UTMD Working Paper
The University of Tokyo
Market Design Center

UTMD-102

Accelerating Automated Auction Design using Sufficiency of Local Incentive Compatibility

Genta Okada
The University of Tokyo

December 3, 2025

Accelerating Automated Auction Design using Sufficiency of Local Incentive Compatibility

Genta Okada*

December 3, 2025

Abstract

Deep learning has emerged as a promising approach for optimal auction design, particularly in complex multi-bidder, multi-item settings. RegretNet (Dütting et al. (2019)) is a notable architecture in this domain, capable of learning revenue-maximizing auctions. However, ensuring Dominant-Strategy Incentive Compatibility (DSIC) in RegretNet, specifically through the estimation of 'regret,' is computationally intensive, posing a barrier to its practical application. This paper proposes a method to accelerate RegretNet training by leveraging the sufficiency of Local Incentive Compatibility (LIC) for DSIC under common auction settings. Instead of minimizing global regret, which requires an expensive search for optimal deviations, we minimize 'local regret', restricting the search to a small neighborhood of the true valuation. This approach significantly reduces the computational burden of regret estimation, primarily by decreasing the number of gradient ascent steps required. Experimental results demonstrate that our approach reduces training time by approximately 80% while maintaining revenue and regret levels comparable to the original RegretNet. This work contributes a more efficient training methodology for RegretNet, thereby enhancing its accessibility for designing optimal auctions.

1 Introduction

Optimal auction design is a cornerstone of economic theory, crucial for allocating scarce resources efficiently. While single-item auctions are well-understood since Myerson (1981), designing optimal mechanisms for multi-bidder, multi-item settings remains a significant

*Graduate School of Economics, the University of Tokyo 7-3-1 Hongo, Bunkyo-ku, Tokyo-to, 113-0033, Japan. E-mail: ogenta0628@g.ecc.u-tokyo.ac.jp

analytical challenge. Known optimal solutions are limited to very specific cases, such as a single bidder with particular valuation distributions (Daskalakis et al. (2017); Manelli and Vincent (2006); Pavlov (2011)) or when bidder valuations are restricted to two possible values (Yao (2017)). The optimal auction is unknown for general multi-bidder, multi-item scenarios, even for simple 2-bidder, 2-item cases.

The analytical intractability of many such problems has spurred the adoption of computational approaches (CONITZER (2002); Sandholm and Likhodedov (2015)). Deep learning, in particular, has emerged as a powerful tool. Dütting et al. (2019) pioneered this by introducing RochetNet and RegretNet, two architectures that apply deep learning to optimal auction design. These methods have successfully replicated theoretically optimal auctions and discovered novel mechanisms in previously unsolved settings, and have been extended to various constraints and problems (Duan et al. (2022); Feng et al. (2018); Golowich et al. (2018); Ivanov et al. (2022); Kuo et al. (2020); Peri et al. (2021); **?**); Rahme et al. (2021); Ravindranath et al. (2021)).

RegretNet is particularly versatile, capable of handling multiple bidders by learning auction mechanisms through direct parameterization of allocation and payment rules using neural networks. It ensures individual rationality (IR) and feasibility by its architecture. However, RegretNet only guarantees approximate dominant-strategy incentive compatibility (DSIC) by penalizing a measure of 'regret'—the utility gain a bidder could achieve by misreporting their valuation. The estimation of this regret, which involves finding a bidder's optimal misreport, typically requires numerous computationally expensive gradient ascent steps for each training sample. This computational bottleneck limits the scalability and practical applicability of RegretNet (**?**). Furthermore, while these gradient descent-based methods aim to find the best misreport, there is no inherent guarantee that they consistently converge to the true global optimum. This could potentially lead to an underestimation of actual regret during training, although the primary focus of this paper is on the computational cost.

Addressing these aspects of regret estimation has been a subject of related research. For instance, **?** proposed ALGnet, where a separate "misreporter" neural network is trained adversarially to generate misreports. Separately, Curry et al. (2020) focused on the accuracy of regret measurement post-training; they developed methods using integer programming to find the certifiable best misreport and thus the true regret of a learned mechanism, offering a rigorous way to evaluate a mechanism independent of gradient-descent-based estimation during training.

This paper introduces a novel approach to accelerate RegretNet training, a key characteristic of which is its strong theoretical underpinning derived from economic mechanism design theory. We leverage the theoretical concept of 'sufficiency of Local Incentive Compatibility (LIC)' (Carroll (2012)). This important result states that under specific, common conditions in auction theory (including additive valuations and convex support of valuation distributions), ensuring LIC is equivalent to ensuring the stronger condition of DSIC. This theoretical foundation provides a principled way to simplify the IC enforcement problem.

