Multivariate Normal Distribution Based Multi-Armed Bandits Pareto Algorithm

Saba Q. Yahyaa, Madalina M. Drugan and Bernard Manderick

Vrije Universiteit Brussel, Department of Computer Science,
Pleinlaan 2, 1050 Brussels, Belgium
{syahyaa,mdurgan,bmanderi}@vub.ac.be

Abstract. In the stochastic multivariate multi-armed bandit, arms generate a vector of stochastic normal rewards, one per objective, instead of a single scalar reward. As a result, there is not only one optimal arm, but there is a set of optimal arms (Pareto front) using Pareto dominance relation. The goal of an agent is to trade-off between exploration and exploitation. Exploration means finding the Pareto front and exploitation means selecting fairly or evenly the optimal arms. We propose annealing-Pareto algorithm that trades-off between exploration and exploitation by using a decaying parameter ϵ_t in combination with Pareto dominance relation. We compare experimentally Pareto-KG, Pareto-UCB1 and annealing-Pareto on multi-objective normal distributions and we conclude that the annealing-Pareto is the best performing algorithm.

Keywords: Multi armed bandit problem, multi objective optimization, annealing algorithm, exploration/exploitation.

1 Introduction

The Multi-Objective Multi-Armed Bandit (MOMAB) problem is a sequential stochastic learning problem. At each time step t, an agent pulls one arm i from an available arm set A and receives a reward vector $\boldsymbol{r}_i$ of the arm i with D variates (or objectives) as feedback signal. The reward vector is drawn from a normal probability distribution vector $N(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2)$, where $\boldsymbol{\mu}_i$ is the true mean vector and $\boldsymbol{\sigma}_i^2$ is the covariance matrix parameters of the arm i. The reward vector $\boldsymbol{r}_i$ that the agent receives from the arm i is independent from all other arms and independent from the past reward vectors of the selected arm i. Moreover, the mean vector of the arm i has *independent* D distributions, i.e. $\boldsymbol{\sigma}^2$ is a *diagonal covariance matrix*. We assume that the true mean vector and covariance matrix of each arm i are unknown parameters to the agent. Thus, by drawing each arm i, the agent maintains estimations of the true mean vector and the diagonal covariance matrix (or the variance vector) which are known as $\hat{\boldsymbol{\mu}}_i$ and $\hat{\boldsymbol{\sigma}}_i^2$, respectively.

The MOMAB problem has a set of Pareto optimal arms (Pareto front) A^*, that are incomparable, i.e. can not be classified using a designed partial order relations. The agent has not to only find the optimal arms (exploring), to minimize the total Pareto loss of not pulling the optimal arms, but also has to play

them fairly (exploiting), to minimize the total unfairness loss. This problem is known as the *trade-off between exploration and exploitation in the multi-objective optimization* [1]. At each time step t, the Pareto loss (or Pareto regret) is the distance between the set mean of Pareto optimal arms and the mean of the selected arm. While, the unfairness loss (or unfairness regret) is the variance in selecting the optimal arms [2]. Thus, the total Pareto regret and the total unfairness regrets are the cumulative summation of the Pareto and unfairness regret over t time steps, respectively. Since, the total unfairness regret grows exponentially on the number of time steps and does not take into account the total number of selecting optimal arms, we propose to compute the unfairness regret using the *entropy measure* [3]. The entropy unfairness regret is a measure of disarray (or disorder) on selecting the optimal arms in the Pareto front A^*.

The Pareto front A^* can be found for example, by using Pareto dominance relation (or Pareto partial order relation) which finds the Pareto front A^* by optimizing directly the Multi-Objective (MO) space [4]. To solve the trade-off between exploration and exploitation problem directly in the MO space, [2] used Upper Confidence Bound (UCB1) [5] policy and [6] used Knowledge Gradient (KG) [7] policy in the MOMAB problem. Both UCB1 and KG policies trade-off between exploration and exploitation by adding an exploration term (or bound) to the estimated mean vector $\hat{\boldsymbol{\mu}}_i$ for each arm i in each objective (or dimension) $d, d \in D$ and select the optimal arms by using Pareto dominance relation. However, the exploration bound of UCB1 for arm i requires only knowledge about that arm, while in case of KG it also requires knowledge about the other arms.

In this paper, we propose annealing-Pareto algorithm that detects the optimal arms in the multi-objective space. The annealing-Pareto controls the trade-off between exploration and exploitation by using a decaying parameter ϵ_t, $\epsilon_t \in (0, 1)$ in combination with the Pareto dominance relation. The decaying parameter ϵ_t has a high value at the beginning of time step t to explore all the available arms and increase the confidence in the estimated means, however, as the time step t increases, the ϵ_t parameter decreases to exploit the arms that have maximum estimated mean. To keep track on all the optimal arms in the Pareto front A^*, at each time step t, the annealing-Pareto uses Pareto dominance relation.

