieve high procurement value while ensuring truthful participation, strict budget feasibility, and computational efficiency. This is challenging because budget constraints tightly restrict the allocation and payment rules, which must simultaneously incentivize truthful participation and keep total payments within a limited budget. Designing budget-feasible mechanisms (BFMs) therefore remains a central challenge in procurement [7–9].

Motivated by this challenge, a prominent line of work has studied BFM design from the perspective of approximation algorithms. Singer [7] pioneered this direction by proposing constant-approximation mechanisms for additive and monotone submodular valuation functions. Building on this foundation, subsequent studies [8–14] have improved approximation bounds and extended theoretical guarantees to broader valuation classes. However, these mechanisms are typically handcrafted for specific valuation classes and optimized for worst-case approximation guarantees, which can make them conservative in practical market instances [7, 15, 16]. These limitations naturally motivate a shift from handcrafted design toward automated, data-driven mechanism design.

Automated mechanism design (AMD) has long sought to automate the design of high-performing mechanisms [17, 18], and recent neural approaches leverage deep learning to optimize empirical performance [19]. This progress naturally raises the possibility of applying neural AMD to BFMs. However, budget-constrained procurement exposes a fundamental gap in existing neural paradigms. RegretNet-like approaches [20–23] represent auction mechanisms as neural networks and seek near-optimal, approximately DSIC solutions via adversarial training, but they provide only approximate DSIC guarantees. AMA-based methods [24–27] ensure exact DSIC through affine maximizer auctions (AMAs), but do not naturally handle strict budget feasibility. A separate line of menu-based neural methods, represented by GemNet [28], obtains a DSIC guarantee through global discretization of the input space and Lipschitz constraints on the menu network. However, this global design suffers from poor scalability, and becomes impractical for BFMs because additional parameters such as valuations and budgets substantially enlarge the discretization domain; restrictive Lipschitz conditions further limit the expressivity of the menu network. These limitations leave open the challenge of learning mechanisms that simultaneously deliver strong empirical performance, exact DSIC, strict budget feasibility, and scalability in budget-constrained procurement.

To address these challenges, we propose BFMNet, a scalable menu-based neural framework for learning truthful BFMs. At a high level, BFMNet decouples expressive neural menu learning from budget-feasibility enforcement. In the first stage, a Transformer-based menu network encodes other sellers' bids, valuations, and the budget to generate high-utility base menus with low empirical budget violations, allowing the learned mechanism to capture rich market structure while guaranteeing DSIC and IR by construction. The key challenge is budget feasibility, which does not automatically follow from self-bid-independent menu generation. To resolve this, BFMNet introduces an ex-post, instance-level menu transformation that adjusts the learned base menus to strictly satisfy the budget constraint while preserving DSIC and IR. This design avoids global discretization of the full input space and removes restrictive Lipschitz constraints, thereby enabling both exact economic guarantees and scalability. To further improve efficiency, we tailor four acceleration strategies, reducing transformation latency by over 99.9% and enabling near real-time inference. We prove that the resulting mechanism satisfies exact DSIC, IR, and budget feasibility, and empirically show that BFMNet achieves strong utility performance under these strict guarantees. Our main contributions are summarized as follows:

- We study learning-based mechanism design for truthful budget-feasible procurement, where strict budget feasibility makes it difficult to simultaneously achieve exact economic guarantees, strong empirical performance, and scalability.
- We propose BFMNet, a scalable menu-based neural framework that separates expressive learning from economic enforcement. BFMNet learns high-quality base menus from market context and provides an ex-post, instance-level menu transformation to enforce budget feasibility while preserving exact DSIC and IR.
- Experiments show that BFMNet improves utility over strong classical baselines by up to 34.1%, remains competitive with neural baselines that provide only approximate guarantees, and scales robustly to larger auction instances under exact economic validity. To support efficient deployment, we further develop four acceleration strategies that reduce transformation latency by over 99.9% and enable near real-time inference.

2 Related Work

Budget-Feasible Mechanisms. The study of BFMs was initiated by Singer [7], who established the first constant-factor approximation mechanisms for additive and monotone submodular valuation functions. Subsequent work has tightened approximation bounds and extended BFM design to broader valuation classes. For additive valuations, Chen et al. [10] improved the approximation ratios to a deterministic $2+\sqrt{2}$ and a randomized 3, Anari et al. [11] achieved an optimal deterministic $e/(e-1)$ approximation for divisible items in large markets, and Gravin et al. [12] obtained a tight deterministic 3-approximation and an optimal randomized 2-approximation. For monotone and non-monotone submodular valuations, further bounds were established and improved in [10, 13, 9]; for subadditive valuations, Dobzinski et al. [8] gave an $O(\log^3 n)$ approximation, later improved to $O(\log n/\log\log n)$ by Balkanski et al. [13]. Despite these theoretical advances, approximation-based BFMs typically require carefully handcrafted designs for specific valuation classes and optimize worst-case guarantees, often resulting in conservative mechanisms with limited empirical utility in practical market instances.

Automated Mechanism Design. Early automated mechanism design (AMD) formulated mechanism design as an optimization problem, often via linear programming over discrete type spaces [17, 18], but suffered from exponential complexity and poor scalability in continuous or high-dimensional settings [29]. Recent neural AMD methods improve the empirical performance of learned mechanisms, but remain limited for budget-constrained procurement. RegretNet [20] and its successors [21–23, 30] optimize mechanisms via adversarial training, but rely on soft incentive constraints and provide only approximate DSIC guarantees, even in budget-constrained settings [31]. AMA-based methods [24–27, 32] ensure exact DSIC by restricting mechanisms to affine maximizer auctions, but do not directly provide strict budget feasibility in procurement. Menu-based methods [33–35], such as GemNet [28], enforce stronger guarantees through global grid discretization and Lipschitz continuity, but suffer from scalability and expressivity limitations; for BFMs, such global discretization must also cover non-strategic profiles such as valuations and budgets, further exacerbating the curse of dimensionality. In contrast, BFMNet learns self-bid-independent menus and enforces budget feasibility through an ex-post, instance-level transformation, achieving exact DSIC, IR, and budget feasibility without global discretization or mandatory Lipschitz constraints.

3 Preliminaries

We study budget-feasible mechanisms (BFMs) in a classical sealed-bid, budget-constrained procurement auction. A buyer with a public budget $B>0$ seeks to procure items, goods, or services from a set of strategic sellers, denoted by $[n]=\{1,\dots,n\}$. Each seller $i\in[n]$ provides an item characterized by a public value v_i to the buyer, while incurring a private cost $c_i\in\mathcal{C}_i\subset\mathbb{R}_{\geq 0}$. We denote the public value profile by $\boldsymbol{v}=(v_1,\dots,v_n)\in\mathbb{R}^n_{\geq 0}$ and the true cost profile by $\boldsymbol{c}=(c_1,\dots,c_n)\in\mathcal{C}$, where $\mathcal{C}=\prod_{i=1}^n\mathcal{C}_i$ is the joint cost space. Since sellers are strategic, they may misreport their private costs by submitting bids $b_i\in\mathcal{C}_i$, possibly with $b_i\neq c_i$. Let $\boldsymbol{b}=(b_1,\dots,b_n)\in\mathcal{C}$ denote the reported bid profile, and let $\boldsymbol{b}_{-i}=(b_1,\dots,b_{i-1},b_{i+1},\dots,b_n)\in\mathcal{C}_{-i}$ denote the bids of all sellers except i. A procurement mechanism $\mathcal{M}=(\boldsymbol{x},\boldsymbol{p})$ consists of an allocation rule $\boldsymbol{x}=(x_1,\dots,x_n):\mathcal{C}\times\mathbb{R}^n_{\geq 0}\times\mathbb{R}_{>0}\to[0,1]^n$ and a payment rule $\boldsymbol{p}=(p_1,\dots,p_n):\mathcal{C}\times\mathbb{R}^n_{\geq 0}\times\mathbb{R}_{>0}\to\mathbb{R}^n_{\geq 0}$. For any profile $(\boldsymbol{b},\boldsymbol{v},B)$, $x_i(\boldsymbol{b},\boldsymbol{v},B)$ and $p_i(\boldsymbol{b},\boldsymbol{v},B)$ specify the allocation and payment for seller i, respectively. For presentation, we use a unified allocation domain $[0,1]^n$, which directly models divisible procurement and the indivisible setting is recovered by restricting allocations to binary outcomes $x_i\in\{0,1\}$. The buyer's utility is evaluated by $V(\boldsymbol{x},\boldsymbol{v})$, where $V:[0,1]^n\times\mathbb{R}^n_{\geq 0}\to\mathbb{R}_{\geq 0}$. Each seller i's utility is defined as:

$$u_i(c_i;\boldsymbol{b},\boldsymbol{v},B)=p_i(\boldsymbol{b},\boldsymbol{v},B)-x_i(\boldsymbol{b},\boldsymbol{v},B)\cdot c_i,\quad\forall i\in[n] \tag{1}$$

We focus on mechanisms that satisfy three key economic desiderata, individual rationality (IR), dominant-strategy incentive compatibility (DSIC), and budget feasibility:

- **Individual Rationality (IR):** A mechanism is individually rational if truthful bidding guarantees non-negative utility for every seller. Formally, for all $i\in[n]$, $\boldsymbol{v}\in\mathbb{R}^n_{\geq 0}$, $B\in\mathbb{R}_{\geq 0}$, $c_i\in\mathcal{C}_i$, and $\boldsymbol{b}_{-i}\in\mathcal{C}_{-i}$,

$$u_i(c_i;(c_i,\boldsymbol{b}_{-i}),\boldsymbol{v},B)\geq 0. \tag{2}$$

- **Dominant-Strategy Incentive Compatibility (DSIC / Truthfulness):** A mechanism is dominant-strategy incentive compatible if truthful reporting maximizes each seller's utility, regardless of the bids submitted by others. Formally, for all $i \in [n]$, $\boldsymbol{v} \in \mathbb{R}^n_{\geq 0}$, $B \in \mathbb{R}_{\geq 0}$, $c_i, b_i \in \mathcal{C}_i$, and $\boldsymbol{b}_{-i} \in \mathcal{C}_{-i}$,

$$u_i(c_i; (c_i, \boldsymbol{b}_{-i}), \boldsymbol{v}, B) \geq u_i(c_i; (b_i, \boldsymbol{b}_{-i}), \boldsymbol{v}, B). \tag{3}$$

- **Budget Feasibility:** A mechanism is (ex-post) budget feasible if the total payment never exceeds the buyer's budget. Formally, for all $\boldsymbol{v} \in \mathbb{R}^n_{\geq 0}$, $B \in \mathbb{R}_{\geq 0}$, and $\boldsymbol{b} \in \mathcal{C}$,

$$\sum_{i \in [n]} p_i(\boldsymbol{b}, \boldsymbol{v}, B) \leq B. \tag{4}$$

Our objective is to learn a general budget-feasible mechanism $\mathcal{M} = (\boldsymbol{x}, \boldsymbol{p})$ that maximizes the buyer's utility while exactly satisfying IR, DSIC, and budget feasibility. These guarantees do not rely on structural assumptions such as additivity or submodularity of $V(\cdot, \cdot)$.

4 Learning Budget-Feasible Mechanisms Through Menu Transformation

In this section, we present BFMNet, a scalable menu-based neural framework for learning truthful BFMs. The key difficulty is that strict budget constraints make it hard to simultaneously achieve strong empirical performance, exact DSIC, (ex-post) budget feasibility, and computational scalability. BFMNet addresses this difficulty through a two-stage design. Section 4.1 first reformulates procurement mechanisms using self-bid-independent menus, which guarantee DSIC and IR by the *Taxation Principle* but do not control the total payment across sellers. Section 4.2 then introduces a permutation-equivariant Transformer that learns expressive base menus from other sellers' bids and non-strategic profiles, using a soft penalty to reduce empirical budget violations during training. Since soft penalties cannot certify exact budget feasibility, Section 4.3 develops an ex-post, instance-level menu transformation, formulated as a mixed-integer linear program (MILP), to enforce the budget constraint while preserving DSIC and IR. Unlike prior menu-based methods with global certification, our instance-level transformation improves scalability, while avoiding discretization over non-strategic profiles and neural Lipschitz-continuity constraints. Finally, Section 4.4 presents four acceleration strategies that exploit the MILP structure to improve transformation efficiency.

4.1 Menu-Based Procurement Formulation

The *Taxation Principle* [33, 36] provides a foundational characterization of DSIC. Specifically, any DSIC mechanism can be equivalently represented by a set of *self-bid-independent* menus coupled with *bidder optimization*. Leveraging this equivalence, we adopt a menu-based representation for procurement auctions, reducing the BFM design problem to learning menu functions. In a menu-based representation, the allocation and payment rules for each seller $i \in [n]$ are encapsulated by a menu function $M_i(\boldsymbol{b}, \boldsymbol{v}, B)$. This function maps the auction profile $(\boldsymbol{b}, \boldsymbol{v}, B)$ to a menu of $K \geq 1$ options:

$$M_i(\boldsymbol{b}, \boldsymbol{v}, B) = \left\{(x_i^1, p_i^1), \ldots, (x_i^K, p_i^K)\right\} \tag{5}$$

where each menu option (x_i^k, p_i^k) specifies a prospective trade, with $x_i^k \in [0, 1]$ denoting the allocation and $p_i^k \in \mathbb{R}$ representing the corresponding payment. The menu profile $\boldsymbol{M} = (M_1, \ldots, M_n)$ thus defines the complete menu-based representation of the procurement auction. Theorem 1 formally characterizes DSIC procurement mechanisms under this framework:

Theorem 1 (Taxation Principle [33, 36])**.** *A procurement mechanism satisfies dominant-strategy incentive compatibility (DSIC) if and only if there exists a menu-based representation $\boldsymbol{M} = (M_1, \ldots, M_n)$ such that for any seller $i \in [n]$: **(1) Self-Bid Independence:** M_i is strictly independent of the seller's own reported bid b_i, i.e., $M_i((c_i, \boldsymbol{b}_{-i}), \boldsymbol{v}, B) = M_i((b_i, \boldsymbol{b}_{-i}), \boldsymbol{v}, B)$ for all $c_i, b_i, \boldsymbol{b}_{-i}$; and **(2) Bidder Optimization:** the realized allocation and payment (x_i^*, p_i^*) maximize the seller's utility from M_i, i.e., $(x_i^*, p_i^*) \in \arg\max_{(x_i^k, p_i^k) \in M_i} (p_i^k - x_i^k \cdot b_i)$.*[3]

Since the menu M_i is independent of seller i's reported bid b_i, truthful bidding ($b_i = c_i$) inherently maximizes seller i's utility and thus guarantees DSIC. However, in the multi-seller setting, the total

[3]Ties among utility-maximizing menu options are broken in favor of the lowest-payment option.

payment induced by n independently optimal menu options may exceed the buyer's budget, i.e., $\sum_{i\in[n]} p_i^* > B$. A naive solution would be to proportionally scale down all allocations and payments to meet the budget constraint. Unfortunately, this approach breaks DSIC: the scaling factor depends on the jointly realized outcome and can therefore be influenced by a seller's report, introducing cross-seller coupling that violates the *self-bid independence* required for truthfulness.

