Experimental Report on Fixed-Budget Best Arm Identification

Xuedong Shang

No Institute Given

Abstract.

1 Introduction

We consider here some fixed-budget and anytime best arm identification algorithms, including Uniform Allocation [1], UCB-E [2], Successive Reject [2], fixed-budget version of UGapE [3], Sequential Halving [4], Thompson Sampling coupled with MPA, Top-Two Thompson Sampling and Top-Two Probability Sampling [5], AT-LUCB [6].

Acronyme

- EDP = Empirical Distribution of Plays
- EBA = Empirical Best Arm
- MPA = Most Played Arm
- SR = Successive Reject
- SHA = Sequential Halving
- TS = Thompson Sampling
- TTTS = Top-Two Thompson Sampling
- TTPS = Top-Two Probability Sampling
- UCB = Upper Confidence Bound
- LCB = Lower Confidence Bound

2 Comparison of Different Algorithms

Experimental Settings We use 7 problem instances proposed by [2] + one hand-made simple problem instance, all settings consider only Bernoulli distributions for the moment (remark that setting 8 below is the same setting as setting 7, but only has a larger budget):

- Setting 0: $\mu = [0.4, 0.5, 0.35, 0.3], \text{budget} = 1000$
- Setting 1: $\mu_{\mathbf{1}} = 0.5, \mu_{\mathbf{2:20}} = 0.4, \text{budget} = 2000$
- Setting 2: $\mu_{\mathbf{1}} = 0.5, \mu_{\mathbf{2:6}} = 0.42, \mu_{\mathbf{7:20}} = 0.38, \text{budget} = 2000$
- Setting 3: $\mu = [0.5, 0.3631, 0.449347, 0.48125839], \text{budget} = 2000$
- Setting 4: $\mu = [0.5, 0.42, 0.4, 0.4, 0.35, 0.35], \text{budget} = 600$
- Setting 5: $\mu_{\mathbf{1}} = 0.5, \mu_{\mathbf{i}} = \mu_{\mathbf{1}} - 0.025i, \forall i \in \{2 \ldots 15\}, \text{budget} = 4000$
- Setting 6: $\mu_{\mathbf{1}} = 0.5, \mu_{\mathbf{2}} = 0.48, \mu_{\mathbf{3:20}} = 0.37, \text{budget} = 6000$
- Setting 7: $\mu_{\mathbf{1}} = 0.5, \mu_{\mathbf{2:6}} = 0.45, \mu_{\mathbf{7:20}} = 0.43, \mu_{\mathbf{7:20}} = 0.38, \text{budget} = 6000$
- Setting 8: $\mu_{\mathbf{1}} = 0.5, \mu_{\mathbf{2:6}} = 0.45, \mu_{\mathbf{7:20}} = 0.43, \mu_{\mathbf{7:20}} = 0.38, \text{budget} = 12000$

Some Implementation Details All algorithms are implemented in Julia.

– UCB-E: UCB-E has two versions of implementations, where the adaptive version estimates online the parameter H_1. And we use EBA as recommendation strategy for UCB-E in this section.
– SR: In SR we need to pay a little attention. Indeed, we need to use *Round-Robin* when we pull arms in each elimination phase. For the moment, we also use EBA as recommendation strategy.
– UGapE: UGapE also faces the problem of unknown H_1 parameter, thus we also have an adaptive version of the algorithm. Note that UGapE has its own recommendation strategy which is not included in EBA, EDP and MPA.
– SHA: SHA also has two versions of implementations. The first one which I call *SHA with Refresh* means that at each elimination phase, we will discard the statistics of each arm we obtained from previous elimination rounds. While the second one which I call *SHA without Refresh* means that we will keep the statistics at each elimination phase. Note that we also need to use Round-Robin when we pull arms in each elimination phase as in SR. The recommendation strategy is also EBA here.
– AT-LUCB: The recommendation strategy of AT-LUCB is also a bit of different from other algorithms. Since it's a LUCB-type algorithm, we need to pull two arms (one based on UCBs and one based on LCBs), thus at each time step we will recommend two times the same arm using EBA strategy.
– TS with MPA: In TS, we draw parameter samples from the posterior instead of computing the optimal action probabilities, since the latter one requires a much larger computational effort.

