A NOTATIONAL TABLE

notation	meaning
$\mathcal{A}$	set of agents
$\mathcal{I}$	set of incentives
$\mathcal{P} = \mathcal{A} \times \mathcal{I}$	allowed agent-incentive pairs
Θ_a	state (type) space of agent a
$P_{a,i}$	transition probability kernel
$\beta_a^{(t)}$	agent a's type distribution at epoch t
$\pi_{a,i}$	stationary distribution of $(a,i) \in \mathcal{P}$
$\mu_{a,i}$	expected reward from $(a,i) \in \mathcal{P}$
τ_k	number of iterations matching offered in epoch k, $\tau_k = \tau_0 + \zeta k, \zeta > 0$
$r_{a,i}^{\theta_a}$	random reward
$\mathcal{T}_r(a,i,\theta_a)$	agent a's reward distribution
$\boldsymbol{r}_{a,i}^{\theta}$	time-averaged reward during epoch k $\boldsymbol{r}_{a,i}^{\theta_a(k)} = \frac{1}{\tau_k}\sum_{t=t_k}^{t_{k+1}-1} r_{a,i}^{\theta_a(t)}$
b_{ξ_l}	maximum number of edges of class ξ_l
G^*	greedy matching on weights $(\mu_{a,i})$
g_j^*	the edge having the j–th largest weight $(\mu_{a,i})$ in G^*.
$i^*(a)$	incentive agent a is matched to in G^*
L_j^*	set of $(a,i) \in \mathcal{P}$ that become infeasible when g_j^* is added to matching but not before that
S_a	set of edges (a,i) such that $\mu_{a,i} \leq \mu_{a,i^*(a)}$
$\mathcal{S}$	$\bigcup_{a\in\mathcal{A}} S_a$
m	number of agents & incentives
n	the total number of epochs
$\theta_a(t)$	state of agent a at the beginning of epoch t
$C_{a,i}, \rho_{a,i}$	constants specific to each edge (a,i)
C_*	$\max_{(a,i)\in\mathcal{P}\setminus\mathcal{S}} C_{a,i}$
$R^{\alpha}(n)$	regret of given matching policy α at the end of n epochs
$T_{a,i}(n)$	number of times edge (a,i) selected in first n epochs
$R_{a,j}^{\theta,k}$	reward on edge (a,i) when selected for the k–th time given θ_a
$\bar{R}_{a,j}^{k}$	average reward on first k times (a,i) is selected, i.e., $\frac{1}{k}\sum_{i=1}^{k} R_{a,j}^{\theta,k}$
$\theta_a(t_{a,i}^l)$	agent a's state at the beginning of epoch l
$X_{a,i}^k$	$R_{a,i}^{\theta,k} - \mathbb{E}[R_{a,i}^{\theta,k}\|\mathcal{F}_{a,i}^{k-1}]$
$Y_{a,i}^k$	$\sum_{j=1}^{k} X_{a,i}^j$ a martingale
$Q_{a,i}(k)$	$\frac{C_{a,i}}{2}\left(\frac{1}{\zeta+\tau_0} + \frac{1}{\zeta}\log\left(1+\frac{k\zeta}{\tau_0}\right)\right)$
$c_{a,j}^k(t)$	upper confidence parameter for edge (a,j) after being selected for k times
$u_{a,j}^k(t)$	average reward plus upper confidence parameter for (a,j), i.e., $\bar{R}_{a,j}^k + c_{a,j}^k$

B PROOFS

B.1 PROOF OF THEOREM 1

Proof Our proof relies on what is referred to in the matching literature as a *charging argument*. In simple terms, we take each edge belonging to the benchmark M^* and identify a corresponding edge in G^* whose weight is larger than that of the benchmark edge. This allows us to charge the weight of the original edge to an edge in G^*. During the charging process, we ensure that no more than three edges in M^* are charged to each edge in G^*. This gives us an approximation factor of three.

Suppose that an edge (a,i) belongs to M^* but not G^*. This implies that the edge (a,i) was removed from the set E' at some iteration during the course of Algorithm 1. Moreover, as per the algorithm, this removal can happen in one of two ways: (i) via Line 7, in which case there exists some edge (a,i') or (a',i) that was selected to G^* ahead of (a,i), and (ii) via Line 8 in which case b_{ξ_j} edges belonging to class $\xi_j = c(a,i)$ were added to G^* before (a,i), as a result of which the capacity constraint for that class was met. Based on this, we divide the analysis into two cases.

Case I: Removal via Line 7. Without loss of generality, suppose that (a',i) is the edge added to G^* during the iteration in which (a,i) is removed. Then, by definition, since $(a',i) = \arg\max_{(a'',i'')\in E'} w(a'',i'')$ before the removal of (a,i) from E', we infer that

$$w(a,i) \leq w(a',i) \tag{1}$$

Case II: Removal via Line 8. In this case, since the class $\xi_j = c(a,i)$ has reached its capacity limit, and since the greedy algorithm selects edges in the decreasing order of weight, it must be the case that for every $(a',i') \in G^* \cap \xi_j$, we have that

$$w(a,i) \leq w(a',i').$$

Since $G^* \cap \xi_j$ contains exactly b_{ξ_j}, we can average the above equation over the edges in $G^* \cap \xi_j$ to get that

$$w(a,i) \leq \frac{1}{b_{\xi_j}} \sum_{(a',i')\in G^*\cap\xi_j} w(a',i'). \tag{2}$$

Finally, we note that if edge (a,i) belongs to both the greedy matching and M^*, we can simply 'charge the weight of (a,i)' to itself.

Now we can complete the proof by summing (1) and (2) over all the edges in M^*. Formally, let $M^* = M_1^* \cup M_2^*$ such that M_1^* denotes the set of edges that are present in both M^* and G^* as well as the edges that fall under the first case. Similarly, let M_2^* denote the edges that fall

under the second case. Summing 1 over all of the edges in M_1^*, we get that

$$\sum_{(a,i)\in M_1^*} w(a,i) \leq 2 \sum_{(a,i)\in G^*} w(a,i). \quad (3)$$

The factor of two in the right hand side comes from the fact that for any given edge (a,i) in G^*, at most two edges in M_1^* can be charged to this edge. Indeed, the only edges that can be charged to (a,i) must contain either the node a or the node i and in a matching, each node can appear in at most one edge. Next, summing (2) over all of the edges in M_2^*, we get that

$$\begin{aligned}\sum_{(a,i)\in M_2^*} w(a,i) &= \sum_{\xi_j\in\mathcal{C}} \sum_{(a,i)\in M_2^*\cap\xi_j} w(a,i) \\ &\leq \sum_{\xi_j\in\mathcal{C}} \sum_{(a,i)\in\xi_j\cap G^*} w(a,i) \\ &= \sum_{(a,i)\in G^*} w(a,i). \quad (4)\end{aligned}$$

To see why this is the case, first observe that in (2), for each edge in class ξ_j belonging to M_2^*, all of the edges in class ξ_j in matching G^* appear in the right hand side with coefficient $\frac{1}{b_{\xi_j}}$. By definition,there are at most b_{ξ_j} edges of class ξ_j in M^* and exactly b_{ξ_j} edges of this class belong to G^*—if this were not the case, Line 8 of Algorithm 1 would not be used. To conclude, the coefficient for each edge in the right hand side is increased by $\frac{1}{b_{\xi_j}}$ for every edge in $M_2^*\cap\xi_j$, and summing over all edges, we get a coefficient of one, therefore validating (4).

Summing (3) and (4), concludes the proof. ■

B.2 PROOF OF PROPOSITION 1

Properties of Markov Chains Before decomposing the regret, we briefly digress to recall some classic results on mixing of Markov chains. For an ergoidic (i.e. irreducible and aperiodic) transition matrix on a finite state space Θ, let π be its stationary distribution and $\tilde{P}$ denote the time reversal of its transition matrix P—that is,

$$\tilde{P}(\theta,\theta') = \frac{\pi(\theta')P(\theta',\theta)}{\pi(\theta)}.$$

The time reversal kernel $\tilde{P}$ is also ergodic with stationary distribution π. Define the *multiplicative reversiblization* $M(P)$ of P by $M(P) = P\tilde{P}$ which is a reversible transition matrix itself. The eigenvalues of $M(P)$ are real and non-negative so that the second largest eigenvalue $\lambda_1(M)\in[0,1]$ (Fill, 1991). Define *chi-squared distance* from stationary at time n by

$$\chi_n^2 = \sum_\theta \frac{(\pi_n(\theta)-\pi(\theta))^2}{\pi(\theta)}.$$

where $\pi_n = \sum_\theta \pi_0(\theta)P^n(\theta,\cdot)$.