4.2 Learning Self-Bid-Independent Base Menus

We parameterize our budget-feasible menu network, BFMNet, by θ. BFMNet adopts a permutation-equivariant Transformer architecture [37] to capture complex market context among sellers; architectural details are provided in Appendix A. To strictly enforce *self-bid independence*, BFMNet takes only the other sellers' bids $\boldsymbol{b}_{-i}$ and non-strategic profiles, i.e., the valuation profile $\boldsymbol{v}$ and budget B, as inputs:

$$M_i(\boldsymbol{b}_{-i}, \boldsymbol{v}, B; \theta) = \left\{(x_i^1(\theta), p_i^1(\theta)), \ldots, (x_i^{K-1}(\theta), p_i^{K-1}(\theta))\right\} \tag{6}$$

To guarantee exact individual rationality (IR), we append a fixed default zero-option, i.e., $(x_i^K, p_i^K) = (0, 0)$. We train BFMNet to maximize empirical buyer utility while discouraging budget violations through the following loss:

$$\mathcal{L}(\theta) = -\frac{1}{|L|}\sum_{\boldsymbol{b}\in[L]} V\left(\left(\sum_{k\in[K]} z_1^k x_1^k, \ldots, \sum_{k\in[K]} z_n^k x_n^k\right), \boldsymbol{v}\right) + \lambda \cdot \frac{1}{|L|}\sum_{\boldsymbol{b}\in[L]} \max\left(0, \sum_{i\in[n]}\sum_{k\in[K]} z_i^k p_i^k - B\right) \tag{7}$$

Here, L denotes a training set of procurement instances, whose sample complexity is analyzed in Appendix B. The first component is the negative empirical buyer utility, whereas the second imposes a penalty on budget violations. The penalty weight $\lambda > 0$ acts as a Lagrange multiplier and is gradually increased throughout training.

The term z_i^k denotes the relaxed probability that seller i selects the k-th menu option. It is computed through a differentiable SoftMax transformation over seller i's menu utilities:

$$z_i^k(c_i; \boldsymbol{b}, \boldsymbol{v}, B) = \text{SoftMax}_k(\tau \cdot u_i^1(c_i; \boldsymbol{b}, \boldsymbol{v}, B), \ldots, \tau \cdot u_i^K(c_i; \boldsymbol{b}, \boldsymbol{v}, B)) \tag{8}$$

where $u_i^k(b_i; \boldsymbol{b}, \boldsymbol{v}, B) = p_i^k - x_i^k b_i$. The scaling factor $\tau > 0$ controls the sharpness of the SoftMax approximation to the discrete $\arg\max$ selection rule. We initialize τ with a small value and gradually increase it during training, thereby progressively tightening the approximation. During inference, we replace the SoftMax rule with exact $\arg\max$ *bidder optimization*. We further establish the convergence rate of the approximation error induced by this relaxation.

Theorem 2 (SoftMax Approximation Error)**.** *In a menu-based procurement auction with a fixed finite menu profile $\boldsymbol{M} = (M_1, \ldots, M_n)$, under the assumptions stated in Appendix C, if the buyer's valuation is 1-Lipschitz in the allocation vector under the ℓ_1 norm, the expected absolute errors in both buyer utility and total payment induced by the SoftMax approximation are $O(1/\tau)$ as $\tau \to \infty$.*

Theorem 2 quantifies the approximation error induced by the SoftMax relaxation under these assumptions. Explicit bounds and the proof are provided in Appendix C.

4.3 Menu Transformation for Exact Budget Feasibility

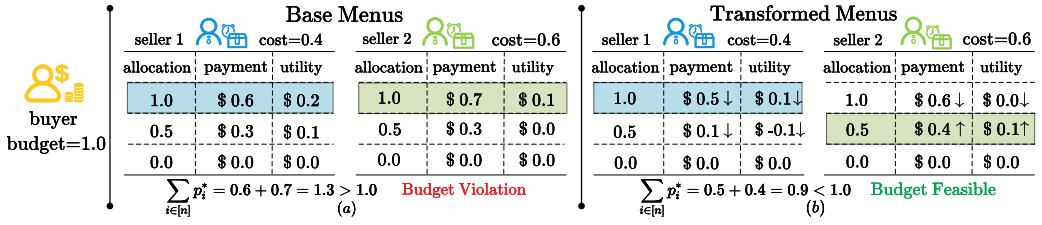


Figure 1: Illustration of the ex-post menu transformation. (a) Under the initial base menus, sellers' utility-maximizing choices may cause total payments to exceed the budget. (b) The ex-post menu transformation strictly guarantees budget feasibility.

BFMNet learns expressive base menus with a soft budget penalty, but *bidder optimization* may induce residual budget violations. To strictly enforce budget feasibility, we introduce an ex-post, instance-level menu transformation. Unlike prior menu-based methods that require discretization over all possible input profiles, our transformation adjusts the learned menus separately for each auction instance. As illustrated in Figure 1, the learned base menus may induce total payments that exceed the budget, whereas the transformed menus strictly satisfy the budget constraint. This ex-post, instance-wise design avoids the curse of dimensionality inherent in global discretization and eliminates the need to discretize non-strategic profiles.

To construct the transformation, we discretize only the continuous bid space $\mathcal{C}$ into a uniform grid $\mathcal{G} = \prod_{i=1}^{n} \mathcal{G}_i$ with step size $\epsilon > 0$. Given an instance $(\boldsymbol{b}, \boldsymbol{v}, B)$, each bid b_i is rounded to its nearest grid point $\tilde{b}_i \in \mathcal{G}_i$. To preserve self-bid independence, seller i's base menu is generated as $M_i(\tilde{\boldsymbol{b}}_{-i}, \boldsymbol{v}, B)$, depending only on other sellers' grid bids and the non-strategic profiles. Although seller i's menu is independent of its own report, its report affects the menus generated for other sellers and can thereby change their payment. Thus, enforcing budget feasibility must hold over seller i's entire bid space $\mathcal{C}_i$. To achieve this, we introduce two key safeguards at every grid point $b_{i,g} \in \mathcal{G}_i$: *safety payment* and *safety margin*.

Safety payment. The first safeguard controls the total payment at grid points. For each grid point $b_{i,g}$, we assign seller i a safe budget share $sp_{i,g} = B \cdot \frac{\tilde{p}^*_{i,g}}{\tilde{p}^*_{i,g} + \sum_{j \neq i} \tilde{p}^*_j(b_{i,g})}$, where $\tilde{p}^*_{i,g}$ is seller i's selected payment at $b_{i,g}$, and $\tilde{p}^*_j(b_{i,g})$ is seller j's selected payment under the same counterfactual grid bid of seller i.[4] This normalization allocates the budget proportionally to the payments induced at $b_{i,g}$. Therefore, if every selected payment is capped by its corresponding safety payment, the resulting total payment is at most B.

Safety margin. The second safeguard extends this guarantee from grid bids to continuous bids. We impose a safety margin $sm = \epsilon/2$ between the selected option and all alternatives. Since menu utilities are linear in the bid and allocations lie in $[0, 1]$, this margin ensures that the selected option remains optimal for any continuous bid in the local neighborhood $[b_{i,g} - \epsilon/2, b_{i,g} + \epsilon/2]$. Thus, the budget-feasible option at the grid point remains valid throughout the surrounding continuous region.

We formulate the two safeguards as a big-M mixed-integer linear program (MILP). Intuitively, the MILP minimally perturbs the learned payments to enforce both safety constraints while retaining as much of the base menus' empirical utility as possible. For each seller $i \in [n]$, it optimizes payment adjustments Δp_i^k over all grid points:

$$
\begin{aligned}
\min \quad & \sum_{k \in [K-1]} |\Delta p_i^k| \\
\text{s.t.} \quad & \sum_{k \in [K]} z_g^k = 1, \quad z_g^k \in \{0,1\}, && \forall g \in [|\mathcal{G}_i|], \\
& U_g \le p_i^k - \Delta p_i^k - x_i^k b_{i,g} + (1 - z_g^k)\Gamma, && \forall k \in [K], \forall g \in [|\mathcal{G}_i|], \\
& U_g \ge p_i^k - \Delta p_i^k - x_i^k b_{i,g} + (1 - z_g^k) sm, && \forall k \in [K], \forall g \in [|\mathcal{G}_i|], \\
& p_i^k - \Delta p_i^k \le sp_{i,g} + (1 - z_g^k)\Gamma, && \forall k \in [K], \forall g \in [|\mathcal{G}_i|].
\end{aligned}
\tag{9}
$$

Here, Γ is a sufficiently large constant, and the default zero-option is kept unchanged, i.e., $\Delta p_i^K = 0$. The binary variable z_g^k indicates whether option k is selected at grid point $b_{i,g}$. When $z_g^k = 1$, the big-M terms vanish, so U_g equals the utility of the optimal menu option and its transformed payment is capped by $sp_{i,g}$. When $z_g^k = 0$, the constraints are relaxed, except that the utility of the optimal option must exceed that of option k by the safety margin sm. The program is always feasible because sufficiently reducing the payments of all regular options makes the unchanged zero-option optimal at every grid point.

Theorem 3 (Economic Guarantees)**.** *The mechanism obtained by applying the ex-post transformation to the self-bid-independent base menus satisfies exact individual rationality (IR), dominant-strategy incentive compatibility (DSIC), and budget feasibility for any auction instance.*

[4] If the total payment in the denominator is zero, we set $sp_{i,g} = 0$.

Theorem 3 summarizes the economic guarantees of BFMNet. The proof follows from the taxation principle and the two safeguards introduced above, the *safety margin* and the *safety payment*. Detailed formal proofs are deferred to Appendix D.

4.4 Acceleration Strategies for Menu Transformation

The proposed ex-post, instance-level menu transformation provides exact economic guarantees, but directly solving the resulting MILPs can be computationally expensive due to the large number of binary variables and constraints. To improve efficiency, we introduce four acceleration strategies that exploit the structure of menu transformation in procurement auctions. Inspired by GemNet [28], these strategies progressively reduce the number of active binary variables and redundant constraints. As shown in Table 5, applying them sequentially reduces the average runtime by over 99.96%, enabling efficient ex-post transformation in practice. We detail the four strategies below.

Parallel-Transformation. Our transformation design makes each seller's menu transformation independent of the transformed menus of other sellers. This allows the seller-specific MILPs to be solved in parallel, thereby avoiding the sequential-processing bottleneck that would otherwise become increasingly severe as the number of sellers grows.

Zero-Violation. The soft budget penalty used during training often produces base menus that already satisfy the required safeguards. For such sellers, no payment adjustment is needed, and the MILP can be safely bypassed. Specifically, for seller i, we skip the solver if the selected payment satisfies $p^*_{i,g} \leq sp_{i,g}$ and the selected option has margin $u^*_i(b_{i,g}) - u^k_i(b_{i,g}) \geq sm$ over every non-selected option $k \neq k^*_g$ at every grid point $g \in [|\mathcal{G}_i|]$.

IR-Screening. Some base menu options yield negative utilities at most grid points and are often not selected after transformation. By imposing non-negative payment adjustments, $\Delta p^k_i \geq 0$, the transformation can only reduce payments and thus cannot make these options more attractive. We can therefore remove their associated binary variables z^k_g and replace them with deterministic safety-margin constraints. This conservative screening substantially simplifies the MILP in practice, while causing only negligible utility loss empirically.

Adaptive Keep-Choice. We further refine the Keep-Choice (KC) strategy of GemNet [28]. KC fixes the pre-transformation optimal option at a grid point, thereby removing the binary variables z^k_g and replacing the big-M constraints with linear margin constraints. Unlike prior point-wise implementations, we exploit the fact that each pairwise utility-gap constraint is affine in the bid. Therefore, if the constraint holds at both endpoints of an interval whose grid points share the same retained option, it holds throughout the interval. This interval-based reduction replaces redundant point-wise constraints with a small number of boundary constraints, yielding orders-of-magnitude reductions in variables and constraints and enabling near real-time transformation.

5 Experiments

We conduct an empirical evaluation of our proposed mechanism across diverse valuation functions and bid distributions. Our experiments validate three core dimensions: (1) strong utility performance under exact economic guarantees; (2) scalability in large-scale settings; and (3) the efficacy of our MILP acceleration strategies.

Setup. We train BFMNet with Adam using a learning rate of 3×10^{-4} and a batch size of $2{,}048$. The bid space is discretized into 101 grid points with step size $\epsilon = 0.01$. The SoftMax scaling factor τ is initialized to 5 and increased by $1.15\times$ every 30 epochs, while the budget penalty λ is updated every 10 epochs. For $\tau > 10{,}000$, we select the highest-utility base-menu model with a pre-transformation violation rate below 0.5%, and apply MILP transformation using the Gurobi optimizer [38]. We set $K \leq 10$ for standard markets ($n \leq 10$) and $K \leq 25$ for large-scale markets ($n \leq 35$), and report experiment results under both divisible and indivisible item settings. Note that for indivisible procurement, each seller's menu consists of a fixed no-trade option $(0, 0)$ and a trade option $(1, p_i)$, with the neural network determining only the payment p_i. All experiments were run on an Intel Xeon Silver 4310 CPU and dual NVIDIA RTX 5090 GPUs.

Baselines. Depending on the setting, we compare against three baseline classes. (1) **Upper bounds**: *OPT*(*Fractional*) and *OPT* maximize buyer utility without DSIC constraints and

therefore serve only as non-strategic references. (2) **Neural AMD baselines**: in divisible settings, we evaluate *RegretNet* [20] and the Transformer-based *CITransNet* [22]. These methods optimize utility using soft regret and budget penalties, and we report their expected regret $rgt := \frac{1}{n}\sum_{i\in[n]} \max_{\boldsymbol{b}}[u_i(c_i;(b_i,\boldsymbol{b}_{-i}),\boldsymbol{v},B) - u_i(c_i;(c_i,\boldsymbol{b}_{-i}),\boldsymbol{v},B)]$ and budget violation $bfv := \max(0, \sum_{i\in[n]} p_i - B)$. For indivisible procurement, we do not include RegretNet or CITransNet because these baselines are trained in a continuous relaxation framework and do not directly yield matched deterministic integral mechanisms without additional rounding or redesign. Moreover, their economic constraints are penalty-based and therefore do not provide exact DSIC or strict budget feasibility. Notably, GemNet [28] is infeasible for BFMs, as global discretization over bids, valuations, and budgets already induces more than 2×10^8 MILPs for $n = 2$ with 100 grid points per dimension. (3) **Classical truthful baselines**: for additive settings we use *Deterministic Threshold* [11], *Greedy Additive* [10], *Posted Pricing* [15], and *Deterministic Pruning* [12]; for monotone submodular settings we use *Greedy Submodular* [10], *Posted Pricing* [15] and *TripleEagleDet* [9]; and for non-monotone submodular settings we use *Clock Auction* [13] and *TripleEagleNm* [9]. These classical truthful mechanisms form the main matched comparisons in indivisible procurement.