Thompson Sampling with MPA In this part, all the plots are averaged on 10000 trials of experiments and we plot the trend of simple regret for each algorithm. In this part we include only Uniform Sampling, UCB-E, SR, UGapE, SHA, AT-LUCB and TS with MPA. The objective is to see how TS with MPA performs compared to state-of-the-art fixed-budget algorithms.

From Fig. 6 to 9 we can see that TS with MPA performs almost as well as AT-LUCB, and beats other algorithms (note that we don't really take into account UCB-E which seems to be always the best, since we cannot know H_1 in advance). These are actually some relatively difficulty problem instances in this report.

TODO: log-scale figures.

Top-Two Thompson Sampling and Top-Two Probability Sampling For TTTS and TTPS, we now need to compute the optimal action probabilities. Actually, in the original TTTS algorithm, we don't need to compute these probabilities since we can always use the parameter sampling from the posterior trick. However, since we are now in a best arm identification setting, and a plausible recommendation strategy for TTTS and TTPS would be recommending at each time step the arm with the largest optimal action probability. Thus in both cases, we need to compute these probabilities. And in practice, we use the Cubature package of Julia to complete this task.

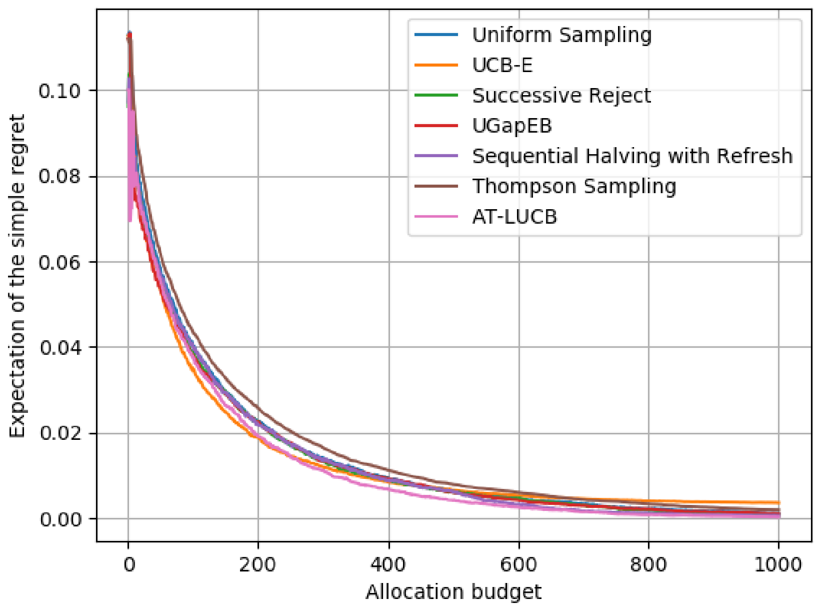


Fig. 1: Problem Setting 0

For the moment, all the plots in this part are averaged on 10000 trials. From Fig. 10 to 18 we can see that Top-Two Thompson Sampling performs almost always the best (except the cheating UCB-E).

3 Comparison of Different Recommendation Strategies

TODO

References

1. Bubeck, S., Munos, R., & Stoltz, G. (2009, October). Pure exploration in multi-armed bandits problems. In International conference on Algorithmic learning theory (pp. 23-37). Springer, Berlin, Heidelberg.
2. Audibert, J. Y., & Bubeck, S. (2010, June). Best arm identification in multi-armed bandits. In COLT-23th Conference on Learning Theory-2010 (pp. 13-p).
3. Gabillon, V., Ghavamzadeh, M., & Lazaric, A. (2012). Best arm identification: A unified approach to fixed budget and fixed confidence. In Advances in Neural Information Processing Systems (pp. 3212-3220).
4. Karnin, Z., Koren, T., & Somekh, O. (2013, February). Almost optimal exploration in multi-armed bandits. In International Conference on Machine Learning (pp. 1238-1246).