Our primary contribution is a RegretNet variant that, guided by this sufficiency result, focuses on minimizing 'local regret'—deviations within a small neighborhood of the true valuation. This local regret is estimated efficiently with a single gradient ascent step initialized at the true valuation. By targeting LIC rather than global DSIC directly and relying on the theoretical sufficiency guarantee, we significantly reduce the computational cost of the regret estimation step. Through simulations, we empirically demonstrate that this approach greatly reduces training time without a significant loss in revenue or the quality of IC guarantees compared to the original RegretNet.

While using local regret offers substantial computational savings, its differing nature and scale compared to global regret can introduce new challenges in tuning the penalty coefficients within RegretNet's original augmented Lagrangian training method. These parameters, already known for their sensitivity (Ivanov et al. (2022)), might require meticulous and setting-specific adjustments when applied to local regret. To address this potential issue and ensure stable and effective training with local regret, we adopt the 'regret budget' method, as proposed by Ivanov et al. (2022). This method explicitly defines an acceptable aggregate level of regret and dynamically adjusts a single Lagrange multiplier to meet this budget. Applying this existing technique to our local regret framework aims to provide a more robust mechanism for managing the revenue-regret trade-off, thereby mitigating the difficulties associated with fine-tuning multiple sensitive hyperparameters for the novel local regret measure.

The remainder of this paper is organized as follows: Section 2 covers preliminaries on the auction setting, desirable mechanism properties, RegretNet, and the regret budget method. Section 3 details our proposed method using the sufficiency of LIC. Section 4 presents experimental results. Section 5 concludes the paper.

2 Preliminaries

2.1 Auction Setting

We consider an auction with a set of n bidders $N = \{1, \dots, n\}$ and a set of m heterogeneous, indivisible items $M = \{1, \dots, m\}$ to be allocated. Each bidder $i \in N$ has a private valuation function $v_i : 2^M \to \mathbb{R}_+$, where $v_i(S)$ is the value bidder i derives from obtaining the bundle of items $S \subseteq M$. In this work, we focus on settings where bidders have **additive valuations**: $v_i(S) = \sum_{j \in S} v_{ij}$, or **unit-demand valuations**: $v_i(S) = \max_{j \in S} v_{ij}$, where $v_{ij} = v_i(\{j\})$ is bidder i's value for item j. Under the constraint that the sum of the allocation probabilities of each item for a bidder can not exceed one, unit-demand valuations are equivalent to additive valuations. We denote bidder i's valuation vector as $v_i = (v_{i1}, \ldots, v_{im}) \in V_i \subseteq \mathbb{R}^m_+$. Bidders have quasi-linear utility: if bidder i receives items yielding total value $v_i(x_i)$ (where x_i represents the allocation to bidder i) and pays p_i, their utility is $u_i = v_i(x_i) - p_i$. With probabilistic allocations, if x_{ij} is the probability bidder i receives item j, their utility is $\sum_{j \in M} v_{ij} x_{ij} - p_i$.

Each bidder's valuation v_i is drawn independently from a distribution F_i with support V_i. We denote the joint valuation profile as $v = (v_1, \ldots, v_n) \in V = \prod_{i \in N} V_i$, and v_{-i} as the valuations of bidders other than i. The distributions F_i are common knowledge, while the realized valuations v_i are private information.

By the revelation principle, we can restrict our attention to direct mechanisms where bidders report their valuations (bids) $b = (b_1, \ldots, b_n)$, and the mechanism determines an allocation rule $g : V \to [0,1]^{nm}$ and a payment rule $p : V \to \mathbb{R}^n$. Here, $g_{ij}(b)$ denotes the probability that bidder i receives item j given bid profile b, and $p_i(b)$ is the payment made by bidder i.

2.2 Desirable Properties of Mechanisms

An auction mechanism (g, p) should ideally satisfy the following properties:

- **Feasibility:** No item is allocated to more than one bidder: for any item $j \in M$ and any bid profile $b \in V$, $\sum_{i \in N} g_{ij}(b) \leq 1$. For a unit-demand utility, the allocation probability is normalized for each bidder so that the total allocation probability to each bidder is at most one.

- **Dominant-Strategy Incentive Compatibility (DSIC):** Truthful reporting is an optimal strategy for each bidder, regardless of other bidders' reports. For all $i \in N$, $v \in V$, and any misreport $b_i \in V_i$:

$$\sum_{j \in M} v_{ij} g_{ij}(v_i, v_{-i}) - p_i(v_i, v_{-i}) \geq \sum_{j \in M} v_{ij} g_{ij}(b_i, v_{-i}) - p_i(b_i, v_{-i}).$$

- **Individual Rationality (IR):** Each bidder's utility from participating truthfully is non-negative:

$$\sum_{j \in M} v_{ij} g_{ij}(v) - p_i(v) \geq 0 \quad \text{for all } i \in N, v \in V.$$

The seller's objective is to design a mechanism (g, p) satisfying Feasibility, DSIC, and IR that maximizes their expected revenue: $E_v[\sum_{i \in N} p_i(v)]$.

