

A Appendix

A.1 Proof of Theorem 3

Define $D(I)$ to be the information sets of player i reachable from I (including I). Define $\sigma|_{D(I) \rightarrow \sigma'}$ to be a strategy profile equal to σ except in the information sets in $D(I)$ where it is equal to σ' . The **full counterfactual regret** is:

$$R_{i,\text{full}}^T(I) = \max_{\sigma' \in \Sigma_1} \sum_{t=1}^T \pi_{-i}^{\sigma^t}(I) (u_i(\sigma^t|_{D(I) \rightarrow \sigma'}, I) - u_i(\sigma^t, I)) \quad (8)$$

Again, we define $R_{1,\text{full}}^{T,+}(I) = \max(R_{1,\text{full}}^T(I), 0)$. Moreover, we define $\text{succ}_i^\sigma(I'|I, a)$ to be the probability that I' is the next information set of player i visited given that the action a was just selected in information set I , and σ is the current strategy. If σ implies that I is unreachable because of an action of player i , that action is changed to allow I to be reachable. Define $\text{Succ}_i(I, a)$ to be the set of all possible next information sets of player i visited given that action $a \in A(I)$ was just selected in information set I . Define $\text{Succ}_i(I) = \bigcup_{a \in A(I)} \text{Succ}_i(I, a)$.

The following lemma describes the relationship between full and immediate counterfactual regret.

Lemma 5 $R_{i,\text{full}}^T(I) \leq R_{i,\text{imm}}^T(I) + \sum_{I' \in \text{Succ}_i(I)} R_{i,\text{full}}^{T,+}(I')$

Proof:

$$R_{i,\text{full}}^T(I) = \frac{1}{T} \max_{a \in A(I)} \max_{\sigma' \in \Sigma_1} \sum_{t=1}^T \pi_{-i}^{\sigma^t}(I) \left(u_i(\sigma^t|_{I \rightarrow a}, I) - u_i(\sigma^t, I) + \sum_{I' \in \text{Succ}_i(I, a)} \text{succ}_i^\sigma(I'|I, a) (u_i(\sigma^t|_{(D(I) \rightarrow \sigma'), I'}) - u_i(\sigma^t, I')) \right) \quad (9)$$

$$R_{i,\text{full}}^T(I) \leq \frac{1}{T} \max_{a \in A(I)} \max_{\sigma' \in \Sigma_1} \sum_{t=1}^T \pi_{-i}^{\sigma^t}(I) (u_i(\sigma^t|_{I \rightarrow a}, I) - u_i(\sigma^t, I)) + \frac{1}{T} \max_{a \in A(I)} \max_{\sigma' \in \Sigma_1} \sum_{t=1}^T \pi_{-i}^{\sigma^t}(I) \sum_{I' \in \text{Succ}_i(I, a)} \text{succ}_i^\sigma(I'|I, a) (u_i(\sigma^t|_{(D(I) \rightarrow \sigma'), I'}) - u_i(\sigma^t, I')) \quad (10)$$

The first part of the expression on the right hand side is the immediate regret. For the second, we know that $\pi_{-i}^{\sigma^t}(I) \text{succ}_i^\sigma(I'|I, a) = \pi_{-i}^{\sigma^t}(I')$, and that $u_i(\sigma^t|_{D(I) \rightarrow \sigma'}, I') = u_i(\sigma^t|_{D(I') \rightarrow \sigma'}, I')$.

$$R_{i,\text{full}}^T(I) \leq R_{i,\text{imm}}^T(I) + \frac{1}{T} \max_{a \in A(I)} \max_{\sigma' \in \Sigma_1} \sum_{t=1}^T \sum_{I' \in \text{Succ}_i(I, a)} \pi_{-i}^{\sigma^t}(I') (u_i(\sigma^t|_{(D(I') \rightarrow \sigma'), I'}) - u_i(\sigma^t, I')) \quad (11)$$

$$R_{i,\text{full}}^T(I) \leq R_{i,\text{imm}}^T(I) + \max_{a \in A(I)} \sum_{I' \in \text{Succ}_i(I, a)} \frac{1}{T} \max_{a \in A(I)} \max_{\sigma' \in \Sigma_1} \sum_{t=1}^T \pi_{-i}^{\sigma^t}(I') (u_i(\sigma^t|_{(D(I') \rightarrow \sigma'), I'}) - u_i(\sigma^t, I')) \quad (12)$$

$$R_{i,\text{full}}^T(I) \leq R_{i,\text{imm}}^T(I) + \max_{a \in A(I)} \sum_{I' \in \text{Succ}_i(I, a)} R_{i,\text{full}}^T(I') \quad (13)$$