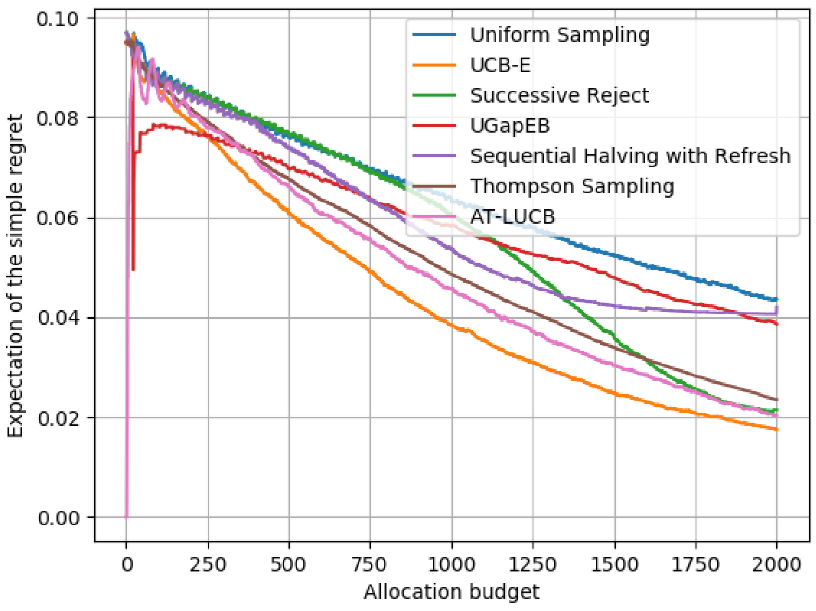


Fig. 2: Problem Setting 1

5. Russo, D. (2016, June). Simple bayesian algorithms for best arm identification. In Conference on Learning Theory (pp. 1417-1418).
6. Jun, K. S., & Nowak, R. D. (2016, June). Anytime Exploration for Multi-armed Bandits using Confidence Information. In ICML (pp. 974-982).

