Regret minimization in gambler bandits

Manos Theodosis
Harvard University
etheodosis@seas.harvard.edu

Max Guo
Harvard University
mguo@college.harvard.edu

I. Introduction

Several interesting applications can be formulated as multiarmed bandit problems (Robbins, 1952; Slivkins, 2019). In multiarmed bandits, an agent is "pulling" arms (much like in a casino setting) and recording a probabilistic reward; the objective of the agent is usually to identify the best of the available arms (in terms of mean reward), usually with a minimal sample cost, or to minimize *regret*: in simple terms, it is desirable that the difference between the reward of the optimal choice and the reward of the actual choice that the agent made over the different rounds is as small as possible. However, traditional formulations of multiarmed bandits (Even-Dar et al., 2006; Jamieson and Nowak, 2014; Karnin et al., 2013) focused solely on the optimization of the objective (either minimizing regret or identifying the best arm), ignoring the potential costs that might be associated with arm pulls. Indeed, in real world scenarios, the possible actions an agent can take are usually associated with costs (Bouneffouf and Rish, 2013); it is therefore natural to analyze a framework for multiarmed bandit problems that incorporates costs. While the introduction of costs is not new as a multiarmed bandit setting (Ding et al., 2013; Zhou and Tomlin, 2018), in most practical applications there is a caveat: rewards and costs are tied and allocated from the same pool. A classical such example is *research development* departments where the costs (investment in new technologies) and rewards (company revenue) are tied and in order for further investments in research projects, a suitable budget must have been obtained.

In terms of applicability, regret minimization is closer to a real-world scenario: the process of minimizing regret is implicitly identifying the best arm, while at the same time incurring the minimum possible cost. In order to formalize regret, consider a set of possible actions arms) $\mathcal{A}$ and their mean rewards μ_a for $a \in \mathcal{A}$, we can define the regret at round T as

$$\boldsymbol{R}(T) = T \cdot \mu_{a^*} - \sum_{t=1}^{T} \mu_{a_t}, \tag{1}$$

where a^* denotes the *best* arm in the set $\mathcal{A}$ (the one with the maximum mean reward) and the sequence $\{a_t\}_{t=1}^T$ denotes the arm choices for the different rounds (which is usually denoted as the arm choice under a policy π). Regret has been analyzed for multitudes of multiarmed bandit problems (Bubeck and Cesa-Bianchi, 2012), including linear and contextual bandits (Slivkins, 2019); competing bandits (Mansour et al., 2018); adversarial bandits (Slivkins, 2019; Lattimore and Szepesvári, 2020); and dueling bandits (ZhouYue and Joachims, 2009; Yue et al., 2012). In this work, we are interested in analyzing regret for a budget constrained, possibly variable cost bandit problem (Ding et al., 2013; Zhou and Tomlin, 2018) where the costs and rewards are not separate entities, but rather the share a common budget. This setting has been recently proposed (Perotto et al., 2019) and is referred to as *surviving* or *gambler* bandits. The definition of regret in this setting is not straightforward, as agents need to account for the chance of being *ruined* (Coolidge, 1909).

The rest of this work is organized as follows: Section 2 studies in more depth the related work of constrained bandits, and specifically the approaches specializing in surviving bandits. In Section 3 the setting of surviving bandits is formally defined and our notation is introduce. Section 4 ponders upon what the proper definition of regret is

in that setting, and discusses the shortcomings of the proposed formulation of Perotto et al. (2021a). An attempt to address the surviving bandit setting is introduced in Section 5 by modifying existing algorithms (such as UCB and Thompson sampling) in order to incorporate the risk of ruin and account for the remaining budget, with an experimental validation outlined in Section 6. Finally, we conclude in Section 7.