Because the game is perfect recall, given distinct $a, a' \in A(I)$, $\text{Succ}_i(I, a)$ and $\text{Succ}_i(I, a')$ are disjoint. If we define, $\text{Succ}_i(I) = \bigcup_{a \in A(I)} \text{Succ}_i(I, a)$, then:

$$R_{i,\text{full}}^T(I) \leq R_{i,\text{imm}}^T(I) + \sum_{I' \in \text{Succ}_i(I)} R_{i,\text{full}}^{T,+}(I') \quad (14)$$

■

We prove Theorem 3 by using a lemma that can be proven recursively:

Lemma 6 $R_{i,\text{full}}^T(I) \leq \sum_{I' \in D(I)} R_{i,\text{imm}}^{T,+}(I)$.

Proof: We prove this for a particular game recursively on the size of $D(I)$. Observe that if an information set has no successors, then Lemma 5 proves the result. We use this as a basis step. Also, observe that $D(I) = \{I\} \cup \bigcup_{I' \in \text{Succ}_i(I)} D(I')$, and that if $I' \in \text{Succ}_i(I)$, then $I \notin D(I')$, implying $|D(I')| < D(I)$. Thus, by induction we can establish that:

$$R_{i,\text{full}}^T(I) \leq R_{i,\text{imm}}^T(I) + \sum_{I' \in \text{Succ}_i(I)} \sum_{I'' \in \text{Succ}_i(I')} R_{i,\text{imm}}^{T,+}(I'') \quad (15)$$

$$\leq R_{i,\text{imm}}^{T,+}(I) + \sum_{I' \in \text{Succ}_i(I)} \sum_{I'' \in \text{Succ}_i(I')} R_{i,\text{imm}}^{T,+}(I'') \quad (16)$$

Because the game is perfect recall, for any distinct $I', I'' \in \text{Succ}_i(I)$, $D(I')$ and $D(I'')$ are disjoint. Therefore:

$$R_{i,\text{imm}}^{T,+}(I) + \sum_{I' \in \text{Succ}_i(I)} \sum_{I'' \in \text{Succ}_i(I')} R_{i,\text{imm}}^{T,+}(I'') = \sum_{I' \in D(I)} R_{i,\text{imm}}^{T,+}(I') \quad (17)$$

The result immediately follows. \blacksquare

Proof (of Theorem 3): If $P(\emptyset) = i$, then $R_{i,\text{full}}^T(\{\emptyset\}) = R_i^T$, and the theorem follows from Lemma 6. If this is not the case, then we can simply add a new information set at the beginning of the game, where player i only has one action. \blacksquare

A.2 Regret Matching

Blackwell’s approachability theorem when applied to minimizing regret is known as **regret matching**. In general, regret matching can be defined in a domain where there are a fixed set of actions A , a function $u^t : A \rightarrow \mathbf{R}$, and on each round a distribution over the actions p^t is selected.

Define the regret of not playing action $a \in A$ until time T as:

$$R^t(a) = \frac{1}{T} \sum_{t=1}^T u^t(a) - \sum_{a \in A} p^t(a) u^t(a) \quad (18)$$

and define $R^{t,+}(a) = \max(R^t(a), 0)$. To apply regret matching, one chooses the distribution:

$$p^t(a) = \begin{cases} \frac{R^{t-1,+}(a)}{\sum_{a' \in A} R^{t-1,+}(a')} & \text{if } \sum_{a' \in A} R^{t-1,+}(a') > 0 \\ \frac{1}{|A|} & \text{otherwise} \end{cases} \quad (19)$$

Theorem 7 *If $|u| = \max_{t \in \{1 \dots T\}} \max_{a, a' \in A} (u^t(a) - u^t(a'))$, the regret of the regret matching algorithm is bounded by:*

$$\max_{a \in A} R^t(a) \leq \frac{|u| \sqrt{|A|}}{\sqrt{T}} \quad (20)$$

Blackwell’s original result [?] focused on the case where an action (or vector) is chosen at random (instead of a distribution over actions) and gave a probabilistic guarantee. The result above focuses on the distributions selected, and is more applicable to a scenario where a probability is selected instead of an action.

For a proof, see [5].

