Learning for Edge-Weighted Online Bipartite Matching with Robustness Guarantees

PENGFEI LI, University of California, Riverside
JIANYI YANG, University of California, Riverside
SHAOLEI REN, University of California, Riverside

ACM Reference Format:
Pengfei Li, Jianyi Yang, and Shaolei Ren. 2023. Learning for Edge-Weighted Online Bipartite Matching with Robustness Guarantees. 1, 1 (May 2023), 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 PROBLEM FORMULATION

In this extended abstract, we discuss our recent work [1] that uses reinforcement learning to solve edge-weighted online bipartite matching with robustness guarantees. Edge-weighted online bipartite matching is a classic online problem with numerous applications, including scheduling tasks to servers, displaying advertisements to online users, recommending articles/movies/products, among many others. The agent matches items (a.k.a. vertices) between two sets $\mathcal{U}$ and $\mathcal{V}$ to gain as high total rewards as possible. Suppose that $\mathcal{U}$ is fixed and contains *offline* items $u \in \mathcal{U}$, and that the *online* items $v \in \mathcal{V}$ arrive sequentially: in each time slot, an online item $v \in \mathcal{V}$ arrives and the weight/reward information $\{w_{uv} \mid w_{u,\min} \le w_{uv} \le w_{u,\max}, u \in \mathcal{U}\}$ is revealed, where w_{uv} represents the reward when the online item v is matched to each offline $u \in \mathcal{U}$. We denote one problem instance by $\mathcal{G} = \{\mathcal{U}, \mathcal{V}, \mathcal{W}\}$, where $\mathcal{W} = \{w_{uv} \mid u \in \mathcal{U}, v \in \mathcal{V}\}$. We denote $x_{uv} \in \{0, 1\}$ as the matching decision indicating whether u is matched to v. Also, any offline item $u \in \mathcal{U}$ can be matched up to c_u times, where c_u is essentially the capacity for offline item u known to the agent in advance. The goal is to maximize the total collected reward $\sum_{v \in \mathcal{V}, u \in \mathcal{U}} x_{uv} w_{uv}$. With a slight abuse of notations, we denote $x_v \in \mathcal{U}$ as the index of item in $\mathcal{U}$ that is matched to item $v \in \mathcal{V}$. The set of online items matched to $u \in \mathcal{U}$ is denoted as $\mathcal{V}_u = \{v \in \mathcal{V} \mid x_{uv} = 1\}$.

For better presentation, we focus on the no-free-disposal setting, while the free-disposal setting is also studied in [1]. Specifically, the **offline** problem with no free disposal can be expressed as:

$$\max_{x_{uv} \in \{0,1\}, u \in \mathcal{U}, v \in \mathcal{V}} \sum x_{uv} w_{uv}, \quad \text{s.t.,} \quad \sum_{v \in \mathcal{V}} x_{uv} \le c_u, \text{ and } \sum_{u \in \mathcal{U}} x_{uv} \le 1, \forall u \in \mathcal{U}, v \in \mathcal{V}$$

where the constraints specify the offline item capacity limit and each online item $v \in \mathcal{V}$ can only be matched up to one offline item $u \in \mathcal{U}$. Given an online algorithm α, we use $f_u^{\alpha}(\mathcal{G})$ to denote the total reward collected for offline item $u \in \mathcal{U}$, and $R^{\alpha}(\mathcal{G}) = \sum_{u \in \mathcal{U}} f_u^{\alpha}(\mathcal{G})$ to denote the total collected reward. We aim to maximize the average reward subject to worst-case robustness guarantees for each problem instance as formalized below:

$$\max \mathbb{E}_{\mathcal{G}} \left[R^{\alpha}(\mathcal{G}) \right], \quad \text{s.t.} \quad R^{\alpha}(\mathcal{G}) \ge \rho R^{\pi}(\mathcal{G}) - B, \quad \forall \mathcal{G},$$

where the expectation $\mathbb{E}_{\mathcal{G}} \left[R^{\alpha}(\mathcal{G}) \right]$ is over the randomness $\mathcal{G} = \{\mathcal{U}, \mathcal{V}, \mathcal{W}\}$.