2.3 RegretNet

Proposed by Dütting et al. (2019), RegretNet employs two neural networks: a parameterized allocation network $g^w : \mathbb{R}^{nm} \to [0,1]^{nm}$ and a parametrized payment fraction network $\tilde{p}^w : \mathbb{R}^{nm} \to [0,1]^n$.

- **Feasibility** is enforced by applying a softmax function across bidders for each item's allocation probabilities output by g^w.
- **IR** is enforced by computing payment for bidder i as $p_i^w(b) = \tilde{p}_i^w(b) \sum_{j \in M} b_{ij} g_{ij}^w(b)$.
- **DSIC** is approximated by minimizing an expected regret term for each bidder i:

$$rgt_i = E_v \left[\max_{b_i \in V_i} \left(\sum_{j \in M} v_{ij} g_{ij}^w(b_i, v_{-i}) - p_i^w(b_i, v_{-i}) \right) - \left(\sum_{j \in M} v_{ij} g_{ij}^w(v_i, v_{-i}) - p_i^w(v_i, v_{-i}) \right) \right].$$

The training objective is to maximize revenue subject to small regret, using an augmented Lagrangian method with a loss function:

$$L(w, \lambda, \rho) = -\sum_i \tilde{p}_i + \sum_i \lambda_i \widetilde{rgt}_i + \frac{\rho}{2} \sum_i (\widetilde{rgt}_i)^2.$$

(where $\tilde{p}_i$ and $\widetilde{rgt}_i$ are empirical payment and regret estimated over mini-batches).

Instead of the original augmented Lagrangian training with manually tuned λ_i and ρ (which is described in Appendix B), we adopt the regret budget method for training, as detailed in Section 2.4.

A key challenge is estimating rgt_i. This involves finding the optimal misreport b_i^* for each sampled (v_i, v_{-i}), approximated using multiple (e.g., $\Gamma = 25$ in the original paper) gradient ascent steps on bidder i's utility with respect to b_i. This procedure is computationally intensive and a major bottleneck in training RegretNet.

2.4 Regret budget

The original RegretNet training employs an augmented Lagrangian method involving multiple per-bidder Lagrange multipliers λ_i and a shared quadratic penalty coefficient ρ. These hyperparameters often require careful, setting-specific tuning and can make the revenue-regret trade-off difficult to interpret directly (**?**). To provide a more robust and interpretable way to manage this trade-off, especially when dealing with potentially different scales or characteristics of local regret measures introduced later, we adapt the 'regret budget' method from Ivanov et al. (2022). This approach aims to maximize revenue subject to an explicit constraint that the average regret does not exceed a predefined budget R_{max}.

The constrained optimization problem is formulated as:

$$\min_{w} -E_v \left[\sum_{i \in N} p_i^w(v) \right] \quad \text{s.t.} \quad \sum_{i \in N} rgt_i \leq R_{max}.$$

The Lagrangian for this problem is:

$$L_{budget}(w, \gamma) = -E_v \left[\sum_{i \in N} p_i^w(v) \right] + \gamma \left(\sum_{i \in N} rgt_i - R_{max} \right),$$

where $\gamma \geq 0$ is the single Lagrange multiplier associated with the aggregate regret constraint. The effective loss for updating w is thus:

$$\tilde{L}_{budget}(w, \gamma) = -\sum_{i} \tilde{p}_i + \gamma \left(\sum_{i \in N} \widetilde{rgt}_i - R_{max} \right), \tag{1}$$

The training process alternates between updating the network parameters w by minimizing $\tilde{L}_{budget}$ (primal update via gradient descent) and updating the Lagrange multiplier γ (dual update via gradient ascent on (1) with respect to γ). The update rule for γ is:

$$\gamma \leftarrow \max \left(0, \gamma + \eta_\gamma \left(\log \left(\sum_{i \in N} \widetilde{rgt}_i \right) - \log \left(\sum_{i \in N} \tilde{p}_i \right) - \log(R_{max}) \right) \right), \tag{2}$$

where η_γ is the learning rate for the dual variable γ. In (2), we normalize the regret estimate by revenue, following Ivanov et al. (2022). To further guide the learning process, an annealing schedule for R_{max} is also employed. Training can start with a relatively high R_{max}^{start} to encourage exploration for high-revenue solutions, even if they initially violate the target regret. R_{max} is then gradually annealed towards a desired final budget R_{max}^{end} during training:

$$R_{max} \leftarrow \max(R_{max}^{end}, R_{mult} \cdot R_{max}),$$

where $R_{mult} < 1$ is a multiplicative factor controlling the annealing speed. This strategy allows the network to first prioritize revenue and then increasingly focus on satisfying the regret constraint.