Proposition 1 ((Fill, 1991)). *Let P be an ergodic transition matrix on a finite state space Θ and let π be the stationary distribution. Then $4\|\pi_n - \pi\|^2 \leq \big(\lambda_1(M)\big)^n\chi_0^2$. Furthermore, $\max_{\pi_0\in\mathcal{P}(\Theta)} \big\|\sum_\theta P^n(\theta,\cdot)\pi_0(\theta) - \pi(\cdot)\big\|^2 \leq \frac{1}{4}\frac{(1-\min_\theta\pi(\theta))^2}{\min_\theta\pi(\theta)}\big(\lambda_1(M)\big)^n$.*

where $\mathcal{P}(\Theta)$ us the space of probability distributions on Θ[1].

From the perspective of a general epoch mixing policy α, the above proposition provides a bound on how close the distribution on types for the Markov chain is after τ_k time steps has elapsed when edge (a,i) is chosen.

Lemma 1. *Consider an arbitrary epoch mixing policy α that selects a matching $\alpha(k)$ during the k–th epoch for τ_k iterations. For each arm $(a,i)\in\alpha(k)$, there exists a constant $C_{a,i}>0$ such that*

$$\big|\mathbb{E}\big[\mu_{a,i} - \boldsymbol{r}_{a,i}^{\theta_a(k)}\big]\big| \leq \frac{C_{a,i}}{\tau_k} \quad (5)$$

The proof is a direct consequence of Proposition 1.

Proof Noting that $\mu^j = \sum_\theta r_\theta^j \pi^j(\theta)$, a direct application of Proposition 1 gives us the following:

$$\begin{aligned}&\left|\mathbb{E}\left[\textstyle\sum_\theta r_\theta^j\pi^j(\theta) - \frac{1}{\tau_k}\sum_{t=t_k}^{t_{k+1}-1} r_{\theta,t}^j \middle| \theta_{t_k}\right]\right| \\ &\leq \tfrac{1}{\tau_k}\textstyle\sum_{t=t_k}^{t_{k+1}-1}\sum_\theta\left|(\pi^j(\theta)-\beta_t(\theta))\right| \\ &\leq \tfrac{1}{\tau_k}\textstyle\sum_{t=t_k}^{t_{k+1}-1}\sum_\theta\left|(\pi^j(\theta)-\sum_{\theta'}P_j^{t-t_k}(\theta',\theta)\beta_{t_k}(\theta'))\right| \\ &= \tfrac{1}{\tau_k}\textstyle\sum_{t=t_k}^{t_{k+1}-1}\|\pi^j(\cdot)-\sum_{\theta'}P_j^{t-t_k}(\theta',\cdot)\beta_{t_k}(\theta')\|_1 \\ &\leq \tfrac{1}{\tau_k}\textstyle\sum_{t=t_k}^{t_{k+1}-1} C_j\lambda_j^{t-t_k} = \frac{C_j(1-\lambda_j^{\tau_k})}{\tau_k(1-\lambda_j)}\end{aligned}$$

This is simply because of the fact that the expected reward is less than 1 by construction, the triangle inequality, and Fubini's theorem (Folland, 2007, Theorem 2.37). ■

We also remark that Proposition 1 also implies that this bound holds for all $\beta_a^{(k)}$ (i.e. the distribution of agent a's type at the beginning of epoch k) and hence, is independent of the algorithm α.

Proof [Proposition 1] Consider the expression for regret from Definition 1:

$$R^\alpha(n) = n\sum_{g_j^*\in G^*}\mu_{g_j^*} - \sum_{k=1}^n\sum_{(a,i)\in\alpha(k)}\mathbb{E}[\boldsymbol{r}_{a,i}^\theta],$$

[1] We remark that the bound in the above equation is easily computed by noting that χ_n^2 is always bounded above by $(\min_\theta\pi(\theta))^{-1}(1-\min_\theta\pi(\theta))^2$.

By adding and subtracting $\sum_{(a,i)\in\mathcal{P}} T^{\alpha}_{a,i}(n)\mu_{a,i}$ from the above equation, the cumulative regret can be written as:

$$\begin{aligned}
R^{\alpha}(n) &= n\sum_{a\in\mathcal{A}}\mu_{a,i^*(a)} - \sum_{(a,i)\in\mathcal{P}} T^{\alpha}_{a,i}(n)\mu_{a,i} \\
&\quad + \sum_{(a,i)\in\mathcal{P}} T^{\alpha}_{a,i}(n)\mu_{a,i} - \sum_{k=1}^{n}\sum_{(a,i)\in\alpha(k)} \boldsymbol{r}^{\theta_a(k)}_{a,i} \\
&= \sum_{(a,i)\in\mathcal{P}} T^{\alpha}_{a,i}(n)\mu_{a,i^*(a)} - \sum_{(a,i)\in\mathcal{P}} T^{\alpha}_{a,i}(n)\mu_{a,i} \\
&\quad + \sum_{(a,i)\in\mathcal{P}} T^{\alpha}_{a,i}(n)\mu_{a,i} \\
&\quad - \sum_{k=1}^{n}\sum_{(a,i)\in\mathcal{P}} \mathbb{1}\{(a,i)\in\alpha(k)\}\boldsymbol{r}^{\theta_a(k)}_{a,i} \\
&= \sum_{(a,i)\in\mathcal{P}} T^{\alpha}_{a,i}(n)(\mu_{a,i^*(a)} - \mu_{a,i}) \\
&\quad + \sum_{k=1}^{n}\sum_{(a,i)\in\mathcal{P}} \mathbb{1}\{(a,i)\in\alpha(k)\}\big(\mu_{a,i} - \boldsymbol{r}^{\theta_a(k)}_{a,i}\big)
\end{aligned} \tag{6}$$

where $\mathbb{1}(\cdot)$ is the indicator function—e.g., $\mathbb{1}\{(a,i)\in\alpha(k)\}$ is one when the edge (a,i) belongs to the matching $\alpha(k)$. In the term $\sum_{(a,i)\in\mathcal{P}} T^{\alpha}_{a,i}(n)\mu_{a,i^*(a)}$, $\mu_{a,i^*(a)}$ appears exactly n times. Although one would expect the matching chosen by the policy (at least in the initial stages) to be sub-optimal compared to the benchmark greedy matching, it is highly possible that some individual edges (arms) may outperform those in the greedy matching. To account for this, we separate the edges in $\mathcal{P}$ into the sub-optimal edges and the super-optimal ones. Formally, for any given $a\in\mathcal{A}$, define the set of sub-optimal edges S_a as follows:

$$S_a = \{(a,i) \mid \mu_{a,i^*(a)} \geq \mu_{a,i}\ \forall i\in\mathcal{I}\}.$$

Suppose that $\mathcal{S} = \bigcup_{a\in\mathcal{A}} S_a$. Then, the regret bound in Equation (6) can be simplified by ignoring the contribution of the terms in $\mathcal{P}\setminus\mathcal{S}$. That is, since $\mu_{a,i^*(a)} < \mu_{a,i}$ for all $(a,i)\in\mathcal{P}\setminus\mathcal{S}$, we have that:

$$\begin{aligned}
R^{\alpha}(n) &\leq \sum_{(a,i)\in\mathcal{S}} T^{\alpha}_{a,i}(n)(\mu_{a,i^*(a)} - \mu_{a,i}) \\
&\quad + \sum_{k=1}^{n}\sum_{(a,i)\in\mathcal{P}} \mathbb{1}\{(a,i)\in\alpha(k)\}\big(\mu_{a,i} - \boldsymbol{r}^{\theta_a(k)}_{a,i}\big).
\end{aligned} \tag{7}$$

Next, we separate the second term above into the contribution of the edges in $\mathcal{S}$ and those in $\mathcal{P}\setminus\mathcal{S}$. That is, $\sum_{k=1}^{n}\sum_{(a,i)\in\mathcal{P}} \mathbb{1}\{(a,i)\in\alpha(k)\}\big(\mu_{a,i} - \boldsymbol{r}^{\theta_a(k)}_{a,i}\big)$ can be written as:

$$\begin{aligned}
&\sum_{k=1}^{n}\sum_{(a,i)\in\mathcal{S}} \mathbb{1}\{(a,i)\in\alpha(k)\}\big(\mu_{a,i} - \boldsymbol{r}^{\theta_a(k)}_{a,i}\big) \\
&+ \sum_{k=1}^{n}\sum_{(a,i)\in\mathcal{P}\setminus\mathcal{S}} \mathbb{1}\{(a,i)\in\alpha(k)\}\big(\mu_{a,i} - \boldsymbol{r}^{\theta_a(k)}_{a,i}\big)
\end{aligned} \tag{8}$$

