Removing Sampling Bias in Networked Stochastic Approximation

Raaz Dwivedi
Department of Electrical Engineering
IIT Bombay
Mumbai, India
Email: dwivediraaz@gmail.com

Vivek S. Borkar
Department of Electrical Engineering
IIT Bombay
Mumbai, India
Email: borkar.vs@gmail.com

***Abstract*—We consider a stochastic approximation implemented on a network of computing elements corresponding to the nodes of a connected graph wherein each node polls one or more of its neighbors at random and pulls the relevant data from there. A blind implementation suffers from 'sampling bias' whereby each node's contribution to the computation gets weighed by its frequency of being polled. We propose a modified step size schedule that works around this problem. As an example, we propose a modification of an existing scheme for reputation systems that removes such a bias therein.**

I. Introduction

We consider a classical stochastic approximation algorithm implemented on a network of processors placed on a connected graph. The processors poll their neighbors at random instants to pull their data, leading to a complicated asynchronous behavior. This can lead to a sampling bias that can adversely affect the convergence properties of the algorithm. By extending the ideas of an earlier work on asynchronous stochastic approximation [2] we propose a modification of the step-size schedule that works around this problem. As an illustration, we analyze a reputation system motivated by [11].

The paper is organized as follows. The next section highlights the difficulties caused by sampling bias in a very general set-up and describes our fix for the same. Section III first describes a reputation system from [11] and indicates how the aforementioned problem is very much present in this scheme. It then proposes an alternative scheme where the proposed modification can be effectively used to circumvent the problem. Section IV presents some supporting numerical experiments. Following some remarks in Section V, Section VI concludes by discussing the results.

II. Removing Sampling Bias

Consider a connected directed or undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with node set $\mathcal{V}, |\mathcal{V}| = d$, and edge set $\mathcal{E}$. $\mathcal{N}(i) \subset \mathcal{V}$ is the set of neighbors of $i \in \mathcal{V}$. Each node i corresponds to a computing element that performs the following iteration at each time $n, n \geq 0$.

$$x_i(n+1) = x_i(n) + a(n)\left[\sum_{j \in \mathcal{N}(i)} \xi_{ij}(n)\left(h_{ij}(x(n), Y(n)) + M_i(n+1)\right)\right]. \quad (1)$$

Here we assume the following: for $n \geq 0$ and

$$\mathcal{F}_n := \sigma(x(m), M(m), Y(m), \xi_{ij}(m), m \leq n, i, j \in \mathcal{V}),$$

- $\{Y(n)\}$ is a process taking values in a finite state space S and satisfying:
$$P(Y(n+1) = j | \mathcal{F}_n) = p_{x(n)}(j|Y(n)),\ j \in S, n \geq 0,$$
for a parametrized family of transition probabilities $\{p_x(\cdot|\cdot)\}, x \in \mathcal{R}^d$, on S that is Lipschitz in x (the *Markov* noise),
- $\{M(n)\}$ is a square-integrable sequence adapted to $\{\mathcal{F}_n\}$ satisfying for $n \geq 0$,
$$E[M(n+1)|\mathcal{F}_n] = 0$$
$$E[\|M(n+1)\|^2 | \mathcal{F}_n] \leq K(1 + \|x(n)\|^2),$$
for some $K > 0$ (the *Martingale* noise),
- $a(n) > 0$ satisfy
$$\sum_n a(n) = \infty,\ \sum_n a(n)^2 < \infty, \quad (2)$$
- $h_{ij}(\cdot) : \mathcal{R}^d \mapsto \mathcal{R}$ are Lipschitz, and,
- $\{\xi_{ij}(n)\}$ are independent $\{0,1\}$-valued random variables.

If one ignores the $\xi_{ij}(n)$'s, this is essentially the standard stochastic approximation algorithm with 'drift' h_{ij}, step-size schedule $\{a(n)\}$, and Martingale and Markov noises $\{M(n)\}, \{Y(n)\}$ resp., satisfying standard conditions - see, e.g., chapters 2 and 6 of [3]. The $\{\xi_{ij}(n)\}$ are indicator random variables with the interpretation: $\xi_{ij}(n) = 1 \iff i$ polls $j \in \mathcal{N}(i)$ at time n. We shall assume that

$$\sup_n \|x(n)\| < \infty \text{ a.s.}, \quad (3)$$

which usually needs a separate verification - see Theorem 9, p. 75, [3], for a sufficient condition. Our focus here is on the effect of the sampling variables $\{\xi_{ij}(n)\}$, so we do not discuss this and some other standard issues in stochastic

approximation in any detail.