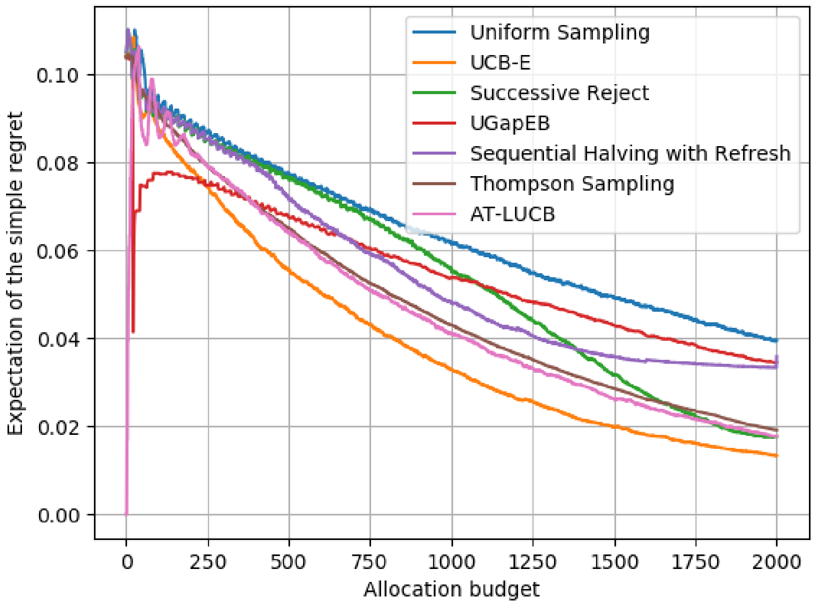


Fig. 3: Problem Setting 2

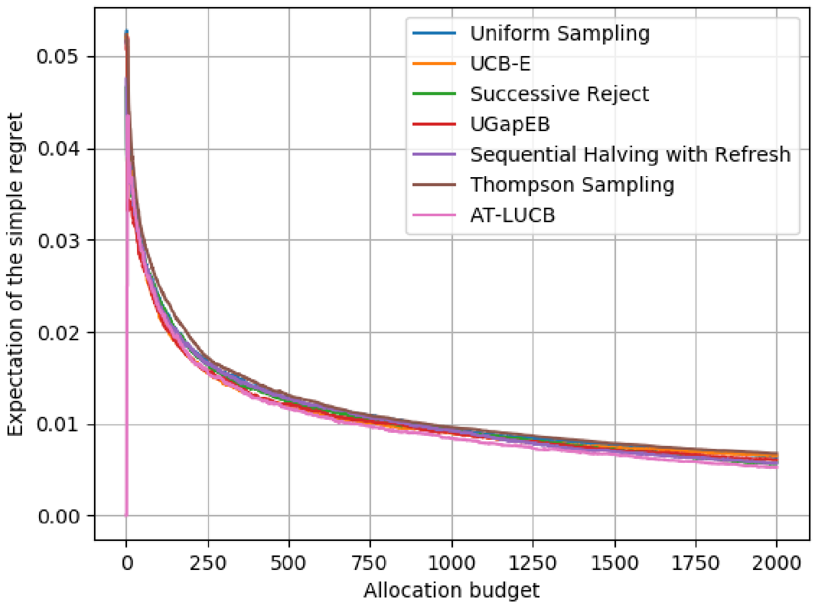


Fig. 4: Problem Setting 3

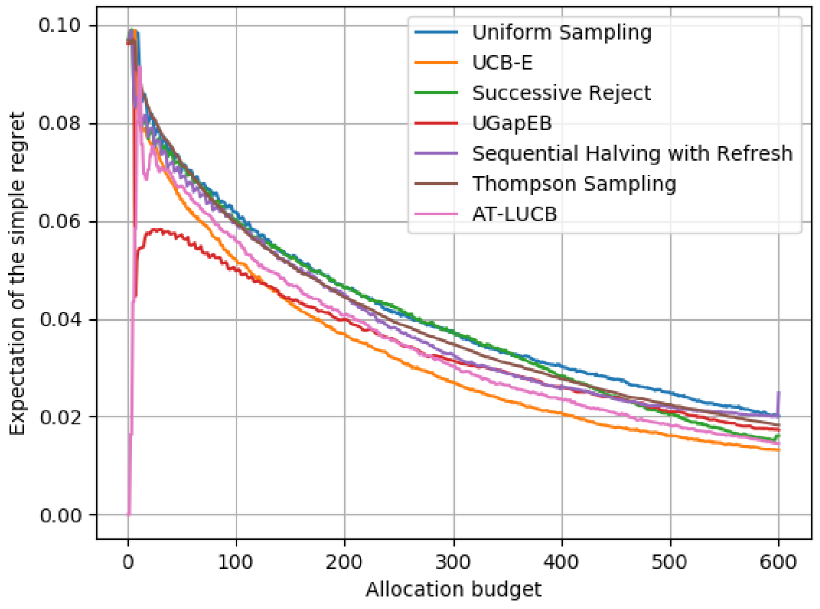