The rest of the paper is organized as follows: In Section 2 we introduce the multivariate normal multi-armed bandit problem. In Section 3 we present the MOMAB algorithms for normal multivariate distributions. In Section 4 we present the performance measure in the MOMAB problem. In Section 5 we introduce the annealing-Pareto algorithm in normal distribution. In Section 6, we describe the experiments set up followed by experimental results. Finally, we conclude and discuss future work.

2 Multi Objective Normal Distributions Multi Armed Bandits Problem

Let us consider the MOMABs problems with $|A| \geq 2$ arms and with *independent* D objectives per arm. At each time step t, the agent selects one arm i

and receives a reward vector $\boldsymbol{r}_i$. The reward vector $\boldsymbol{r}_i$ is drawn from a corresponding normal probability distribution $N(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2)$ with unknown mean parameter vector $\boldsymbol{\mu}_i$, $\boldsymbol{\mu}_i = [\mu_i^1, \cdots, \mu_i^D]^T$ and unknown variance parameter vector $\boldsymbol{\sigma}_i$, $\boldsymbol{\sigma}_i = [\sigma_i^1, \cdots, \sigma_i^D]^T$, where T is the transpose. Thus, by drawing each arm i, the agent maintains estimate of the mean parameter vector $\hat{\boldsymbol{\mu}}_i$ and the variance $\hat{\boldsymbol{\sigma}}_i^2$ parameter vector, and computes the number of times N_i arm i is drawn. The agent updates the estimated mean $\hat{\mu}_i^d$, the estimated variance $\hat{\sigma}_i^{2,d}$ of the selected arm i in each dimension $d, d \in D$ and the number of times N_{i+1} arm i has been selected as follows [8]:

$$N_{i+1} = N_i + 1, \quad \hat{\mu}_{i+1}^d = (1 - \frac{1}{N_{i+1}})\, \hat{\mu}_i^d + \frac{1}{N_{i+1}} r_{t+1}^d \tag{1}$$

$$\hat{\sigma}_{i+1}^{2,d} = \frac{N_{i+1} - 2}{N_{i+1} - 1}\, \hat{\sigma}_i^{2,d} + \frac{1}{N_{i+1}} (r_{t+1}^d - \hat{\mu}_i^d)^2 \tag{2}$$

where $\hat{\mu}_{i+1}^d$ is the updated estimated mean, and $\hat{\sigma}_{i+1}^{2,d}$ is the updated estimated variance of the arm i in the dimension d and r_{t+1}^d is the observed reward of the arm i in the dimension d.

When the objectives are conflicting with one another then the mean component μ_i^d of arm i corresponding with objective d, $d \in D$, can be better than the component μ_j^d of another arm j but worse if we compare the components for another objective d': $\mu_i^d > \mu_j^d$ but $\mu_i^{d'} < \mu_j^{d'}$ for objectives d and d', respectively. The agent has a set of optimal arms (Pareto front) A^* which can be found by the Pareto dominance relation (or Pareto partial order relation).

The Pareto dominance relation finds the Pareto front A^* directly in the multi-objective MO space [4]. It uses the following relations between the mean vectors of two arms. We use i and j to refer to the mean vector (estimated mean vector or true mean vector) of arms i and j, respectively:

Arm i dominates or is better than j, $i \succ j$, if there exists at least one objective d for which $i^d \succ j^d$ and for all other objectives d' we have $i^{d'} \succeq j^{d'}$. Arm i is incomparable with j, $i \parallel j$, if and only if there exists at least one objective d for which $i^d \succ j^d$ and there exists another objective d' for which $i^{d'} \prec j^{d'}$. Arm i is not dominated by j, $j \not\succ i$, if and only if there exists at least one objective d for which $j^d \prec i^d$. This means that either $i \succ j$ or $i \parallel j$.

Using the above relations, Pareto front $A^*, A^* \subset A$ be the set of arms that are not dominated by all other arms. Moreover, the optimal arms in A^* are incomparable with each other.

3 Multi Objective Multi Armed Bandits Algorithms in Normal Distribution

Pareto-UCB1 [2] and Pareto-KG [6] trade-off between exploration and exploitation by combination one-objective, Multi-Armed Bandits (MAB) algorithms (or policies) with Pareto dominance relation.