The regret budget method alleviates the burden of tuning multiple, often interdependent, λ_i and ρ parameters with the more interpretable setting of R_{max} (and its schedule) and the single dual learning rate η_γ. This can lead to a more straightforward hyperparameter tuning process.

3 Proposed Method: Accelerating RegretNet via Sufficiency of LIC

Our approach aims to reduce the computational cost of enforcing IC in RegretNet by leveraging the concept of Local Incentive Compatibility (LIC) and its sufficiency for DSIC under certain conditions.

3.1 Local Incentive Compatibility (LIC)

LIC is a weaker notion of incentive compatibility. A mechanism (g, p) is LIC if for every bidder i, true valuation v_i, and valuations of others v_{-i}, there exists some neighborhood $\mathcal{N}(v_i)$ around v_i such that bidder i has no incentive to misreport to any $b_i \in \mathcal{N}(v_i)$. More formally, as defined in the context of this work: a mechanism (g, p) is LIC if for all $v \in V$, $i \in N$, there exists an $\epsilon > 0$ such that for all $b_i \in \mathcal{B}_\epsilon(v_i) = \{b_i' \in V_i | d(v_i, b_i') < \epsilon\}$ (where d is Euclidean distance):

1. $\sum_{j \in M} v_{ij} g_{ij}(v_i, v_{-i}) - p_i(v_i, v_{-i}) \geq \sum_{j \in M} v_{ij} g_{ij}(b_i, v_{-i}) - p_i(b_i, v_{-i})$.
2. $\sum_{j \in M} b_{ij} g_{ij}(b_i, v_{-i}) - p_i(b_i, v_{-i}) \geq \sum_{j \in M} b_{ij} g_{ij}(v_i, v_{-i}) - p_i(v_i, v_{-i})$.

The first condition states that truthful reporting is locally optimal for the true type v_i. The second condition ensures that the reported type b_i also (locally) prefers its outcome over the outcome of v_i. [1]

[1] Although this reciprocal condition that the neighboring type also prefers its own allocation and payment is not included in the more common definition of LIC, it is necessary for Carroll (2012)'s result.

3.2 Sufficiency of LIC

LIC is 'sufficient' for DSIC if any mechanism satisfying LIC also satisfies DSIC.

Proposition 1 (Corollary of Proposition 1 from Carroll, 2012)**.** For auctions where bidders have additive valuations and the support of their valuation distributions V_i is convex for all $i \in N$, any mechanism satisfying LIC is also DSIC.

This result is pivotal: if these conditions (additive valuations [2], convex support) hold, ensuring LIC is equivalent to ensuring DSIC.

3.3 Accelerating Regret Estimation with Local Regret

Given the sufficiency of LIC under the specified conditions, we propose to modify RegretNet to target LIC instead of global DSIC directly during training. We define empirical "local regret" ($\widetilde{lrgt}_i$) as:

$$\widetilde{lrgt}_i = \frac{1}{L}\sum_{l=1}^{L}\left[\max_{b_i \in \text{local search from } v_i^k}\left(\sum_{j\in M} v_{ij}^k g_{ij}(b_i, v_{-i}^k) - p_i(b_i, v_{-i}^k)\right) - \left(\sum_{j\in M} v_{ij}^k g_{ij}(v^k) - p_i(v^k)\right)\right].$$

where L is the batch size. Our proposed modification to RegretNet's training is twofold:

1. Replace the global regret $\widetilde{rgt}_i$ with this local regret $\widetilde{lrgt}_i$ in the augmented Lagrangian loss function (1).

2. To estimate $\widetilde{lrgt}_i$, we perform only a **single gradient ascent step** to find a potentially profitable misreport b_i. This gradient step is **initialized at the bidder's true valuation** v_i.

This contrasts with standard RegretNet, which uses multiple (usually 25) gradient steps. The expected benefit is a multiplicative reduction in computational cost.