Table 1: Results under **additive** valuation $V(\boldsymbol{x},\boldsymbol{v}) = \sum_{i\in[n]} x_i v_i$. Profiles: $bid, value \sim U[0,1]$, $budget \sim U[2,4]$.

Setting	Method	$n=2$			$n=4$			$n=6$			$n=8$			$n=10$		
		val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓
Divisible	**OPT** (Fractional)	1.0016	N/A	-	1.9841	N/A	-	2.8902	N/A	-	3.6241	N/A	-	4.2190	N/A	-
	CITransNet	1.0016	0.0083	0.0002	1.7746	0.0016	0.0006	2.2669	0.0013	0.0010	2.6499	0.0011	0.0007	3.0702	0.0018	0.0008
	RegretNet	0.9952	0.0111	0.0052	1.7063	0.0073	0.0004	2.1946	0.0023	0.0009	2.5983	0.0019	0.0008	2.9799	0.0024	0.0066
	Deterministic Threshold	0.8636	-	-	1.4822	-	-	1.9610	-	-	2.3360	-	-	2.6577	-	-
	Ours	**1.0015**	-	-	**1.7706**	-	-	**2.2589**	-	-	**2.6150**	-	-	**2.9534**	-	-
Indivisible	**OPT**	1.0016	N/A	-	1.9674	N/A	-	2.6344	N/A	-	3.5182	N/A	-	4.0870	N/A	-
	Greedy Additive	0.6680	-	-	0.8512	-	-	1.7374	-	-	2.5184	-	-	2.9633	-	-
	Deterministic Pruning	0.8569	-	-	1.5147	-	-	1.8914	-	-	2.2292	-	-	2.5538	-	-
	Posted Pricing	**1.0016**	-	-	1.6240	-	-	2.0487	-	-	2.3585	-	-	2.6381	-	-
	Ours	1.0015	-	-	**1.7708**	-	-	**2.2651**	-	-	**2.6093**	-	-	**2.9684**	-	-

Table 2: Results under **monotone submodular** valuation $V(\boldsymbol{x},\boldsymbol{v}) = \sqrt{\sum_{i\in[n]} x_i v_i}$. Profiles: $bid \sim U[0,1]$, $value \sim U[0,2]$, $budget \sim U[2,4]$.

Setting	Method	$n=2$			$n=4$			$n=6$			$n=8$			$n=10$		
		val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓
Divisible	**OPT** (Fractional)	1.3776	N/A	-	1.9707	N/A	-	2.3833	N/A	-	2.6752	N/A	-	2.8882	N/A	-
	CITransNet	1.3776	0.0075	0.0001	1.8477	0.0011	0.0010	2.1177	0.0010	0.0009	2.2823	0.0012	0.0009	2.4748	0.0019	0.0017
	RegretNet	1.3234	0.0040	0.0000	1.8086	0.0034	0.0005	2.0941	0.0024	0.0014	2.2651	0.0011	0.0014	2.4212	0.0021	0.0016
	Ours	**1.3776**	-	-	**1.8476**	-	-	**2.1024**	-	-	**2.2791**	-	-	**2.4158**	-	-
Indivisible	**OPT**	1.3776	N/A	-	1.9620	N/A	-	2.3567	N/A	-	2.6344	N/A	-	2.8418	N/A	-
	Greedy Submodular	1.1300	-	-	1.2564	-	-	1.3051	-	-	1.3302	-	-	1.3468	-	-
	TripleEagleDet	1.1300	-	-	1.3976	-	-	1.5972	-	-	1.7479	-	-	1.8766	-	-
	Posted Pricing	**1.3776**	-	-	1.7792	-	-	1.9932	-	-	2.1403	-	-	2.2590	-	-
	Ours	**1.3776**	-	-	**1.8485**	-	-	**2.1004**	-	-	**2.2770**	-	-	**2.4097**	-	-

Table 3: Results under **non-monotone submodular** valuation $V(\boldsymbol{x},\boldsymbol{v}) = \left(\sum_{i\in[n]} v_i x_i\right)\left(\sum_{i\in[n]} v_i - \sum_{i\in[n]} v_i x_i\right)$. Profiles: $bid \sim U[0,1]$, $value \sim U[0,1]$, $budget \sim U[1,2]$.

Setting	Method	$n=2$			$n=4$			$n=6$			$n=8$			$n=10$		
		val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓
Divisible	**OPT** (Fractional)	0.2917	N/A	-	1.0796	N/A	-	2.3759	N/A	-	4.1564	N/A	-	6.4193	N/A	-
	CITransNet	0.2917	0.0006	0.0000	1.0770	0.0004	0.0002	2.3690	0.0011	0.0012	4.0156	0.0027	0.0025	6.2031	0.0027	0.0057
	RegretNet	0.2917	0.0005	0.0000	1.0677	0.0004	0.0002	2.2983	0.0004	0.0005	3.7981	0.0010	0.0008	5.9386	0.0010	0.0039
	Ours	**0.2869**	-	-	**1.0507**	-	-	**2.3341**	-	-	**3.9857**	-	-	**6.0328**	-	-
Indivisible	**OPT**	0.2499	N/A	-	1.0652	N/A	-	2.3639	N/A	-	4.1339	N/A	-	6.3752	N/A	-
	Clock Auction	**0.2499**	-	-	1.0006	-	-	1.9503	-	-	3.1099	-	-	4.5259	-	-
	TripleEagleNm	**0.2499**	-	-	1.0057	-	-	1.9865	-	-	3.1346	-	-	4.4570	-	-
	Ours	0.2492	-	-	**1.0550**	-	-	**2.3209**	-	-	**3.9601**	-	-	**5.9774**	-	-

Utility Under Exact Economic Guarantees. Tables 1, 2, and 3 summarize the performance of our mechanism across diverse valuation settings. In our results, **bold** values indicate the highest utility among strictly feasible mechanisms, i.e., mechanisms satisfying exact DSIC and budget feasibility,

while underlined values denote the highest utility across all baselines excluding the non-strategic OPT. The results demonstrate that our mechanism significantly outperforms all classical truthful baselines while maintaining exact IR, DSIC and budget feasibility. Specifically, in the most challenging **non-monotone submodular** discrete setting ($n = 10$), our mechanism achieves a utility of 5.9774, representing a 34.1% improvement over the state-of-the-art *TripleEagleNm* (4.4570). Similarly, for **monotone submodular** discrete valuations ($n = 10$), our mechanism outperforms *TripleEagleDet* by 28.4% (2.4097 vs. 1.8766). Even in **additive** settings, our approach consistently delivers robust performance gains over traditional approximation mechanisms. Furthermore, compared with deep learning baselines such as *RegretNet* and *CITransNet*, which provide only approximate economic guarantees, our approach achieves highly competitive utility while eliminating their residual economic violations. For instance, in the $n = 10$ continuous additive market, *RegretNet* incurs a budget violation of 0.0066 and a regret of 0.0024, whereas our mechanism rigorously enforces exact DSIC and budget feasibility through the ex-post menu transformation. This demonstrates the ability of our framework to achieve high utility without compromising strict economic guarantees.

Table 4: Scalability evaluation in **large-scale** settings ($n = 15$ to 35) under submodular valuation $V(\boldsymbol{x}, \boldsymbol{v}) = \sqrt{\sum_{i\in[n]} x_i v_i}$ with menu size $K \leq 25$. Profiles: $bid \sim U[0, 1]$, $value \sim U[0, 1]$, and $budget \sim U[0.25n, 0.5n]$.

Setting	Method	$n = 15$			$n = 20$			$n = 25$			$n = 30$			$n = 35$		
		val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓
Divisible	**OPT** (Fractional)	2.6174	N/A	-	3.0331	N/A	-	3.3943	N/A	-	3.7240	N/A	-	4.0285	N/A	-
	RegretNet	2.2298	0.0013	0.0016	2.5643	0.0011	0.0073	2.8609	0.0012	0.0104	3.1211	0.0012	0.0075	3.3640	0.0011	0.0086
	Ours	**2.2214**	-	-	**2.5769**	-	-	**2.8506**	-	-	**3.1053**	-	-	**3.3918**	-	-
Indivisible	**OPT**	2.6013	N/A	-	3.0207	N/A	-	3.3843	N/A	-	3.7156	N/A	-	4.0213	N/A	-
	Greedy Submodular	0.9678	-	-	0.9755	-	-	0.9804	-	-	0.9836	-	-	0.9859	-	-
	TripleEagleDet	1.7712	-	-	2.0854	-	-	2.3669	-	-	2.6261	-	-	2.8663	-	-
	Ours	**2.2258**	-	-	**2.5829**	-	-	**2.8310**	-	-	**3.1188**	-	-	**3.4011**	-	-

Scalability in Large-Scale Settings. Table 4 demonstrates the robust scalability of our mechanism as the number of sellers increases from $n = 15$ to 35. We omit *CITransNet* from this comparison due to its prohibitive training cost, which exceeds 50 hours for $n = 15$. In indivisible item settings, our mechanism significantly outperforms classical truthful baselines across all scales. At $n = 35$, our mechanism achieves a utility of 3.4011, representing an 18.7% improvement over *TripleEagleDet* and a 245.0% improvement over *Greedy Submodular*. Furthermore, while *RegretNet* exhibits increasing budget violations as the market scales, reaching 0.0086 at $n = 35$, our mechanism maintains strict budget feasibility and DSIC through its instance-level transformation. Remarkably, in large-scale divisible settings, our mechanism even surpasses the raw utility of *RegretNet*, e.g., 3.3918 vs. 3.3640 at $n = 35$. This advantage arises because our menu transformation does not rely on Lipschitz continuity constraints, enabling BFMNet to leverage a more expressive Transformer architecture and sustain strong performance in large-scale settings.

Table 5: Complexity reduction of the MILP solver for a 35-seller divisible submodular setting ($K = 25, \epsilon = 0.01$, 101 grid points). Profiles: bid, $value \sim U[0, 1]$, and $budget \sim U[0.25n, 0.5n]$. Strategies are added incrementally.

Strategy	# MILP	Per Instance Complexity (Mean ± Std)		
		# Constraints	# Binary Variables	Run Time (s)
Nothing	35	268,695	88,375	9.8950 ± 0.4205
+ Parallel-Transformation	1	7,677	2,525	0.2881 ± 0.0359
+ Zero-Violation	0.991	$7{,}613 \pm 698$	$2{,}504 \pm 230$	0.2832 ± 0.0437
+ IR-Screening	0.991	$4{,}766 \pm 998$	$1{,}080 \pm 459$	0.2187 ± 0.0627
+ Adaptive Keep-Choice	0.991	118 ± 158	3 ± 42	0.0043 ± 0.0137
Overall Reduction	**-97.17%**	**-99.96%**	**-99.99%**	**-99.96%**

Computational Efficiency and Acceleration Strategies. Table 5 highlights the effectiveness of our proposed acceleration strategies on 35-seller submodular procurement instances. The base formulation, denoted as Nothing, requires over $268,000$ constraints and takes nearly 10 seconds per instance. By incrementally applying Parallel-Transformation, Zero-Violation, IR-Screening, and Adaptive Keep-Choice, we reduce the number of constraints by 99.96% and the number of binary

variables by 99.99%. Consequently, the average runtime drops by 99.96% to only 0.0043 seconds per instance. These results demonstrate the substantial impact of our acceleration strategies, enabling practical deployment in near real-time auction settings.

Supplementary Experimental Results. Appendix E provides further empirical analyses, including: (1) robustness to complex valuations and cost distributions, such as Beta and irregular distributions; (2) utility retention after menu transformation; (3) out-of-setting generalization analyses; and (4) empirical verification of exact DSIC and strict budget feasibility.

6 Discussion and Future Work

In this paper, we introduced BFMNet, a scalable and expressive neural framework tailored for designing budget-feasible mechanisms in procurement auctions. By decoupling mechanism design into Transformer-based generation of self-bid-independent base menus and instance-level ex-post menu transformation, BFMNet addresses the fundamental scalability and expressivity bottlenecks of existing neural AMD paradigms. Theoretically, we proved that our mechanism strictly guarantees exact dominant-strategy incentive compatibility (DSIC), individual rationality (IR), and budget feasibility. Empirically, BFMNet significantly outperforms classical truthful baselines across diverse valuation settings and achieves highly competitive buyer utility compared with state-of-the-art neural baselines, while completely eliminating their inherent economic violations. Furthermore, our tailored MILP acceleration strategies reduce the computational overhead by over 99.9%, enabling near real-time deployment in large-scale markets.

Limitations. Despite its strong theoretical guarantees and empirical performance, our framework has a few limitations. First, while our acceleration strategies successfully bypass MILP optimization in the vast majority of cases, the strict economic guarantees still ultimately depend on external MILP solvers. This dependency makes our approach less suitable for deployment in highly resource-constrained environments. Second, the mechanism's performance is inherently sensitive to the discretization step size ϵ. This creates a fundamental trade-off: a smaller ϵ yields a finer grid and potentially higher utility, but also increases the computational overhead and solving time during the transformation phase.

Future Work. An interesting avenue for future research is extending BFMNet to multi-item settings, where each seller has multiple items available for procurement. However, multi-item environments are significantly more complex and naturally require much larger menu sizes to capture combinatorial preferences. Furthermore, our current *Adaptive Keep-Choice* acceleration method cannot be trivially extended to multi-item domains, which would substantially increase transformation time. Overcoming these computational bottlenecks and scaling BFMNet to combinatorial and multi-item budget-feasible mechanisms constitute a promising direction for future work.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 62402102, 62472093, 62572120, and 62572246; the Natural Science Foundation of Jiangsu Province under Grant Nos. BK20230024 and BK20241275; and the Jiangsu Province Talent Research Consortium. The authors declare no competing interests.

References

[1] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '03, page 137–146, New York, NY, USA, 2003. Association for Computing Machinery.

[2] Kai Han, Keke Huang, Xiaokui Xiao, Jing Tang, Aixin Sun, and Xueyan Tang. Efficient algorithms for adaptive influence maximization. *Proc. VLDB Endow.*, 11(9):1029–1040, May 2018.