We can now use Lemma 1 to bound the difference between the empirical rewards and the stationary reward during any given epoch. Suppose that $\tau_0 \geq 1$ and $\tau_k = \tau_0 + \zeta k$ with ζ a non-zero natural number[2]. An application of Lemma 1 and the tower property of expectation allows us to bound the first term above, i.e., suppose that $T_1 = \mathbb{E}\big[\sum_{k=1}^{n}\sum_{(a,i)\in\mathcal{S}} \mathbb{1}\{(a,i)\in\alpha(k)\}\big(\mu_{a,i} - \boldsymbol{r}^{\theta_a(k)}_{a,i}\big)\big]$. Then,

$$\begin{aligned}
T_1 &= \mathbb{E}_{\alpha}\big[\sum_{k=1}^{n}\sum_{(a,i)\in\mathcal{S}} \mathbb{1}\{(a,i)\in\alpha(k)\}\mathbb{E}\big[\\
&\qquad \mu_{a,i} - \boldsymbol{r}^{\theta_a(k)}_{a,i}\big|\theta_a(k)\big]\big] \\
&\leq \mathbb{E}_{\alpha}\big[\sum_{k=1}^{n}\sum_{(a,i)\in\mathcal{S}} \mathbb{1}\{(a,i)\in\alpha(k)\}\frac{C_{a,i}}{\tau_k}\big] \\
&\leq \mathbb{E}_{\alpha}\big[\sum_{(a,i)\in\mathcal{S}}\sum_{k=1}^{n} \mathbb{1}\{(a,i)\in\alpha(k)\}\frac{C_{a,i}}{\tau_0}\big] \\
&\leq \sum_{(a,i)\in\mathcal{S}} \frac{C_{a,i}}{\tau_0}\mathbb{E}_{\alpha}[T^{\alpha}_{j}(n)]
\end{aligned} \tag{9}$$

where we use the notation $\mathbb{E}_{\alpha}$ to emphasize that this expectation is now dependent only on the algorithm where the number of times an arm is chosen is a random variable. Analogously, bound the second term of Equation 8, i.e., $T_2 = \mathbb{E}\Big[\sum_{k=1}^{n}\sum_{(a,i)\in\mathcal{P}\setminus\mathcal{S}} \mathbb{1}\{(a,i)\in\alpha(k)\}\big(\mu_{a,i} - \boldsymbol{r}^{\theta_a(k)}_{a,i}\big)\Big]$

$$\begin{aligned}
T_2 &\leq \mathbb{E}_{\alpha}\big[\sum_{k=1}^{n}\sum_{(a,i)\in\mathcal{P}\setminus\mathcal{S}} \mathbb{1}\{(a,i)\in\alpha(k)\}\frac{C_{a,i}}{\tau_k}\big] \\
&\leq \sum_{k=1}^{n}\frac{1}{\tau_k}\sum_{(a,i)\in\mathcal{P}\setminus\mathcal{S}} C_{a,i}\mathbb{E}_{\alpha}\big[\mathbb{1}\{(a,i)\in\alpha(k)\}\big] \\
&\leq C_*\sum_{k=1}^{n}\frac{1}{\tau_k}\sum_{(a,i)\in\mathcal{P}} \mathbb{E}_{\alpha}\big[\mathbb{1}\{(a,i)\in\alpha(k)\}\big] \\
&\leq mC_*\sum_{k=1}^{n}\frac{1}{\tau_k},
\end{aligned}$$

[2]There are other choices for the sequence $\{\tau_k\}$; e.g., $\tau_k = a^k\tau_0$. The choice we make allows for tighter bounds.

where $C_* = \max_{(a,i)\in\mathcal{P}\setminus\mathcal{S}} C_{a,i}$. Note that for any given epoch k, our policy selects at most m edges in the matching and therefore, $\sum_{(a,i)\in\mathcal{P}\setminus L} \mathbb{E}_\alpha\big[\mathbb{1}\{(a,i)\in\alpha(k)\}\big] \leq \sum_{(a,i)\in\mathcal{P}} \mathbb{E}_\alpha\big[\mathbb{1}\{(a,i)\in\alpha(k)\}\big] \leq m$. Finally, we can bound the harmonic summation using the fact that $\tau_k = \tau_0 + \zeta k$:

$$\begin{aligned} T_2 &\leq mC_* \sum_{k=1}^{n} \frac{1}{\tau_k} \\ &\leq mC_* \frac{1}{\zeta}\Big(1 + \int_{\tau_0/\zeta}^{n-1+\tau_0/\zeta} \frac{1}{x} dx\Big) \\ &\leq mC_* \frac{1}{\zeta}\Big(1 + \log\Big(\frac{n-1}{\tau_0} + 1\Big)\Big) \end{aligned} \tag{10}$$

Recall from the definition of the marginal infeasibility sets in Equation (1) that for any given $(a,i) \in \mathcal{P} \setminus \mathcal{G}^*$, there exists a unique edge $g_j^* \in G^*$ such that $(a,i) \in L_j^*$. Define $L^{-1}(a,i) \coloneqq g_j^* \in G^*$ such that $(a,i) \in L_j^*$. Now, we can define the reward gap for any given edge $(a,i) \in \mathcal{P}$ as follows:

$$\begin{aligned} \Delta_{a,i} &= \mu_{a,i^*(a)} - \mu_{a,i} && \text{if } (a,i) \in \mathcal{S} \\ &= \mu_{L^{-1}(a,i)} - \mu_{a,i} && \text{if } (a,i) \in (\mathcal{P}\setminus G^*)\setminus\mathcal{S} \\ &= \mu_{g_{j-1}^*} - \mu_{g_j^*} && \text{if } (a,i) = g_j^* \text{ for } j \geq 2. \end{aligned}$$

Going back to our regret lower bound in (7) and decomposing the second term using (9) and (10), we get the main proposition. ■

B.3 PROOF OF THEOREM 2

Before proving Theorem 2, we state some useful supplementary lemmas.

Lemma 2 (Azuma-Hoeffding Inequality (Azuma, 1967; Hoeffding, 1963))**.** *Suppose $(Z^k)_{k\in\mathbb{Z}_+}$ is a martingale with respect to the filtration $(\mathcal{F}^k)_{k\in\mathbb{Z}_+}$ having bounded differences, i.e., there are finite, non-negative constants c^k, $k \geq 1$ such that $|Z^k - Z^{k-1}| < c^k$ almost surely. Then for all $t > 0$*

$$P(Z^k - \mathbb{E}Z^k \leq -t) \leq \exp\left(-\frac{t^2}{2\sum_{k=1}^{N}(c^k)^2}\right).$$

We define some notation that is useful for the following lemma as well the proof of Theorem 2. Consider the MG-EUCB algorithm described in Algorithm 2. Let $R_{a,i}^{\theta,j}$ be the cumulative reward received when arm (a,i) is chosen for the j–th time where we include θ in the subscript to note the state-dependence of the random reward. That is, $R_{a,i}^{\theta,j} = \boldsymbol{r}_{a,i}^{\theta_a(t_{a,i}^j)}$ where, by an abuse of notation, $t_{a,i}^j$ denotes the time instance at which edge (a,i) is pulled for the j–th time and $\theta_a(t_{a,i}^j)$ denotes the state of agent a during that epoch.

Define the filtration $\mathcal{F}_{a,i}^k = \sigma(R_{a,i}^{\theta,1}, \ldots, R_{a,i}^{\theta,k}, \theta_a(t_1^j), \ldots, \theta_a(t_k^j))$—that is, the smallest σ-algebra generated by the random variables $(R_{a,i}^{\theta,1}, \ldots, R_{a,i}^{\theta,k}, \theta_a(t_{a,i}^1), \ldots, \theta_a(t_{a,i}^k))$. Let $X_{a,i}^k = R_{a,i}^{\theta,k} - \mathbb{E}[R_{a,i}^{\theta,k}|\mathcal{F}_{a,i}^{k-1}]$ and $Y_{a,i}^k = \sum_{j=1}^k X_{a,i}^j$. We have that $Y_{a,i}^k$ is a martingale since $\mathbb{E}[Y_{a,i}^{k+1}|\mathcal{F}_{a,i}^k] = \mathbb{E}[X_{a,i}^{k+1}|\mathcal{F}_{a,i}^k] + \mathbb{E}[Y_{a,i}^k|\mathcal{F}_{a,i}^k] = Y_{a,i}^k$ (since $Y_{a,i}^k$ is $\mathcal{F}_{a,i}^k$–measurable by construction) and $\mathbb{E}[|Y_{a,i}^k|] < \infty$ (rewards are bounded). Moreover, the boundedness of the rewards also implies the martingale $Y_{a,i}^k$ has bounded differences. Indeed, $|Y_{a,i}^k - Y_{a,i}^{k-1}| = |X_{a,i}^k| \leq 1$ almost surely since rewards are normalized to be on the interval $[0,1]$, without loss of generality. Now, we are ready to show an upper bound on the difference in the empirical reward and the stationary state rewards.