4 Experiments

4.1 Experimental Setup

We implemented our proposed method and the baseline RegretNet using Tensorflow. The neural network architecture for both allocation and payment networks consisted of three to six hidden layers with 100 ReLU units each. Training was conducted for 80 epochs

[2] Note that this additive formulation includes unit-demand settings.

using the Adam optimizer with a learning rate of 10^{-3} and a batch size of 128. The total number of training samples was 640,000. Training was conducted using the regret budget method described in Section 2.4. R_{mult} was chosen such that R_{max} converges to R^{end}_{max} in approximately two-thirds of the training iterations, following Ivanov et al. (2022). Evaluation is based on 10000 samples, and optimal misreports are estimated using 1000 gradient descent steps from 1000 random initial values, reporting the one that achieves the largest regret.

We considered four valuation settings:

- **(a):** 2 bidders, 2 items, additive valuation; $v_{ij} \sim$ i.i.d. Uniform$(0, 1)$.
- **(b):** 2 bidders, 5 items, additive valuation; $v_{ij} \sim$ i.i.d. Uniform$(0, 1)$.
- **(c):** 3 bidders, 10 items, additive valuation; $v_{ij} \sim$ i.i.d. Uniform$(0, 1)$.
- **(d):** 1 bidder, 2 items, additive valuation; $v_{11} \sim$ i.i.d. Uniform$(4, 16)$, $v_{12} \sim$ i.i.d. Uniform$(4, 7)$.
- **(e):** 2 bidders, 2 items, unit-demand valuation; $v_{ij} \sim$ i.i.d. Uniform$(0, 1)$.

For (a), (b), (c), (e), the theoretical optimal is unknown; for (d), it was found by Daskalakis et al. (2017).

4.2 Baselines and Proposed Method

- **Baseline (RegretNet-Standard):** Original RegretNet; $\Gamma = 25$ gradient ascent steps for regret estimation.
- **Proposed (RegretNet-Local):** Modified RegretNet; $\Gamma = 1$ gradient ascent step for local regret estimation, initialized at v_i.

4.3 Evaluation Metrics

- **Expected Seller Revenue.**
- **Average Regret:** Calculated post-training using an extensive search (2,000 gradient steps from 1,000 random starts).
- **Training Time:** Total wall-clock time for 80 epochs.

Table 1: Performance Comparison of RegretNet-Standard and RegretNet-Local

Setting	Method	Revenue	Avg. Regret	Time (h)
(a)	RegretNet-Standard	0.887	0.0004	**2.94**
	RegretNet-Local	0.882	0.0007	**0.58**
(b)	RegretNet-Standard	2.310	0.0015	**3.41**
	RegretNet-Local	2.257	0.0016	**0.68**
(c)	RegretNet-Standard	5.635	0.0022	**6.51**
	RegretNet-Local	5.675	0.0061	**1.27**
(d)	RegretNet-Standard	9.896	0.0072	**1.87**
	RegretNet-Local	9.826	0.0077	**0.52**
	Optimal	9.781	0	-
(e)	RegretNet-Standard	0.717	0.0004	**2.94**
	RegretNet-Local	0.720	0.0009	**0.57**

4.4 Results

The results in Table 1 demonstrate the effectiveness of RegretNet-Local. RegretNet-Local significantly reduced training time, approximately five times accelerating it across all settings. Revenue generated by RegretNet-Local is comparable to, and in some instances slightly higher than, RegretNet-Standard. Average regret values for RegretNet-Local remain very small (on the order of 10^{-4} to 10^{-3} fraction of the revenue), indicating good approximate DSIC properties. Overall, RegretNet-Local successfully accelerates training while maintaining strong performance.

5 Conclusion

This paper addressed the computational challenge in training RegretNet. We proposed RegretNet-Local, leveraging the sufficiency of LIC for DSIC in settings with additive valuations and convex valuation supports. Our modification involves estimating local regret using a single gradient ascent step initialized at the true valuation. Experiments showed that RegretNet-Local reduces training time greatly while maintaining comparable seller revenue and low regret levels. This work demonstrates that incorporating theoretical insights can lead to more efficient deep learning algorithms for economic problems, making the deep learning approach to auction design more computationally feasible.

Future research could explore applicability to more general valuation settings or investigate theoretical guarantees of the single gradient step for local regret estimation.

References

Carroll, Gabriel. 2012. "When are local incentive constraints sufficient?" *Econometrica* 80 (2): 661–686.

CONITZER, V. 2002. "Complexity of Mechanism Design." In *Proc. of the 18th Conf. on Uncertainty in Artificial Intelligence (UAI), 2002.*

Curry, Michael, Ping-Yeh Chiang, Tom Goldstein, and John Dickerson. 2020. "Certifying strategyproof auction networks." *Advances in Neural Information Processing Systems* 33 4987–4998.

Daskalakis, Constantinos, Alan Deckelbaum, and Christos Tzamos. 2017. "Strong Duality for a Multiple-Good Monopolist." *Econometrica* 85 735–767.