3.1 Pareto-UCB1 in Normal distribution

Pareto-UCB1 is the extension of the UCB1 policy [5] to the MOMABs. Pareto-UCB1 plays initially each arm i once. At each time step t, it estimates the mean vector of each of the multivariate arms i, i.e. $\hat{\boldsymbol{\mu}}_i = [\hat{\mu}_i^1, \cdots, \hat{\mu}_i^D]^T$ and adds to each dimension d an upper confidence bound which represents the *exploration bound* ExpB_i^d, $\text{ExpB}_i^d = \sqrt{(2\ln(t\sqrt[4]{D|A^*|}))/N_i}$ in the dimension d to trade-off between exploration and exploitation, where D is the number of objectives, $|A^*|$ is the number of optimal arms, and N_i is the number of times arm i has been selected. Pareto-UCB1 uses a Pareto dominance relation, Section 2 to find the Pareto-UCB1 optimal arm set A^*_{UCB1}. Thus, for all the non-optimal arms $k \notin A^*_{UCB1}$ there exists a Pareto optimal arm $j \in A^*_{UCB1}$ that is not dominated by the arms k, i.e. $\hat{\boldsymbol{\mu}}_k + \mathbf{ExpB}_k \nprec \hat{\boldsymbol{\mu}}_j + \mathbf{ExpB}_j$, where $\mathbf{ExpB}_j$, $\mathbf{ExpB}_j = [\text{ExpB}_j^1, \cdots, \text{ExpB}_j^D]$ is the exploration bound vector of the arm j. Pareto-UCB1 selects uniformly randomly one of the arms in the set A^*_{UCB1}. The idea is to select most of the times one of the optimal arm in the Pareto front, $i \in A^*$. An arm $j \notin A^*$ that is closer to the Pareto front according to metric measure is more selected than the arm $k \notin A^*$ that is far from A^*. After pulling the chosen arm i, Pareto-UCB1, updates the estimated mean $\hat{\boldsymbol{\mu}}_i$ vector, the number of times arm i is chosen N_i and computes the Pareto and the unfairness regrets.

3.2 Pareto-KG in Normal distribution

Pareto-KG is the extension of the KG policy [7] to the MOMABs. Pareto-KG plays each arm initial *Steps*. At each time step t, Pareto-KG calculates an exploration bound $\mathbf{ExpB}_i$, $\mathbf{ExpB}_i = [\text{ExpB}_i^1, \cdots, \text{ExpB}_i^D]^T$ for each arm i. The exploration bound of arm i depends on the estimated mean of all arms and on the estimated standard deviation of the arm i. The exploration bound of arm i for dimension d (ExpB_i^d) is calculated as follows:

$$\text{ExpB}_i^d = (L-t) * |A| D * v_i^d, \quad v_i^d = \hat{\bar{\sigma}}_i^d \; x \left(\begin{array}{c} \hat{\mu}_i^d - \max\limits_{j \neq i,\, j \in A} \hat{\mu}_j^d \\ -\left| \dfrac{\hat{\mu}_i^d - \max\limits_{j \neq i,\, j \in A} \hat{\mu}_j^d}{\hat{\bar{\sigma}}_i^d} \right| \end{array} \right), \quad \forall_{d \in D} \quad (3)$$

where v_i^d is the index of an arm i for dimension d, L is the horizon of experiment which is the total number of time steps, $|A|$ is the total number of arms, and $\hat{\bar{\sigma}}_i^d$, $\hat{\bar{\sigma}}_i^d = \hat{\sigma}_i^d / \sqrt{N_i}$ is the root mean square error of an arm i for dimension d. After computing the exploration bound for each arm, Pareto-KG sums the exploration bound of arm a with the corresponding estimated mean. Thus, Pareto-KG selects the optimal arms j that are not dominated by all other arms $k, k \in |A|$ using Pareto dominance relations, $\hat{\boldsymbol{\mu}_k} + \mathbf{ExpB}_k \nprec \hat{\boldsymbol{\mu}}_j + \mathbf{ExpB}_j$, Section 2 Pareto-KG chooses uniformly randomly one of the optimal arms in A^*_{KG}, where A^*_{KG} is the Pareto-KG optimal arm set. After pulling the chosen arm i, Pareto-KG, updates the estimated mean $\hat{\boldsymbol{\mu}}_i$, and the estimated variance $\hat{\boldsymbol{\sigma}}_i^2$ vectors, the number of times arm i is chosen N_i and computes the Pareto and the unfairness regrets.