Lemma 3. *Given aperiodic, irreducible Markov chains $P_{a,i}$ with corresponding stationary distributions $\mu_{a,i}$ for each $(a,i) \in \mathcal{P}$ and mixing sequence $\{\tau_k\}$ such that $\tau_k = \tau_0 + \zeta k$, $\tau_0 \geq 1$, we have that*

$$\begin{aligned} &\Big|\mathbb{E}\Big[\mu_{a,i} - \tfrac{1}{k}\textstyle\sum_{j=1}^k \mathbb{E}[R_{a,i}^{\theta,j}|\mathcal{F}_{a,i}^{j-1}]\Big]\Big| \\ &\leq \tfrac{C_{a,i}}{2k}\Big(\tfrac{1}{\zeta+\tau_0} + \tfrac{1}{\zeta}\log\Big(1 + \tfrac{k\zeta}{\tau_0}\Big)\Big) \end{aligned} \tag{11}$$

The proof of the above lemma follows a similar line of reasoning as Lemma 1.

Proof Since Θ is a finite set with finite elements (i.e. $|x| < \infty$ for all $x \in \Theta$), we are able to use analogous reasoning as was used in Proposition 1 along with the Markov property on the conditional expectation $\mathbb{E}[R_i^j|\mathcal{F}_{i-1}^j]$ to bound $\mu^j - \frac{1}{k}\sum_{i=1}^k \mathbb{E}[R_i^j|\mathcal{F}_{i-1}^j]$ by $\frac{L_j(k)}{k}$ for some constant $L_j(k)$. Indeed, the quantity $V = \Big|\mathbb{E}\big[\mu^j - \frac{1}{k}\sum_{i=1}^k \mathbb{E}[R_i^j|\mathcal{F}_{i-1}^j]\big]\Big|$ can be simplified as follows:

$$\begin{aligned} V &= \Big|\tfrac{1}{k}\textstyle\sum_{i=1}^k \Big(\mathbb{E}\Big[\mu^j - \mathbb{E}[R_i^j|\mathcal{F}_{i-1}^j]\Big]\Big)\Big| \\ &\leq \tfrac{1}{k}\textstyle\sum_{i=1}^k \mathbb{E}\Big[\sum_\theta r_\theta^j \pi^j(\theta) - \\ &\qquad \mathbb{E}\big[(\tau_i^j)^{-1}\sum_{t=t_i^j}^{t_{i+1}^j-1} r_{\theta,t}^j \Big| \mathcal{F}_{i-1}^j\big]\Big] \\ &\leq \tfrac{1}{k}\textstyle\sum_{i=1}^k \mathbb{E}\Big[(\tau_i^j)^{-1}\sum_{t=t_i^j}^{t_{i+1}^j-1}\sum_\theta |\pi^j(\theta) - \beta_t(\theta)|\Big] \\ &\leq \tfrac{1}{k}\textstyle\sum_{i=1}^k \tfrac{C_j}{2}\mathbb{E}\Big[(\tau_i^j)^{-1}\sum_{t=t_i^j}^{t_{i+1}^j-1}(\lambda_j)^{(t-t_i^j)}\Big] \\ &\leq \tfrac{1}{k}\textstyle\sum_{i=1}^k \tfrac{C_j}{2}\mathbb{E}\Big[(\tau_i^j)^{-1}(1-(\lambda_j)^{\tau_i^j})(1-\lambda_j)^{-1}\Big] \end{aligned}$$

$\leq \frac{1}{k}\frac{C_j}{2}\frac{1}{1-\lambda_j}\sum_{i=1}^{k}\mathbb{E}\big[(\tau_i^j)^{-1}\big]$,

where we have used the fact that the reward bounded almost surely on $[0,1]$. Now, $1/\tau_{ij}$ is a random variable with respect to the algorithm since at the i–th pull of arm j we do not know *a priori* what iteration of the algorithm we are on. However, at the i–th pull of arm, we do know that the algorithm is at least at the i–th iteration. Hence, $\sum_{i=1}^{k}\mathbb{E}\big[(\tau_i^j)^{-1}\big] \leq \sum_{i=1}^{k}(\tau_0+\zeta i)^{-1}$. Now, for any $a \geq 1$ and positive integer k, we have that $\sum_{i=a}^{a+k}(i)^{-1} \leq \frac{1}{a} + \log(1+\frac{k}{a})$. Indeed, rewrite the summation in the lemma statement as $\sum_{i=a}^{a+k} i^{-1} = a^{-1}+\sum_{i=a+1}^{a+k} i^{-1}$ and apply the fundamental inequality, $(i)^{-1} \leq \int_{i-1}^{i} x^{-1}dx$, which holds for any $i \geq 1$, repeatedly for $i = a+1, a+2, \ldots, a+k$ so that we have a telescoping summation of integrals—i.e.

$$\begin{aligned}\textstyle\sum_{i=a}^{a+k}\frac{1}{i} &= \frac{1}{a} + \textstyle\sum_{i=a+1}^{a+k}\frac{1}{i} \\ &\leq \frac{1}{a} + \int_{a}^{a+k}\frac{1}{x}dx = \frac{1}{a} + \log\left(\frac{a+k}{a}\right).\end{aligned}$$

Thus, $\sum_{i=1}^{k}(\tau_0+\zeta i)^{-1} \leq (\zeta+\tau_0)^{-1} + \frac{1}{\zeta}\log\left(1+\frac{k\zeta}{\tau_0}\right)$ so that (11) holds. ■

Proof [Theorem 2] We begin by formalizing the choice of the UCB parameter $c_{a,i}^k(t)$—it is crucial that this parameter reflects the error due to both the Markov chain and the randomness of rewards. Applying Lemma 3 to our problem, we observe that the average error stemming from the randomness in the user state after k pulls of the edge (a,i) can be written as:

$$\begin{aligned}&\left|\mathbb{E}\left[\mu_{a,i} - \tfrac{1}{k}\textstyle\sum_{j=1}^{k}\mathbb{E}[R_{a,i}^{\theta,j}|\mathcal{F}_{a,i}^{j-1}]\right]\right| \\ &\leq \tfrac{C_{a,i}}{2k}\left(\tfrac{1}{\zeta+\tau_0} + \tfrac{1}{\zeta}\log\left(1+\tfrac{k\zeta}{\tau_0}\right)\right)\end{aligned}$$

Based on this, for each edge (a,i) and 'pull count' k, we define the constant $Q_{a,i}(k)$

$$Q_{a,i}(k) = \frac{C_{a,i}}{2}\left(\frac{1}{\zeta+\tau_0} + \frac{1}{\zeta}\log\left(1+\frac{k\zeta}{\tau_0}\right)\right).$$

Finally, we can now define the confidence parameter as follows:

$$c_{a,i}^k(t) = Q_{a,i}(k)/k + \sqrt{\frac{6}{k}\log(t) + \frac{4}{k}\log(m)}.$$

Coming back to the proof of Theorem 2, our primary goal is to map every selection of a sub-optimal edge to a condition on the relative empirical rewards between edges that can then be resolved using Azuma-Hoeffding inequality. Applying Lemma 1, we see that if MATCH-GREEDY does not return the benchmark matching G^* at epoch t and instead returns a matching $\alpha(t) \neq G^*$, at least one of the above conditions must fail. Alternatively, this implies that one of the following two (inverse) conditions must be true:

1. $\mathbb{1}\{\exists j < j' |\ \big(u_{g_{j'}^*}(t) > u_{g_j^*}(t)\big) \wedge (g_{j'}^* \in \alpha(t))\}$
2. $\mathbb{1}\{\exists j, (a,i) \in L_j^* |\ \big(u_{g_j^*}(t) < u_{a,i}(t)\big) \vee ((a,i) \in \alpha(t))\} = 1$

To express the above conditions in a concise manner, let us augment the sets L_j^* to include edges from the greedy matching. Specifically, for all $1 \leq j \leq m-1$, let $L_j^+ = L_j^* \cup \{g_{j+1}^*\}$ and $L_m^+ = L_m^*$. Observe that $\bigcup_j L_j^+ = \mathcal{P} \setminus g_1^*$. Now, we can formally say that if the matching returned by the UCB algorithm during iteration t (call this matching $\alpha(t)$) does not coincide with the greedy matching, then

$$\begin{aligned}&\mathbb{1}\{\exists 1 \leq j \leq m, (a,i) \in L_j^+ |\ u_{g_j^*}(t) \\ &\quad < u_{a,i}(t) \wedge (a,i) \in \alpha(t)\} = 1. \qquad (12)\end{aligned}$$