Duan, Zhijian, Jingwu Tang, Yutong Yin, Zhe Feng, Xiang Yan, Manzil Zaheer, and Xiaotie Deng. 2022. "A context-integrated transformer-based neural network for auction design." In *International Conference on Machine Learning*, 5609–5626, PMLR.

Dütting, Paul, Zhe Feng, Harikrishna Narasimhan, David Parkes, and Sai Srivatsa Ravindranath. 2019. "Optimal auctions through deep learning." In *International Conference on Machine Learning*, 1706–1715, PMLR.

Feng, Zhe, Harikrishna Narasimhan, and David C Parkes. 2018. "Deep learning for revenue-optimal auctions with budgets." In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems*, 354–362.

Golowich, Noah, Harikrishna Narasimhan, and David C Parkes. 2018. "Deep Learning for Multi-Facility Location Mechanism Design.." In *IJCAI*, 261–267.

Ivanov, Dmitry, Iskander Safiulin, Igor Filippov, and Ksenia Balabaeva. 2022. "Optimal-er auctions through attention." *Advances in Neural Information Processing Systems* 35 34734–34747.

Kuo, Kevin, Anthony Ostuni, Elizabeth Horishny, Michael J Curry, Samuel Dooley, Ping-yeh Chiang, Tom Goldstein, and John P Dickerson. 2020. "Proportionnet: Balancing fairness and revenue for auction design with deep learning." *arXiv preprint arXiv:2010.06398.*

Manelli, Alejandro M, and Daniel R Vincent. 2006. "Bundling as an optimal selling mechanism for a multiple-good monopolist." *Journal of Economic Theory* 127 (1): 1–35.

Myerson, Roger B. 1981. "Optimal auction design." *Mathematics of operations research* 6 (1): 58–73.

Pavlov, Gregory. 2011. "Optimal mechanism for selling two goods." *The BE Journal of Theoretical Economics* 11 (1): .

Peri, Neehar, Michael Curry, Samuel Dooley, and John Dickerson. 2021. "Preferencenet: Encoding human preferences in auction design with deep learning." *Advances in Neural Information Processing Systems* 34 17532–17542.

Rahme, Jad, Samy Jelassi, Joan Bruna, and S Matthew Weinberg. 2021. "A permutation-equivariant neural network architecture for auction design." In *Proceedings of the AAAI conference on artificial intelligence*, Volume 35. 5664–5672.

Ravindranath, Sai Srivatsa, Zhe Feng, Shira Li, Jonathan Ma, Scott D Kominers, and David C Parkes. 2021. "Deep learning for two-sided matching." *arXiv preprint arXiv:2107.03427*.

Sandholm, Tuomas, and Anton Likhodedov. 2015. "Automated design of revenue-maximizing combinatorial auctions." *Operations Research* 63 (5): 1000–1025.

Yao, Andrew Chi-Chih. 2017. "Dominant-strategy versus bayesian multi-item auctions: Maximum revenue determination and comparison." In *Proceedings of the 2017 ACM Conference on Economics and Computation*, 3–20.

A Proof Theorem 1

The proof follows the logic presented by Carroll (2012). We include this for completeness.

Let (g, p) be an arbitrary mechanism satisfying LIC, and consider any $v \in V$, $i \in N$, and $b_i \in V_i$. We need to show that

$$\sum_{j \in M} v_{ij} g_{ij}(v_i, v_{-i}) - p_i(v_i, v_{-i}) \geq \sum_{j \in M} v_{ij} g_{ij}(b_i, v_{-i}) - p_i(b_i, v_{-i}). \tag{3}$$

For any $\alpha \in [0, 1]$, let $v_i^\alpha = (1 - \alpha)v_i + \alpha b_i$. By convexity of V_i, we have $v_i^\alpha \in V_i$. (Note: In the following, we will denote v_{ij}^α as the j-th component of v_i^α, and for brevity, when v_{-i} is fixed, we write $g_{ij}(v_i^\alpha)$ for $g_{ij}(v_i^\alpha, v_{-i})$ and $p_i(v_i^\alpha)$ for $p_i(v_i^\alpha, v_{-i})$.)

LIC implies that for any α, α' such that $v_i^{\alpha'}$ is in a small neighborhood of v_i^α:

$$\sum_{j \in M} v_{ij}^\alpha g_{ij}(v_i^\alpha) - p_i(v_i^\alpha) \geq \sum_{j \in M} v_{ij}^\alpha g_{ij}(v_i^{\alpha'}) - p_i(v_i^{\alpha'})$$

and

$$\sum_{j \in M} v_{ij}^{\alpha'} g_{ij}(v_i^{\alpha'}) - p_i(v_i^{\alpha'}) \geq \sum_{j \in M} v_{ij}^{\alpha'} g_{ij}(v_i^\alpha) - p_i(v_i^\alpha).$$

We denote this pair of local incentive compatibility conditions holding between v_i^α and $v_i^{\alpha'}$ as $\alpha \leftrightarrow \alpha'$.