A.3 Proof of Theorem 4

Observe that Equation 7 is an implementation of regret matching. Moreover, observe that for all $I \in \mathcal{I}_i$, $a \in A(I)$, $\pi_{\sigma^t}^{\sigma^t}(u_i(\sigma^t|_{I \rightarrow a}, I) - u_i(\sigma^t, I)) \leq \Delta_{u,i}$. Therefore, Theorem 7 states that the counterfactual regret of that node will be less than $\Delta_{u,i} \sqrt{|A(I)|} / \sqrt{T} \leq \Delta_{u,i} |A_i| / \sqrt{T}$. Summing over all $I \in \mathcal{I}_i$ yields the result.

A.4 Poker-Specific Implementation

We need to iterate over all of the information sets reachable given the joint bucket sequence, and compute probabilities and regrets. In order to do this swiftly, we represent the data in each information set in a “player view tree”: in other words, we never explicitly represent every state in the abstracted game: instead, we represent the information sets for each player in its own tree, with each node n being one of four types:

- **Bucket Nodes:** nodes representing where information about the cards is observed. Has a child node (an opponent or player node) for each different class that could be observed at that point.
- **Opponent Nodes:** nodes representing where the opponent takes an action. Has a child node for each action.
- **Player Nodes:** nodes representing where the current player takes an action. Contains the average regret with respect to each action, the total probability for each action until this point, and a child node for each action (either an opponent, bucket, or terminal node). There is an implicit information set associated with this node, which we will write as $I(n)$.
- **Terminal Nodes:** nodes where the game ends due to someone folding or a showdown. Given the probability of a win, loss, and tie, has sufficient information to compute an expected utility for the hand given that the node was reached.

Each player observes different pieces of information about the game, and therefore travels to a different part of its tree during the computation. Our algorithm recurses over both trees in a paired fashion. Before we begin, define $u'_i(\sigma, I) = \pi_{-i} u_i(\sigma, I)$. For each node in the trees, there will be a value $u_i(\sigma, n)$ which we use in order to compute the values $u_i(\sigma, I)$ and $u_i(\sigma, I, a)$, which is the expected value given information set I is reached and action a is taken.

Algorithm 1 WALKTREES(r_1, r_2, b, p_1, p_2)

Require: A node r_1 for an information set tree for player 1.
Require: A node r_2 for an information set tree for player 2.
Require: A joint bucket sequence b .
Require: A probability p_1 of player 1 playing to reach the node.
Require: A probability p_2 of player 2 playing to reach the node.
Ensure: The utility $u_i(\sigma, r_i)$ for player 1 and player 2.

- 1: **if** r_1 is a player node (meaning r_2 is an opponent node) **then**
- 2: Compute $\sigma_1(I(r_1))$ according to Equation 7.
- 3: **for** Each action $a \in A(I(r_1))$ **do**
- 4: Find the associated child of c_1 of r_1 and c_2 of r_2 .
- 5: Compute $u_1(\sigma, I(r_1), a)$ and $u_2(\sigma, r_2, a)$ from WALKTREES($c_1, c_2, b, p_1 \times \sigma_1(I(r_1))(a), p_2$).
- 6: **end for**
- 7: Compute $u_1(\sigma, I(r_1)) = \sum_{a \in A(I(r_1))} \sigma_1(I(r_1))(a) u_1(\sigma, I(r_1), a)$.
- 8: **for** Each action $a \in A(I(r_1))$ **do**
- 9: $R_1(I, a) = \frac{1}{T+1} (TR_1(I, a) + p_2(u_1(\sigma, I(r_1), a) - u_1(\sigma, I(r_1))))$
- 10: **end for**
- 11: Set $u_1(\sigma, r_1) = u_1(\sigma, I(r_1))$
- 12: Compute $u_2(\sigma, r_2) = \sum_{a \in A(I(r_1))} \sigma_1(I(r_1))(a) u_2(\sigma, r_2, a)$.
- 13: **else if** r_2 is a player node (meaning r_1 is an opponent node) **then**
- 14: do (opposite of above)
- 15: **else if** r_1 is a bucket node **then**
- 16: Choose the child c_1 of r_1 according to the class in b for player 1 on the appropriate round and the child c_2 of r_2 similarly.
- 17: Find $u_1(\sigma, c_1)$ and $u_2(\sigma, c_2)$ from WALKTREES(c_1, c_2, b, p_1, p_2).
- 18: Set $u_1(\sigma, r_1) = u_1(\sigma, c_1)$ and $u_2(\sigma, r_2) = u_2(\sigma, c_2)$.
- 19: **else if** r_1 is a terminal node **then**
- 20: Find $u_1(\sigma, r_1)$ and $u_2(\sigma, r_2)$, the utility of each player if this node is actually reached.
- 21: **end if**

A.5 Poker-Specific Analysis

We first analyze the non-sampling algorithm from Section 3, significantly tightening the presented regret bounds for the specific case of poker games. We then give a regret analysis for the sampling implementation described in Section 4 and used in the experimental results presented in Section 5.