[3] Jingwen Zhang, Yuezhou Wu, and Rong Pan. Incentive mechanism for horizontal federated learning based on reputation and reverse auction. In *Proceedings of the Web Conference 2021*, WWW '21, page 947–956, New York, NY, USA, 2021. Association for Computing Machinery.

[4] Xiang Liu, Hau Chan, Minming Li, Xianlong Zeng, Chenchen Fu, and Weiwei Wu. Care: Compatibility-aware incentive mechanisms for federated learning with budgeted requesters. In *IEEE INFOCOM 2025 - IEEE Conference on Computer Communications*, pages 1–10, 2025.

[5] Amazon mechanical turk. `https://www.mturk.com`. Accessed: 2026-04-27.

[6] Xiang Zhang, Guoliang Xue, Ruozhou Yu, Dejun Yang, and Jian Tang. Truthful incentive mechanisms for crowdsourcing. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 2830–2838, 2015.

[7] Yaron Singer. Budget feasible mechanisms. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 765–774, 2010.

[8] Shahar Dobzinski, Christos H. Papadimitriou, and Yaron Singer. Mechanisms for complement-free procurement. In *Proceedings of the 12th ACM Conference on Electronic Commerce*, EC '11, page 273–282, New York, NY, USA, 2011. Association for Computing Machinery.

[9] Kai Han, You Wu, He Huang, and Shuang Cui. Triple eagle: Simple, fast and practical budget-feasible mechanisms. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 33894–33911. Curran Associates, Inc., 2023.

[10] Ning Chen, Nick Gravin, and Pinyan Lu. *On the Approximability of Budget Feasible Mechanisms*, pages 685–699.

[11] Nima Anari, Gagan Goel, and Afshin Nikzad. Mechanism design for crowdsourcing: An optimal 1-1/e competitive budget-feasible mechanism for large markets. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 266–275, 2014.

[12] Nick Gravin, Yaonan Jin, Pinyan Lu, and Chenhao Zhang. Optimal budget-feasible mechanisms for additive valuations. *ACM Trans. Econ. Comput.*, 8(4), October 2020.

[13] Eric Balkanski, Pranav Garimidi, Vasilis Gkatzelis, Daniel Schoepflin, and Xizhi Tan. *Deterministic Budget-Feasible Clock Auctions*, pages 2940–2963.

[14] Jixian Zhang, Yi Zhang, Hao Wu, and Weidong Li. An ordered submodularity-based budget-feasible mechanism for opportunistic mobile crowdsensing task allocation and pricing. *IEEE Transactions on Mobile Computing*, 23(2):1278–1294, 2024.

[15] Eric Balkanski and Jason D. Hartline. Bayesian budget feasibility with posted pricing. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 189–203, Republic and Canton of Geneva, CHE, 2016. International World Wide Web Conferences Steering Committee.

[16] Xiang Liu, Hau Chan, Minming Li, and Weiwei Wu. Budget feasible mechanisms: A survey.

[17] Vincent Conitzer and Tuomas Sandholm. Complexity of mechanism design. *CoRR*, cs.GT/0205075, 2002.

[18] Vincent Conitzer and Tuomas Sandholm. Self-interested automated mechanism design and implications for optimal combinatorial auctions. In *Proceedings of the 5th ACM Conference on Electronic Commerce*, EC '04, page 132–141, New York, NY, USA, 2004. Association for Computing Machinery.

[19] Michael J. Curry. Automated mechanism design: A survey.

[20] Paul Duetting, Zhe Feng, Harikrishna Narasimhan, David Parkes, and Sai Srivatsa Ravindranath. Optimal auctions through deep learning. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1706–1715. PMLR, 09–15 Jun 2019.

[21] Michael Curry, Ping-yeh Chiang, Tom Goldstein, and John Dickerson. Certifying strategyproof auction networks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 4987–4998. Curran Associates, Inc., 2020.

[22] Zhijian Duan, Jingwu Tang, Yutong Yin, Zhe Feng, Xiang Yan, Manzil Zaheer, and Xiaotie Deng. A context-integrated transformer-based neural network for auction design. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 5609–5626. PMLR, 17–23 Jul 2022.

[23] Dmitry Ivanov, Iskander Safiulin, Igor Filippov, and Ksenia Balabaeva. Optimal-er auctions through attention. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 34734–34747. Curran Associates, Inc., 2022.

[24] Kevin Roberts. The characterization of implementable choice rules. *Aggregation and revelation of preferences*, 12(2):321–348, 1979.

[25] Anton Likhodedov and Tuomas Sandholm. Approximating revenue-maximizing combinatorial auctions. In *Proceedings of the 20th National Conference on Artificial Intelligence - Volume 1*, AAAI'05, page 267–273. AAAI Press, 2005.

[26] Michael J. Curry, Tuomas Sandholm, and John P. Dickerson. Differentiable economics for randomized affine maximizer auctions. *CoRR*, abs/2202.02872, 2022.

[27] Zhijian Duan, Haoran Sun, Yurong Chen, and Xiaotie Deng. A scalable neural network for dsic affine maximizer auction design. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 56169–56185. Curran Associates, Inc., 2023.

[28] Tonghan Wang, Yanchen Jiang, and David C. Parkes. Gemnet: Menu-based, strategy-proof multi-bidder auctions through deep learning, 2024.

[29] Vincent Conitzer. *Computational aspects of preference aggregation*. PhD thesis, Carnegie Mellon University, 2006. Chapter 6.

[30] Neehar Peri, Michael Curry, Samuel Dooley, and John Dickerson. Preferencenet: Encoding human preferences in auction design with deep learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 17532–17542. Curran Associates, Inc., 2021.

[31] Zhe Feng, Harikrishna Narasimhan, and David C. Parkes. Deep learning for revenue-optimal auctions with budgets. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '18, page 354–362, Richland, SC, 2018. International Foundation for Autonomous Agents and Multiagent Systems.

[32] Michael Curry, Vinzenz Thoma, Darshan Chakrabarti, Stephen McAleer, Christian Kroer, Tuomas Sandholm, Niao He, and Sven Seuken. Automated design of affine maximizer mechanisms in dynamic settings. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(9):9626–9635, Mar. 2024.

[33] Peter J. Hammond. Straightforward individual incentive compatibility in large economies. *The Review of Economic Studies*, 46(2):263–282, 04 1979.

[34] Weiran Shen, Pingzhong Tang, and Song Zuo. Computer-aided mechanism design: designing revenue-optimal mechanisms via neural networks. *CoRR*, abs/1805.03382, 2018.

[35] Paul Dütting, Zhe Feng, Harikrishna Narasimhan, David C. Parkes, and Sai Srivatsa Ravindranath. Optimal auctions through deep learning: Advances in differentiable economics. *J. ACM*, 71(1), February 2024.

[36] J.C. Rochet. The taxation principle and multi-time hamilton-jacobi equations. *Journal of Mathematical Economics*, 14(2):113–128, 1985.

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[38] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2026.

[39] Jad Rahme, Samy Jelassi, Joan Bruna, and S. Matthew Weinberg. A permutation-equivariant neural network architecture for auction design. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(6):5664–5672, May 2021.

[40] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

A BFMNet Architecture

In this section, we describe the architecture of the menu network, BFMNet. BFMNet serves as a conditional menu generator that produces high-quality base menus, which are then used as strong initial solutions for the subsequent ex-post menu transformation. Unlike GemNet [28], our ex-post transformation framework does not require Lipschitz continuity of the underlying neural network. This decoupling between expressive menu generation and economic enforcement allows us to avoid restrictive Lipschitz constraints and adopt more flexible neural architectures. As illustrated in Figure 2, BFMNet is built upon a permutation-equivariant Transformer architecture [37]. The attention-based design enables the network to capture rich interactions among sellers and market-level context, while permutation equivariance ensures that the generated menus respect the exchangeability of sellers [39]. This property also makes the architecture naturally compatible with varying market sizes, supporting the out-of-setting generalization evaluated in our experiments.

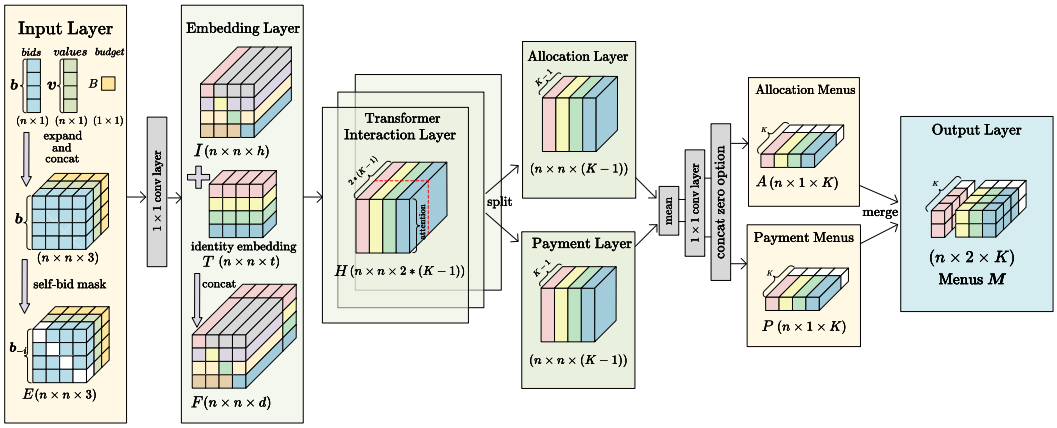


Figure 2: Architecture of BFMNet. BFMNet generates base menus for budget-constrained procurement auctions from the sellers' bid profile $\boldsymbol{b}$, valuation profile $\boldsymbol{v}$, and global budget B. For each seller i, a self-bid masking operation constructs the bid-independent context $\boldsymbol{b}_{-i}$, ensuring that the generated menu for seller i is independent of its own reported bid. A permutation-equivariant attention module captures high-order interactions among sellers and market-level features in $(\boldsymbol{b}_{-i}, \boldsymbol{v}, B)$. A zero option with zero allocation and zero payment is appended to each menu to guarantee exact individual rationality (IR).

A.1 Input Layer

The input layer takes as input the bid profile $\boldsymbol{b} \in \mathbb{R}^n_{\geq 0}$, the valuation profile $\boldsymbol{v} \in \mathbb{R}^n_{\geq 0}$, and the global budget $B \in \mathbb{R}_{\geq 0}$. We first construct a seller-pair interaction tensor of size $n \times n \times 3$ through expansion and concatenation, where each entry encodes the contextual information used for generating seller-specific menus. To enforce *self-bid independence* at the architectural level, we apply a structural self-bid mask that removes seller i's own bid b_i from the context used to generate its menu. As a

result, the candidate menu options for seller i depend only on the competing sellers' bids $\boldsymbol{b}_{-i}$ and the non-strategic profiles $\boldsymbol{v}$ and B. This architectural constraint aligns the menu network with the self-bid-independence requirement of the taxation principle, which is essential for preserving DSIC. The resulting input feature tensor is denoted by $E \in \mathbb{R}^{n\times n\times 3}$.

A.2 Embedding Layer

To extract higher-order features, the input feature tensor E is mapped into a high-dimensional latent space via a 1×1 convolutional layer, resulting in $I \in \mathbb{R}^{n\times n\times h}$. Subsequently, to enable the model to distinguish sellers with heterogeneous bid distributions, we concatenate I with an identifier embedding tensor $T \in \mathbb{R}^{n\times n\times t}$. This integration provides a seller-specific context, allowing the network to capture the individual impact of each seller on the global budget allocation, ultimately forming the augmented feature tensor $F \in \mathbb{R}^{n\times n\times d}$.

A.3 Transformer Interaction Layer

Within the interaction layer, we deploy a series of permutation-equivariant Transformer blocks with multi-head self-attention to capture complex nonlinear correlations among bids, valuations, and the global budget. This design effectively models the competitive dynamics among sellers under a shared budget constraint. Our architecture stacks three Transformer layers to enhance representational capacity. Following the Transformer blocks, a split operation is first employed to decouple the hidden features into distinct allocation and payment branches. Within each branch, 1×1 convolutions and spatial mean pooling operations are applied to aggregate the interactive context and project the features into dimension $K-1$. Sigmoid and Softplus activations are then applied to these $K-1$ generated items to ensure valid allocation ($x_i \in [0,1]$) and non-negative payments ($p_i \geq 0$), respectively. Finally, we append a hard-coded default zero option ($x_i^K = 0, p_i^K = 0$) to satisfy exact IR constraints, yielding the complete allocation menu $A \in \mathbb{R}^{n\times K}$ and payment menu $P \in \mathbb{R}^{n\times K}$.

A.4 Output Layer

In the output layer, the allocation menu A and payment menu P are merged to construct the final base menu set $\boldsymbol{M} \in \mathbb{R}^{n\times 2\times K}$ containing K options for each seller. This set serves as the input for the subsequent ex-post menu transformation, providing a highly refined initial solution while maintaining foundational economic properties, i.e., *self-bid independence* and exact IR prior to the MILP payment transformation.

B Sample Complexity

We provide a sample-complexity bound that simultaneously controls two generalization gaps: the gap between empirical and expected buyer utility, and the gap between empirical and expected total payment. We use covering numbers to measure the capacity of the procurement mechanism class $\mathcal{M}$. We define the $\ell_{\infty,1}$-distance between any two mechanisms $(\boldsymbol{x}, \boldsymbol{p}), (\boldsymbol{x}', \boldsymbol{p}') \in \mathcal{M}$ as:

$$\max_{\boldsymbol{b}} \left(\sum_{i\in[n]} |x_i(\boldsymbol{b}, \boldsymbol{v}, B) - x_i'(\boldsymbol{b}, \boldsymbol{v}, B)| + \sum_{i\in[n]} |p_i(\boldsymbol{b}, \boldsymbol{v}, B) - p_i'(\boldsymbol{b}, \boldsymbol{v}, B)| \right) \tag{10}$$

For any $r > 0$, let $\mathcal{N}_{\infty,1}(\mathcal{M}, r)$ denote the r-covering number, i.e., the minimum number of balls of radius r required to cover $\mathcal{M}$ under the $\ell_{\infty,1}$-distance. For the sample-complexity analysis, assume that the valuation function is 1-Lipschitz in the allocation vector under the ℓ_1 norm, i.e.,

$$|V(\boldsymbol{x}, \boldsymbol{v}) - V(\boldsymbol{x}', \boldsymbol{v})| \leq \|\boldsymbol{x} - \boldsymbol{x}'\|_1, \qquad \forall\, \boldsymbol{x}, \boldsymbol{x}' \in [0,1]^n.$$

We also assume that bids are uniformly bounded, such that $\max_{i\in[n]} b_i \leq 1$, and the sellers' reported bids are drawn independently from prior distributions F_i, with the joint distribution denoted by

$F = \prod_{i=1}^{n} F_i$. We introduce the following variance upper bounds over $\mathcal{M}$:

$$\begin{aligned}\sigma^2_{\text{utility}} &:= \sup_{(\boldsymbol{x}^\theta, \boldsymbol{p}^\theta) \in \mathcal{M}} \operatorname{Var}_{\boldsymbol{b} \sim F}\left[V(\boldsymbol{x}^\theta(\boldsymbol{b}, \boldsymbol{v}, B), \boldsymbol{v})\right], \\ \sigma^2_{\text{payment}} &:= \sup_{(\boldsymbol{x}^\theta, \boldsymbol{p}^\theta) \in \mathcal{M}} \operatorname{Var}_{\boldsymbol{b} \sim F}\left[\sum_{i \in [n]} p_i^\theta(\boldsymbol{b}, \boldsymbol{v}, B)\right].\end{aligned} \tag{11}$$

Let $\sigma^2 = \max\{\sigma^2_{\text{utility}}, \sigma^2_{\text{payment}}\}$. We obtain the following theorem.