Pareto-UCB1 and Pareto-KG control the trade-off between exploration and exploitation by adding an exploration bound ExpB_i^d to the estimated mean μ_i^d of

each arm i in each objective d. The added exploration bound ExpB_i^d for the arm i in the objective d by Pareto-KG depends on the estimated mean of all available arms in the objective d and on the root mean square error $\hat{\sigma}_i^d$ of the arm i, i.e. each objective has different exploration bound. While, the added exploration bound ExpB_i^d for the arm i in the dimension d by Pareto-UCB1 depends only on the arm i, i.e. each objective has the same exploration bound.

4 Performance Measure

In the MOMAB, the agent has not only to find the Pareto front A^* (or exploring the optimal arms), but also has to play them fairly (or exploiting) the optimal arms). As a result, there are two regret measures.

Pareto regret measure (R_{Pareto}) [2] measures the distance between a mean vector of an arm i that is pulled at time step t and the Pareto front A^*. Pareto regret R_{Pareto} is calculated by finding firstly the virtual distance dis^*. The virtual distance dis^* is defined as the minimum distance that is added to the mean vector of the pulled arm $\boldsymbol{\mu}_t$ at time step t in each dimension to create a virtual mean vector $\boldsymbol{\mu}_t^*$, $\boldsymbol{\mu}_t^* = \boldsymbol{\mu}_t + \boldsymbol{\varepsilon}^*$ that is incomparable with all the arms in Pareto set A^*, i.e. $\boldsymbol{\mu}_t^* || \boldsymbol{\mu}_i \; \forall_{i \in A^*}$. Where $\boldsymbol{\varepsilon}^*$ is a vector, $\boldsymbol{\varepsilon}^* = [dis^{*,1}, \cdots, dis^{*,D}]^T$. Then, the Pareto regret R_{Pareto}, $R_{Pareto} = dis(\boldsymbol{\mu}_t, \boldsymbol{\mu}_t^*) = dis(\boldsymbol{\varepsilon}^*, \mathbf{0})$ is the distance between the mean vector of the virtual arm $\boldsymbol{\mu}_t^*$ and the mean vector of the pulled arm $\boldsymbol{\mu}_t$ at time step t, where dis, $dis(\boldsymbol{\mu}_t, \boldsymbol{\mu}_t^*) = (\sum_{d=1}^{D} (\mu_t^{*,d} - \mu_t^d)^2)^{(1/2)}$ is the Euclidean distance. Thus, the regret of the Pareto front is 0 for optimal arms, i.e. the mean of the optimal arm coincides itself.

The unfairness regret metric is the *Shannon's entropy measure* [3] which is a measure of disorder (or disarray) on the Pareto front A^*. The higher the entropy, the higher the disorder. At time step t, the Shannon regret is $R_{SE}(t)$, $R_{SE}(t) = -\frac{1}{N_{|A^*|}(t)} \sum_{i^* \in A^*} p_{i^*}(t) \ln(p_{i^*}(t))$, where $p_{i^*}(t)$, $p_{i^*}(t) = {}^{N_{i^*}(t)}/_{N(t)}$ is the probability of selecting an optimal arm i^* at time step t, where $N_{i^*}(t)$ is the number of times the optimal arm i^* has been selected and $N(t)$ is the number of times all arms $i = 1, \cdots, A$ have been selected at time step t, and $N_{|A^*|}(t)$ is the number of times the optimal arms, $i^* = 1, \cdots, |A^*|$ have been selected at time step t.

5 The Annealing-Pareto Algorithm

Annealing-Pareto algorithm has a specific mechanism to control the trade-off between exploration and exploitation. It uses an exponential decay ϵ_t, $\epsilon_t = {}^{\epsilon_{decay}^t}/_{(|A|D)}$, where ϵ_{decay} is the decay parameter and Pareto dominance relation. At the beginning of time step t, ϵ_t has a high value to explore all the available arms. As the time step t is increased, ϵ_t has a low value to exploit only the optimal arms. To keep track on all the optimal arms in the Pareto front A^*, the annealing-Pareto uses Pareto dominance relation. The decay parameter ϵ_{decay}, $\epsilon_{decay} \in (0, 1)$, when $\epsilon_{decay} = 0$ means the annealing-Pareto is a fully Pareto dominance relation and when $\epsilon_{decay} = 1$ means the annealing-Pareto uses a fixed exponential decay. The pseudocode of the annealing-Pareto is given in Algorithm 1.