Let A be the set of all $\alpha \in [0, 1]$ such that there exists a finite sequence $0 = \alpha_0 < \alpha_1 < \cdots < \alpha_K = \alpha$ where $\alpha_k \leftrightarrow \alpha_{k+1}$ for all $k = 0, \ldots, K - 1$. If $\alpha \in A$ and there exists an α' such that $\alpha < \alpha' \leq 1$ and $\alpha \leftrightarrow \alpha'$, then clearly $\alpha' \in A$ (by extending the sequence for α with α').

Let $\bar{\alpha} = \sup A$. Since $0 \leftrightarrow 0$ (trivially), $0 \in A$, so $\bar{\alpha} \geq 0$. If $\bar{\alpha} = 0$, then $\bar{\alpha} \in A$. If $\bar{\alpha} > 0$, then by the definition of LIC, for any α sufficiently close to $\bar{\alpha}$ (with $\alpha < \bar{\alpha}$), we have $\bar{\alpha} \leftrightarrow \alpha$. Since $\bar{\alpha} = \sup A$, there must exist such an $\alpha \in A$ arbitrarily close to $\bar{\alpha}$. Thus, we can "chain" the local incentive compatibility from 0 to α and then from α to $\bar{\alpha}$ (if $\alpha \neq \bar{\alpha}$), implying $\bar{\alpha} \in A$. Therefore, $\bar{\alpha} \in A$ always holds.

Now, assume for contradiction that $\bar{\alpha} < 1$. By LIC, for $\alpha_N > \bar{\alpha}$ and sufficiently close to $\bar{\alpha}$, we have $\bar{\alpha} \leftrightarrow \alpha_N$. Since $\bar{\alpha} \in A$, this implies $\alpha_N \in A$. But $\alpha_N > \bar{\alpha}$, which contradicts $\bar{\alpha} = \sup A$. Thus, we must have $\bar{\alpha} = 1$. This means $1 \in A$.

So, there exists a sequence $0 = \alpha_0 < \alpha_1 < \cdots < \alpha_K = 1$ such that $\alpha_k \leftrightarrow \alpha_{k+1}$ for all $k = 0, \ldots, K - 1$. This means for each $k = 0, \ldots, K - 1$:

$$\sum_{j \in M} v_{ij}^{\alpha_k} (g_{ij}(v_i^{\alpha_k}) - g_{ij}(v_i^{\alpha_{k+1}})) - (p_i(v_i^{\alpha_k}) - p_i(v_i^{\alpha_{k+1}})) \geq 0 \tag{4}$$

and

$$\sum_{j\in M} v_{ij}^{\alpha_{k+1}}(g_{ij}(v_i^{\alpha_{k+1}}) - g_{ij}(v_i^{\alpha_k})) - (p_i(v_i^{\alpha_{k+1}}) - p_i(v_i^{\alpha_k})) \geq 0. \tag{5}$$

Multiply Eq. (4) by α_{k+1} and Eq. (5) by α_k and summing them leads to:

$$\sum_{j\in M}(\alpha_{k+1}v_{ij}^{\alpha_k} - \alpha_k v_{ij}^{\alpha_{k+1}})(g_{ij}(v_i^{\alpha_k}) - g_{ij}(v_i^{\alpha_{k+1}})) - (\alpha_{k+1}-\alpha_k)(p_i(v_i^{\alpha_k}) - p_i(v_i^{\alpha_{k+1}})) \geq 0.$$

We have

$$\begin{aligned}
\alpha_{k+1}v_{ij}^{\alpha_k} - \alpha_k v_{ij}^{\alpha_{k+1}} &= \alpha_{k+1}((1-\alpha_k)v_{ij} + \alpha_k b_{ij}) - \alpha_k((1-\alpha_{k+1})v_{ij} + \alpha_{k+1}b_{ij}) \\
&= (\alpha_{k+1}(1-\alpha_k) - \alpha_k(1-\alpha_{k+1}))v_{ij} + (\alpha_{k+1}\alpha_k - \alpha_k\alpha_{k+1})b_{ij} \\
&= (\alpha_{k+1} - \alpha_{k+1}\alpha_k - \alpha_k + \alpha_k\alpha_{k+1})v_{ij} \\
&= (\alpha_{k+1} - \alpha_k)v_{ij}.
\end{aligned}$$