II. Related works

Traditional multiarmed bandit settings (Robbins, 1952; Slivkins, 2019) are frequently ill-suited for real-world applications as they disregard the existence of costs that are usually associated with agents' actions (Bouneffouf and Rish, 2013). The *variable costs* setting for multiarmed bandits, where arms have non-uniform, possibly probabilistic, costs associated with them, has been moderately studied (Ding et al., 2013; Zhou and Tomlin, 2018). In this setting, an allocated budget is constraining the sampling of the arms, with each having a variable; an optimal policy needs to guarantee that the budget suffices for the desired arm pulls. The setting was first introduced by Ding et al. (2013), who acknowledged that in practice arms are usually associated with different costs and pointed out the gap in the literature. In their work, they analyzed regret bounds of UCB-like algorithms in the variable costs setting, which required significant deviations from the analyses of the no-cost settings. Most recently, Zhou and Tomlin (2018) examined the variable cost setting under a *multiple play* constraint, where K out of the possible N arms have to be pulled at every round. Zhou and Tomlin (2018) analyzed the regret in the stochastic case and also derived regret bounds for an adversarial setting by extending the `Exp3` algorithm.

Most relevant to this work is the idea of *surviving bandits*, *gambler bandits*, and the notion of *risk of ruin* in multiarmed bandit problems. The setting was recently introduced as an open problem by Perotto et al. (2019); in many realistic scenarios the set of actions available to an agent are associated with nontrivial, and possibly probabilistic, costs. However, unlike the setting introduced by Ding et al. (2013), in surviving bandits the rewards and costs of the actions are tied, which can lead to ruin. In a follow-up work by the same authors that tries to address the problem posed (Perotto et al., 2021a), the risk of ruin is experimentally examined in a simplified setting where the rewards are -1 or $+1$. This is equivalent to having *uniform*, deterministic costs of 1 and probabilistic rewards from the set $\{0, 2\}$. Surviving bandits, in which rewards and costs are a common resource, is not a new concept in computer science or mathematics; in fact, the setting is directly connected to the very rich literature on gambler's ruin (Coolidge, 1909; Harik et al., 1999), which is famously modeled as a random walk and is the basis of queueing theory. Recently, Perotto et al. (2021b) considered a variation of a random walk closest to the bandit setting, where the agent can decide if they want to stop the game at an earlier point, and claim the rewards thus far, or if they want to continue playing; the analysis introduces various strategies which incorporate the probability of being ruined into their decision making. This setting can be thought of a special case of surviving bandits where the number of arms is 1, and the main objective of the agent is again to estimate the success probability (or mean) of the arm.

III. Problem Definition

Our problem definition is based on but not identical to that of Perotto et al. (2021b). We assume that we have a set $\mathcal{A}$ of k arms that follow reward distributions $\mathcal{F} = \{f_1, \ldots, f_k\}$ where a random variable $X_i \sim f_i$ has probability mass function $P(X_i = 1) = p_i$, and $P(X_i = -1) = 1 - p_i$ for all i: that is, the arms have a *uniform* cost of 1 and a probabilistic reward in the set $\{0, 2\}$. While this setting might seem restrictive, analyzing and even *defining* regret while incorporating the risk of ruin is highly nontrivial. Agents are allocated an initial budget of b_0, and an agent's budget at time t is denoted b_t. If $b_t > 0$, at time step t the agent may choose arm a_t and receive reward $r(a_t)$; however, if the budget is 0 then the agent is ruined and is no longer able to pull arms. After a successful arm pull, the observed reward $r(a_t)$ is added to the agent's budget, i.e., $b_{t+1} = b_t + r(a_t)$. As a function of the initial budget and

the observed rewards up to a time horizon t, the current budget can be expressed as:

$$b_t = b_0 + \sum_{i=1}^{t} r(a_t). \tag{2}$$

The initial budget b_0 and the current budget b_t are integral in the analysis and the regret of surviving bandits, much like in the original (Coolidge, 1909) gambler's ruin problem. Indeed, if $b_t = 0$ for some t, we say that the agent has been ruined. Suppose that $r_\pi(a_t)$ is the reward we receive at time step t after pulling arm a_t following a policy π. Consider the event $S_{\pi,t}$

$$S_{\pi,t} = \bigcap_{s=0}^{t} \left[b_0 + \sum_{i=0}^{s} r_\pi(a_i) > 0 \right] \tag{3}$$

to be the event that the agent has not been ruined by time step t following policy π. This event requires the budget at every time step, up to t to have a *strictly* positive value. Then, let π^* be the policy that always pulls the best arm a^* at every time step up to t. If we have a set time horizon T, then we see that policy π^* maximizes the expected budget at time T (and note that the remaining budget is 0 if the agent gets ruined does not survive to time step T). One possible way to formulate regret in the surviving bandit setting is to cast it as the problem of finding the policy π_{opt} that minimizes the difference between the expected budget at time T when following the best policy π^* and the expected budget from the proposed policy:

$$\begin{aligned} \pi_{opt} &= \arg\min_{\pi} P(S_{\pi^*,T})\mathbb{E}[b_{\pi^*,T}|S_{\pi^*,T}] - P(S_{\pi,T})\mathbb{E}[b_{\pi,T}|S_{\pi,T}] \\ &= \arg\max_{\pi} P(S_{\pi,T})\mathbb{E}[b_{\pi,T}|S_{\pi,T}] \end{aligned} \tag{4}$$