Fig. 5: Problem Setting 4

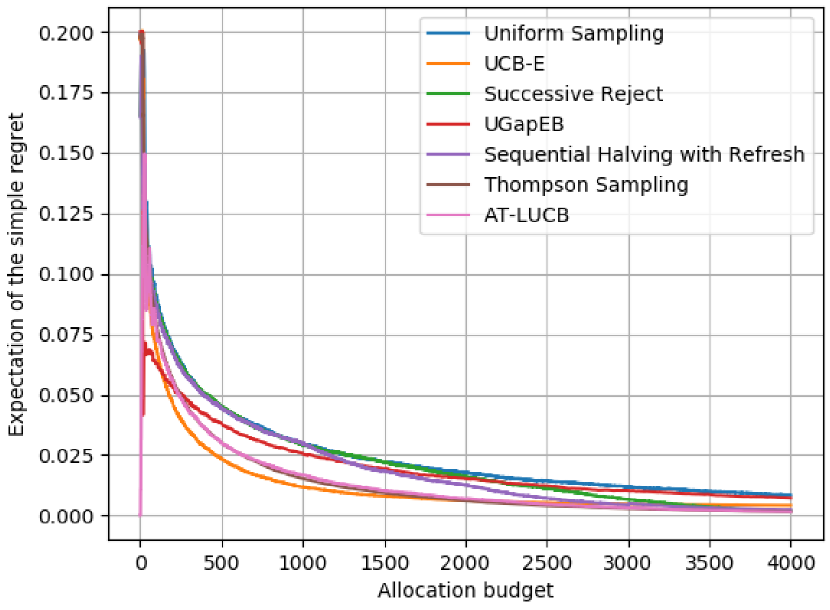


Fig. 6: Problem Setting 5

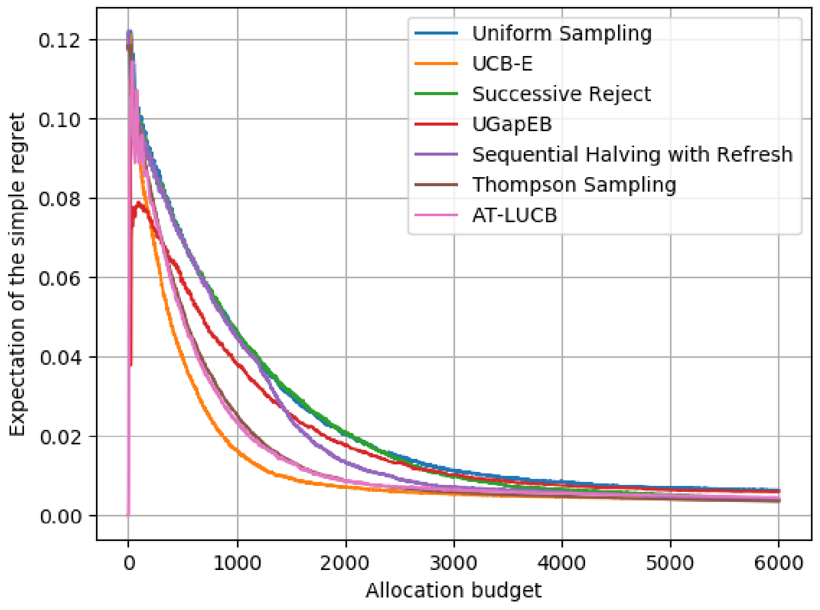


Fig. 7: Problem Setting 6