The analysis of Chapter 6 and 7 of [3] suggests that under these hypotheses, (1) will almost surely track the asymptotic behavior of the ordinary differential equation (o.d.e. for short)

$$\dot{x}_i(t) = \sum_{j \in \mathcal{N}(i)} \lambda_{ij}(t) h_{ij}(x(t)),\ 1 \leq i \leq d. \tag{4}$$

where $\lambda_{ij}(t) \geq 0$ reflect the 'instantaneous relative frequencies' with which the respective polling events (of j by i) take place. (This is an intuitive statement, see *ibid.* for a mathematically rigorous account.)

If $\xi_{ij}(n) \equiv 1\ \forall\ i, j, n$, then $\lambda_{ij}(t) \equiv 1\ \forall\ i, j, n$, as well and the above o.d.e. becomes

$$\dot{x}_i(t) = \sum_{j} h_{ij}(x(t)),\ 1 \leq i \leq d. \tag{5}$$

This would correspond to a *synchronous* iterate, corresponding to the ideal algorithm. The way (1) comes about is when there are communication constraints that force a random polling as above. But the asymptotic behavior of (4) can be quite different from that of (5), implying possibly undesirable behavior of the algorithm.

To motivate our modification of the above scheme, we recall the framework of [2]. In [2], the iteration analyzed was

$$x_i(n+1) = x_i(n) + a(n)\xi_i(n)\left(h_i(x(n)) + M_i(n+1)\right). \tag{6}$$

where $\xi_i(n) = 1$ if the ith component is updated at time n and 0 otherwise.

Define $\nu(i,n) := \sum_{m=0}^{n} \xi_i(m),\ n \geq 0$. Assume that:

1) There exists $\delta > 0$ such that $\forall i$,

$$\liminf_{n \uparrow \infty} \frac{\nu(i,n)}{n} \geq \delta \text{ a.s.} \tag{7}$$

2) $\{a(n)\}$ satisfy, for $A(n) := \sum_{m=0}^{n} a(m), c \in (0,1)$,

$$\sup_n \frac{a(\lfloor yn \rfloor)}{a(n)} < \infty\ \forall\ y \in (0,1), \tag{8}$$

$$\frac{A(\lfloor yn \rfloor)}{A(n)} \stackrel{n \uparrow \infty}{\rightarrow} 1 \text{ uniformly in } y \in (c,1]. \tag{9}$$

Under these conditions, it was shown that if (3) holds, then (6) a.s. tracks the asymptotic behavior of the o.d.e. $\dot{x}(t) = \frac{1}{d} h(x(t))$, which is same as that of the o.d.e. $\dot{x}(t) = h(x(t))$ because they are merely time-scaled versions of each other.

Remark 1 In [2], an additional complication is considered, viz., communication delays. We do not include it here because one can handle it by completely analogous arguments as in *ibid.* There are additional hypotheses on $\{a(n)\}$ therein expressly for the purpose of analyzing the effect of delays. We have skipped them here.

Motivated by this, we analogously define $\nu(i,j,n) := \sum_{m=0}^{n} \xi_{ij}(m),\ n \geq 0$. Assume that (7) holds with $\nu(i,j,n)$ replacing $\nu(i,n)$ and (8), (9) also hold. Replace (1) by

$$x_i(n+1) = x_i(n) + \left[\sum_{j \in \mathcal{N}(i)} a(\nu(i,j,n))\xi_{ij}(n)\left(h_{ij}(x(n), Y(n)) + M_i(n+1)\right)\right]. \tag{10}$$

Arguing exactly as in [2], we can then conclude:

Theorem 1 Under (3), (10) a.s. tracks the asymptotic behavior of the o.d.e.

$$\dot{x}_i(t) = \sum_{j} h_{ij}(x(t)),\ 1 \leq i \leq d,\ t \geq 0. \tag{11}$$

We conclude this section with an observation that while different step-sizes for different components have been used in [2] and elsewhere for possibly other reasons (see, e.g., the '*two time-scale stochastic approximation*' of [3], Section 6.1), this is perhaps the first instance where different step-sizes have been used in the *same* component to achieve a prescribed goal.