We will use the notation $\bar{R}_{a,i}^k = \frac{1}{k}\sum_{j=1}^{k} R_{a,i}^{\theta,j}$. Since Proposition 1 provides an upper bound for the regret in terms of the number of times each (sub-optimal) edge is chosen, it suffices to bound the quantity $T_{a',i'}(n)$, which is the number of times our UCB algorithm selects the edge (a',i') given that $(a',i') \in \mathcal{S}$—i.e. $\mu_{a',i'} < \mu_{a',i^*(a')}$. Note that by definition, for any $(a,i) \in \mathcal{S}$, the edge (a',i') does not belong to the greedy benchmark matching G^*. Suppose that ℓ denotes an arbitrary integer (to be formalized later). Then, we have that:

$$\begin{aligned}T_{a',i'}(n) &= 1 + \textstyle\sum_{t=m+1}^{n}\mathbb{1}\{(a',i') \in \alpha(t)\} \\ &\leq 1 + \textstyle\sum_{t=m+1}^{n}\mathbb{1}\{\exists j, (a,i) \in L_j^+ |\ u_{g_j^*}(t) \\ &\quad < u_{a,i}(t) \wedge (a,i) \in \alpha(t)\} \quad \text{(from (12))} \\ &\leq 1 + \textstyle\sum_{t=m+1}^{n}\sum_{j=1}^{m}\sum_{(a,i)\in L_j^+}\mathbb{1}\{u_{g_j^*}(t) \leq \\ &\quad u_{a,i}(t) \wedge (a,i) \in \alpha(t)\} \\ &= 1 + \textstyle\sum_{j=1}^{m}\sum_{(a,i)\in L_j^+}\sum_{t=m+1}^{n}\mathbb{1}\{u_{g_j^*}(t) \\ &\quad \leq u_{a,i}(t) \wedge (a,i) \in \alpha(t)\} \\ &\leq 1 + \textstyle\sum_{j=1}^{m}\sum_{(a,i)\in L_j^+}\big(\ell \\ &\quad + \textstyle\sum_{t=m+1}^{n}\mathbb{1}\{u_{g_j^*}(t) \leq u_{a,i}(t) \\ &\quad \wedge (a,i) \in \alpha(t) \wedge T_{a,i}(t) > \ell\}\big) \\ &\leq 1 + \textstyle\sum_{j=1}^{m}\sum_{(a,i)\in L_j^+}\big(\ell \\ &\quad + \textstyle\sum_{t=m+1}^{n}\mathbb{1}\{u_{g_j^*}(t) \leq u_{a,i}(t) \\ &\quad \wedge T_{a,i}(t) > \ell\}\big) \\ &\leq \ell m^2 + \textstyle\sum_{j=1}^{m}\sum_{(a,i)\in L_j^+}\sum_{t=m+1}^{n}\mathbb{1}\{u_{g_j^*}(t) \\ &\quad \leq u_{a,i}(t) \wedge T_{a,i}(t) > \ell\} \\ &\leq \ell m^2 + \textstyle\sum_{j=1}^{m}\sum_{(a,i)\in L_j^+}\sum_{t=m+1}^{n}\big(\end{aligned}$$

$$
\begin{aligned}
&\mathbb{1}\{\min_{0<s<t} u^s_{g^*_j}(t) \le \max_{\ell\le k<t} u^k_{a,i}(t)\}\big)\\
&\le \ell m^2 + \textstyle\sum_{j=1}^m \sum_{(a,i)\in L_j^+} \big(\\
&\quad \textstyle\sum_{t=m+1}^n \sum_{s=1}^{t-1}\sum_{k=\ell}^{t-1} \mathbb{1}\{u^s_{g^*_j}(t) \le u^k_{a,i}(t)\}\big)\\
&= \ell m^2 + \textstyle\sum_{j=1}^m \sum_{(a,i)\in L_j^+} \sum_{t=m+1}^n \sum_{s=1}^{t-1} \big(\\
&\quad \textstyle\sum_{k=\ell}^{t-1} \mathbb{1}\{\bar{R}^s_{g^*_j} + c^s_{g^*_j}(t) \le \bar{R}^k_{a,i} + c^k_{a,i}(t)\}\big)
\end{aligned}
$$

Now, $\bar{R}^s_{g^*_j} + c^s_{g^*_j}(t) \le \bar{R}^k_{a,i} + c^k_{a,i}(t)$ implies that atleast one of the following must hold:

$$
\begin{aligned}
\bar{R}^s_{g^*_j} &\le \mu_{g^*_j} - c^s_{g^*_j}(t) &(13)\\
\bar{R}^k_{a,i} &\ge \mu_{a,i} + c^k_{a,i}(t) &(14)\\
\mu_{g^*_j} &< \mu_{a,i} + 2c^k_{a,i}(t) &(15)
\end{aligned}
$$

Indeed, suppose that all three of the above inequalities are false. Then, $u^s_{g^*_j}(t) = \bar{R}^s_{g^*_j} + c^s_{g^*_j}(t) > \mu_{g^*_j} \ge \mu_{a,i} + 2c^k_{a,i}(t) > \bar{R}^k_{a,i} + c^k_{a,i}(t) = u^k_{a,i}(t)$, which is, of course, a contradiction. Hence, if $\bar{R}^s_{g^*_j} + c^s_{g^*_j}(t) \le \bar{R}^k_{a,i} + c^k_{a,i}(t)$, then at least one of (13)–(15) holds. We bound the probability of events (13) and (14) using the Azuma-Hoeffding inequality in Lemma 2 and find an ℓ such that (15) is always false for every $j, (a,i) \in L_j^+$.

Towards this end, we apply Lemma 2 to the martingale $(Y^k_{a,i})_{k\in\mathbb{Z}_+}$. Note that by the law of conditional expectations, $\mathbb{E}[Y^k_{a,i}] = 0$ so that Lemma 2 implies that for each arm (a,i) and any $t > 0$, $P(Y^k_{a,i} \le -t) \le \exp(-t^2/(2k))$.

We need to relate the random variable $Y^k_{a,i}$ to the difference of the empirical mean of the average cumulative reward from its true value for each arm so that we can bound this difference. Consider the event

$$
\begin{aligned}
\omega &= \big\{\mu_{g^*_j} - \bar{R}^s_{g^*_j} \ge \gamma\big\}\\
&= \big\{\mu_{g^*_j} - \tfrac{1}{s}\textstyle\sum_{l=1}^s \mathbb{E}[R^{\theta,l}_{g^*_j}|\mathcal{F}^{l-1}_{g^*_j}]\\
&\quad + \tfrac{1}{s}\textstyle\sum_{l=1}^s \mathbb{E}[R^{\theta,l}_{g^*_j}|\mathcal{F}^{l-1}_{g^*_j}] - \bar{R}^s_{g^*_j} \ge \gamma\big\}\\
&= \big\{\mu_{g^*_j} - \tfrac{1}{s}\textstyle\sum_{l=1}^s \mathbb{E}[R^{\theta,l}_{g^*_j}|\mathcal{F}^{l-1}_{g^*_j}] - \tfrac{1}{s}Y^s_{g^*_j} \ge \gamma\big\}
\end{aligned}
$$

where we have added and subtracted the random variable $\frac{1}{s}\sum_{l=1}^s \mathbb{E}[R^{\theta,l}_{g^*_j}|\mathcal{F}^{l-1}_{g^*_j}]$. By Lemma 3,

$$
\begin{aligned}
\omega &\subset \big\{\tfrac{1}{s}Q_{g^*_j}(s) - \tfrac{1}{s}Y^s_{g^*_j} \ge \gamma\big\}\\
&= \big\{\tfrac{1}{s}Y^s_{g^*_j} \le \tfrac{1}{s}Q_{g^*_j}(s) - \gamma\big\}.
\end{aligned}
$$

Hence,

$$
\begin{aligned}
P\big(\mu_{g^*_j} - \bar{R}^s_{g^*_j} \ge \gamma\big) &\le P\Big(\frac{1}{s}Y^s_{g^*_j} \le \frac{1}{s}Q_{g^*_j}(s) - \gamma\Big)\\
&\le \exp\big(-\frac{1}{2}s\big(\gamma - \frac{1}{s}Q_{g^*_j}(s)\big)^2\big)
\end{aligned}
$$

so that with $\gamma = c^s_{g^*_j}(t) = \sqrt{\frac{6}{s}\log t + \frac{4}{s}\log m} + \frac{1}{s}Q_{g^*_j}(s)$, we have,

$$
P\left(\mu_{g^*_j} - \bar{R}^s_{g^*_j} \ge c^s_{g^*_j}(t)\right) \le t^{-3}m^{-2}.
$$

Therefore, it follows that $P(\bar{R}^s_{g^*_j} \le \mu_{g^*_j} - c^s_{g^*_j}(t)) \le t^{-3}m^{-2}$ and $P(\bar{R}^k_{a,i} \ge \mu_{a,i} + c^k_{a,i}(t)^*) \le t^{-3}m^{-2}$ which imply that (13) and (14) occur with very low probability.