As initialization step, Algorithm 1 plays each arm i once to estimate the corresponding mean vector $\hat{\boldsymbol{\mu}}_i$ and the ϵ-Pareto optimal arm set A^*_ϵ contains all the arms in the arm set A. At each time step t, Algorithm 1 trades-off between exploration and exploitation by using the decay parameter ϵ_{decay} in the exponential decay ϵ_t (step: 4). In each objective d, $d \in D$, the Algorithm 1 detects the optimal arm in that objective $i^{*,d}$, $i^{*,d} = \text{argmax}_{i=1,\cdots,A}\, \hat{\mu}_i^d$, where $\hat{\mu}_i^d$ is the estimated mean for arm i in the dimension d (step: 7). Algorithm 1 selects all the arms in the objective d that have estimated mean between $[\hat{\mu}^{*,d} - \epsilon_t, \hat{\mu}^{*,d}]$ and include them in the corresponding selected arm set S^d (steps: 8-12), where $\hat{\mu}^{*,d}$, $\hat{\mu}^{*,d} = \max_{i \in A} \hat{\mu}_i^d$ is the estimated mean of the optimal arm $i^{*,d}$ in the objective d. Algorithm 1 constructs the total selected arm set $S(t)$ at time step t by reunion of the selected arm set (step: 14). To keep track on the Pareto front A^*, the Algorithm 1 uses Pareto dominance relation (step: 17) on the arms j that are elements in the previous ϵ-Pareto optimal arm set $A^*_\epsilon(t-1)$ and are not element in the total selected arm set $S(t)$. If the arm j is not dominated by all other arms, then this arm will be added to the total selected arm set $S(t)$ (step: 18). Algorithm 1 updates its ϵ-Pareto optimal arm set $A^*_\epsilon(t)$ to be the total selected arm set $S(t)$ (step: 21). It pulls uniformly at random one of the arms i^* that is an element in the ϵ-Pareto optimal arm set $A^*_\epsilon(t)$ (step: 22), observes the corresponding reward vector $\boldsymbol{r}_{i^*}$ and updates its estimated mean vector $\hat{\boldsymbol{\mu}}_{i^*}$ and the number of times N_{i^*} arm i^* is selected (step: 23). Then, it calculates the Pareto and unfairness regrets. This procedure is repeated until the end of playing L time steps which is the horizon of an experiment.

In Fig. (1), the dynamic of the algorithm is illustrated on 2-objective 5-armed bandit. The optimal arms a_1^*, a_2^*, and a_3^* have the means μ_1^*, μ_2^* and μ_3^*, respectively. The non-optimal arms a_4, and a_5 have the means μ_4 and μ_5, respectively. At the beginning of time step, $t = 1$ the total selected arm set $S(t)$ almost contains all the arms (optimal and non-optimal arms), and the ϵ-Pareto optimal arm set A^*_ϵ contains all the arms as shown in subfigure a. As the time step increases, $S(t)$ contains some of the optimal arms, i.e. a_2^* as shown in subfigure b and c, therefore, to maintain all the Pareto front, the algorithm constructs its updated ϵ-Pareto optimal arm set $A^*_\epsilon(t)$ to be the set that contains the non dominated arms (a_1^* and a_3^*) in the previous $A^*_\epsilon(t-1)$ and the arms in the set $S(t)$.

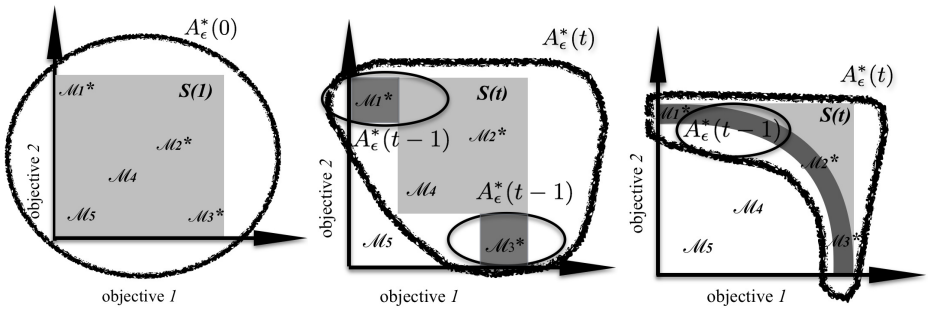
6 Experiments

In this section, we experimentally compare Pareto-UCB1, Pareto-KG and annealing-Pareto. The performance measures are: 1) the cumulative average regret at each time step which are the average of M experiments. 2) the cumulative average unfairness at each time step which are the average of M experiments.