However, an optical policy is not straightforwardly defined in surviving bandits, as it is not clear exactly how much "regret" an agent should incur if they are ruined. Indeed, there are a couple other metrics of success in the survival bandit problem. One could ponder the best arm identification problem, where the best arm is defined as that which maximizes the expected reward at time T (which is assumed to be 0 if the bandit does not survive until then), $\mathbb{E}[r_\pi(a_t)]$. Not that is is distinctly different from the optimization of the overall *budget* and results in a much greedier agent. A more conservative agent could also maximize the probability of survival for a finite time horizon T, i.e. $P(S_{\pi,T})$, while completely ignoring the acquired rewards (or, equivalently, try to maximize the time that the agent is active and not ruined). As a final consideration, an agent could try to maximize the probability pulling the optimal arm at time t, which is equivalent to maximizing $P_\pi(a_t = a^*)$.

IV. Thoughts on Perotto et al. (2021a)

However, it is unclear how to adapt the notion of regret to incorporate the chance of ruin. One attempt to model the regret for this setting was introduced by Perotto et al. (2021a) in the form of the *expected normalized relative regret*

$$\ell_{h,\pi} = \frac{\omega_{h,\pi}}{\omega_h^*} \cdot \sum_{i=1}^{k} \left[\frac{p^* - p_i}{p^*} \cdot \frac{\mathbb{E}[N_{i,h}]}{h} \right] + \left(\frac{\omega_h^* - \omega_{h,\pi}}{\omega_h^*} \right), \tag{5}$$

where the first term corresponds to the "classic" normalized regret formulation and the second term models the regret due to ruin. However, if we refactor a few terms we get

$$\begin{aligned}
\ell_{h,\pi} &= \frac{\omega_{h,\pi}}{\omega_h^*} \cdot \sum_{i=1}^{k} \left[\frac{p^* - p_i}{p^*} \cdot \frac{\mathbb{E}[N_{i,h}]}{h} \right] + \left(\frac{\omega_h^* - \omega_{h,\pi}}{\omega_h^*} \right) \\
&= \frac{\omega_{h,\pi}}{\omega_h^*} \cdot \left[\sum_{i=1}^{k} \frac{\mathbb{E}[N_{i,h}]}{h} - \sum_{i=1}^{k} \frac{p_i}{p^*} \cdot \frac{\mathbb{E}[N_{i,h}]}{h} \right] + \left(\frac{\omega_h^* - \omega_{h,\pi}}{\omega_h^*} \right) \\
&= \frac{\omega_{h,\pi}}{\omega_h^*} \cdot \underbrace{\left[\frac{1}{h} \sum_{i=1}^{k} \mathbb{E}[N_{i,h}] - \frac{1}{hp^*} \sum_{i=1}^{k} p_i \cdot \mathbb{E}[N_{i,h}] \right]}_{\text{familiar form of normalized regret}} + \left(\frac{\omega_h^* - \omega_{h,\pi}}{\omega_h^*} \right) \\
&= \frac{\omega_{h,\pi}}{\omega_h^*} \cdot \left[1 - \frac{1}{hp^*} \sum_{i=1}^{k} p_i \cdot \mathbb{E}[N_{i,h}] \right] + \left(\frac{\omega_h^* - \omega_{h,\pi}}{\omega_h^*} \right) \\
&= 1 - \frac{\omega_{h,\pi}}{\omega_h^*} \cdot \frac{1}{hp^*} \sum_{i=1}^{k} p_i \cdot \mathbb{E}[N_{i,h}] \\
&= \frac{1}{\omega_h^*} \cdot \frac{1}{hp^*} \left(\omega_h^* \cdot hp^* - \omega_{h,\pi} \cdot \sum_{i=1}^{k} p_i \cdot \mathbb{E}[N_{i,h}] \right).
\end{aligned} \tag{6}$$

The formulation of (6) is identical the normalized regret, with the exception of the multiplicative factors ω_h^* and $\omega_{h,\pi}$, denoting the probability of survival by time h when following the optimal arm, or a policy π, respectively. Note that this formulation is similar to that of (4); however, the latter considers the expected budget *given* that the agent is not ruined by time T. In contrast, the formulation of (6) does not account for that, and instead uses the unconditional expectation.