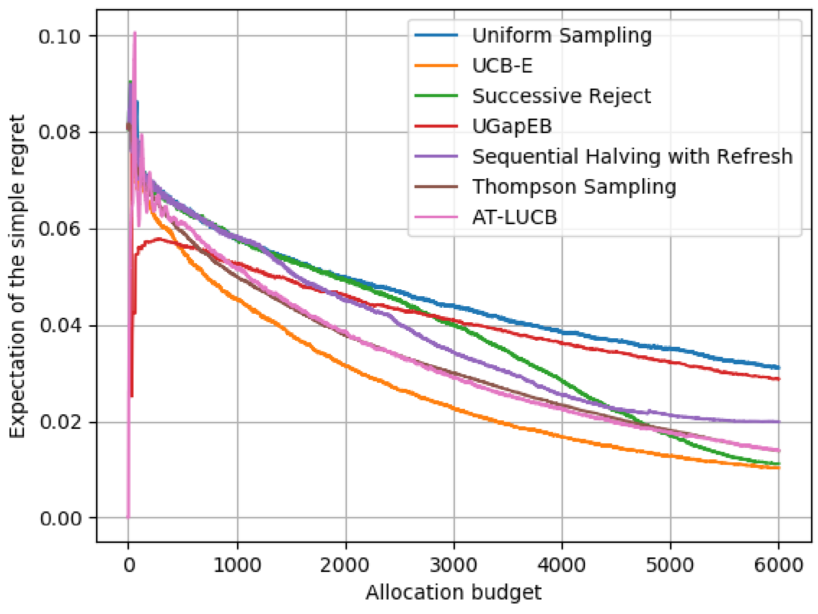


Fig. 8: Problem Setting 7

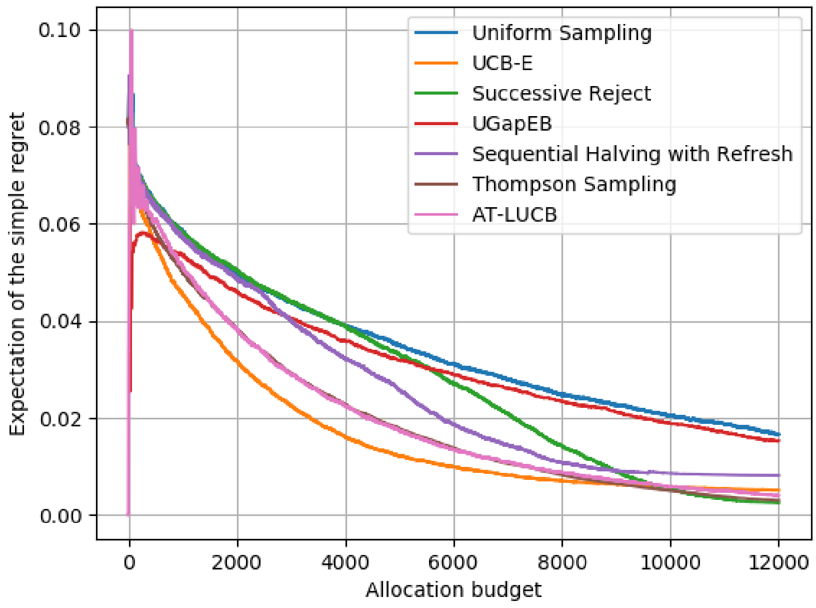
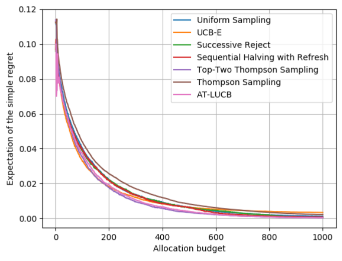
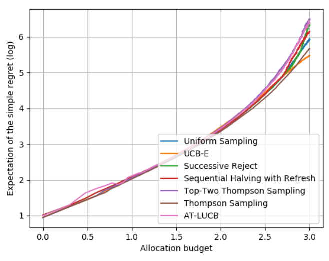