III. A Reputation System

In [11], the problem of individually rating a committee of experts based on their track record was addressed as follows. We label the experts as $\{1, \cdots, d\}$. Suppose the initial ratings of the experts (from now, we use the term reputation in place of rating) are all equal, say $p_0^i = \frac{1}{d}\ \forall i$. At each time $t+1$, the reputation p_t^i of an active expert i is updated based on his prediction $x_t^i \in [0,1]$ of observation $y_t \in \{0,1\}$, $\{y_t\}$ i.i.d., according to the update rule

$$\begin{aligned} p_{t+1}^i &= p_t^i \frac{x_t^i}{\hat{y}_t} \text{ if } i \in E_t,\ y_t = 1, \\ &= p_t^i \frac{1 - x_t^i}{1 - \hat{y}_t} \text{ if } i \in E_t,\ y_t = 0, \\ &= p_t^i \text{ if } i \notin E_t, \end{aligned} \tag{12}$$

where $E_t :=$ the set of experts active at time t and $\hat{y}_t$ is their weighted prediction given by

$$\hat{y}_t := \frac{\sum_{i \in E_t} p_t^i x_t^i}{\sum_{i \in E_t} p_t^i}. \tag{13}$$

Almost sure convergence of p_t^i to 1 as $t \uparrow \infty$ for the best expert[1] was established under the assumption that the distribution of $I\{i \in E_t\}, t \geq 0$, is stationary and symmetric in i. The latter condition is necessitated by the fact that otherwise the frequency with which an expert i participates (i.e., $i \in E_t$) influences her rating, favoring in particular the experts who participate more often. This is not desirable in most situations, for the best experts may opine only occasionally. On the other hand, the above fix of assuming a symmetric participation

[1] if unique, the scheme oscillates between best experts otherwise.

may also be unreasonable. Motivated by this, we propose an alternative scheme as follows:

$$p^i_{t+1} = \Gamma\Big(p^i_t\Big[1 + a(\nu(i,t))I\{i \in E_t\}w^i_t - \sum_j a(\nu(j,t))I\{j \in E_t\}p^j_t w^j_t)\Big]\Big), \quad (14)$$

where $w^j_t := y_t x^j_t + (1 - x^j_t)(1 - y_t)$ and $\nu(j,t) := \sum_{m=0}^{t} I\{j \in E_m\}$. $\Gamma(\cdot)$ is the projection onto the d-dimensional probability simplex S. Also y_t need not be binary any more, it can take any value in $[0,1]$. We shall assume E_t to be i.i.d. Let $z_i := E[w^i_t], 1 \leq i \leq d$, and without loss of generality, let $z_1 > z_j, j \neq 1$. What we have done is to replace the multiplicative scaling in (12) by an additive normalization, thereby replacing ratios of expressions involving indicator random variables by additive counterparts thereof that are easier to analyze. This scheme does not have the interpretation of a network algorithm as in the preceding section because it is only one entity polling the rest. Nevertheless, we have picked this example because it is a simple and clear instance where sampling bias is a 'clear and present danger'.

By definition it is clear that w^i_t takes a high value if x^i_t and y_t are 'similar', that is the value of w^i_t is large if both x^i_t and y_t are high or both of them are low. And so it seems reasonable to say that a good expert must have a high value of z_i. We use this observation to define a convention - an expert i is better than an expert j if $z_i > z_j$. And thus in our model expert 1 is the best expert (as $z_1 > z_j, j \neq 1$). We have the following result:

Theorem 2 $p^1_t \overset{t\uparrow\infty}{\rightarrow} 1$ a.s.