V. Proposed Approach

We now present two algorithms, Modified UCB and Modified Thompson Sampling, which are variants of the standard multiarmed bandit algorithms. We show experimentally that, under some metrics, they may outperform their counterparts by taking their budget limitations into consideration.

V.I. Modified UCB

Recall the UCB algorithm from Slivkins (2019), which in our setting translates to the following:

Algorithm 1 UCB algorithm

Require: Initial budget b_0

while $b_t > 0$ **do**

Pick arm a which maximizes $\mathrm{UCB}_t(a)$:

$$\mathrm{UCB}_t(a) = \hat{\mu}_t(a) + c\sqrt{\frac{\log(t)}{n_t(a)}}, \tag{7}$$

$b_t \leftarrow b_{t-1} + r(a)$

end while

Note c is some constant to be tuned. However, note that this criteria has no regard for the budget. This may lead to

undesirable behavior. For example, our agent may continuously explore even in the case that the budget is small. We propose a modified version of the UCB algorithm that takes into the budget by modifying the UCB criterion:

Algorithm 2 Modified UCB algorithm

Require: Initial budget b_0

while $b_t > 0$ **do**

Pick arm a which maximizes $\mathrm{UCB}_t^{\text{budget}}(a)$:

$$\mathrm{UCB}_t^{\text{budget}}(a) = \hat{\mu}_t(a) + \min\left(1, \frac{b_t}{b_0}\right) \cdot c\sqrt{\frac{\log(t)}{n_t(a)}} \tag{8}$$

$b_t \leftarrow b_{t-1} + r(a)$

end while

The $\min\left(1, \frac{b_t}{b_0}\right)$ term controls for the budget. We consider the budget to be large enough if it is at least the initial budget, in which case this algorithm behaves exactly according to the original UCB algorithm. However, if the budget is smaller than the original budget, our agent reduces the exploration term in order to prioritize survival. In the limit, as $b_t \to 0$, our algorithm approaches the purely greedy algorithm.

V.II. Modified Thompson Sampling

Recall the Thompson sampling algorithm from Slivkins (2019) applied to our scenario of Bernoulli rewards:

Algorithm 3 Thompson Sampling algorithm

Require: Initial budget b_0

Initialize the prior reward distribution $\mathbb{P}_a$ of each arm a as $\mathrm{Beta}(1, 1)$.

while $b_t > 0$ **do**

For each arm, draw mean reward vector μ_a^t from prior distribution $\mathbb{P}_a$.

Pick arm a with the maximum μ_a^t.

Observe reward for arm a, find posterior distribution of $\mathbb{P}_a$ and set as new prior for arm a.

$b_t \leftarrow b_{t-1} + r(a)$

end while

In our setting, due to Beta-Binomial conjugacy, the reward distribution for any arm at any given point in time will be a Beta distribution. Note that, similar to the UCB algorithm, this algorithm does not take into consideration the budget. This may lead to situations in which we have a low budget but arms with large variance are chosen by chance, which may lead to a greater chance of ruin.

This motivates our proposed modified Thompson Sampling algorithm:

Algorithm 4 Modified Thompson Sampling algorithm

Require: Initial budget b_0, hyperparameter c.

Initialize the prior reward distribution $\mathbb{P}_a$ of each arm a as $\text{Beta}(1, 1)$.

while $b_t > 0$ **do**

For each arm, draw mean reward vector μ_a^t from modified distributions $\mathbb{P}'_a$ such:

$$\mathbb{P}_a \sim \text{Beta}(\alpha, \beta) \implies \mathbb{P}'_a \sim \text{Beta}(k\alpha, k\beta), \quad k = \max\left(1, \frac{b_0}{\max(1, b_t - c)}\right)$$

Pick arm a with the maximum μ_a^t.