Now, we choose ℓ to be the largest integer such that (15) is always false. Indeed, we choose it such that

$$
\begin{aligned}
&\mu_{g^*_j} - \mu_{a,i} - 2c^k_{a,i}(t)\\
&> \mu_{g^*_j} - \mu_{a,i} - 2\left(\tfrac{Q_{a,i}(\ell)}{\ell} + \sqrt{\tfrac{6\log t}{\ell} + \tfrac{4\log m}{\ell}}\right) > 0.
\end{aligned}
$$

Plugging in $Q_{a,i}(\ell)$, we have

$$
\begin{aligned}
&\Delta_{a,i} - 2\Big(\tfrac{C_{a,i}}{2\ell}\left(\tfrac{1}{\zeta+\tau_0} + \tfrac{1}{\zeta}\log\left(1 + \tfrac{\ell\zeta}{\tau_0}\right)\right)\\
&\quad + \sqrt{\tfrac{1}{\ell}6\log t + \tfrac{1}{\ell}4\log m}\Big) > 0. \qquad (16)
\end{aligned}
$$

Let $\tilde{\ell} = \ell\zeta/\tau_0$ so that

$$
\begin{aligned}
&\Delta_{a,i} - 2\Big(\tfrac{C_{a,i}}{2\tau_0}\left(\tfrac{1}{\tilde{\ell}}\tfrac{\zeta}{\zeta+\tau_0} + \tfrac{1}{\tilde{\ell}}\log\left(1+\tilde{\ell}\right)\right)\\
&\quad + \sqrt{\tfrac{6\log t}{\ell} + \tfrac{4\log m}{\ell}}\Big) > 0.
\end{aligned}
$$

Since $1/x < 1/\sqrt{x}$ and $1/x\log(1+x) < 1/\sqrt{x}$ on $[1,\infty)$, we have that

$$
\frac{1}{\tilde{\ell}}\frac{\zeta}{\zeta+\tau_0} + \frac{1}{\tilde{\ell}}\log\left(1+\tilde{\ell}\right) < \frac{\zeta}{\zeta+\tau_0}\frac{1}{\sqrt{\tilde{\ell}}} + \frac{1}{\sqrt{\tilde{\ell}}}
$$

so that (16) reduces to finding the largest integer ℓ such that

$$
\begin{aligned}
&\Delta_{a,i} - 2\Big(\frac{C_{a,i}}{2\tau_0}\left(\frac{\zeta}{\zeta+\tau_0}\frac{\sqrt{\tau_0}}{\sqrt{\ell\zeta}} + \frac{\sqrt{\tau_0}}{\sqrt{\ell\zeta}}\right)\\
&\quad + \frac{\sqrt{6\log t + 4\log m}}{\sqrt{\ell}}\Big) > 0
\end{aligned}
$$

Rearranging and squaring terms, we get that (15) is false for

$$
\ell \ge \Big\lceil \frac{4}{\Delta^2_{a,i}}\big(\frac{\rho_{a,i}}{\sqrt{\tau_0}} + \sqrt{6\log n + 4\log m}\big)^2\Big\rceil. \qquad (17)
$$

In the above equation, $\rho_{a,i}$ is the edge-specific constant

$$
\rho_{a,i} = \big(\frac{\zeta}{\zeta+\tau_0} + 1\big)\frac{C_{a,i}}{2\sqrt{\zeta}}.
$$

In fact, we require that (15) be false for all $1 \le j \le m$ and $(a,i) \in L_j^+$. Therefore, we set the parameter ℓ to be

the maximum of the right hand side of (17). Formally, define (a^*, i^*) to be the edge in $\mathcal{P} \setminus g_1^*$ that maximizes the right hand side of (17). That is, for a given instance,

$$(a^*, i^*) = \text{argmax}_{(a_1,i_1)\in\mathcal{P}\setminus g_1^*} \left\lceil \frac{4}{\Delta^2_{a_1,i_1}} \left(\frac{\rho_{a_1,i_1}}{\sqrt{\tau_0}} + \sqrt{6\log n + 4\log m} \right)^2 \right\rceil \tag{18}$$

Then, by defining ℓ as follows, we are assured that Equation 17 holds for all $1 \le j \le m$ and $(a, i) \in L_j^+$.

$$\ell = \left\lceil \frac{4}{\Delta^2_{a^*,i^*}} \left(\frac{\rho_{a^*,i^*}}{\sqrt{\tau_0}} + \sqrt{6\log n + 4\log m} \right)^2 \right\rceil \tag{19}$$

Hence, we can bound the number of plays of our original sub-optimal arm (a', j') as follows:

$$\begin{aligned}
\mathbb{E}[T_{a',i'}(n)] &\le \ell m^2 + \sum_{j=1}^{m} \sum_{(a,i)\in L_j^+} \sum_{t=m+1}^{n} \\
&\quad \sum_{s=1}^{t-1} \sum_{k=\ell}^{t-1} \big(P(\bar{R}^s_{g_j^*} \le \mu_{g_j^*} - c^s_{g_j^*}(t)) \\
&\quad + P(\bar{R}^k_{a,i} \ge \mu_{a,i} + c^k_{a,i}(t)) \big) \\
&\le \left\lceil \frac{4m^2}{\Delta^2_{a^*,i^*}} \Big(\frac{\rho_{a^*,i^*}}{\sqrt{\tau_0}} + \sqrt{6\log n + 4\log m} \Big)^2 \right\rceil \\
&\quad + \sum_{(a,i)\in\mathcal{P}} \sum_{t=1}^{n} \sum_{s=1}^{t} \sum_{k=1}^{t} 2t^{-3}m^{-2} \\
&\le \frac{4m^2}{\Delta^2_{a^*,i^*}} \Big(\frac{\rho_{a^*,i^*}}{\sqrt{\tau_0}} + \sqrt{6\log n + 4\log m} \Big)^2 \\
&\quad + 2(1 + \log(n)).
\end{aligned}$$

■

As a direct consequence of Theorem 2, we can bound the regret of the MatchGreedy-EpochUCB policy.

Corollary 1 (Regret Bound for UCB)**.** *Consider α as the MatchGreedy-EpochUCB algorithm and suppose that $\tau_k = \tau_0 + \zeta k$ with $\tau_0 \ge 1$. The regret bound is*

$$\begin{aligned}
R^\alpha(n) \le \sum_{(a,i)\in\mathcal{S}} &\left(\frac{4m^2}{\Delta^2_{a^*,i^*}} \big(\frac{\rho_{a^*,i^*}}{\sqrt{\tau_0}} + \sqrt{6\log n + 4\log m} \big)^2 \right. \\
&\left. + 2(1 + \log(n)) \right) \left(\Delta_{a,i} + \frac{C_{a,i}}{\tau_0} \right) \\
&+ \frac{mC_*}{\zeta} \left(1 + \log\left(\frac{\zeta(n-1)}{\tau_0} + 1 \right) \right),
\end{aligned}$$

where (a^*, i^*) is an edge defined in (18) *and ρ_{a^*,i^*} and $C_{a,i}$ are edge-specific constants.*

C UCB ALGORITHM

C.1 INITIAL PLAY OF UCB ALGORITHM

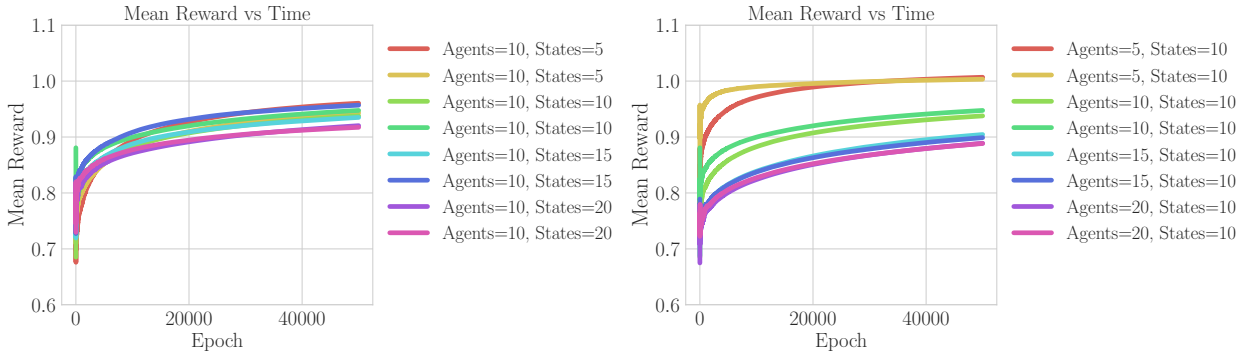
Since the UCB algorithm estimates the average reward for each edge (a, i), it is customary to initialize a preliminary round where each arm is played exactly once. In the absence of any capacity constraints (e.g., $b_{\xi_l} = m$ for all $\xi_l \in \mathcal{C}$), it is easy to compute a sequence of m matchings so that every edge in $\mathcal{P}$ belongs to exactly one of these matchings. We now present a procedure that achieves the same effect even in the presence of arbitrary capacity constraints.

