Accelerated ε-Greedy Multi Armed Bandit Algorithm for Online Sequential-Selections Applications

Khosrow Amirizadeh*, Rajeswari Mandava
Computer Vision Lab., School of Computer Sciences, Universiti Sains Malaysia (USM), 11800 Penang, Malaysia.

* Corresponding author. Tel.: 604-6532157; email: Khosrowamirizadeh@yahoo.com.
Manuscript submitted September 12, 2014; accepted March 8, 2015.
doi: 10.17706/jsw.10.3.239-249

Abstract: Current algorithms for solving multi-armed bandit (MAB) problem in stationary observations often perform well. Although this performance may be acceptable with accurate parameter settings, most of them degrade under non stationary observations. We setup an incremental ε-greedy model with stochastic mean equation as its action-value function which is more applicable to real-world problems. Unlike the iterative algorithms suffering from step size dependency, we propose an adaptive step-size model (ASM) to introduce adaptive MAB algorithm. The proposed model employs ε-greedy approach as action selection policy. In addition, a dynamic exploration parameter ε is introduced to be ineffective by increasing decision maker's intelligence. The proposed model is empirically evaluated and compared with existing algorithms including the standard ε-greedy, Softmax, ε-decreasing and UCB-Tuned models under stationary as well as non stationary situations. ASM not only addresses concerns in parameter dependency problem but also performs either comparable or better than mentioned algorithms. Applying these enhancements to the standard ε-greedy reduce the learning time which is more attractive to the wide range of on-line sequential selection-based applications such as autonomous agents, adaptive control, industrial robots and forecasting trend problems in management and economics domains.

Key words: Enhanced MAB, adaptive incremental learning, MAB empirical evaluations, setting-free step-size model.

1. Introduction

Multi armed bandit (MAB) is one of the common classical problem in statistical decision making, adaptive control engineering and machine learning. MAB is a framework to study a learning task where an agent is expected to make successive selections without any knowledge about benefit of the selection made. In a general case, it contains a set of options often referred as actions with a hidden reward for each one. The agent or decision maker faces a row of these options and decides which one must be selected such that, the cumulative reward will be maximized. Maximizing this cumulative reward is equivalent to minimizing the regret which is the difference between total rewards of optimal selections and so far cumulative reward relating to all selections performed by the agent [1], [2].

Several algorithms are presented in [2]-[7] to solve the MAB problem based on regret analysis or sample-mean analysis that estimates the 'true value' of each action. Unfortunately, most of these algorithms are evaluated theoretically. Although all theoretical proofs are good, their application to real-world problems, often arise some difficulties. As an example, in some algorithms that need to approximate the variance of the reward (e.g. some upper confidence bound models) there is considerable drop in their performance when such

algorithms are operating under high variances [8,] [9]. Authors in [10], also, concluded this issue. They said, empirical results may be different where, the algorithms operate under real-world conditions, while the models are built based on the best asymptotic guarantees. An extensive empirical study that compares MAB algorithms with different settings has been conducted by Kuleshov and Precup [8]. They concluded that, some theoretical guarantees do not ensure good performance in real-world applications. Besides, performance of some algorithms is limited to specific settings. However, they found that, models based on ε-greedy approach are still attractive and are more applicable to real-world tasks. According to the results reported by authors in [8], performance of some MAB algorithms dramatically decreases when different settings are applied. However, non stationary observations are absent in these comparisons as well as the effect of high variance of reward. In adaptive control and adaptive pattern recognition tasks, we usually encounter environments which are effectively non-stationary. These are some of the concerns related to MAB algorithms that are usually invisible in theoretical assessments.

Another concern, in the incremental MAB models, is the step-size dependency. As an example, Bubeck & Cesa-Bianchi [2] introduced an on-line mirror descent (OMD) algorithm based on gradient descent method that minimizes total loss incurred. OMD uses step size η and similar to all other gradient-based algorithms, the performance of OMD is dependent on fine tuning of η. Subsequently, they applied stochastic estimation of the gradient of the loss function to make the second gradient descent model, namely OSMD. Although, both models are theoretically sound, the step size dependency problem remains. Here, it may be concluded that, except two key parameters, variance of the reward and number of options, that may affect the efficiency of MAB algorithms [8], parameter dependency and type of observations are two other important factors. These problems are only observed in exhaustive empirical evaluations. To the knowledge of the authors, these issues are yet to be addressed by the research community.

This paper aims to minimize these concerns by presenting an incremental ε-greedy model utilizing a stochastic-mean as action-value function coupled with an automatic computation of the step-size. This called ASM that presents two modifications. The first modification is the adaptive step size computation and the second modification is the dynamic computation of the exploration rate ε in order to more balance the exploration-exploitation trade-off. From this study, it is observed that, ASM is more stable under different variance without any step-size dependency and mostly convergence faster than other models.