Theorem 4. *Fix $\delta, \mu \in (0, 1)$. For any procurement mechanism $(\boldsymbol{x}^\theta, \boldsymbol{p}^\theta) \in \mathcal{M}$, if the sample size L of the training set $S \sim F^L$ satisfies:*

$$L \geq \frac{18\sigma^2 + 2n\mu}{\mu^2}\left(\ln\frac{4}{\delta} + \ln \mathcal{N}_{\infty,1}\left(\mathcal{M}, \frac{\mu}{3}\right)\right) \tag{12}$$

then with probability at least $1 - \delta$ over the draw of training the set S of L samples from F, the following two inequalities hold simultaneously for all procurement mechanisms $(\boldsymbol{x}^\theta, \boldsymbol{p}^\theta) \in \mathcal{M}$:

$$\begin{aligned}&\left|\mathbb{E}_{\boldsymbol{b} \sim F}\left[\sum_{i \in [n]} p_i^\theta(\boldsymbol{b}, \boldsymbol{v}, B)\right] - \frac{1}{L}\sum_{\ell \in [L]}\sum_{i \in [n]} p_i^\theta(\boldsymbol{b}^{(\ell)}, \boldsymbol{v}, B)\right| \leq \mu, \\ &\left|\mathbb{E}_{\boldsymbol{b} \sim F}\left[V(\boldsymbol{x}^\theta(\boldsymbol{b}, \boldsymbol{v}, B), \boldsymbol{v})\right] - \frac{1}{L}\sum_{\ell \in [L]} V(\boldsymbol{x}^\theta(\boldsymbol{b}^{(\ell)}, \boldsymbol{v}, B), \boldsymbol{v})\right| \leq \mu.\end{aligned} \tag{13}$$

Proof. The proof combines covering numbers [40, 22] with a generalization lemma based on concentration inequalities (Lemma 1). We first define the covering numbers for the buyer utility and payment function classes.

Covering Numbers $\mathcal{N}_{\infty,1}(\mathcal{V}, r)$ and $\mathcal{N}_\infty(\mathcal{V}_i, r)$: Let $\mathcal{V}_i$ be the class of marginal value functions representing the value of the buyer obtains from seller i over the mechanism space $\mathcal{M}$, defined as:

$$\mathcal{V}_i = \left\{V_i(\boldsymbol{x}, \boldsymbol{v}) : \mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0} \to \mathbb{R}_{\geq 0} \mid V_i(\boldsymbol{x}(\boldsymbol{b}, \boldsymbol{v}, B), \boldsymbol{v}) = V(\boldsymbol{x}(\boldsymbol{b}, \boldsymbol{v}, B), \boldsymbol{v}) - V(\boldsymbol{x}_{-i}(\boldsymbol{b}, \boldsymbol{v}, B), \boldsymbol{v})\right\}. \tag{14}$$

Similarly, let $\mathcal{V}$ be the family of buyer valuation profiles over $\mathcal{M}$. The $\ell_{\infty,1}$-distance between two buyer valuation profiles $V(\boldsymbol{x}, \boldsymbol{v})$ and $V'(\boldsymbol{x}, \boldsymbol{v})$ is defined as:

$$\max_{\boldsymbol{b} \sim F} |V(\boldsymbol{x}(\boldsymbol{b}, \boldsymbol{v}, B), \boldsymbol{v}) - V'(\boldsymbol{x}(\boldsymbol{b}, \boldsymbol{v}, B), \boldsymbol{v})|. \tag{15}$$

Let $\mathcal{N}_{\infty,1}(\mathcal{V}, r)$ denote the r-covering number for $\mathcal{V}$ under this distance. Furthermore, we define the ℓ_∞-distance between $V_i(\boldsymbol{x}, \boldsymbol{v})$ and $V_i'(\boldsymbol{x}, \boldsymbol{v})$ as $\max_{\boldsymbol{b} \sim F} |V_i(\boldsymbol{x}(\boldsymbol{b}, \boldsymbol{v}, B), \boldsymbol{v}) - V_i'(\boldsymbol{x}(\boldsymbol{b}, \boldsymbol{v}, B), \boldsymbol{v})|$, and let $\mathcal{N}_\infty(\mathcal{V}_i, r)$ denote the r-covering number for $\mathcal{V}_i$ under the ℓ_∞-distance.

Covering Numbers $\mathcal{N}_{\infty,1}(\mathcal{P}, r)$ and $\mathcal{N}_\infty(\mathcal{P}_i, r)$: Let $\mathcal{P}_i$ be the class of payment functions for seller i over $\mathcal{M}$, defined as:

$$\mathcal{P}_i = \left\{p_i : \mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0} \mid \boldsymbol{p} \in \mathcal{P}, \text{ and } p_i \text{ is the } i\text{-th component of } \boldsymbol{p}\right\}. \tag{16}$$

Based on the definitions above, we denote the r-covering number for the payment profile class $\mathcal{P}$ under the $\ell_{\infty,1}$-distance as $\mathcal{N}_{\infty,1}(\mathcal{P}, r)$, and the r-covering number for the individual payment class $\mathcal{P}_i$ under the ℓ_∞-distance as $\mathcal{N}_\infty(\mathcal{P}_i, r)$.

Lemma 1 (Generalization via Bernstein's Inequality)**.** *Let $S = \{z_1, \ldots, z_L\} \sim F^L$ be a dataset of L samples drawn i.i.d. from a distribution F over the domain $\mathcal{Z}$. Assume that for all functions $f \in \mathcal{F}$ and all $z \in \mathcal{Z}$, we have $f(z) \in [a, b]$. We define the ℓ_∞-distance between any two functions $f, f' \in \mathcal{F}$ as $\max_{z \in \mathcal{Z}} |f(z) - f'(z)|$, and let $\mathcal{N}_\infty(\mathcal{F}, r)$ denote the r-covering number of $\mathcal{F}$ under this ℓ_∞-distance.*

Define the expected value $L_F(f) = \mathbb{E}_{z \sim F}[f(z)]$ and the empirical average $L_S(f) = \frac{1}{L}\sum_{\ell \in [L]} f(z_\ell)$. Let $\sigma^2 := \sup_{f \in \mathcal{F}} \operatorname{Var}_{z \sim F}(f(z))$ be the uniform variance upper bound. Then, for any $\mu > 0$, we have:

$$\mathbb{P}_{S \sim F^L}\left[\exists f \in \mathcal{F} : |L_S(f) - L_F(f)| > \mu\right] \leq 2\mathcal{N}_\infty\left(\mathcal{F}, \frac{\mu}{3}\right)\exp\left(-\frac{L\mu^2}{18\sigma^2 + 2(b-a)\mu}\right). \tag{17}$$

Proof. Let $\mathcal{F}_r$ be a minimal r-cover of $\mathcal{F}$ under the ℓ_∞-distance, such that $|\mathcal{F}_r| = \mathcal{N}_\infty(\mathcal{F}, r)$. For any $f \in \mathcal{F}$, let $f_r \in \mathcal{F}_r$ denote its closest function in the cover. By definition, for all $z \in \mathcal{Z}$, we have $|f(z) - f_r(z)| \leq r$.

Setting $r = \mu/3$, we obtain:

$$
\begin{aligned}
&\mathbb{P}_{S\sim F^L}\left[\exists f \in \mathcal{F}, |L_S(f) - L_F(f)| > \mu\right] \\
&\leq \mathbb{P}_{S\sim F^L}\left[\exists f \in \mathcal{F}, |L_S(f) - L_S(f_r)| + |L_S(f_r) - L_F(f_r)| + |L_F(f_r) - L_F(f)| > \mu\right] \\
&\leq \mathbb{P}_{S\sim F^L}\left[\exists f \in \mathcal{F}, r + |L_S(f_r) - L_F(f_r)| + r > \mu\right] \\
&\leq \mathbb{P}_{S\sim F^L}\left[\exists f_r \in \mathcal{F}_r, |L_S(f_r) - L_F(f_r)| > \frac{\mu}{3}\right] \\
&\leq \sum_{f_r \in \mathcal{F}_r} \mathbb{P}_{S\sim F^L}\left[|L_S(f_r) - L_F(f_r)| > \frac{\mu}{3}\right] \quad \text{(Union Bound)} \qquad (18) \\
&\leq 2|\mathcal{F}_r| \exp\left(-\frac{L(\mu/3)^2}{2\sigma^2 + \frac{2}{3}(b-a)(\mu/3)}\right) \quad \text{(Bernstein's Inequality)} \\
&\leq 2\mathcal{N}_\infty\left(\mathcal{F}, \frac{\mu}{3}\right) \exp\left(-\frac{L\mu^2}{18\sigma^2 + 2(b-a)\mu}\right).
\end{aligned}
$$

□

The following two lemmas (Lemma 2 and Lemma 3) establish the relationship between the covering number of the mechanism class and those of the payment and buyer utility function classes.

Lemma 2. *The covering number of the payment class $\mathcal{P}$ is bounded by the covering number of the mechanism class $\mathcal{M}$:*

$$\mathcal{N}_{\infty,1}(\mathcal{P}, \mu) \leq \mathcal{N}_{\infty,1}(\mathcal{M}, \mu). \qquad (19)$$

Proof. By the definition of the covering number for the auction mechanism class $\mathcal{M}$, there exists a covering set $\hat{\mathcal{M}}$ for $\mathcal{M}$ of size $|\hat{\mathcal{M}}| \leq \mathcal{N}_{\infty,1}(\mathcal{M}, \mu)$. This implies that for any mechanism $(\boldsymbol{x}, \boldsymbol{p}) \in \mathcal{M}$, there exists an approximating mechanism $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{p}}) \in \hat{\mathcal{M}}$ such that for any $(\boldsymbol{b}, \boldsymbol{v}, B)$:

$$\sum_{i\in[n]} |x_i(\boldsymbol{b}, \boldsymbol{v}, B) - \hat{x}_i(\boldsymbol{b}, \boldsymbol{v}, B)| + \sum_{i\in[n]} |p_i(\boldsymbol{b}, \boldsymbol{v}, B) - \hat{p}_i(\boldsymbol{b}, \boldsymbol{v}, B)| \leq \mu.$$

As a result, we can have $\hat{\mathcal{P}} = \{\hat{\boldsymbol{p}} \mid (\hat{\boldsymbol{x}}, \hat{\boldsymbol{p}}) \in \hat{\mathcal{M}}\}$. It follows that for any payment rule $\boldsymbol{p} \in \mathcal{P}$, there must exist a corresponding $\hat{\boldsymbol{p}} \in \hat{\mathcal{P}}$ such that for any $(\boldsymbol{b}, \boldsymbol{v}, B)$:

$$\sum_{i\in[n]} |p_i(\boldsymbol{b}, \boldsymbol{v}, B) - \hat{p}_i(\boldsymbol{b}, \boldsymbol{v}, B)| \leq \mu. \qquad (20)$$

Therefore, we have $\mathcal{N}_{\infty,1}(\mathcal{P}, \mu) \leq \mathcal{N}_{\infty,1}(\mathcal{M}, \mu)$. □

Lemma 3. *The covering number of the valuation profile class $\mathcal{V}$ is bounded by the covering number of the mechanism class $\mathcal{M}$:*

$$\mathcal{N}_{\infty,1}(\mathcal{V}, \mu) \leq \mathcal{N}_{\infty,1}(\mathcal{M}, \mu). \qquad (21)$$

Proof. By the definition of the covering number for the auction mechanism class $\mathcal{M}$, there exists a covering set $\hat{\mathcal{M}}$ for $\mathcal{M}$ of size $|\hat{\mathcal{M}}| \leq \mathcal{N}_{\infty,1}(\mathcal{M}, \mu)$. This implies that for any mechanism $(x, p) \in \mathcal{M}$, there exists a mechanism $(\hat{x}, \hat{p}) \in \hat{\mathcal{M}}$ such that for any $(\boldsymbol{b}, \boldsymbol{v}, B)$:

$$\sum_{i\in[n]} |x_i(\boldsymbol{b}, \boldsymbol{v}, B) - \hat{x}_i(\boldsymbol{b}, \boldsymbol{v}, B)| + \sum_{i\in[n]} |p_i(\boldsymbol{b}, \boldsymbol{v}, B) - \hat{p}_i(\boldsymbol{b}, \boldsymbol{v}, B)| \leq \mu.$$

To measure the capacity of the valuation function class $\mathcal{V}$, we construct an induced set $\hat{\mathcal{V}} = \{V(\hat{\boldsymbol{x}}, \boldsymbol{v}) \mid (\hat{\boldsymbol{x}}, \hat{\boldsymbol{p}}) \in \hat{\mathcal{M}}\}$. Clearly, the cardinality of this set satisfies $|\hat{\mathcal{V}}| \leq |\hat{\mathcal{M}}|$. By the assumed 1-Lipschitz

continuity of V in the allocation vector, we have, for any $(\boldsymbol{b}, \boldsymbol{v}, B)$:

$$\begin{aligned}
&|V(\boldsymbol{x}(\boldsymbol{b}, \boldsymbol{v}, B), \boldsymbol{v}) - V(\hat{\boldsymbol{x}}(\boldsymbol{b}, \boldsymbol{v}, B), \boldsymbol{v})| \\
\leq &\sum_{i \in [n]} |x_i(\boldsymbol{b}, \boldsymbol{v}, B) - \hat{x}_i(\boldsymbol{b}, \boldsymbol{v}, B)| \\
\leq &\sum_{i \in [n]} |x_i(\boldsymbol{b}, \boldsymbol{v}, B) - \hat{x}_i(\boldsymbol{b}, \boldsymbol{v}, B)| + \sum_{i \in [n]} |p_i(\boldsymbol{b}, \boldsymbol{v}, B) - \hat{p}_i(\boldsymbol{b}, \boldsymbol{v}, B)| \\
\leq &\mu.
\end{aligned} \tag{22}$$