Algorithm 1 Computation of disjoint matchings that play each arm once

```
function MATCHINGS-INITIALPLAY(P)
    E ← P                                  ▷ Edges not yet selected
    i ← 1                                  ▷ Index for current matching
    while E ≠ ∅ do
        F ← E    ▷ Feasible set for current matching
        M ← ∅
        while F ≠ ∅ do
            Select any (a, i) ∈ F
            if M ∪ (a, i) does not violate (P1) then
                M ← M ∪ (a, i)
            else
                F ← F \ (a, i).
            end if
        end while
        M_i ← M, i ← i + 1, E ← E \ M.
    end while
    return M_1, M_2, ..., M_{i-1}
end function
```

Informally, in some iteration i, the above algorithm greedily selects edges for matching M_i without violating the capacity constraints. When no additional edge can be added to M_i—a maximal matching—we move on to the next iteration.

Unfortunately, the number of matchings returned by this procedure can be quite large—in the worst case this can be as large as m^2, where m is the number of agents or incentives. However, for more reasonable instances such as the ones considered in our simulations, we observe that the number of initial matchings required to play each edge at least once is much closer to the lower bound of m.



(a) (b)

Figure 1: Figure 1a presents results demonstrating how the performance of our algorithm varies with the number of states given that the number of agents and incentives is fixed for two instances of each configuration. Figure 1b shows how the performance of the algorithm varies with the number of agents and incentives given that the number of states is fixed for two instances of each configuration.

Algorithm 2 Environment Implementation for Pulling a Matching (Set of Arms)

function INCENT(M, t_n, n τ_0,ζ)
  $r_{a,i}^{t_n} \leftarrow 0 \quad \forall (a,i) \in M$
  for $t \in [t_n, t_n + \tau_0 + \zeta n - 1]$ **do**
    for $(a,i) \in M$ **do**
      offer incentive i to agent a
      receive reward $r_{a,i}^{\theta_a,t}$
      $r_{a,i}^{t_n} \leftarrow r_{a,i}^{\theta,t} + r_{a,i}^{t_n}$
    end for
  end for
  return $(r_{a,i}^{t_n})_{(a,i)\in M}$
end function

D ADDITIONAL EXPERIMENTS

D.1 COMPARISON OF TRADITIONAL UCB AND MG-EUCB FOR SIMPLE EXAMPLE

We return to the simple two-agent two-incentive instance depicted in Figure 1. We ignore the capacity constraints by assuming that there is a single class C_1 such that every edge belongs to this class and $b_{C_1} = 2$. Clearly, this instance only admits two unique matchings $M^* = \{(a_1, i_1), (a_2, i_2)\}$—the optimum matching—and $M = \{(a_1, i_2), (a_2, i_1)\}$—the sub-optimal matching.

As discussed previously, any traditional bandit approach that ignores the evolution of agent rewards would converge to the sub-optimal matching, i.e., M. To see why, observe that every time the algorithm selects the matching M , both the agents' states are reset to θ_1 . Following this, when the algorithms 'explores' the optimum matching, the reward consistently happens to be zero since the agents are in state θ_1. Owing to this, the traditional approach largely underestimates the rewards for the (edges in the) optimum matching and converges to M.

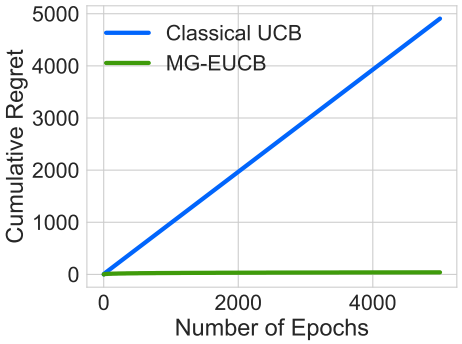
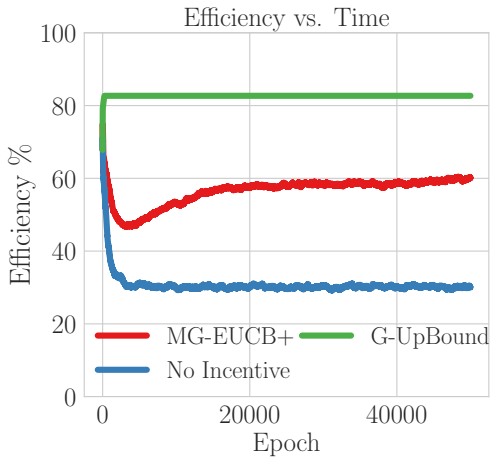


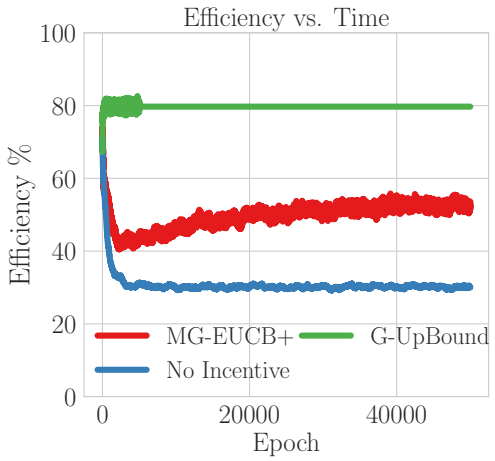
Figure 3: Comparison of the performance of classical UCB algorithms for matching problems versus the MatchGreedy-EpochUCB algorithm for the example depicted in Figure 1.The length of horizon was $n = 5000$.

To validate this experimentally, we compare the performance of our MatchGreedy-EpochUCB algorithm described in Algorithm 2 to a conventional implementation of the UCB algorithm for matching problems (e.g., as in (Chen et al., 2016; Gai et al., 2011; Kveton et al., 2015)). More specifically, we consider an implementation that runs for a total of $\sum_{i=1}^{k} \tau_k$ for some suitable set of parameters—in each iteration, the algorithm selects a matching based on the empirical rewards and the confidence bound. The iterations are then divided into rewards for convenience and the time-average reward in each epoch is computed and plotted alongside the same metric for the MG-EUCB algorithm in Figure 3.

Our simulations support our prior conclusions. For ex-



(a) Static Demand



(b) Random Demand

Figure 2: Bike-share experiments with utility model: Figures 2a and 2b compare the efficiency of the bike-share system with two demand models and a utility based behavioral model under incentive matchings selected by MG-EUCB+ with upper and lower bounds given by the system performance when the incentive matching is given by computing the optimal greedy matching at each epoch based on the current state information and when no incentives are offered respectively.

ample, after 5000 epochs, the classical UCB algorithm selects the sub-optimal matching over 99% of the time. Owing to this reason, the classical algorithm has a regret that grows linearly with the length of the horizon whereas the regret of our algorithm is almost zero for this instance.

D.2 ADDITIONAL SYNTHETIC EXPERIMENTS

In our synthetic simulations we fixed the number of agents, incentives, and states equally as $m = |\mathcal{A}| = |\mathcal{I}| = |\Theta_a| = 10$. We now present results in Figure 1 evaluating how the performance of our algorithm varies with each of these parameters. In Figure 1a, we observe that when the number of agents and incentives is fixed, the number of states has a negligible impact on the rate of convergence to the optimal solution. This indicates that within this range of states the Markov chains mix rapidly and the edge dependent constants in the regret bound do not significantly factor in. We find in Figure 1b, as predicted by our regret bounds, the convergence slows as the number of agents in the problem increases.