The paper is organized as follows: Section 2 introduces the mathematical model of MAB and relevant algorithms. The proposed adaptive step size model ASM are presented in Section 3. Implementations, comparisons and experimental evaluations are reported in Section 4 and finally, the paper ends with the conclusion section.

2. MAB Framework and Different Approaches to Solve It

Let $A = \{a_1, a_2 \dots a_N\}$ is a set of N usable actions whose reward distribution set is $D = \{d_{a_1}, d_{a_2} \dots d_{a_N}\}$. The set of expected value of these distributions is $M = \{Q^*(a_1), Q^*(a_2) \dots Q^*(a_N)\}$ and set of variances is $S = \{\sigma_{rew}(a_1), \sigma_{rew}(a_2) \dots \sigma_{rew}(a_N)\}$. The observation sequences follow an independent and identical distribution (i.i.d). In non stationary observation, these expected values may vary with time. After choosing an action a, k_a times, instant estimation of 'actual value', namely $Q^*(a)$, at time step k is obtained through the sample-mean equation $Q_k(a) = 1/k_a \sum_{i=1}^{k_a} r_i(a)$. Here, $r_i(a)$ is the instant reward at step i. Finding the best estimate of $Q^*(a)$ is the main objective. This means that $lim_{k\to\infty} Q_k(a) = Q^*(a)$. On the other hand, the expected regret at step k may be defined through expression $R_k = k\, Q_{a^*} - \sum_{i=1}^{k} Q(a_i)$, where the best action is defined by $a^* = argmax_a Q(a)$. Clearly, in this approach, the goal is to minimize the expected regret. Since with non stationary observations, the optimal expected value does not decrease with time, the regret analysis may not a suitable approach; consequently, we focus on the first objective.

In this paper, we replace the sample-mean equation by the stochastic-mean equation and compute its step-size optimally to make an adaptive incremental ε-greedy algorithm. We can adapt this incremental MAB models to

other existing action selection policies for comparisons and evaluations. The incremental algorithm below computes the stochastic-mean of the observation is used to estimate the 'actual value' of each action:

$$Q_{k+1}(a) = Q_k(a) + \eta_k \,.\, [r_{k+1}(a) - Q_k(a)] \tag{1}$$

where $Q_{k+1}(a)$ is the estimation of the mean related to the action a. The step size η_k must be chosen from domain $0 < \eta_k \leq 1$. In both stationary and non stationary environments, the step size must be precisely tuned to achieve the best estimate. The term in the brackets is the temporal difference error. Although, the sample-mean procedure for computing the cumulative reward is more attractive and simple to implement, Eq. (1) is more suitable to study this estimate with non-stationary observations. It is necessary to define the action selection policy and the step-size strategy for this incremental model to form the main structure of the incremental MAB algorithm. These are listed as follows:

2.1 ε- Greedy Policy and Polynomial Step-Size

Based on ε-greedy policy, an agent exploits current information to select an action with highest $Q_k(a)$ greedy with probability $1 - \varepsilon$ and it explores a few actions randomly with probability ε. Here, the exploration parameter ε is fixed. Therefore, the value estimation is defined based on the Eq. (1) with the polynomial step size function [11] that is $\eta_k = 1/k^{\beta}$ where k is the step counter and $0 < \beta \leq 1$.

2.2 ε- Decreasing Policy and Polynomial Step-Size

With the passage of time as the agent gets more intelligent, exploration rate could be reduced instead of a fixed value is used. This idea is implemented in some variant of ε − greedy approach. Vermorel *et al*. [10] reported function $\varepsilon_k = \min\{1, \frac{\varepsilon_0}{k}\}$ where $\varepsilon_0 > 0$. We use a gradually decreasing function Eq. (2) which is introduced by authors in [9] and again utilize Eq. (1) with the polynomial step-size to estimate value functions.

$$\varepsilon_k = \log k \,/k \tag{2}$$

2.3 UCB-Tuned Policy and Polynomial Step-Size

Some authors have addressed tasks to define an upper confidence band (UCB) to decrease the cumulative regret [3], [4], [7]-[9]. UCB approaches consider the number of times ~~an~~ action has been selected after k rounds, namely k_a plus cumulative reward as greedy criterion. Subsequently, the policy that select an action may be $a_k^* = argmax_a(Q_k(a) + \sqrt{\frac{2ln\,k}{k_a}}$). The UCB-Tuned model [3], [9] considers both empirical mean and variance of each action at every step. We use Eq. (3) that introduces UCB-Tuned action selection policy and employ Eq. (1) with polynomial step-size for value function estimation.

$$a_k^* = argmax_a(Q_k(a) + \sqrt{\left(\frac{ln\,k}{k_a}\right)\min(0.25\,, V_k(a))} \tag{3}$$

where, $V_k(a) = \sigma_k^2(a) + \sqrt{\frac{2ln\,k}{k_a}}$ and the variance is $\sigma_k^2(a) = \frac{1}{k}\sum_i^k (r_i - Q_i(a))^2$.