A.5.1 Non-Sampling Algorithm

In Section 3, we discussed Blackwell's Approachability Theorem being applied in every information set. The disadvantage of such an algorithm is that every iteration involves a walk across the entire game tree. The

advantage of such an algorithm is that it converges really quickly in terms of iterations. In this analysis, we focus on poker.

If we can bound the difference in any two counterfactual utilities at every information set, we can achieve a bound on the overall regret, because Blackwell's Approachability Theorem gives a guarantee based upon this. In particular, after T time steps, if the bound for the counterfactual utility at an information set is $\Delta_{u,1}(I)$, and there are $|A(I)|$ actions, then the counterfactual regret is bounded by:

$$R_1^T(I) \leq \frac{\Delta_{u,1}(I)\sqrt{|A(I)|}}{\sqrt{T}} \quad (21)$$

By Theorem 3, this means the average overall regret is bounded by:

$$R_1^T \leq \sum_{I \in \mathcal{I}_1} \frac{\Delta_{u,1}(I)\sqrt{|A(I)|}}{\sqrt{T}} \quad (22)$$

First of all, define $\Delta_{u,1}$ to be the overall range of utilities in limit poker (48 small bets/hand). In particular, given $\pi_0(I)$ (the probability of chance acting to reach a node), $\Delta_{u,1}(I) \leq \pi_0(I)\Delta_{u,1}$. In limit one could be more precise, because any information set that begins with both players checking on the pre-flop has a tighter limit on the maximum won or lost, but bounding based on chance nodes is more crucial. In the next step, we leverage the structure of poker: in particular, the fact that all actions are observable. Define \mathcal{B}_1 to be the set of all betting sequences where the first player has to act: in particular, \mathcal{B}_1 can be considered a partition of the information sets \mathcal{I}_1 (such that each $B \in \mathcal{B}_1$ is a set of information sets). Note that, for all $B \in \mathcal{B}_1$:

$$\sum_{I \in B} \pi_0(I) = 1 \quad (23)$$

Moreover, observe that we can define $A(B)$ to be the set of actions available at any information set in B . Applying these concepts to the equation:

$$R_1^T \leq \sum_{B \in \mathcal{B}_1} \frac{\sqrt{|A(B)|}\Delta_{u,1}}{\sqrt{T}} \quad (24)$$

$$R_1^T \leq \frac{\Delta_{u,1}}{\sqrt{T}} \sum_{B \in \mathcal{B}_1} \sqrt{|A(B)|} \quad (25)$$

Thus, *increasing the size of the card abstraction does not affect the rate of convergence*. This is not as surprising as one might think: if one imagined n independent algorithms minimizing regret, each with a bound on their utility of $\Delta_{u,1}$, then one would expect that the theoretical bound on the average of the algorithms would closely resemble the theoretical bound on the average of one particular algorithm. This is very similar to what was leveraged in this section. However, the number of information sets does have an affect on the cost of an iteration: each game state in the abstraction must be traversed in every iteration. This is the primary motivation for WALKTREES.

A.5.2 Sampling Algorithm

In order to analyze WALKTREES, we focus on two different measures of regret:

1. \hat{R} , the regret measured by the algorithm.
2. R , the underlying regret (if all states were visited every iteration).