D.3 ADDITIONAL BIKE-SHARE DESCRIPTION AND EXPERIMENTS

In this section we provide further motivation for the bike-sharing problem as a matching problem, more detail on our problem setup, as well as additional experimental results. Bike-share programs must deal with varying spatio-temporal demand to ensure that a high percentage of demand is met in order to satisfy customers and maximize profit. To avoid both pile-ups of bikes at popular destinations and depletion of bikes at stations with high demand, bike sharing companies manually replenish and manipulate the spatial supply of bikes. This is costly to companies and an alternative is to attempt to incentivize users to alter their paths in order to balance the spatial supply of bikes in such a way that meets future demand. A successful incentive system could reduce the need for manually replenishing the supply of bikes at stations, saving money and time as a result.

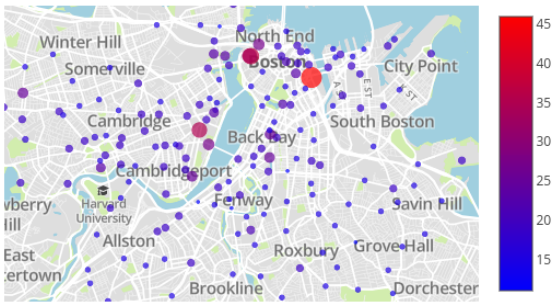


Figure 4: Heatmap of the scaled initial supply of the Boston Hubway stations. Each bubble indicates the location of a station and are scaled in size and colored according to the number of bikes available at the station.

We consider the bike-share problem as a repeated game in our simulations. Specifically, at each epoch users move into the system seeking a bike from a station

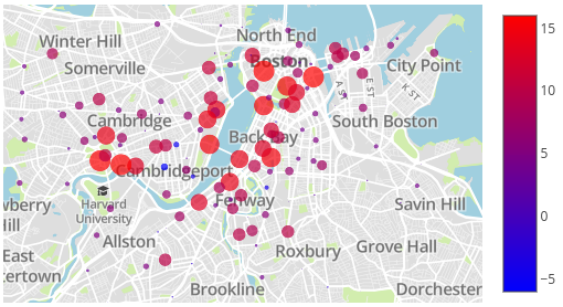


Figure 5: This heatmap shows the spatial reduction in the number of rejections at each station in epoch 20000 from epoch 1000 corresponding to the result in Figure 3a. Positive numbers indicate how many fewer rejections occurred at the station at the later epoch than the earlier epoch. We observe a global reduction spatially in rejections nearly uniformly.

while simultaneously users transition from the location in which they picked up a bike to a location where they drop off the bike. In our simulations we allow the spatial supply of bikes to evolve based on the transitions of bikes between stations. We begin each simulation with the supply at each station given by the data scaled by a factor of two. As a result we have over 6000 agents in the system that can move between close to 200 stations.

We experimented with static and random demand models using quantities derived from the data. In the static demand model we set the demand between a directed pair of stations at each epoch to be the empirical mean of the number of transitions between the stations within 12PM–1PM at each day over June, 2017 – August, 2017. In our random demand model we used the empirical means as the parameter of a Poisson distribution from which we sampled the demand at each epoch for each directed pair of stations. To justify this choice we have included several representative probability mass functions for the demand between stations and the Poisson distributions that were fit to them in Figure 6. We also applied goodness of fit tests to ensure this was a realistic modeling choice.

In our simulations we considered two behavioral models of the users in the system that govern how rewards are produced as well as the probability of a user accepting an incentive. As touched upon previously, in our bike-share model, associated with the state of a user are a distance threshold parameter and a parameter of a Bernouilli distribution. The distance threshold gives the maximum distance a user is willing to be re-routed and is drawn uniformly at random for each state in $[0, 4000]$ meters. The Bernouilli parameter gives the probability that a user will accept an incentive below its distance threshold for a particular state and is drawn uniformly at random in $[0, 1]$. In the primary behavioral model we consider based on a Bernouilli distribution presented in Figure 3, if the distance between the two stations of the proposed incentive is less than the threshold parameter associated with an agent's state the agent will accept the incentive with probability p and give a reward of one, otherwise the incentive will be rejected and a reward of zero will be given. We also investigate a utility-based model; this model is the same as the Bernouilli based model with the slight modification that if an incentive is accepted following a successful realization of the Bernouilli draw, a reward is given that is proportional to the difference in distance between the threshold associated with a users state and the distance between the station the user intended to go to and the station of the proposed incentive.

We now give an overview of our results and the additional experiments we present in this section. We make two key favorable observations from the simulations in Figure 3 in which we investigated static and random demand with the Bernoulli behavioral model. First, compared to a naive baseline of the convergence of the system without any incentives our algorithm is able to increase the efficiency of the system approximately 40% with the static demand model. Furthermore, the extension to random demand does not reduce the performance significantly. When comparing to an upper bound on performance we observe that our algorithm leads the system to approach this limit.

The mean matching rewards presented in Figure 3c can be interpreted as the mean number of incentives that are accepted and equivalently the mean of users re-routed. This result indicates that on average less than 1% of users are matched to an incentive. This is a highly desirable property as it means we only need to influence a small part of the population in order to get significant performance gains. As a result, most users will only benefit from the incentive system, while from the planners perspective the minuscule cost of incentivizing only a small portion of the population is a beneficial.

We now show the results in Figure 2 of the static and random demand in combination with the utility based behavioral model. We generally draw the same conclusions as from Figure 3 with somewhat lower performance for the system. This is an expected result as the users are more sensitive to the extra distance they must travel due to an incentive and they are therefore more difficult to incentivize. We note that we observed looking at the additional distances traveled due to an accepted incentive, that users under the utility based model do travel modestly less additional distance as a result of accepting an incentive than when we used the Bernouilli based model.

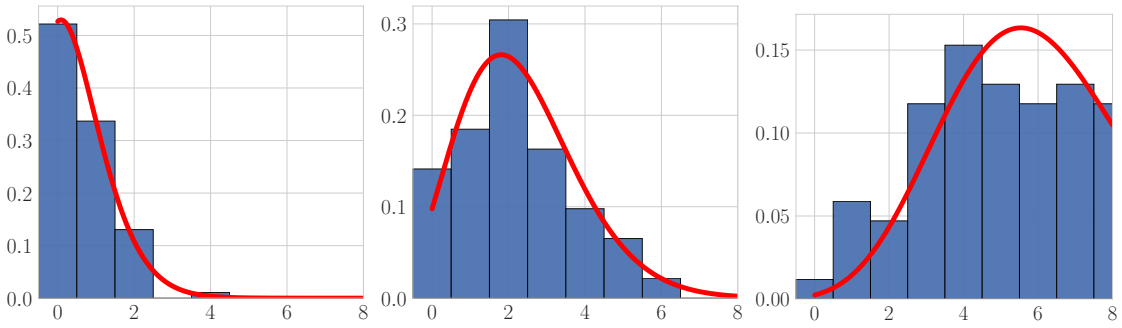


Figure 6: Each empirical probability mass function in the figure gives the probability on the number of users that transitioned between a pair of stations in the Boston Hubway dataset between 12PM–1PM each day between June, 2017 – August, 2017. The red lines show the Poisson distribution that we fit to the distributions that we sampled from to generate random demand at each epoch of the simulation.

E IMPLEMENTATION DETAILS

We make a small modification to the number of iterations within an epoch to reduce computation time of the MG-EUCB algorithm. Specifically when the time-averaged reward has changed by no more than 5×10^{-4} between consecutive iterations for 200 iterations in a row—indicating the time averaged reward has converged—we end the epoch early. We find that this leads to the number of iterations in an epoch being roughly in the range of 1000-1500. We observe this leads to a negligible change in the mean and cumulative rewards of the algorithm while significantly speeding up computation over a large horizon.

F Discussion

In this work we developed a bandit algorithm for matching incentives to users, whose preferences are unknown a priori and evolving dynamically in time, in a resource constrained environment. We theoretically analyzed the problem and derived logarithmic gap-dependent regret bounds. There are several interesting future lines of work that we believe are worth pursuing.

In this work, under the MDP dynamics we only investigated the combinatorial optimization problem of resource constrained matching and our proof techniques relied on the properties of the greedy matching paradigm. In future work, we are interested in attempting to extend this work to arbitrary combinatorial optimization problems with constraints in the case that the designer is allowed oracle access to solve the optimization problem, as has been done in the case without dynamics (Kveton et al., 2015; Wen et al., 2015).

The resource constraints that we considered were static over time. It is often the case that constraints of this form are time-varying or coupled over the decision-making horizon. A prominent example in online resource allocation is the Adwords problem. Due to the practical significance, we plan to explore if our model can be adapted to capture this richer class of constraints.

Finally, we would like to make our model increasingly realistic from the designer's and agents' perspectives. From the designer's point of view, this would be to incorporate incentive compatibility and fairness constraints. From the perspective of the agent, beyond the MDP dynamics, strategic behavior will be important to model and assess the impacts of going forward.