2.4 ε- Greedy Policy and Adaptive Step-Size Model OSA

A combination of an automatic step size routine and ε-greedy policy may form an adaptive incremental MAB algorithm. A good survey that compares step size computation models has been presented in [11]. Among these, the model OSA optimally computes the step-size. Due to similarity of this approach with our proposed model, this combination is made for comparisons.

2.5 Softmax Action Selection Policy

Based on the Boltzmann distribution, an agent selects the best action with respect to the probability order that is proportional to the average of the cumulative reward [1]. At each round, the probability of all actions must be computed and the action with higher probability must be selected. This is expressed as follows:

$$P_k(a) = e^{Q_k(a)/\tau} / \sum_{b=1}^{N} e^{Q_k(b)/\tau} \quad (4)$$

Here, $P_k(a)$ is the probability of selecting an action a at step k. The parameter $\tau \epsilon R^+$ is called temperature that controls the randomize behavior of the choice. We use Eq. (4) as the action selection policy and employ Eq. (1) with the polynomial step-size for value function estimation.

3. The Proposed Adaptive Step Size Model (ASM)

The general approach behind the proposed model is the steepest descent (SD) optimization approach. The iterative model stated in Eq. (1) computes the current estimate of the 'actual value' related to the selected action a is referred to derive the proposed model. Sequence $\{\eta_k\}$ is a series of positive scalar gains or the step sizes that plays an important role in this iterative equation. Convergence of the Eq. (1) is guaranteed while the step size is set based on the following assumptions:

$$\sum_{k=0}^{\infty} \eta_k \to \infty \ , \ \sum_{k=0}^{\infty} \eta_k^{r>1} < \infty \ , \ \lim_{k\to\infty} \eta_k \to 0 \quad (5)$$

With any step size that follows the conditions in Eq. (5), Eq. (1) can converge to the optimum point almost surely [1, 2, 12], whereas how it converges is our concern. In steepest descent technique, the next search point is chosen in a direction that optimizes the objective function [14]. Assume that a quadratic function $J = E[e_k^2]$ where $e = Q^* - Q_k$ is to be optimized. Now, the trajectories towards the optimum point, namely Q^*, based on the Gradient descent method may be defined as [12, 13]:

$$Q_{k+1} = Q_k + \eta_k g_k \quad (6)$$

This line search in gradient descent method will be successful if the parameters step size η_k and g_k are carefully defined [12]. g_k is the gradient of the objective function in opposite direction. In the steepest descent method the step size η_k may be chosen as follows:

$$\eta_k = arg\ min_{\eta>0}\ J(Q_k + \eta g_k) \quad (7)$$

From optimally criterion we have:

$$\frac{\partial J(Q_k+\eta g_k)}{\partial \eta_k} = 0 \quad (8)$$

$$\frac{\partial J(Q_k)}{\partial Q_k} = \lim_{k\to\infty} E[Q^* - Q_k] = \lim_{k\to\infty} E[Q^*] - Q_k = R_k - Q_k \quad (9)$$

We assume that $\lim_{k\to\infty} E[Q^*] = R_k$. Applying chain rule in Eq. (8) helps to find the optimum value of η_k at each step as follows:

$$\frac{\partial J(Q_{k+1})}{\partial \eta_k} = \frac{\partial J(Q_{k+1})}{\partial Q_{k+1}} \frac{\partial Q_{k+1}}{\partial \eta_k} = 0 \ , \quad (10)$$

$$\frac{\partial J(Q_{k+1})}{\partial Q_{k+1}} = R_{k+1} - Q_{k+1}, \{\text{From Eq.(9)}\}$$

$$\frac{\partial Q_{k+1}}{\partial \eta_k} = r_{k+1} - Q_k, \quad \{\text{From Eq.(1)}\}$$

We substitute two fractions of Eq. (10) by the equivalent values above, the result is:

$$(R_{k+1} - Q_{k+1})(r_{k+1} - Q_k) = 0$$

Replacing Q_{k+1} with Eq. (1):

$$[R_{k+1} - (Q_k + \eta_k\ (r_{k+1} - Q_k))\ ]\ \ [r_{k+1} - Q_k] = 0$$

Finally, after reordering the elements, the step size at each step is computed by:

$$\eta_k = \frac{R_{k+1}r_{k+1} - R_{k+1}Q_k - r_{k+1}Q_k + Q_kQ_k}{r_{k+1}.r_{k+1} - r_{k+1}.Q_k - r_{k+1}.Q_k + Q_k.Q_k} \tag{11}$$

In order to compute the step size η_k, current R_{k+1} is iteratively estimated using following stochastic equation:

$$R_{k+1} = R_k + \left(\frac{k_a}{k_a+1}\right)(r_{k+1} - R_k) \tag{12}$$

Here R_k is the current estimation of the expected value Q^*, and r_{k+1} is the 'current reward' at step k. The term $\{k_a/(k_a+1)\}$ may assist the error $(r_{k+1} - R_k)$ damp to zero. Since R_k will be closer to Q_k , numerator is smaller than denominator and then $\eta_k < 1$ and consequently $\eta_k \to 0$. Besides, the sum of this fraction $\sum_{k=1}^{\infty}\eta_k$ will be infinity. From this, it is inferred that the step size with a decreasing rate drops to its minimum point and hence justifies the conditions in Eq. (5). This behavior is later confirmed in the empirical evaluation as shown in Fig. 5, Section IV. Thus these computations prove the convergence of ASM. At each step k, the step size η_k is adaptively computed. Besides the simplicity and parameter independency are two additional advantages of this procedure. Due to these linear mathematical operations, the computational complexity is the same as the standard model which is stated in Eq. (1). The optimal computation of the step size is the first enhancement in this paper.