In this implementation, the range of counterfactual utilities can be $\Delta_{u,1}$ in almost every state. Define $C^T(I)$ to be the number of times an information set I was visited until time T : in particular, how many times the bucket sequence that makes I reachable was selected. Blackwell's Approachability Theorem yields us:

$$\hat{R}_1^T(I) \leq \frac{\Delta_{u,1}(I)\sqrt{|A(I)|}\sqrt{C^T(I)}}{T} \quad (26)$$

Observe that for any $B \in \mathcal{B}_1$ (see Section A.5.1), $\sum_{I \in B} C^T(I) = T$. Define $Y = \max_{B \in \mathcal{B}_1} |B|$, in other words the number of card partitions on the river. Then $\sum_{I \in B} \sqrt{C^T(I)} \leq \sqrt{YT}$.

$$\sum_{I \in B} \hat{R}_1^T(I) \leq \sum_{I \in B} \frac{\Delta_{u,1}|A(I)|\sqrt{C^T(I)}}{T} \quad (27)$$

$$\sum_{I \in B} \hat{R}_1^T(I) \leq \frac{\sqrt{|A(B)|}\Delta_{u,1}\sqrt{Y}}{\sqrt{T}} \hat{R}_1^T \leq \frac{\Delta_{u,1}\sqrt{Y}}{\sqrt{T}} \sum_{B \in \mathcal{B}_1} \sqrt{|A(B)|} \quad (28)$$

$$\hat{R}_1^T \leq \frac{\Delta_{u,1}\sqrt{Y}}{\sqrt{T}} |\mathcal{B}_1| \sqrt{|A_1|} \quad (29)$$

Thus, the regret bound has increased by a factor of \sqrt{Y} : however, the computation per round has decreased by a factor of nearly Y^2 , resulting in a dramatic overall gain, so long as R and \hat{R} are similar.

This last portion is tricky: since the algorithm is randomized, we cannot guarantee that every information set is reached, let alone that it has converged. Therefore, instead of proving a bound on the absolute difference of R and \hat{R} , we focus on proving a probabilistic connection.

In particular, we will focus on the similarity of the counterfactual regret ($R^T(I)$ and $\hat{R}^T(I)$) in every node. In particular, we will focus on the similarity of the counterfactual regret of a particular action at a particular time ($r_1^t(I, a)$ and $\hat{r}_1^t(I, a)$). Define $\text{Reach}^t(I)$ to be true if I is reachable given the actions of nature at time t . Formally:

$$r_1^t(I, a) = \pi_{-1}^{\sigma^t}(I) (u_1(\sigma^t|_{I \rightarrow a}, I) - u_1(\sigma^t, I)) \quad (30)$$

$$\hat{r}_1^t(I, a) = \begin{cases} \frac{r_1^t(I, a)}{\pi_0(I)} & \text{if } \text{Reach}^t(I) \\ 0 & \text{otherwise} \end{cases} \quad (31)$$

It is the case that $\mathbf{E}[r_1^t(I, a) - \hat{r}_1^t(I, a)] = 0$. These are the elementary components of $R_1^T(I)$ and $\hat{R}_1^T(I)$, because:

$$R_1^T(I) = \frac{1}{T} \max_{a \in A(I)} \sum_{t=1}^T r_1^t(I, a) \quad (32)$$

$$\hat{R}_1^T(I) = \frac{1}{T} \max_{a \in A(I)} \sum_{t=1}^T \hat{r}_1^t(I, a) \quad (33)$$

We bound the expected squared difference between $\sum_{I \in \mathcal{I}_1} R_1^T(I)$ and $\sum_{I \in \mathcal{I}_1} \hat{R}_1^T(I)$ in order to prove that they are close, because for any random variable X :

$$\Pr[|X| \geq k \sqrt{\mathbf{E}[X^2]}] \leq \frac{1}{k^2} \quad (34)$$

by Markov's Inequality.

$$\mathbf{E}[(\sum_{I \in \mathcal{I}_1} (R_1^T(I) - \hat{R}_1^T(I)))^2] \leq |\mathcal{I}_1| \sum_{I \in \mathcal{I}_1} \mathbf{E}[(R_1^T(I) - \hat{R}_1^T(I))^2] \quad (35)$$

This is because, for all $a_1 \dots a_k \in \mathbf{R}$, $(\sum_{i=1}^k a_i)^2 \leq k \sum_{i=1}^k a_i^2$. Finally:

$$(R_1^T(I) - \hat{R}_1^T(I))^2 = \left(\frac{1}{T} \max_{a \in A(I)} \sum_{t=1}^T r_1^t(I, a) - \frac{1}{T} \max_{a \in A(I)} \sum_{t=1}^T \hat{r}_1^t(I, a) \right)^2 \quad (36)$$

$$(R_1^T(I) - \hat{R}_1^T(I))^2 \leq \frac{1}{T^2} \max_{a \in A(I)} \left(\sum_{t=1}^T r_1^t(I, a) - \sum_{t=1}^T \hat{r}_1^t(I, a) \right)^2 \quad (37)$$

$$(R_1^T(I) - \hat{R}_1^T(I))^2 \leq \frac{1}{T^2} \sum_{a \in A(I)} \left(\sum_{t=1}^T r_1^t(I, a) - \sum_{t=1}^T \hat{r}_1^t(I, a) \right)^2 \quad (38)$$

$$\mathbf{E}[(R_1^T(I) - \hat{R}_1^T(I))^2] \leq \frac{1}{T^2} \sum_{a \in A(I)} \sum_{t=1}^T \mathbf{E}[(r_1^t(I, a) - \hat{r}_1^t(I, a))^2] \quad (39)$$