Therefore, we have $\mathcal{N}_{\infty,1}(\mathcal{V}, \mu) \leq \mathcal{N}_{\infty,1}(\mathcal{M}, \mu)$. □

For any $\mu, \delta \in (0, 1)$, let $\sigma^2 = \max\{\sigma^2_{\text{utility}}, \sigma^2_{\text{payment}}\}$. When the sample size L satisfies the following condition:

$$L \geq \frac{18\sigma^2 + 2n\mu}{\mu^2} \left(\ln \frac{4}{\delta} + \ln \mathcal{N}_{\infty,1} \left(\mathcal{M}, \frac{\mu}{3} \right) \right), \tag{23}$$

we can derive the final probability bounds by integrating the preceding lemmas. By combining Lemma 1 and Lemma 2, we establish the generalization error bound for the total payment. Specifically, for a sample set S drawn i.i.d. from distribution F^L, the probability that there exists a mechanism $(\boldsymbol{x}^\theta, \boldsymbol{p}^\theta) \in \mathcal{M}$ whose expected total payment deviates from its empirical average by more than μ is bounded as follows:

$$\begin{aligned}
&\mathbb{P}_{S \sim F^L} \left[\exists (\boldsymbol{x}^\theta, \boldsymbol{p}^\theta) \in \mathcal{M}, \left| \sum_{i \in [n]} \mathbb{E}[p_i^\theta(\boldsymbol{b}, \boldsymbol{v}, B)] - \frac{1}{L} \sum_{\ell \in [L]} \sum_{i \in [n]} p_i^\theta(\boldsymbol{b}^{(\ell)}, \boldsymbol{v}, B) \right| > \mu \right] \\
&\leq 2\mathcal{N}_\infty \left(\mathcal{P}, \frac{\mu}{3} \right) \exp \left(- \frac{L\mu^2}{18\sigma^2_{\text{payment}} + 2(b-a)\mu} \right) \\
&\leq 2\mathcal{N}_{\infty,1} \left(\mathcal{M}, \frac{\mu}{3} \right) \exp \left(- \frac{L\mu^2}{18\sigma^2 + 2n\mu} \right) \\
&\leq \frac{\delta}{2}.
\end{aligned} \tag{24}$$

Similarly, by combining Lemma 1 and Lemma 3, we obtain the probability bound for the buyer's utility (valuation):

$$\begin{aligned}
&\mathbb{P}_{S \sim F^L} \left[\exists (\boldsymbol{x}^\theta, \boldsymbol{p}^\theta) \in \mathcal{M}, \left| \mathbb{E}[V(\boldsymbol{x}, \boldsymbol{v})] - \frac{1}{L} \sum_{\ell \in [L]} V(\boldsymbol{x}^{(\ell)}, \boldsymbol{v}) \right| > \mu \right] \\
&\leq 2\mathcal{N}_\infty \left(\mathcal{V}, \frac{\mu}{3} \right) \exp \left(- \frac{L\mu^2}{18\sigma^2_{\text{utility}} + 2(b-a)\mu} \right) \\
&\leq 2\mathcal{N}_{\infty,1} \left(\mathcal{M}, \frac{\mu}{3} \right) \exp \left(- \frac{L\mu^2}{18\sigma^2 + 2n\mu} \right) \\
&\leq \frac{\delta}{2}.
\end{aligned} \tag{25}$$

Applying the union bound, the probability that at least one of the above violations occurs is at most $\frac{\delta}{2} + \frac{\delta}{2} = \delta$. Therefore, with probability at least $1 - \delta$, equation 13 hold. We complete the proof of Theorem 4.

□

C Error Bounds of SoftMax Approximation

During the training phase, we approximate the discrete *bidder optimization*, i.e., the *argmax* selection from each seller's menu, using a *SoftMax* operation with scaling factor $\tau > 0$.

In this section, we fix the public parameters and a finite menu profile $\boldsymbol{M} = (M_1, \ldots, M_n)$, where each menu has K options with allocations in $[0, 1]$. Bids are drawn from a joint distribution whose k-th marginal has a density f_k on $[0, 1]$ satisfying $0 \le f_k \le \mathcal{X}_k$. Each menu contains no duplicate options, and its utility-maximizing option is unique almost everywhere on $[0, 1]$.

As in Appendix B, we assume that the buyer's valuation is 1-Lipschitz in the allocation vector under the ℓ_1 norm:

$$|V(\boldsymbol{x}, \boldsymbol{v}) - V(\boldsymbol{x}', \boldsymbol{v})| \le \|\boldsymbol{x} - \boldsymbol{x}'\|_1, \qquad \boldsymbol{x}, \boldsymbol{x}' \in [0, 1]^n. \tag{26}$$

Let $\boldsymbol{x}^\tau(\boldsymbol{b})$ and $\boldsymbol{p}^\tau(\boldsymbol{b})$ denote the SoftMax-weighted allocation and payment vectors, and let $\boldsymbol{x}^*(\boldsymbol{b})$ and $\boldsymbol{p}^*(\boldsymbol{b})$ denote those selected by exact bidder optimization. All expectations below are over bids, with the menus and public parameters held fixed.

Theorem 5 (Approximation Error)**.** *Under the assumptions above, for every $\tau > 0$, the following bounds hold:*

1. *The expected absolute error in buyer utility satisfies*

$$\mathbb{E}_{\boldsymbol{b}}[|V(\boldsymbol{x}^\tau(\boldsymbol{b}), \boldsymbol{v}) - V(\boldsymbol{x}^*(\boldsymbol{b}), \boldsymbol{v})|] \le \sum_{k \in [n]} \mathcal{X}_k \left(\frac{2}{\tau} A_k(M_k) + K(K-1) e^{-\tau I_k} \right). \tag{27}$$

2. *The expected absolute error in total payment satisfies*

$$\mathbb{E}_{\boldsymbol{b}} \left[\left| \sum_{k \in [n]} p_k^\tau(\boldsymbol{b}) - \sum_{k \in [n]} p_k^*(\boldsymbol{b}) \right| \right] \le \sum_{k \in [n]} P_k \mathcal{X}_k \left(\frac{2}{\tau} A_k(M_k) + K(K-1) e^{-\tau I_k} \right). \tag{28}$$

Here, P_k, $A_k(M_k)$, and I_k are defined below. For the fixed menu profile, both expected absolute errors are $\mathcal{O}(1/\tau)$ as $\tau \to \infty$.

Proof. We first bound the payment error contributed by a single seller $k \in [n]$. Let $P_k = \max_{i,j \in [K]} |p_k^i - p_k^j|$ be the maximum payment difference in seller k's menu.

Let $i_k^*(b_k)$ be the index selected by exact bidder optimization, with ties broken in favor of the lowest-payment option, and write $(x_k^*(b_k), p_k^*(b_k)) = (x_k^{i_k^*(b_k)}, p_k^{i_k^*(b_k)})$. We suppress the dependence on b_k when convenient.

For any two items $i, j \in [K]$, define

$$g_k^{i,j}(b_k) := u_k^i(b_k) - u_k^j(b_k) = (p_k^i - p_k^j) - (x_k^i - x_k^j) b_k. \tag{29}$$

Winning Region. Define

$$W_k^i := \{b_k \in [0, 1] \mid u_k^i(b_k) \ge u_k^j(b_k),\ \forall j \in [K]\}. \tag{30}$$

Since utilities are affine in b_k, each W_k^i is a possibly empty closed interval.

Non-degeneracy. Let

$$T_k := \{b_k \in [0, 1] \mid \exists i \ne j,\ u_k^i(b_k) = u_k^j(b_k) = \max_{\ell \in [K]} u_k^\ell(b_k)\}. \tag{31}$$

By assumption, T_k is a Lebesgue null set, so the optimal option is unique almost everywhere.

Adjacent and Non-adjacent Sets. Let

$$\mathcal{E}_k := \big\{\{i, j\} \subseteq [K] \mid i \ne j,\ W_k^i \cap W_k^j \ne \emptyset\big\}. \tag{32}$$

For any $\{i, j\} \in \mathcal{E}_k$, we have $|x_k^i - x_k^j| > 0$: otherwise, a tie would imply identical options, which are excluded. Define

$$A_k(M_k) := \sum_{\{i,j\} \in \mathcal{E}_k} \frac{1}{|x_k^i - x_k^j|}, \qquad I_k := \inf\big\{g_k^{i,j}(b_k) \mid b_k \in W_k^i,\ \{i, j\} \notin \mathcal{E}_k,\ i \ne j\big\}. \tag{33}$$

For every non-adjacent pair and every $b_k \in W_k^i$, the gap $g_k^{i,j}(b_k)$ is strictly positive; equality would imply $b_k \in W_k^j$. Compactness of the nonempty winning regions and finiteness of the menu therefore imply $I_k > 0$ whenever its defining set is nonempty. If that set is empty, we set $I_k = +\infty$ and interpret $e^{-\tau I_k}$ as zero. Empty sums in $A_k(M_k)$ are taken to be zero.

For $j \in [K]$, let

$$q_k^j(b_k) := \frac{e^{\tau u_k^j(b_k)}}{\sum_{\ell\in[K]} e^{\tau u_k^\ell(b_k)}}, \qquad x_k^\tau(b_k) := \sum_{j\in[K]} x_k^j q_k^j(b_k), \qquad p_k^\tau(b_k) := \sum_{j\in[K]} p_k^j q_k^j(b_k). \tag{34}$$

Define

$$\mathrm{Err}_k^{\mathrm{payment}}(\tau) := \mathbb{E}_{b_k}\big[|p_k^\tau(b_k) - p_k^*(b_k)|\big]. \tag{35}$$

For $b_k \notin T_k$, using $\sum_j q_k^j = 1$ gives

$$|p_k^\tau - p_k^*| = \left|\sum_{j\neq i_k^*} (p_k^j - p_k^*) q_k^j\right| \le P_k \sum_{j\neq i_k^*} q_k^j. \tag{36}$$

On W_k^i, we have $q_k^j \le e^{-\tau g_k^{i,j}(b_k)}$. Since the winning regions partition $[0,1]$ up to a null set,

$$\mathrm{Err}_k^{\mathrm{payment}}(\tau) \le P_k \mathcal{X}_k \sum_{i\in[K]} \sum_{j\neq i} \int_{W_k^i} e^{-\tau g_k^{i,j}(b_k)}\, db_k. \tag{37}$$

Let $Y_k^{i,j}(\tau) := \int_{W_k^i} e^{-\tau g_k^{i,j}(b_k)}\, db_k$. We analyze this integral in two cases.

Case 1: Adjacent items ($\{i,j\} \in \mathcal{E}_k$). There exists $m \in W_k^i \cap W_k^j$ with $g_k^{i,j}(m) = 0$. For $b_k \in W_k^i$, $g_k^{i,j}(b_k) = |x_k^i - x_k^j|\,|b_k - m|$.

The change of variables $t = g_k^{i,j}(b_k)$ has absolute Jacobian $1/|x_k^i - x_k^j|$. Extending the transformed integration range to $[0, +\infty)$ gives

$$Y_k^{i,j}(\tau) \le \int_0^{+\infty} \frac{e^{-\tau t}}{|x_k^i - x_k^j|}\, dt = \frac{1}{\tau |x_k^i - x_k^j|}. \tag{38}$$

Case 2: Non-adjacent items ($\{i,j\} \notin \mathcal{E}_k$). If W_k^i is empty, then $Y_k^{i,j}(\tau) = 0$. Otherwise, $g_k^{i,j}(b_k) \ge I_k$ on W_k^i, whose length is at most one, so

$$Y_k^{i,j}(\tau) \le \int_{W_k^i} e^{-\tau I_k}\, db_k \le e^{-\tau I_k}. \tag{39}$$

Combining the two cases yields

$$\mathrm{Err}_k^{\mathrm{payment}}(\tau) \le P_k \mathcal{X}_k \left[\sum_{i\in[K]} \sum_{\substack{j\neq i \\ \{i,j\}\in\mathcal{E}_k}} \frac{1}{\tau |x_k^i - x_k^j|} + \sum_{i\in[K]} \sum_{\substack{j\neq i \\ \{i,j\}\notin\mathcal{E}_k}} e^{-\tau I_k} \right]. \tag{40}$$

Each unordered adjacent pair occurs twice in the first sum. Thus,

$$\sum_{(i,j):\{i,j\}\in\mathcal{E}_k} \frac{1}{\tau |x_k^i - x_k^j|} = 2 \sum_{\{i,j\}\in\mathcal{E}_k} \frac{1}{\tau |x_k^i - x_k^j|} = \frac{2}{\tau} A_k(M_k). \tag{41}$$

The number of non-adjacent ordered pairs is at most $K(K-1)$. Consequently,

$$\mathrm{Err}_k^{\mathrm{payment}}(\tau) \le P_k \mathcal{X}_k \left(\frac{2}{\tau} A_k(M_k) + K(K-1) e^{-\tau I_k} \right). \tag{42}$$

Because each fixed-menu outcome depends only on its seller's bid, the triangle inequality and the single-seller bounds give

$$\mathbb{E}_{\boldsymbol{b}}\left[\left|\sum_{k\in[n]} p_k^{\tau}(\boldsymbol{b}) - \sum_{k\in[n]} p_k^{*}(\boldsymbol{b})\right|\right] \le \sum_{k\in[n]} \mathrm{Err}_k^{\mathrm{payment}}(\tau) \le \sum_{k\in[n]} P_k \mathcal{X}_k \left(\frac{2}{\tau} A_k(M_k) + K(K-1)e^{-\tau I_k}\right). \tag{43}$$

We next bound the buyer utility error. Since allocations lie in $[0,1]$,

$$|x_k^{\tau} - x_k^{*}| = \left|\sum_{j\neq i_k^{*}} q_k^j (x_k^j - x_k^{*})\right| \le \sum_{j\neq i_k^{*}} q_k^j. \tag{44}$$

The 1-Lipschitz assumption implies

$$|V(\boldsymbol{x}^{\tau}, \boldsymbol{v}) - V(\boldsymbol{x}^{*}, \boldsymbol{v})| \le \sum_{k\in[n]} |x_k^{\tau} - x_k^{*}| \le \sum_{k\in[n]} \sum_{j\neq i_k^{*}} q_k^j. \tag{45}$$

The same winning-region integration bounds give

$$\mathbb{E}_{b_k}\left[\sum_{j\neq i_k^{*}} q_k^j\right] \le \mathcal{X}_k \sum_{i\in[K]} \sum_{j\neq i} Y_k^{i,j}(\tau) \le \mathcal{X}_k \left(\frac{2}{\tau} A_k(M_k) + K(K-1)e^{-\tau I_k}\right). \tag{46}$$

Taking expectations and summing over sellers therefore yields

$$\mathbb{E}_{\boldsymbol{b}}[|V(\boldsymbol{x}^{\tau}(\boldsymbol{b}), \boldsymbol{v}) - V(\boldsymbol{x}^{*}(\boldsymbol{b}), \boldsymbol{v})|] \le \sum_{k\in[n]} \mathcal{X}_k \left(\frac{2}{\tau} A_k(M_k) + K(K-1)e^{-\tau I_k}\right). \tag{47}$$

For the fixed finite menu profile, $A_k(M_k)$ and P_k are finite, and each nonempty comparison set has $I_k > 0$. Thus both expected absolute errors are $\mathcal{O}(1/\tau)$ as $\tau \to \infty$.