Since, the temporal error decreases with time, it implies that the agent gains sufficient knowledge and hence, the exploration rate ε may be reduced gradually to increase the bias towards exploitation. A suitable rule simulating this behavior is used as a basis to propose dynamic exploration criteria. $TD = R_k - Q_k$ is the expected temporal difference error. The TD error is high at the start of the learning task and after the agent is better trained, this TD error decreases. At this stage the agent is expected to make more greedy selections instead of random selections. Consequently, the exploration parameter ε should be low. To achieve this, the following functions $f\ (TD, s)$ and $\varepsilon\ (a)$ are introduced as follows:

$$f(TD,\ s) = \left.(1 - e^{\frac{-|TD|}{s}})\right/(1 + e^{\frac{-|TD|}{s}}) \tag{13}$$

$$\varepsilon(a) = \varepsilon(a) + \delta_k.(f(TD, s) - \varepsilon(a)) \tag{14}$$

$f(TD, s)$ is a bipolar sigmoid function that takes any value from negative infinity to positive infinity as input, and produces an output with values between 0 and 1. The parameters s and $\delta_k = 1/k$ define the sensitivity and step size to smooth the error in the iterative expression Eq. (14). The function $\varepsilon(a)$ is the rate of exploration when an action a is selected. The parameter s determines the sensitivity of ε (low value of s is related to high rate of change in ε and a high value of s produces low rate of change in ε). Linking the parameter s to the reward variance of each action $s \cong {\sigma^2}_{rew}$, links the exploration probability ε with ${\sigma^2}_{rew}$. Through this definition, ${\sigma^2}_{rew}$ has a direct control on the performance of the algorithm. It may be estimated as:

$$\sigma^2_{rew,k}(a) = \frac{1}{k_a}\sum_i^{k_a}(r_i - Q_i(a))^2 \tag{15}$$

This dynamic computation of ε is the second enhancement in the proposed model. The linear complexity of the standard model Eq. (1) is still preserved. However a small amount of additional time is required to compute

'expected reward', current 'exploration parameter ε' and the current 'variance'. Algorithm 1 presents the proposed adaptive step size model ASM:

Algorithm 1. Adaptive Step size Model (ASM).

For $k = 1$ to Plays
  Select a_k^*;
  Receive Reward $r_{k+1}(a_k^*)$;
  Compute $R_k(a_k^*)$; // {Eq. (12)}
  Compute $\eta_k(a_k^*)$; // Eq. (11)}
  Update $Q_k(a_k^*)$; // {Eq. (1)}
  Update $\sigma^2_{rew,k}(a_k^*)$; // {Eq.(15)
  Compute $f(TD, s)$; // {Eq. (13)}
  Update $\varepsilon(a_k^*)$; // {Eq. (14)}
End

In this section, we introduced adaptive step size model (ASM) to employ in Eq. (1) and estimate the value functions in MAB problem. The main objective is to maximize the cumulative rewards and the number of optimal selections especially, under different variances and non stationary observations. Two major contributions in the proposed algorithm are: 1) introducing the adaptive step size model and 2) presenting the dynamic exploration rate. We also applied this approach in the incremental MAB with UCB Policy which is reported in [14].

4. Experimental Results and Discussion

This section is devoted to experimental evaluation of the methods reviewed in Section II as well as the proposed model ASM. We named these models by ε-greedy (Sec 2.1), ε-decreasing (Sec. 2.2), UCB-Tuned (Sec. 2.3), OSA (Sec. 2.4), Softmax (Sec. 2.5) and ASM.