Observe reward for arm a, find posterior distribution of $\mathbb{P}_a$ and set as new prior.

$b_t \leftarrow b_{t-1} + r(a)$

end while

Note that the mean of $\mathbb{P}_a$ and $\mathbb{P}'_a$ are always equal. However, the variance of $\mathbb{P}_a$ is approximately k times the variance of $\mathbb{P}'_a$. If b_t is small enough, then k is larger than 1, and the variance of $\mathbb{P}'_a$ is smaller than the variance of $\mathbb{P}_a$. In other words, we are drawing from distributions with much smaller variance when we have a smaller budget, which leads to more exploitation and less exploration. This idea, if we did not modify the denominator of $\frac{b_0}{\max(1, b_t - c)}$ for empirical performance reasons, also results in a purely greedy strategy in the limit as $b_t \to 0$. When the budget is large, then $k = 1$, and we return to the normal Thompson Sampling algorithm.

VI. Current Empirical Results

We implemented Modified UCB (Algorithm 2) and Modified Thompson Sampling (Algorithm 4) and other policies (ε-greedy, normal Thompson Sampling, and normal UCB) for comparison. We then conducted simulations according to the problem definition in Section 3. We used with $k = 10$ arms, with probability parameters uniform from 0.3 to 0.7. The initial budget was 10. We let $T = 5000$ and ran 100 experiments total for each policy, and the results are shown in Figure 1.

There are a few points of interest from Figure 1. Before we do so, note that it appears the performance according to any metric is generally correlated with performance in other metrics, but there are some differences in the relative orderings between metrics. This tells us that the metrics are not equivalent to one another, and it is worth considering each metric individually.

First, note the distinct dominance of Modified Thompson Sampling over vanilla Thompson Sampling in every metric. This is especially clear in the survival rate metric, where the difference is at least 6%. This experimentally verifies that, as desired, modified Thompson Sampling successfully avoids ruin by being exploitative in low budget situations. This advantage affects each of the other metrics, which take into consideration the number of surviving agents at each point in time.

Second, note the slight superiority in the performance of Modified UCB over vanilla UCB in the survival rate metric. This, too, demonstrates the ability for Modified UCB to avoid ruin in low budget situations by limiting exploration. However, in the other three metrics the two algorithms are almost indistinguishable.

Finally, we note that all of the more sophisticated approaches of exploration lead to better survival rates than the ε-greedy algorithm. However, this may be due to the certain parameters we picked for each of the algorithms.

This final concession of our choice of parameters being a little bit arbitrary extends to other questions that naturally arise regarding our results. For example, the major advantage modified Thompson Sampling seems to have over modified UCB is less interpretable because of the hyperparameters involved in both. We did not tune the hyperparameters