Fig. 9: Problem Setting 8

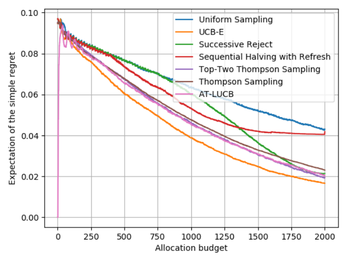


(a) linear scale

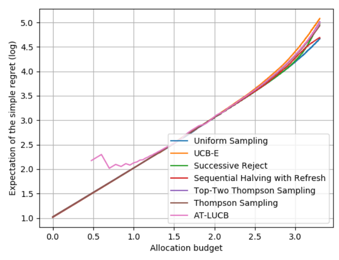


(b) log scale

Fig. 10: Problem Setting 0

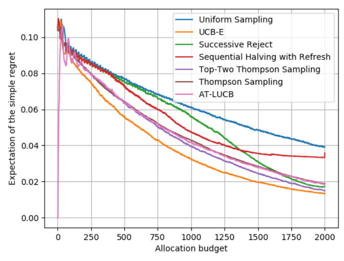


(a) linear scale

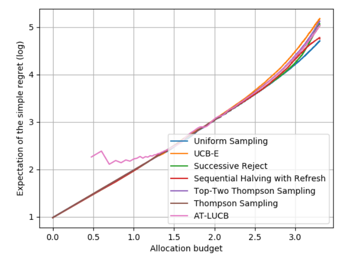


(b) log scale

Fig. 11: Problem Setting 1

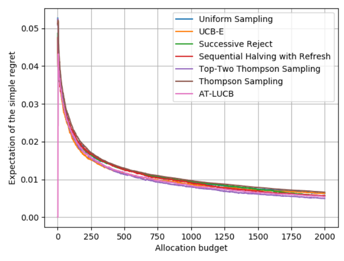


(a) linear scale

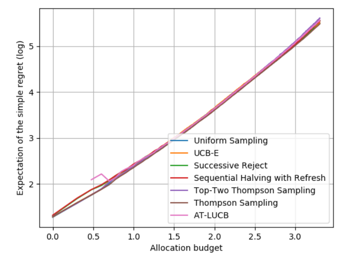


(b) log scale

Fig. 12: Problem Setting 2

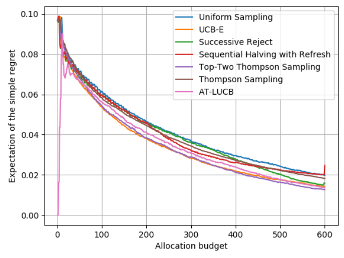


(a) linear scale

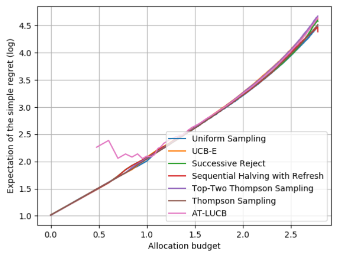


(b) log scale

Fig. 13: Problem Setting 3

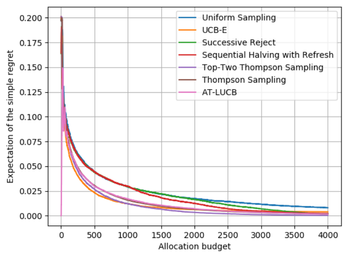


(a) linear scale

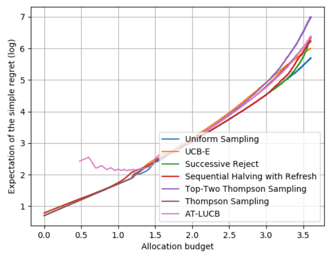


(b) log scale

Fig. 14: Problem Setting 4

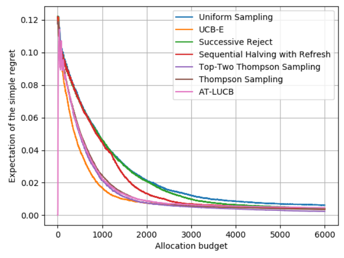


(a) linear scale

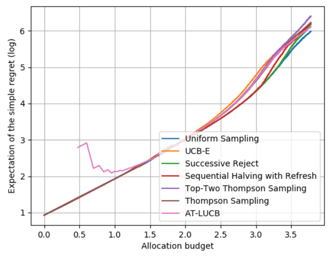


(b) log scale

Fig. 15: Problem Setting 5

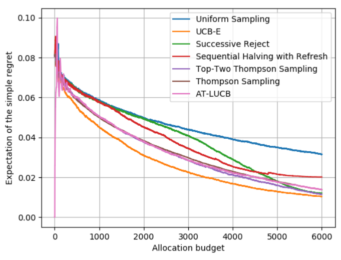


(a) linear scale

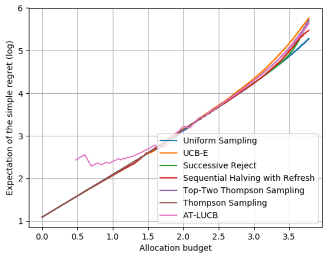


(b) log scale

Fig. 16: Problem Setting 6

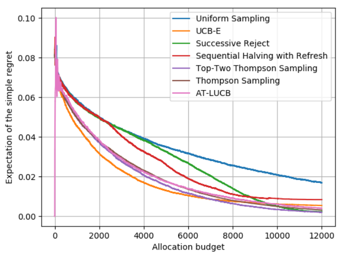


(a) linear scale

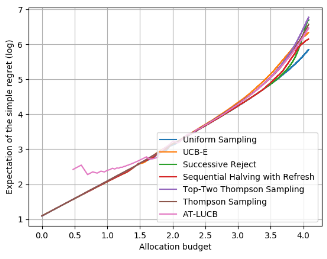


(b) log scale

Fig. 17: Problem Setting 7



(a) linear scale



(b) log scale

Fig. 18: Problem Setting 8