Algorithm 2. Incremental MAB Algorithm.

```
Define #Repeats=2000, #Plays=2000;
Define totalcumulativeReward(#Repeats, #Plays);
Define #optimalSelection (#Repeats, #Plays);
For z = 1 to #Repeats
  For k = 1 to #Plays
    Select a_k^*; {based on the relevant policy and model}
    If this is a correct selection increment #optimalSelction(z , k)
    Receive Reward ; {based on stationary / non-stationary cases}
    Update CumulativeReward (z , k) with current reward ;
    Update Value function Q_k ; { based on Eq. (1) }
  End
End
Plot Mean(#optimalSelection)
Plot Mean(totalcumulativeReward)
```

Each method and their setting is run for 2000 plays that each one is again repeated 2000 times to get an appropriate average over these independent runs. The general settings are: the number of arms/actions is N=5 and 20. In the stationary case, all rewards are taken from a normal distribution with mean $Q^* \sim N(0,1)$ and standard deviations $\sigma_{rew} \in \{0.1, 1, 5\}$. The reward function is $rew = Q^* + \sigma_{rew} * Rand$, that is separately computed for each action, after the action is selected. The $Rand$ function also gives a random number from a

normal distribution with mean 0 and standard deviation 1. At start, variance is set to the normal value of 1 ($\sigma_{rew} = 1$) and it is changed in subsequent experiments. In the non stationary observations case, this rule is again used except that Q^* at each round is changed by $Q^* \sim N(Q^*, \sigma_{Q^*})$ where $Q^*(a) = Q^*(a) + \sigma_{Q^*} * Rand$. At first, both σ_{Q^*} and Q^* are set to 1 for all arms/actions and then Q^* is changed incrementally so that the value of Q^*, for each arm, behaves as an independent random walk to simulate a non-stationary situation. The MATLAB 2012a is used for programming and drawing plots. Algorithm 2 can increase our understanding about these implementations.

The exploration parameter ε is set to 0.1 for fixed-epsilon approaches ε-greedy (Sec. 2.1) and OSA (Sec. 2.4). The initial value of parameter ϑ_0 (in OSA) is 0.3. Other procedures use the general polynomial step size function $\eta_k = 1/k^{\beta}$ that β sets to 0.3 in stationary case and is 0.5, 0.5, 0.1 for ε-greedy, ε-decreasing and UCB-Tuned respectively in non stationary cases. Parameter tau in Softmax is set to be 0.2. All experiments are run on a PC notebook with Core i2, 2.00 GHz, L2 Cache 2 MB, BUS 667 MHz and 2 GB DDR2 RAM.

2.1 Percentage of Optimal Action Selections and Amount of Cumulative Reward under Stationary Observations

In this experiment at each step, the 'best action' is defined by the reward distribution, and the 'selected action', chosen by the algorithm is compared. If these two are the same, it is denoted as 'optimal action selection'. The percentage of the optimal action selections is computed and plotted for all plays. This percentage will be different based on different σ_{rew} values. Fig. 1 left shows the average of the cumulative rewards with $\sigma_{rew} = 1$. The plot shows that all algorithms produce nearly equal cumulative rewards, however ASM and UCB-Tuned collect more than other models for example OSA and ε-greedy. It is noted that, ASM operates without any step-size setting. The percent of optimal selections is depicted on the right side of Fig. 1. Most of models achieved close to 90% optimal selection.

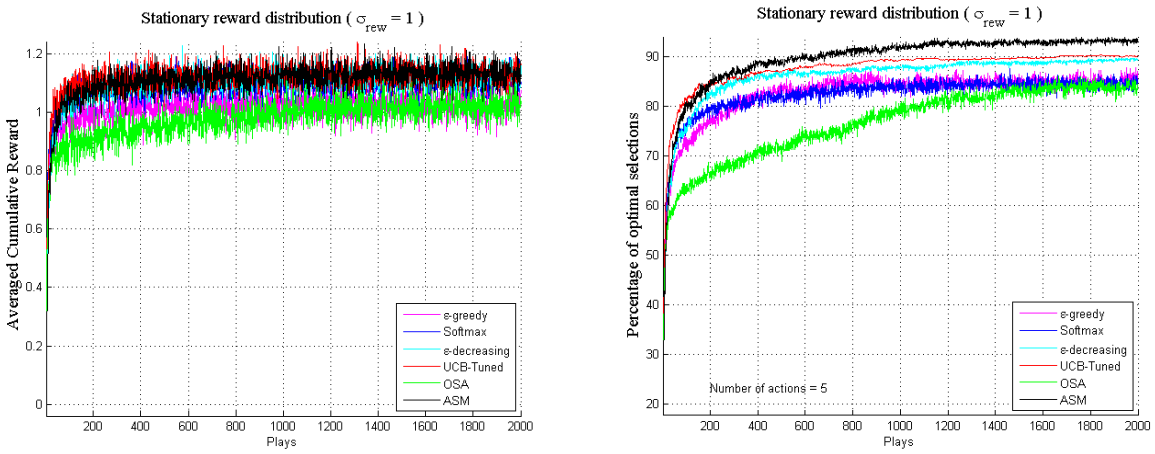

Fig. 1. Average of cumulative reward over 2000 plays (left) and percent of optimal selections (right). All method used iterative model in Eq. (1) to estimate their value function.