2 ALGORITHM

To guarantee robustness (i.e., ρ-competitive against a given expert for any $\rho \in [0, 1]$), we propose a novel algorithm called LOMAR. Our key novelty is the design of a robust constraint which serves as

Authors' addresses: Pengfei Li, University of California, Riverside; Jianyi Yang, University of California, Riverside; Shaolei Ren, University of California, Riverside.

2023. XXXX-XXXX/2023/5-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Algorithm 1 Inference of Robust Learning-based Online Bipartite Matching (LOMAR)

```
Require: Competitiveness constraint $\rho \in [0,1]$ and $B \geq 0$
for $v = 1$ to $|\mathcal{V}|$ do
    Run the expert $\pi$ and get expert's decision $x_v^\pi$.
    If $x_v^\pi \neq$ skip: $\mathcal{V}^\pi_{x_v^\pi,v} = \mathcal{V}^\pi_{x_v^\pi,v-1} \bigcup \{v\}$,
    $R_v^\pi = R_{v-1}^\pi + w_{x_v^\pi,v}$.
    //Update the virtual decision set and reward of the expert
    $s_u = w_{uv} - h_\theta(I_u, w_{uv}), \forall u \in \mathcal{U}$
    //Run RL model to get score $s_u$ with history information $I_u$
    $\tilde{x}_v = \arg\max_{u \in \mathcal{U}_a \bigcup \{\text{skip}\}} \left\{\{s_u\}_{u \in \mathcal{U}_a}, s_{\text{skip}}\right\}$, with $s_{\text{skip}} = 0$ and $\mathcal{U}_a = \left\{u \in \mathcal{U} \mid |\mathcal{V}_{u,v-1}| < c_u\right\}$.
    //Get RL decision $\tilde{x}_v$
    if Robust constraint in (1) is satisfied then
        Select $x_v = \tilde{x}_v$. //Follow RL
    else if $x_v^\pi$ is available (i.e., $|\mathcal{V}_{x_v^\pi,v-1}| < c_{x_v^\pi}$) then
        Select $x_v = x_v^\pi$. //Follow the expert
    else
        Select $x_v =$ skip.
    end if
    If $x_v \neq$ skip, $\mathcal{V}_{x_v,v} = \mathcal{V}_{x_v,v-1} \bigcup \{v\}$,
    $R_v = R_{v-1} + w_{x_v,v}$.
    //Update the true decision set and reward
end for
```

the condition for online switching. Specifically, by assigning more online items to $u \in \mathcal{U}$ than the expert algorithm at step v, LOMAR can possibly receive a higher cumulative reward than the expert's cumulative reward. But, such advantages are just *temporary*, because the expert may receive an even higher reward in the future by filling up the unused capacity of item u. Thus, to hedge against the future uncertainties, LOMAR chooses the RL decisions only when the following condition is satisfied:

$$R_{v-1} + w_{\tilde{x}_v,v} \geq \rho\Big(R_v^\pi \sum_{u \in \mathcal{U}} \left(|\mathcal{V}_{u,v-1}| - |V^\pi_{u,v}| + \mathbb{I}_{u=\tilde{x}_v}\right)^+ \cdot w_{u,\max}\Big) - B, \tag{1}$$

where $\mathbb{I}_{u=\tilde{x}_v} = 1$ if and only if $u = \tilde{x}_v$ and 0 otherwise, $(\cdot)^+ = \max(\cdot, 0)$, $\rho \in [0,1]$ and $B \geq 0$ are the hyperparameters indicating the desired robustness with respect to the expert algorithm π.

We prove that LOMAR guarantees robustness in terms of ρ-competitiveness against *any* given expert online algorithms for any $\rho \in [0,1]$. To improve the average performance of LOMAR, we also train the RL policy in LOMAR by explicitly taking into account the introduced switching operation. Importantly, to avoid the "no supervision" trap during the initial RL policy training, we propose to approximate the switching operation probabilistically. We also extend LOMAR to the free-disposal setting where each offline item $u \in \mathcal{U}$ can be matched more than c_u times but only the top c_u rewards are counted when more than c_u online items are matched to u. Finally, we offer empirical experiments to demonstrate that LOMAR can improve the average cost (compared to existing expert algorithms) as well as lower the competitive ratio (compared to pure RL-based optimizers).

REFERENCES

[1] Pengfei Li, Jianyi Yang, and Shaolei Ren. Learning for edge-weighted online bipartite matching with robustness guarantees. In *ICML*, 2023.