□

D Economic Guarantees

In this section, we formally establish the theoretical foundation of our proposed framework. We prove that the menu-based ex-post transformed mechanism strictly satisfies individual rationality (IR), dominant-strategy incentive compatibility (DSIC), and budget feasibility.

Theorem 6 (Individual Rationality)**.** *The proposed menu-based ex-post transformed mechanism guarantees individual rationality (IR) for all participating sellers.*

Proof. By design, the transformed menu always retains the default zero option $(0, 0)$, i.e., $\Delta p_i^K = 0$, which yields exactly zero utility. Under *bidder optimization*, a seller $i \in [n]$ who truthfully reports their cost selects the item that maximizes their utility, thereby ensuring a non-negative utility:

$$u_i = \max_{k\in[K]} (p_i^k - \Delta p_i^k - x_i^k \cdot c_i) \ge 0. \tag{48}$$

□

Theorem 7 (Dominant-Strategy Incentive Compatibility (DSIC))**.** *The proposed menu-based ex-post transformed mechanism ensures DSIC in procurement auctions.*

Proof. For each seller $i \in [n]$, both the generation of the base menu M_i and the subsequent payment transformation procedure are constructed independently of seller i's true cost c_i, reported bid b_i, and rounded grid bid $\tilde{b}_i$. Since the mechanism assigns the utility-maximizing menu item through *bidder optimization*, the requirements of the Taxation Principle (Theorem 1) are strictly satisfied, thereby guaranteeing DSIC. □

Lemma 4 (Grid Choice Stability)**.** *For any seller $i \in [n]$, let $* = \arg\max_{k\in[K]} u_i^k(b_{i,g})$ denote the index of the utility-maximizing menu item at grid point $b_{i,g}$. Under the ex-post menu transformation with safety margin $sm = \epsilon/2$, $*$ remains the optimal choice for any continuous bid $b_i \in [b_{i,g} - \epsilon/2, b_{i,g} + \epsilon/2]$.*

Proof. The MILP transformation enforces a safety margin $sm = \epsilon/2$ against any alternative item $k \in [K], k \neq *$, i.e., $u_i^*(b_{i,g}) - u_i^k(b_{i,g}) \geq \epsilon/2$. For any continuous bid $b_i = b_{i,g} + \delta$ where $|\delta| \leq \epsilon/2$, the utility difference becomes:

$$\begin{aligned} u_i^*(b_i) - u_i^k(b_i) &= \big(p_{i,g}^* - x_{i,g}^*(b_{i,g} + \delta)\big) - \big(p_i^k - x_i^k(b_{i,g} + \delta)\big) \\ &= \big(u_i^*(b_{i,g}) - u_i^k(b_{i,g})\big) - \delta(x_{i,g}^* - x_i^k) \\ &\geq \epsilon/2 - |\delta| \cdot |x_{i,g}^* - x_i^k|. \end{aligned} \tag{49}$$

Since the allocation probabilities satisfy $x_i \in [0, 1]$, the allocation difference is bounded by $|x_{i,g}^* - x_i^k| \leq 1$. Given $|\delta| \leq \epsilon/2$, it follows that $u_i^*(b_i) - u_i^k(b_i) \geq 0$. Thus, $*$ remains a utility-maximizing option for any continuous bid $b_i \in [b_{i,g} - \epsilon/2, b_{i,g} + \epsilon/2]$. □

Lemma 5 (Grid Budget Feasibility)**.** *For any discrete bid profile $\tilde{\boldsymbol{b}} \in \mathcal{G}$, the sum of the transformed payments for the optimal menu items across all sellers is bounded by the budget B.*

Proof. For any discrete grid profile $\tilde{\boldsymbol{b}}$, our MILP constraints bound the transformed payment of the chosen item for each seller i by the safety payment $sp_i(\tilde{\boldsymbol{b}})$. Summing these payments across all n sellers yields:

$$\sum_{i\in[n]} \Big(p_i^*(\tilde{\boldsymbol{b}}) - \Delta p_i^*(\tilde{\boldsymbol{b}})\Big) \leq \sum_{i\in[n]} sp_i(\tilde{\boldsymbol{b}}) \leq B. \tag{50}$$

The last inequality follows because the sum of safety payments equals B when the total base-menu payment is positive, and equals zero otherwise. Thus, budget feasibility holds at every discrete grid profile. □

Theorem 8 (Budget Feasibility)**.** *The proposed menu-based ex-post transformed mechanism strictly enforces budget feasibility for any procurement instance.*

Proof. For any continuous bid profile $\boldsymbol{b}$, let $\tilde{\boldsymbol{b}}$ denote its nearest grid profile. By the definition of the grid, each continuous bid b_i falls within $[\tilde{b}_i - \epsilon/2, \tilde{b}_i + \epsilon/2]$.

For each seller i, the transformed menu is identical at $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$, since it depends only on the other sellers' rounded bids and the public parameters. By Lemma 4, the option selected at $\tilde{\boldsymbol{b}}$ remains a utility-maximizing option at $\boldsymbol{b}$. Under the lowest-payment tie-breaking rule, the payment selected at $\boldsymbol{b}$ is therefore no greater than that selected at $\tilde{\boldsymbol{b}}$.

Summing over all sellers and applying Lemma 5 yields:

$$\sum_{i\in[n]} (p_i^*(\boldsymbol{b}) - \Delta p_i^*(\boldsymbol{b})) \leq \sum_{i\in[n]} \Big(p_i^*(\tilde{\boldsymbol{b}}) - \Delta p_i^*(\tilde{\boldsymbol{b}})\Big) \leq B. \tag{51}$$

□

E Supplementary Experiments

This section provides additional empirical results that complement the main experimental findings and further examine the robustness, efficiency, and economic validity of BFMNet. We first evaluate BFMNet under broader experimental settings, including subadditive valuations, asymmetric bid distributions, and irregular bid distributions. We then analyze how the ex-post menu transformation affects the utility of the learned base menus, and provide a fine-grained ablation study of the proposed MILP acceleration strategies in small-scale settings. Finally, we present visual evidence for out-of-setting generalization and empirically verify exact DSIC and strict budget feasibility.

E.1 Robustness to Complex Valuations and Cost Distributions

To examine the robustness of BFMNet beyond the standard experimental settings, we evaluate it under more challenging valuation functions and cost distributions. Table 6 reports the results under a subadditive valuation function. BFMNet preserves exact economic guarantees while achieving utility comparable to soft-penalty neural baselines in divisible settings. In indivisible settings, it substantially outperforms classical truthful baselines, including *Clock Auction* [13] and *Greedy Subadditive* [8].

We further test BFMNet under heterogeneous and irregular cost distributions, which better reflect the variability of real-world procurement markets. Table 7 considers asymmetric Beta distributions, where each seller's cost is sampled from a distinct distribution parameterization. Table 8 evaluates an irregular discontinuous mixture distribution. Across both settings, BFMNet consistently maintains exact DSIC and strict budget feasibility. By contrast, soft-constraint neural baselines, including *RegretNet* and *CITransNet*, exhibit non-zero regret and noticeable budget violations, highlighting the importance of our ex-post enforcement mechanism under distributional complexity.

Table 6: Performance evaluation under **subadditive** valuations. The buyer's valuation is defined as $V(\boldsymbol{x}, \boldsymbol{v}) = \ln(1 + s + 0.5 \sin^2 s)$, where $s = \sum_{i \in [n]} x_i v_i$. Profiles: $bid, value \sim U[0, 1]$, and $budget \sim U[2, 4]$.

Setting	Method	$n = 2$			$n = 4$			$n = 6$			$n = 8$			$n = 10$		
		val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓
Divisible	**OPT** (Fractional)	0.8123	N/A	-	1.1897	N/A	-	1.3896	N/A	-	1.5597	N/A	-	1.6855	N/A	-
	CITransNet	0.8123	0.0084	0.0001	1.1363	0.0039	0.0004	1.2567	0.0009	0.0003	1.3227	0.0007	0.0006	1.4228	0.0014	0.0010
	RegretNet	0.7926	0.0057	0.0007	1.0982	0.0023	0.0004	1.2245	0.0005	0.0005	1.2869	0.0004	0.0005	1.3812	0.0004	0.0006
	Ours	**0.8107**	-	-	**1.1299**	-	-	**1.2462**	-	-	**1.3100**	-	-	**1.3793**	-	-
Indivisible	**OPT**	0.8123	N/A	-	1.1852	N/A	-	1.3761	N/A	-	1.5351	N/A	-	1.6577	N/A	-
	Clock Auction	0.6060	-	-	0.8757	-	-	1.0813	-	-	1.1869	-	-	1.2476	-	-
	Greedy Subadditive	0.5472	-	-	0.7123	-	-	0.7779	-	-	0.8765	-	-	0.8797	-	-
	Ours	**0.8107**	-	-	**1.1299**	-	-	**1.2489**	-	-	**1.3201**	-	-	**1.3851**	-	-

Table 7: Robustness under **asymmetric** bid distributions. Bidders' costs are drawn from heterogeneous Beta distributions, $bid \sim \text{Beta}(\alpha, \beta)$, where α varies uniformly from 1 to 2 and β varies uniformly from 2 to 1 across bidders. The valuation is monotone submodular, $V(\boldsymbol{x}, \boldsymbol{v}) = \sqrt{\sum_{i \in [n]} x_i v_i}$, with $value \sim U[0, 2]$ and $budget \sim U[2, 4]$.

Setting	Method	$n = 2$			$n = 4$			$n = 6$			$n = 8$			$n = 10$		
		val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓
Divisible	**OPT** (Fractional)	1.3791	N/A	-	1.9743	N/A	-	2.3903	N/A	-	2.6752	N/A	-	2.8834	N/A	-
	CITransNet	1.3723	0.0056	0.0001	1.8753	0.0053	0.0006	2.1453	0.0024	0.0009	2.3363	0.0020	0.0029	2.4877	0.0020	0.0020
	RegretNet	1.2063	0.0062	0.0000	1.8139	0.0044	0.0005	2.1340	0.0019	0.0008	2.3203	0.0010	0.0051	2.4515	0.0019	0.0017
	Ours	**1.3707**	-	-	**1.8478**	-	-	**2.1212**	-	-	**2.2610**	-	-	**2.4133**	-	-
Indivisible	**OPT**	1.3791	N/A	-	1.9671	N/A	-	2.3646	N/A	-	2.6341	N/A	-	2.8369	N/A	-
	Greedy Submodular	1.1320	-	-	1.2577	-	-	1.3059	-	-	1.3308	-	-	1.3462	-	-
	TripleEagleDet	1.1320	-	-	1.3625	-	-	1.5376	-	-	1.6719	-	-	1.7797	-	-
	Ours	**1.3707**	-	-	**1.8547**	-	-	**2.1153**	-	-	**2.2585**	-	-	**2.4157**	-	-

Table 8: Robustness under **irregular** bid distributions. Bids are sampled from a discontinuous mixture distribution: $U[0, 1]$ with probability $1/4$ and $U[1, 2]$ with probability $3/4$. The valuation is monotone submodular, $V(\boldsymbol{x}, \boldsymbol{v}) = \sqrt{\sum_{i \in [n]} x_i v_i}$, with $value \sim U[0, 2]$ and $budget \sim U[2, 4]$.

Setting	Method	$n = 2$			$n = 4$			$n = 6$			$n = 8$			$n = 10$		
		val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓	val↑	rgt↓	bfv↓
Divisible	**OPT** (Fractional)	1.3542	N/A	-	1.7609	N/A	-	1.9593	N/A	-	2.0889	N/A	-	2.1871	N/A	-
	CITransNet	1.2451	0.0012	0.0005	1.5039	0.0022	0.0004	1.6693	0.0020	0.0006	1.7342	0.0013	0.0007	1.8294	0.0019	0.0015
	RegretNet	1.2129	0.0041	0.0003	1.4787	0.0012	0.0007	1.6146	0.0007	0.0005	1.7201	0.0008	0.0009	1.7983	0.0011	0.0009
	Ours	**1.2202**	-	-	**1.4675**	-	-	**1.6167**	-	-	**1.7053**	-	-	**1.7702**	-	-
Indivisible	**OPT**	1.3019	N/A	-	1.6504	N/A	-	1.8430	N/A	-	1.9776	N/A	-	2.0826	N/A	-
	Greedy Submodular	1.1318	-	-	1.2564	-	-	1.3059	-	-	1.3310	-	-	1.3463	-	-
	TripleEagleDet	1.1318	-	-	1.2676	-	-	1.3387	-	-	1.3895	-	-	1.4331	-	-
	Ours	**1.2204**	-	-	**1.4575**	-	-	**1.5989**	-	-	**1.6965**	-	-	**1.7698**	-	-

Table 9: Buyer utility of base and transformed menus under divisible allocation. The ex-post menu transformation enforces exact economic guarantees while causing only minor utility loss across valuation functions and cost distributions.

Setting	**Menu Type**	**Number of Sellers (n)**				
		$n = 2$	$n = 4$	$n = 6$	$n = 8$	$n = 10$
Additive	Base Menus	1.0015	1.7730	2.2653	2.6233	2.9648
	Transformed	1.0015	1.7706 ↓	2.2589 ↓	2.6150 ↓	2.9534 ↓
Monotone Submodular	Base Menus	1.3776	1.8489	2.1044	2.2811	2.4207
	Transformed	1.3776	1.8476 ↓	2.1024 ↓	2.2791 ↓	2.4158 ↓
Non-monotone Submodular	Base Menus	0.2869	1.0841	2.3532	3.9942	6.0516
	Transformed	0.2869	1.0507 ↓	2.3341 ↓	3.9857 ↓	6.0328 ↓
Subadditive	Base Menus	0.8107	1.1395	1.2492	1.3149	1.3929
	Transformed	0.8107	1.1299 ↓	1.2462 ↓	1.3100 ↓	1.3793 ↓
Asymmetric	Base Menus	1.3707	1.8916	2.1391	2.3155	2.4474
	Transformed	1.3707	1.8478 ↓	2.1212 ↓	2.2610 ↓	2.4133 ↓
Irregular	Base Menus	1.2212	1.4885	1.6301	1.7109	1.7881
	Transformed	1.2202 ↓	1.4675 ↓	1.6167 ↓	1.7053 ↓	1.7702 ↓

Table 10: Buyer utility of base and transformed menus under indivisible allocation. The transformed menus retain nearly the same utility as the learned base menus while satisfying exact DSIC, IR, and strict budget feasibility.