All bandit algorithms are dependent on the variance of the reward. It is important to know which one is less sensitive to this variance. To gain insight into this case, we change the variance of the reward to 0.1 and 5. Fig. 2 plots these results. From these observations it is noted that, some procedures work well only under situations with lower variances. With low variance, ε-decreasing, UCB-Tuned and ASM handle MAB problem well. OSA and ε-greedy converge slowly to their optimum values. Under higher variance case which is depicted in Fig. 2, ε-greedy and ASM results show better performance. Fig. 2 right plots this case. Although the greedy models seem to make acceptable operations, they need to be again tuned for each variance. Softmax operates well only under normal variance $\sigma_{rew} = 1$ as it is plotted in Fig. 1 right. These results indicate that ASM enhances the performance of ε-greedy MAB model both under high as well as low variances of the reward.

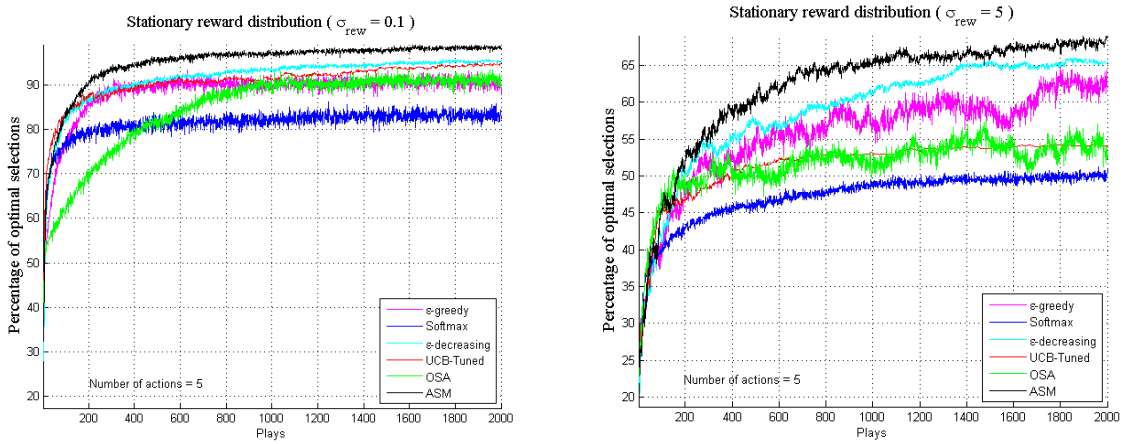

Fig. 2. Percent of optimal action selections for different variances of reward, 0.1 (left) and 5 (right). ASM has acceptable performance under both high and low variances without step-size tuning.

2.2 Percent of Optimal Action Selection under Non-Stationary Observations

One of the main questions is what happens when the distribution of the reward changes during runs. This situation is simulated by varying the mean of the reward distribution. At each time step the mean changes by rule $Q^* = Q^* + \sigma_{Q^*} * Rand$. This makes an incremental trend in the cumulative reward curves. Fig. 3 left depicts this situation. However, as shown in Fig. 3 right, the percentage of optimal action selections under non-stationary observations is less than in stationary case.

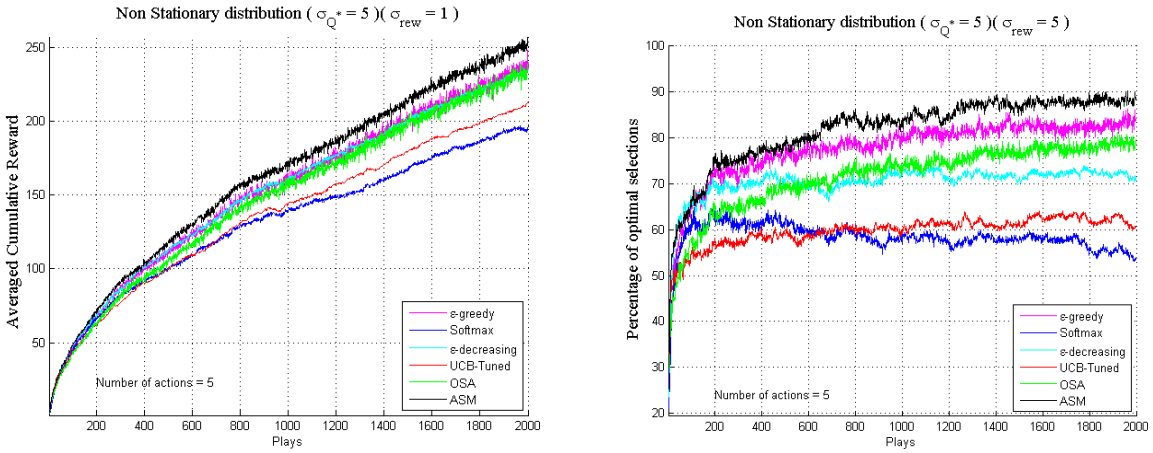

Fig. 3. Average of the cumulative reward over 2000 plays (left) and the percentage of optimal action selections (right). All methods used iterative model Eq. (1) in non-stationary situation.

All models necessitate an additional step size tuning process under non stationary observations except ASM that automatically tunes itself. Naïve ε-greedy model also has moderate performance but it operates both in normal as well as low variances. The performances of Softmax and UCB-Tuned are not acceptable with non-stationary observations. From these results we note that adaptive step size with incremental value function equation may improve the performance of MAB algorithm.