The final step is because if $t \neq t'$, then $\mathbf{E}[(r_1^t(I, a) - \hat{r}_1^t(I, a)) (r_1^{t'}(I, a) - \hat{r}_1^{t'}(I, a))] = 0$. Substituting back into Equation 35:

$$\mathbf{E}[(\sum_{I \in \mathcal{I}_1} (R_1^T(I) - \hat{R}_1^T(I)))^2] \leq \frac{|\mathcal{I}_1|}{T^2} \sum_{I \in \mathcal{I}_1} \sum_{a \in A(I)} \sum_{t=1}^T \mathbf{E}[(r_1^t(I, a) - \hat{r}_1^t(I, a))^2] \quad (40)$$

Recall that $\pi_{-1}^{\sigma^t}(I) = \pi_2^{\sigma^t}(I) \pi_0^{\sigma^t}(I)$. Thus, $|r_1^t(I, a)| \leq \Delta_{u,1} \pi_0^{\sigma^t}$, and $\hat{r}_1^t(I, a) \leq \Delta_{u,1}$. Also, $\Pr[\hat{r}_1^t(I, a) \neq 0] \leq \pi_0(I)$. Finally:

$$\mathbf{E}[(\hat{r}_1^t(I, a) - \hat{r}_1^t(I, a))^2 | \text{Reach}(I)] \leq 2\Delta_{u,1}^2 \quad (41)$$

$$\mathbf{E}[(\hat{r}_1^t(I, a) - \hat{r}_1^t(I, a))^2 | \neg \text{Reach}(I)] \leq 2\pi_0(I) \Delta_{u,1}^2 \quad (42)$$

$$\mathbf{E}[(\hat{r}_1^t(I, a) - \hat{r}_1^t(I, a))^2] \leq 4\pi_0(I) \Delta_{u,1}^2 \quad (43)$$

Thus, substituting back into Equation 40:

$$\mathbf{E}[(\sum_{I \in \mathcal{I}_1} (R_1^T(I) - \hat{R}_1^T(I)))^2] \leq \frac{|\mathcal{I}_1|}{T^2} \sum_{I \in \mathcal{I}_1} \sum_{a \in A(I)} \sum_{t=1}^T 4\pi_0(I) \Delta_{u,1}^2 \quad (44)$$

$$\mathbf{E}[(\sum_{I \in \mathcal{I}_1} (R_1^T(I) - \hat{R}_1^T(I)))^2] \leq \frac{4|\mathcal{I}_1| \Delta_{u,1}^2}{T} \sum_{I \in \mathcal{I}_1} |A(I)| \pi_0(I) \quad (45)$$

(46)

Again, by focusing on \mathcal{B}_i :

$$\mathbf{E}[(\sum_{I \in \mathcal{I}_1} (R_1^T(I) - \hat{R}_1^T(I)))^2] \leq \frac{4|\mathcal{I}_1| \Delta_{u,1}^2}{T} \sum_{B \in \mathcal{B}_1} \sum_{I \in B} |A(I)| \pi_0(I) \quad (47)$$

$$\mathbf{E}[(\sum_{I \in \mathcal{I}_1} (R_1^T(I) - \hat{R}_1^T(I)))^2] \leq \frac{4|\mathcal{I}_1| \Delta_{u,1}^2}{T} \sum_{B \in \mathcal{B}_1} |A(B)| \quad (48)$$

For any $p \in [0, 1]$, with probability at least $1 - p$:

$$R_1^T \leq \frac{2\sqrt{|\mathcal{I}_1| |\mathcal{B}_1| |A_1|} \Delta_{u,1}}{\sqrt{pT}} + \frac{\Delta_{u,1} \sqrt{Y}}{\sqrt{T}} |\mathcal{B}_1| \sqrt{|A_1|} \quad (49)$$