The number of experiments M and the horizon of each experiment L are 1000. The rewards of each arm i in each objective d, $d \in D$ are drawn from normal distribution $N(\boldsymbol{\mu}_i, \boldsymbol{\sigma}^2_{i,r})$ where $\boldsymbol{\mu}_i = [\mu_i^1, \cdots, \mu_i^D]^T$ is the unknown true mean and $\boldsymbol{\sigma}^2_{i,r} = [\sigma^{2,1}_{i,r}, \cdots, \sigma^{2,D}_{i,r}]^T$ is the true unknown variance of the reward. The standard deviation σ_r^d for arms in each objective is set to $0.01, 0.1$ or 1. For

Algorithm 1 (Annealing-Pareto in Normal Distribution)

Input: Horizon of an experiment L; time step t; number of arms $|A|$; number of objectives $|D|$; reward distribution $r \sim N(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$; selected arm set $S^d(t) = \{\ \}\ \forall_d$; decay parameter $\epsilon_{decay} \in (0, 1)$.
Intialize: play each arm i initial steps to estimate its mean vector $\hat{\boldsymbol{\mu}}_i = [\hat{\mu}_i^1, \cdots, \hat{\mu}_i^D]^T$; initial ϵ-Pareto front set $A_\epsilon^*(0) = A$.
For time step $t = 1, \cdots, L$
  Set the decay parameter $\epsilon_t = \epsilon_{decay}^t / (|A||D|)$
  For objective $d = 1, \cdots, D$
    $S^d(t) = \{\phi\}$
    $\hat{\mu}^{*,d} = \max_{1 \le i \le A} \hat{\mu}_i^d$
    For arm $i = 1, \cdots, A$
      If $\hat{\mu}_i^d \in [\hat{\mu}^{*,d} - \epsilon_t, \hat{\mu}^{*,d}]$
        $S^d(t) \leftarrow \{S^d(t), i\}$
      End If
    End For
  End For
  $S(t) \leftarrow S^1(t) \cup S^2(t) \cup \cdots \cup S^D(t)$
  $S_{difference} \leftarrow A_\epsilon^*(t-1) - S(t)$
  For arm $j \in S_{difference}$ do
    If $\hat{\boldsymbol{\mu}}_k \not\succ \hat{\boldsymbol{\mu}}_j, \forall_k \in A$
    $S(t) \leftarrow S(t) \cup j$
    End If
  End For
  $A_\epsilon^*(t) \leftarrow S(t)$
  Select an optimal arm i^* uniformly, at random from $A_\epsilon^*(t)$
  Observe: reward vector $r_{i^*}, r_{i^*} = [r_{i^*}^1, \cdots, r_{i^*}^D]^T$;Update: $\hat{\boldsymbol{\mu}}_{i^*}$; $N_{i^*} \leftarrow N_{i^*} + 1$
End For
Output: Unfairness regret; Pareto regret



a. At $t = 1$ *b.* At $t > 1$ *c.* At $t >> 1$

Fig. 1. The dynamic of the annealing-Pareto algorithm.

Pareto-UCB1 and the annealing-Pareto, each arm is played initially one time, i.e. $Initial = 1$. Pareto-KG needs the estimated variance for each arm, $\hat{\boldsymbol{\sigma}}_i^2$, therefore, each arm is played initially 2 times which is the minimum number to estimate

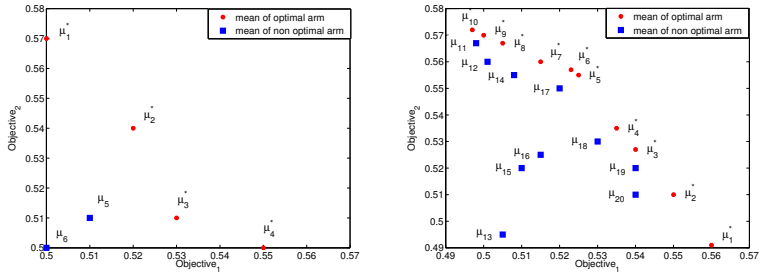


Fig. 2. Non-convex and convex mean vector set. Left figure shows a non-convex set with 2-objective, 6-armed. Right figure shows a convex set with 2-objective, 20-armed.

the variance. To get rid of tuning the parameter ϵ_{decay}, we generate uniformly at random the parameter $\epsilon_{decay} \in (0, 1)$. Shannon entropy measures the unfairness regret, Section 4. For example, for 2-objective, 6-armed with Pareto front $A^* = \{a_1^*, a_2^*, a_3^*, a_4^*\}$, where a_i^* is an optimal arm, Experiment 1. If the number of selecting each arm vector $\boldsymbol{N}$ by an algorithm is $\boldsymbol{N} = [30, 20, 20, 15, 10, 5]^T$ and the optimal number $\boldsymbol{N}^*$ of selecting each arm is $\boldsymbol{N}^* = [25, 25, 25, 25, 0, 0]^T$ at time step $t = 100$ without *initial* steps, then Shannon entropy is 0.0143.