2.3 Stability Under Increasing Number of Actions

The most noticeable point is the dependency of all models on the number of actions/arms. Fig. 4 illustrates the percent of optimal selections when the number of arms increased to $N = 20$, with both stationary (left) and non-stationary (right) observations. Fig. 4 left shows that ASM, UCB-Tuned and ε-greedy operate better than others

models. However, with non-stationary observations, Fig. 4 right, the prominent result belongs to the ASM.

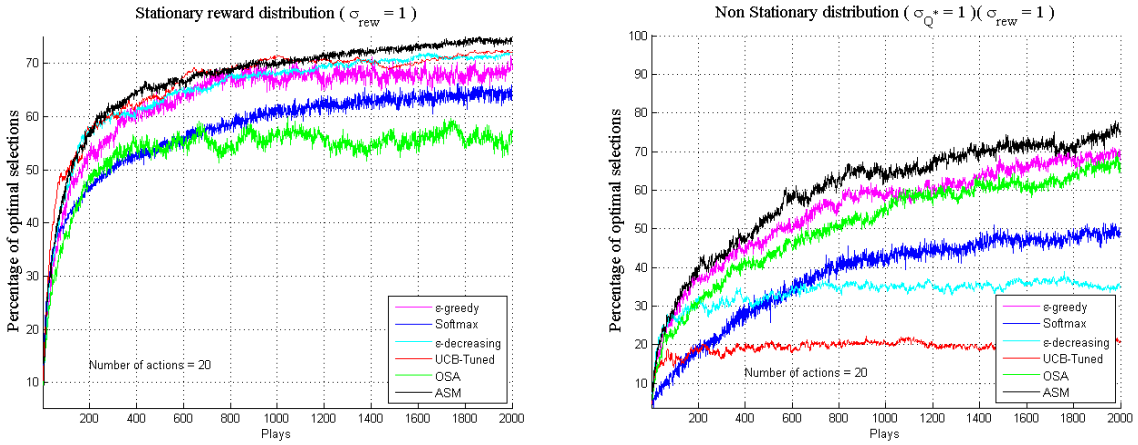

Fig. 4. Percent of optimal action selection with higher number of actions/arms.

2.4 Behavior of the Step Size in ASM

Based on the assumptions in Eq. (5) the step size, for each arm/action, should be decrease gradually. It means $\lim_{k\to\infty} \eta_k \to 0$. However, we know that $\lim_{k\to\infty} (R_k - Q_k) \to 0$. Fig. 5 shows the behavior of the step size in stationary and non-stationary cases. These curves are step size values of action/arm number 4.

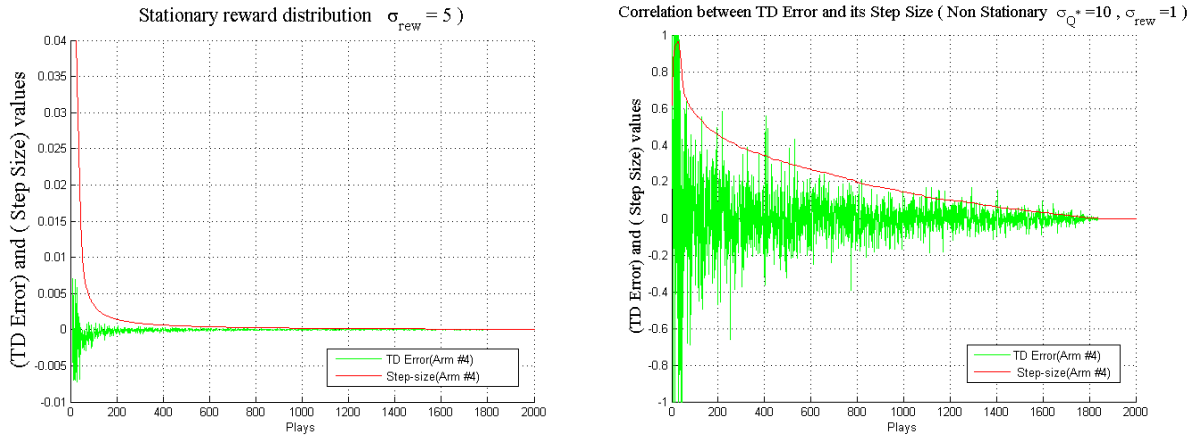

Fig. 5. The step sizes values behave as a gradually decreasing curve. The temporal difference error curve under stationary (left) and non-stationary (right) conditions decrease by time progress.

In both stationary and non stationary cases, as depicted in the figures, the step sizes follow a decreasing rate. We overlay these decreasing plots on corresponds expected temporal error $TD = R_k - Q_k$. The step-size plot curve is the average of the step sizes over 2000 iterations. Since, all of arms are not selected in all steps; the number of plays with the number of selected an action is inconsistent. Hence, as it clear, towards to the end of plays, step size plots, in both graphs, fall on the zero lines that does not mean the step size values are set to zero. In fact, all actions/arms are not selected at each repetition.