Setting	**Menu Type**	**Number of Sellers (n)**				
		$n = 2$	$n = 4$	$n = 6$	$n = 8$	$n = 10$
Additive	Base Menus	1.0015	1.7727	2.2697	2.6169	2.9757
	Transformed	1.0015	1.7708 ↓	2.2651 ↓	2.6093 ↓	2.9684 ↓
Monotone Submodular	Base Menus	1.3776	1.8490	2.1030	2.2805	2.4131
	Transformed	1.3776	1.8485 ↓	2.1004 ↓	2.2770 ↓	2.4097 ↓
Non-monotone Submodular	Base Menus	0.2492	1.0550	2.3234	3.9650	5.9983
	Transformed	0.2492	1.0550	2.3209 ↓	3.9601 ↓	5.9774 ↓
Subadditive	Base Menus	0.8107	1.1311	1.2516	1.3278	1.3962
	Transformed	0.8107	1.1299 ↓	1.2489 ↓	1.3201 ↓	1.3851 ↓
Asymmetric	Base Menus	1.3707	1.8905	2.1390	2.3103	2.4451
	Transformed	1.3707	1.8547 ↓	2.1153 ↓	2.2585 ↓	2.4157 ↓
Irregular	Base Menus	1.2210	1.4615	1.6030	1.6998	1.7744
	Transformed	1.2204 ↓	1.4575 ↓	1.5989 ↓	1.6965 ↓	1.7698 ↓

E.2 Utility Retention after Menu Transformation

All main experimental results report the performance of the transformed menus, since the ex-post menu transformation is the final step that enforces strict budget feasibility while preserving exact DSIC and IR. A natural concern is whether this enforcement step substantially degrades the utility of the learned base menus. To address this concern, we compare the buyer utility before and after menu transformation across all valuation and distributional settings.

Tables 9 and 10 report the results for divisible and indivisible allocations, respectively. Across nearly all settings, the transformed menus retain almost the same utility as the original base menus. The utility decrease is generally very small, even under more challenging settings such as non-monotone submodular valuations, asymmetric cost distributions, and irregular mixture distributions. This indicates that the learned base menus are already close to satisfying the required economic constraints, and the ex-post transformation only needs to make minor payment adjustments in most cases.

These results provide additional evidence that BFMNet does not obtain exact economic guarantees by sacrificing substantial empirical utility. Instead, the transformation acts as a lightweight enforcement layer: it converts high-utility learned base menus into economically valid menus while preserving most of their performance.

E.3 Computational Efficiency and Acceleration Strategies

Table 11 provides a fine-grained ablation study of the proposed MILP acceleration strategies on a 5-seller additive procurement instance. Although the exact ex-post transformation provides the desired economic guarantees, the naive MILP formulation introduces many binary variables and constraints, making direct optimization computationally expensive. By progressively applying Parallel-Transformation, Zero-Violation, IR-Screening, and Adaptive Keep-Choice, the MILP size is reduced by over 99.9% in terms of both constraints and binary variables. As a result, the average transformation time decreases from 2.88 seconds to 0.0038 seconds, demonstrating that the proposed acceleration strategies make exact menu transformation practical even under strict economic enforcement.

Table 11: Complexity reduction of the MILP solver for a 5-seller additive procurement setting with menu size $K = 5$. Costs and values are i.i.d. uniform on $[0, 1]$, and budgets are sampled from $U[2, 4]$. A discretization grid with step size $\epsilon = 0.001$, yielding 1001 grid points per seller, is used. Strategies are added incrementally.

Strategy	# MILP	Per Instance Complexity (Mean $\pm$ Std)		
		# Constraints	# Binary Vars.	Run Time (s)
Nothing	5	80,085	25,025	2.8890 ± 0.1233
+ Parallel-Transformation	1	16,017	5,005	0.5701 ± 0.0564
+ Zero-Violation	0.816	$13{,}075 \pm 6{,}202$	$4{,}086 \pm 1{,}938$	0.4546 ± 0.2215
+ IR-Screening	0.816	$8{,}010 \pm 3{,}976$	$1{,}553 \pm 941$	0.3601 ± 0.2024
+ Adaptive Keep-Choice	0.816	79 ± 424	12 ± 94	0.0038 ± 0.0173
Overall Reduction	**-83.68%**	**-99.90%**	**-99.95%**	**-99.87%**

E.4 Out-of-Setting Generalization

Figure 3 evaluates the out-of-setting generalization capability of BFMNet. To examine whether the learned mechanism captures transferable procurement structure rather than overfitting to a fixed market size, we train a single model only on instances with $n = 6$ sellers and directly evaluate it on markets with $n \in \{2, 4, 8, 10\}$ sellers without retraining. The dashed curves represent out-of-setting evaluation, while the solid curves correspond to in-setting models trained separately for each value of n. Across both divisible and indivisible allocation settings, the dashed curves closely track the solid curves, indicating only minor utility degradation under changes in market size. These results suggest that the permutation-equivariant architecture enables BFMNet to generalize effectively across different numbers of sellers.

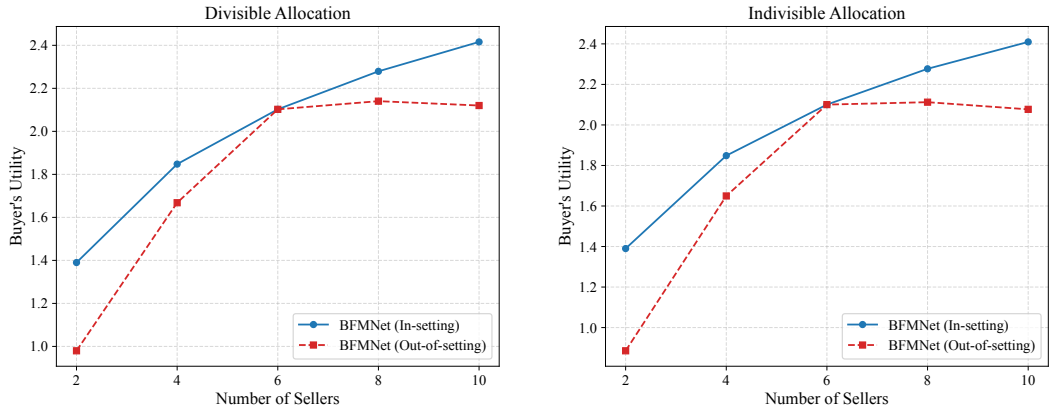


Figure 3: Out-of-setting generalization results. We evaluate BFMNet across varying numbers of sellers. The blue solid lines, BFMNet In-setting, represent models trained and tested for each specific n. The red dashed lines, BFMNet Out-of-setting, represent a single model trained exclusively on $n = 6$ sellers and directly evaluated on other market sizes. We conduct evaluations under both (a) continuous allocation and (b) discrete allocation settings.

E.5 Empirical Verification of Economic Guarantees

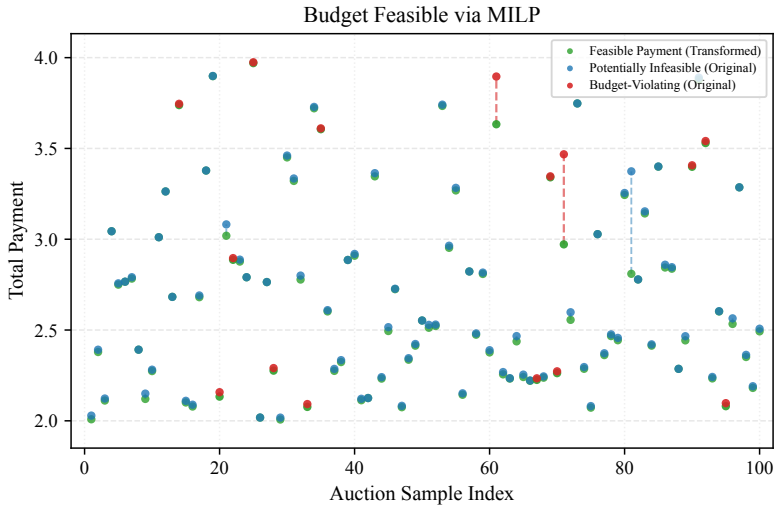
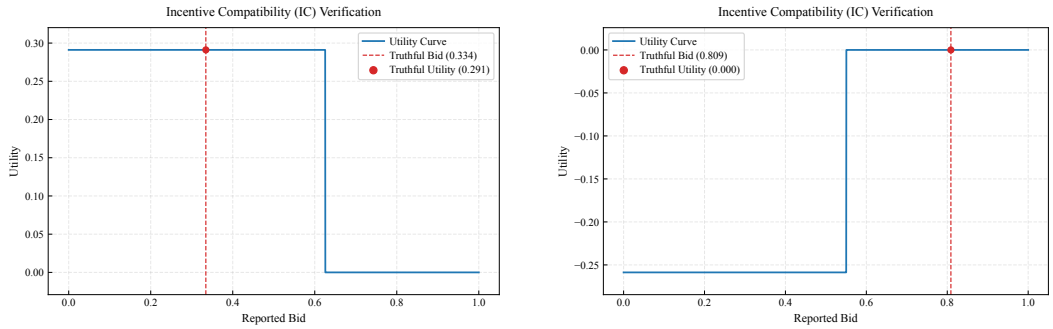


Figure 4: Effectiveness of ex-post menu transformation on budget feasibility. The scatter plot tracks total payment dynamics across 100 sampled auction instances. The red points represent original base menus generated by BFMNet that initially violate the global budget constraint. The blue points indicate instances where the initial total payment is within the budget, but menu transformation is still required to enforce *safety margin* and *safety payment* bounds, which guarantee exact DSIC across the continuous bid space. The green points depict the adjusted payments after our MILP-based ex-post transformation. Crucially, all adjusted payments are strictly capped by the budget limit, empirically validating that our approach guarantees strict budget feasibility without exception.

Figure 4 visually illustrates the necessity and effectiveness of the ex-post menu transformation. The base menus may induce total payments that exceed the budget, as shown by the red points, or lead to potentially infeasible outcomes, as shown by the blue points. After transformation, all adjusted total payments, shown as green points, lie below the budget threshold. This provides empirical evidence that the transformation successfully eliminates budget violations and enforces strict budget feasibility on the tested instances.



(a) Seller with True Cost $c_i = 0.334$

(b) Seller with True Cost $c_i = 0.809$

Figure 5: Empirical verification of dominant-strategy incentive compatibility (DSIC). The blue curves illustrate sellers' ex-post utility against reported bids $b_i \in [0, 1]$, with truthful bids ($b_i = c_i$) explicitly marked. **(a)** For a winning seller ($c_i = 0.334$), truthful reporting achieves a maximal utility of 0.291. Underbidding yields the same utility, while overbidding causes utility to drop to zero. **(b)** For a losing seller ($c_i = 0.809$), truthful utility is 0.000. Overbidding maintains zero utility, whereas underbidding leads to negative utility. Both scenarios empirically confirm that sellers maximize their utility by reporting truthfully, validating the mechanism's DSIC property.

Figure 5 empirically verifies the DSIC property by plotting each seller's ex-post utility against its reported bid. For both winning and losing sellers, truthful reporting, i.e., $b_i = c_i$, maximizes the seller's utility. Underbidding yields no higher utility and may reduce utility, while overbidding causes the seller's utility to drop to zero once the seller is no longer selected. These observations are consistent with the theoretical DSIC guarantee established in Theorem 3.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims in the abstract and introduction accurately reflect the paper's contributions and scope. The paper claims a scalable neural framework for budget-feasible mechanisms with exact DSIC, IR, and budget feasibility, and these claims are supported by the theoretical guarantees in Section 4 and empirical results in Section 5.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We explicitly discuss the limitations of our approach, including its fundamental reliance on external MILP solvers for worst-case execution and the computational trade-offs regarding the discrete step size, in Discussion and Future Work 6.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The paper states the assumptions and formal settings in Section 3 and provides theoretical results in Section 4, including the SoftMax approximation error, economic guarantees, and sample complexity. Complete proofs and additional formal results are provided in Appendices B, C, D.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper discloses the main information needed to reproduce the experiments, including valuation functions, bid/value/budget distributions, baselines, optimizer, learning rate, batch size, grid size, menu size, SoftMax schedule, budget penalty schedule, and MILP solver details in Section 5 and the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The source code for BFMNet is publicly available at `https://github.com/seu-chp-hub/BFMNet`. The repository includes a README.md file with instructions for environment setup, synthetic data generation, model training, and evaluation.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 5 specifies the experimental setup, including the valuation functions, bid/value/budget distributions, optimizer, learning rate, batch size, discretization step, SoftMax temperature schedule, budget penalty schedule, MILP transformation, menu sizes, baselines, and compute platform. Additional architecture and experimental details are provided in the appendices.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

Justification: The main utility tables report average performance over large test sets but do not include error bars or confidence intervals for all utility results, due to the computational cost of training and transforming mechanisms across many valuation settings and market sizes. We report mean and standard deviation for the computational metrics in Table 5 and Table 11.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The specific hardware configurations used for our experiments (an Intel Xeon Silver 4310 CPU and dual NVIDIA RTX 5090 GPUs) and the runtime analyses are provided in Section 5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This paper focuses on theoretical mechanism design and algorithmic optimization, which fully conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: This work has potential positive impacts by improving utility and economic reliability in budget-constrained procurement settings such as crowdsourcing, federated learning, and digital resource allocation. Potential negative impacts may arise if such mechanisms are deployed with biased valuation models or in markets where budget constraints could disadvantage certain sellers; careful domain-specific validation is therefore needed before real-world deployment.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The mechanism models presented in this paper are trained on standard synthetic data and do not involve high-risk assets, pretrained large models, or web-scraped datasets.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We properly cite the relevant literature and open-source solvers utilized in our implementation and numerical experiments.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces new code assets for BFMNet, including training, evaluation, synthetic auction instance generation, and ex-post menu transformation scripts. These assets are publicly available at `https://github.com/seu-chp-hub/BFMNet`, together with documentation on dependencies, usage instructions, experimental configurations, and limitations.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human subjects research, and therefore IRB approval or equivalent review is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as an important, original, or non-standard component.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.