Since $\alpha_{k+1} - \alpha_k > 0$, we can divide the inequality by $(\alpha_{k+1} - \alpha_k)$:

$$\sum_{j\in M} v_{ij}(g_{ij}(v_i^{\alpha_k}) - g_{ij}(v_i^{\alpha_{k+1}})) - (p_i(v_i^{\alpha_k}) - p_i(v_i^{\alpha_{k+1}})) \geq 0. \tag{6}$$

Summing Eq. (6) for $k = 0, \ldots, K-1$:

$$\begin{aligned}
0 &\leq \sum_{k=0}^{K-1}\left[\sum_{j\in M} v_{ij}(g_{ij}(v_i^{\alpha_k}) - g_{ij}(v_i^{\alpha_{k+1}})) - (p_i(v_i^{\alpha_k}) - p_i(v_i^{\alpha_{k+1}}))\right] \\
&= \sum_{j\in M} v_{ij}\sum_{k=0}^{K-1}(g_{ij}(v_i^{\alpha_k}) - g_{ij}(v_i^{\alpha_{k+1}})) - \sum_{k=0}^{K-1}(p_i(v_i^{\alpha_k}) - p_i(v_i^{\alpha_{k+1}})).
\end{aligned}$$

This is a telescoping sum:

$$\begin{aligned}
\sum_{k=0}^{K-1}(g_{ij}(v_i^{\alpha_k}) - g_{ij}(v_i^{\alpha_{k+1}})) &= (g_{ij}(v_i^{\alpha_0}) - g_{ij}(v_i^{\alpha_1})) + \cdots + (g_{ij}(v_i^{\alpha_{K-1}}) - g_{ij}(v_i^{\alpha_K})) \\
&= g_{ij}(v_i^{\alpha_0}) - g_{ij}(v_i^{\alpha_K}).
\end{aligned}$$

Similarly for the payment terms. Since $\alpha_0 = 0$, $v_i^{\alpha_0} = v_i$. Since $\alpha_K = 1$, $v_i^{\alpha_K} = b_i$. Thus, the sum becomes:

$$\sum_{j\in M} v_{ij}(g_{ij}(v_i) - g_{ij}(b_i)) - (p_i(v_i) - p_i(b_i)) \geq 0.$$

Rearranging this gives:

$$\sum_{j \in M} v_{ij} g_{ij}(v_i) - p_i(v_i) \geq \sum_{j \in M} v_{ij} g_{ij}(b_i) - p_i(b_i).$$

This is exactly Eq. (3), which completes the proof. □

B RegretNet Training Algorithm

The RegretNet training algorithm involves iteratively updating network weights and Lagrange multipliers.

Algorithm 1 RegretNet Training

Input: Minibatches $\mathcal{S}_1, \ldots, \mathcal{S}_T$ of size B

Parameters: $\forall t, \rho_t > 0$, $\gamma > 0$, $\eta > 0$, $\Gamma \in \mathbb{N}$, $Q \in \mathbb{N}$

Initialize: $w^0 \in \mathbb{R}^d$, $\lambda^0 \in \mathbb{R}^n$

for $t = 0$ **to** T **do**

Receive minibatch $\mathcal{S}_t = \{v^{(1)}, \ldots, v^{(B)}\}$

Initialize misreport ${v'}_i^{(\ell)} \in V_i, \forall \ell \in [B],\ i \in N$

for $r = 0$ **to** Γ **do**

$\forall \ell \in [B],\ i \in N$:

$${v'}_i^{(\ell)} \leftarrow {v'}_i^{(\ell)} + \gamma \nabla_{v'_i} \left[u_i^w \left(v_i^{(\ell)}; \left(v'_i, v_{-i}^{(\ell)} \right) \right) \right] \Big|_{v'_i = {v'}_i^{(\ell)}}$$

end for

Compute regret gradient: $\forall \ell \in [B], i \in N$:

$$g_{\ell,i}^t = \nabla_w \left[u_i^w \left(v_i^{(\ell)}; \left({v'}_i^{(\ell)}, v_{-i}^{(\ell)} \right) \right) - u_i^w \left(v_i^{(\ell)}; v^{(\ell)} \right) \right] \Big|_{w = w^t}$$

Compute Lagrangian gradient and update w^t:

$$w^{t+1} \leftarrow w^t - \eta \nabla_w L_{\rho_t}(w, \lambda^t) \Big|_{w = w^t}$$

Update Lagrange multipliers once in Q iterations:

if t is a multiple of Q

$$\lambda_i^{t+1} \leftarrow \lambda_i^t + \rho_t\, \widetilde{rgt}_i(w^{t+1}), \quad \forall i \in N$$

else

$$\lambda^{t+1} \leftarrow \lambda^t$$

end for