Non-Convex Mean Vector Set;

Experiment 1. We use the same example in [2], since it is simple to understand and the Pareto mean set contains values close to each others. The number of arms $|A|$ is 6, and the number of objectives $|D|$ is 2. The true mean vector set is $(\boldsymbol{\mu}_1 = [0.55, 0.5]^T, \boldsymbol{\mu}_2 = [0.53, 0.51]^T, \boldsymbol{\mu}_3 = [0.52, 0.54]^T, \boldsymbol{\mu}_4 = [0.5, 0.57]^T, \boldsymbol{\mu}_5 = [0.51, 0.51]^T, \boldsymbol{\mu}_6 = [0.5, 0.5]^T)$, the standard deviation for arms in each objective is set to 0.1. Note that the Pareto front is $A^* = (a_1^*, a_2^*, a_3^*, a_4^*)$ where a_i^* refers to the optimal arm i^*. The suboptimal a_5 is not dominated by the two optimal arms a_1^* and a_4^*, but a_2^* and a_3^* dominates a_5 while a_6 is dominated by all the other mean vectors. Fig. 2 shows a set of 2-objective true mean with a non-convex set.

Experiment 2. We add extra 3 objectives and 14 arms in Experiment 1, resulting in 5-objective, 20-armed, we add 3 optimal arms and 11 dominated arms by all the arms in Pareto front A^*. Pareto front contains 7 optimal arms. Fig. 3 gives the average cumulative Pareto and unfairness regret performances. The y-axis is either the average of the cumulative Pareto or unfairness regret performance. The x-axis is the time steps. Fig. 3 shows the performance of algorithms. The annealing-Pareto is the best algorithm and Pareto-UCB1 is the worst one. Pareto-KG has an intermediate performance.

Convex Mean Vector Set

Experiment 3. With number of objectives D equals 2, number of arms $|A|$ equals 20 and convex Pareto mean set, $(\boldsymbol{\mu}_1 = [.56, .491]^T, \boldsymbol{\mu}_2 = [.55, .51]^T, \boldsymbol{\mu}_3 = [.54, .527]^T, \boldsymbol{\mu}_4 = [.535, .535]^T, \boldsymbol{\mu}_5 = [.525, .555]^T, \boldsymbol{\mu}_6 = [.523, .557]^T, \boldsymbol{\mu}_7 = [.515, .56]^T, \boldsymbol{\mu}_8 = [.505, .567]^T, \boldsymbol{\mu}_9 = [.5, .57]^T, \boldsymbol{\mu}_{10} = [.497, .572]^T, \boldsymbol{\mu}_{11} = [.498, .567]^T, \boldsymbol{\mu}_{12} = [.501, .56]^T, \boldsymbol{\mu}_{13} = [.505, .495]^T, \boldsymbol{\mu}_{14} = [.508, .555]^T, \boldsymbol{\mu}_{15} = [.51, .52]^T, \boldsymbol{\mu}_{16} = [.515, .525]^T, \boldsymbol{\mu}_{17} = [.52, .55]^T, \boldsymbol{\mu}_{18} = [.53, .53]^T, \boldsymbol{\mu}_{19} = [.54, .52]^T, \boldsymbol{\mu}_{20} = [.54, .51]^T)$,

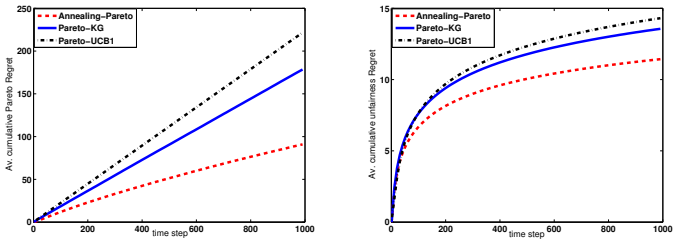


Fig. 3. Performance comparison on 5-objective, 20-armed with non-convex mean vector set. Left sub-figure shows the average cumulative Pareto regret performance. Right sub-figure shows the average cumulative unfairness regret performance.

the standard deviation for arms in each objective is set to 0.1. The Pareto front A^* contains 10 optimal arms, $A^* = (a_1^*, a_2^*, a_3^*, a_4^*, a_5^*, a_6^*, a_7^*, a_8^*, a_9^*, a_{10}^*)$. Fig. 2 shows a set of 2-objective convex true mean vector set.