5. Conclusion

In this article, empirical evaluations of incremental MAB algorithms are presented. Different variances, non stationary observations, number of actions and step-size dependency are set of concerns that may degrade the operation of MAB models. In this study more effective issues on the performance of adaptive applications such as non stationary observations and step size dependency are considered. We conclude that only a few algorithms

are able to maintain their performance under non stationary observations that are more representative in real world applications. More specifically algorithms such as UCB-Tuned are more suitable with stationary observations. However, a few algorithms are able to maintain their efficiency under widely changing variances. The ε-greedy models have acceptable performance in situations with non stationary observations. However their performance under larger set of arms/actions and lower stationary variance are weak. Softmax operates well under normal stationary observations.

For improving adaptability of ε-greedy model in different situations and increasing its performance under above mentioned concerns, the incremental MAB algorithm with stochastic mean equation and an adaptive step size computation is introduced. It called ASM. Enhancing the percentage of optimal action selections and maintaining the performance under these concerns without any parameter dependency are two main objectives that distinguish ASM from other iterative MAB models. Several empirical evaluations have been conducted to evaluate the performance of ASM against the naive ε-greedy, Softmax, ε-decreasing and UCB-Tuned approaches. In these comparisons, it is observed that performance of ASM is either comparable or better than other algorithms. Dynamic calculation of the exploration rate in ASM establishes a balance between the exploration and the exploitation tasks as the agent gets trained after the some plays. These modifications are more attractive for adaptive option selection tasks in control engineering, machine learning and sequential decision making in management and economics domains.

Acknowledgement

This work is supported by the Ministry of Higher Education, Malaysia under the grant (304/CNEURO/652203/K134).

References

[1] Sutton, R. S., & Barto, A. G. (1998). *Reinforcement Learning: An introduction*, Cambridge, MA: MIT Press.

[2] Bubeck, S., & C. N., Bianchi. (2012). *Regret analysis of stochastic and non stochastic multi-armed bandit problems*.

[3] Audibert, J. Y., Munos, R., & Szepesvári, C. (2009). Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science*, *410(19)*, 1876-1902.

[4] Audibert, J., Bubeck, S., & Munos, R. (2010). Best arm identification in multi-armed bandits. *Proceedings of the 23rd International Conference on Learning Theory*.

[5] Scott, S. L. (2010). A modern bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, *26(6)*, 639-658.

[6] Granmo, O. C., & Glimsdal, S. (2013). Accelerated bayesian learning for decentralized two-armed bandit based decision making with applications to the goore game. *Applied Intelligence*, 1-10.

[7] E., Dar, E. Mannor, S., & Mansour Y. (2006). Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems, *Journal of Machine Learning Research*, 1079–1105.

[8] Kuleshov, V., & Precup, D. (2010). Algorithms for the multi-armed bandit problem, *Journal of Machine learning research*, 1-48.

[9] Auer, P., Cesa, B. N., & Fischer, P. (2002). Finite-time analysis of the multi armed bandit problem. *Machine learning*, *47(2-3)*, 235-256.

[10] Vermorel, J., & Mohri, M. (2005). Multi-armed bandit algorithms and empirical evaluation. In *Machine Learning: ECML 2005* (pp. 437-448). Springer Berlin Heidelberg.

[11] George, A. P., & Powell, W. B. (2006). Adaptive step sizes for recursive estimation with applications in approximate dynamic programming. *Journal of Machine Learning, 65(1)*, 167-198.

[12] Nocedal, J., & Wright, S. (1999). *Numerical Optimization*, New York: Springer.

[13] Benveniste, A., Metivier, M., & Priouret, P. (1990). *Adaptive Algorithms and Stochastic Approximations*, New York: Springer.

[14] Amirizadeh, K., & Mandava, R. (2014). Fast iterative model for sequential-selection-based applications, *International Journal of Computers and Technology*, *12(7)*, 3689–3696.

Khosrow Amirizadeh received his B.S. degree in computer engineering from Shiraz University in 1990, Iran and M.S. degree in artificial intelligence from IAU University (Researches and Sciences) in 1998, Tehran, Iran. Currently, he works in intelligent systems lab at Universiti Sains Malaysia (USM) as a research assistant supported by the Ministry of Higher Education, Malaysia. His research interests include adaptive algorithms and reinforcement learning, evolutionary and intelligent control, sequential decision making tasks, intelligent tracking and recognition, medical imaging, Brain fiber tracking modeling and optimization of adaptive learning model.

Rajeswari Mandava received her M. Tech from Indian Institute of Technology, Kanpur in 1980. She received her PhD degrees from University of Wales Swansea, in 1995. She has been an academician at Universiti Sains Malaysia since 1982. Her research interests include pattern recognition, machine intelligence, kernel machines, kernel learning, meta heuristic search, and multi objective optimization.