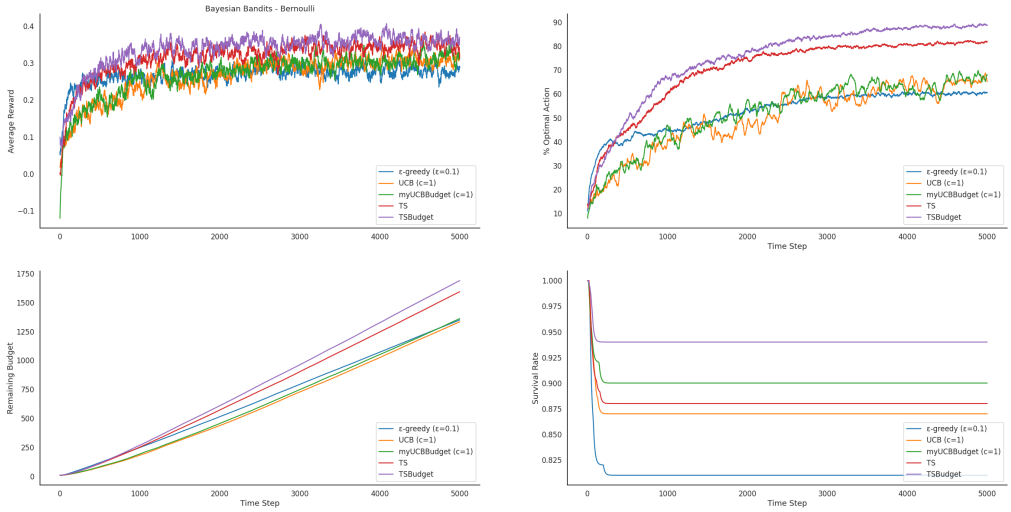

Figure 1: Performance of various algorithms under metrics listed in Section 3. In order from the top left, top right, bottom left, bottom right, the metrics estimate: $E[r_\pi, t]$, $P_\pi(a_t = a^*)$, our metric from Equation (4), and $P(S_{\pi,t})$. Each of the graphs represents an exponented moving average of the actual values for ease of visualization.

because our primary goal was to compare the modified algorithms with their original versions, but this is certainly something that can be done in the future.

VII. Conclusions

Multiarmed bandits have been applied to a variety of settings, from reinforcement learning to feature selection. However, in a lot of their most pertinent applications the existing settings are not able to account for the natural costs that are associated with agents' actions. In this work, we rethought the surviving bandit setting, where the rewards and costs of pulling an arm are sharing a common budget and there is a nontrivial risk of being ruined. We discussed a possible regret formulation, and introduced and evaluated two new algorithms, Modified UCB and Modified Thompson Sampling, that attempt to improve upon existing algorithms which do not explicitly care about budget. We presented preliminary experimental results that validated the effectiveness of our algorithms in avoiding ruin when compared to their standard counterparts. More careful experimentation in the future that could allow for more in-depth and useful comparisons between different algorithms may lead to more insights. Additionally, the formal analysis of the setting when the number of arms is greater than one, as well as an optimal formulation of regret in the setting are still open problems and could also be fruitful avenues for further work.

References

Herbert Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

Aleksandrs Slivkins. Introduction to multi-armed bandits. *Foundations and Trends in Machine Learning*, 12(1-2):1–286, 2019.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7:1079–1105, 2006.

Kevin Jamieson and Robert Nowak. Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting. In *Conference on Information Sciences and Systems*, 2014.

Zohar Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In *International Conference on Machine Learning*, 2013.

Djallel Bouneffouf and Irina Rish. A survey on practical applications of multi-armed and contextual bandits. In *arXiv*, 2013.

Wenkui Ding, Tao Qin, Xu-Dong Zhang, and Tie-Yan Liu. Multi-armed bandit with budget contraint and variable costs. In *AAAI Conference on Artificial Intelligence*, 2013.

Datong Zhou and Claire Tomlin. Budget-constrained multi-armed bandits with multiple plays. In *AAAI Conference on Artificial Intelligence*, 2018.

Sébastian Bubeck and Nicolò Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1):1–122, 2012.

Yishay Mansour, Aleksandrs Slivkins, and Zhiwei Steevn Wu. Competing bandits: Learning under competition. In *Innovations in Theoretical Computer Science*, 2018.

Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020.

Yisong ZhouYue and Thorsten Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. In *International Conference on Machine Learning*, 2009.

Yisong Yue, Josef Broder, Robert Kleinberd, and Thorsten Joachims. The k-armed dueling bandits problem. *Journal of Computer and System Sciences*, 78(5):1538–1556, 2012.

Filipo Studzinski Perotto, Mathieu Bourgais, Laurent Vercouter, and Bruno Castro da Silva. Open problem: Risk of ruin in mutliarmed bandits. In *International Conference on Learning Theory*, 2019.

Julian Lowell Coolidge. The gambler's ruin. *Annals of Mathematics*, 10(4):181–192, 1909.

Filipo Studzinski Perotto, Sattar Vakili, Pratik Gajane, Faghan Yaser, and Mathieu Bourgais. Gambler bandits and the regret of being ruined. In *International Conference on Autonomous Agents and Multiagent Systems*, 2021a.

George Harik, Erick Cantú-Paz, David Goldberg, and Brad Miller. The gambler's ruin problem, genetic algorithms, and the sizing of populations. *Evolutionary Computation*, 7(3):231–253, 1999.

Filipo Studzinski Perotto, Imen Trabelshi, Stéphanie Combettes, Valérie Camps, Elsy Kaddoum, and Nicolas Verstaevel. Deciding when to quit the gambler's ruin game with unknown probabilities. arXiv, 2021b.