Experiment 4. We add extra 3 objectives and 10 arms in Experiment 3, resulting in 5-objective, 20-armed, we add dominated arms by all the arms in A^*. Pareto front A^* still contains 10 optimal arms. Fig. 4 gives the average cumulative Pareto and unfairness regrets and shows the annealing-Pareto performance is the best algorithm, and the Pareto-UCB1 performance is the worst one according to the Pareto regret performance, while according to the unfairness regret performance Pareto-KG is the worst algorithm.

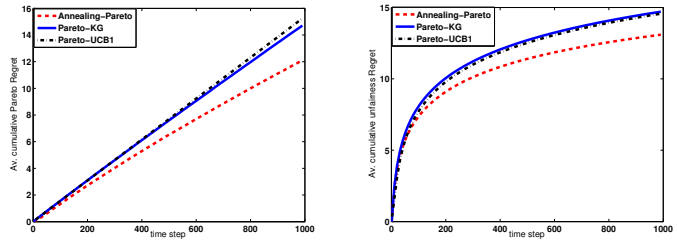


Fig. 4. Performance comparison on 5-objective, 20-armed with convex mean vector set. Left sub-figure shows the average cumulative Pareto regret performance. Right sub-figure shows the average cumulative unfairness regret performance.

From the above experiments, we see that the annealing-Pareto algorithm is the best one according to both the unfairness and Pareto regrets. The intuition is that the annealing-Pareto does not have an exploration term that decreases fast to 0 after time steps to control the trade-off between exploration and exploitation. Instead, the annealing-Pareto has a decay parameter that decreases slowly to 0, this means that the annealing-Pareto explores widely the available arms.

For convex mean vector set, Pareto-KG outperforms Pareto-UCB1 according to the Pareto and unfairness regrets.While, for non-convex mean vector set, Pareto-KG outperforms Pareto-UCB1 according to the Pareto regret and Pareto-UCB1 outperforms Pareto-KG according to the unfairness regret. The intuition is the exploration term. The exploration term for UCB1 depends on the time step t and the number of times N_i arm i is pulled and it will be high if the arm i is less selected. Thus, UCB1 plays fairly the optimal arms because it selects the optimal arms that have either larger estimated mean or larger exploration term. In contrast, the exploration term for KG policy depends on the estimated mean of all other arms and on the estimated variance of arm i. The exploration term is large if the variance of arm i is low, or if the estimated mean of arm i exceeds in the future. Thus, KG selects more efficiently the optimal arms.

7 Conclusion

We introduced the normal MOMAB, Pareto dominance relation, the performance measure in the MOMAB, Pareto-KG and Pareto-UCB1. We proposed the annealing-Pareto algorithm. We proposed using the entropy measure as a performance measure in the MOMAB. We studied empirically the trade-off between exploration and exploitation (or the trade-off for short) in the normal MOMAB. Pareto-KG and Pareto-UCB1 trade-off by using KG and UCB1 policy, respectively. While, the annealing-Pareto trades-off by using a decay parameter. Finally, we compared Pareto-KG, Pareto-UCB1, and the annealing-Pareto and concluded that: the annealing-Pareto is the best algorithm according to both the Pareto and the unfairness regret performance measures.

References

1. Yahyaa, S.Q., Drugan, M.M., Manderick, M.: The Scalarized Multi-Objective Multi-Armed Bandit Problem: An Empirical Study of its Exploration vs. Exploration Tradeoff. In: International Joint Conference on Neural Networks. (2014)
2. Drugan, M.M., Nowe, A.: Designing Multi-Objective Multi-Armed Bandits Algorithms: A study. In: International Joint Conference on Neural Networks. (2013)
3. Sethna, J.: Statistical Mechanics: Entropy, Order Parameters and Complexity. Oxford University Press, (2006)
4. Zitzler, E. and et al.: Performance Assessment of Multiobjective Optimizers: An Analysis and Review. J. IEEE Transactions on Evolutionary Computation 7, 117–132 (2002)
5. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-Time Analysis of the Multiarmed Bandit Problem. J. Machine Learning 47(2-3), 235–256 (2002)
6. Yahyaa, S.Q., Drugan, M.M., Manderick, B.: Knowledge Gradient for Multi-Objective Multi-Armed Bandit Algorithms. In: International Conference on Agents and Artificial Intelligence (ICAART). (2014)
7. Ryzhov, I.O., Powell, W.B., Frazier, P.I.: The Knowledge Gradient Policy for a General Class of Online Learning Problems. In: Operation Research, (2011)
8. Powell, W.B.: Approximate Dynamic Programming: Solving the Curses of Dimensionality. John Willey and Sons, New York, USA, (2007)