Proof Since S is bounded, (3) is free. Arguing as in [2], the limiting o.d.e. is

$$\dot{p}^i(t) = cp^i(t)\left(z_i - \sum_j p^j(t)z_j\right),\ 1 \leq i \leq d, \quad (15)$$

for some $c > 0$. This is a special case of the celebrated replicator dynamics of mathematical biology and is seen to converge to the Dirac measure at 1. (See, e.g., the proof of Lemma 4, p. 14, of [4], which in fact considers a more complicated situation.) It may be noted that while the analysis of the preceding section is for a scheme without the projection operator Γ, this makes no difference for following reasons: All one needs to do is to replace the limiting o.d.e. by its projected version (see section 5.4 of [3]), which in the present case is exactly the same as the original o.d.e. because the simplex S is invariant under this o.d.e. □

IV. Numerical Experiments

In this section we present some simulation results for the reputation system considered in section III that verify the theoretical result that we have proved in Section III, Theorem 2. Several cases were considered and all were consistent with Theorem 2. Two cases with figures depicting convergence of p_i's are displayed below.

In the simulations, we label the iterations by index $n \geq 0$ and thus n plays the role of t. For a given i, we generate $I(i \in E_n)$ in an i.i.d. fashion for different i. For projection Γ, in simulations we use a slight modification followed by normalization

$$\begin{aligned} \text{First calculate, } p_i(n+1) &= \max(\epsilon, Y),\ \forall i, \\ \text{And then normalize, } p_i(n+1) &= \frac{p_i(n+1)}{\sum_j p_j(n+1)}, \end{aligned} \quad (16)$$

where Y denotes the argument of Γ in RHS of (14) with n replacing t and $\epsilon = 10^{-6}$ is a small number to prevent the algorithm from accidentally getting stuck at a lower dimensional face of the probability simplex (Note that these are invariant sets for the iteration). For clarity we depict the variation of p_i against iterations for only three experts with largest z_i values. Using the convention that an expert i is better than expert j if $z_i > z_j$, we call these three experts as 'Best 3 Experts'. If required, we refer to the unique best expert as expert i^*. We also define $\nu_i := E(I(i \in E_n))$ (note the close relation to $\nu(i,t)$).

The convergence rate of $p_{i^*}(n)$ to 1 depends upon values of z_{i^*} and ν_{i^*} relative to other z_i's and ν_i's resp. and can be boosted by changing the step size schedule from $a(n) = \frac{1}{n}$ to $a(n) = \frac{1}{[n/K]+1}$ where [x] denotes greatest integer not greater than x and K is a suitably chosen large integer. We use the modified step schedule with $K = 100$ for both cases. This schedule continues to satisfy conditions (2), but has a slower decrease, leading to faster convergence at the expense of somewhat higher fluctuations. (This is a standard trade-off in stochastic approximation.) We provide two figures for each case, one of them depicting transience (and fluctutations because of the modification for faster convergence) and other showing the convergence result.

Case 1 For a reputation system with 20 experts, we generate $x_i(n)$'s and $y(n)$ as independent random variables uniformly distributed in $[0,1]$ with randomly pre-assigned means. Fig. 1 shows that $p_{i^*}(n)$ is far from 1 because i^* has not opined sufficient number of times to be identified as the best expert. While Fig. 2 shows that finally the iterates converge to Dirac measure 1, with value 1 for the expert with highest z_i, though ν_{i^*} is 'approximately half' of the second best expert.

Case 2 We simulate a reputation system with 10 experts. Here, we directly generate w_i's. The z_i values are pre-assigned deterministically with one best expert i^* such that $z_{i^*} = 2z_i,\ \forall i \neq i^*$. However, the best expert is '10 times less likely' to opine than any other expert. That is, $\nu_{i^*} = \frac{1}{10}\nu_i, \forall i \neq i^*$. We assign such ratios to demonstrate that the algorithm is in fact successful in removing the sampling bias. Fig. 3 shows that initially there are great fluctuations but eventually $p_{i^*}(n)$

does converge to 1 as evident from Fig. 4. As compared to the previous case, the number of iterations for convergence are much larger because of the 'very rare' opining by the 'best expert' .

Fig. 1. Case 1: Transient Behavior of Best 3 Experts' Reputation

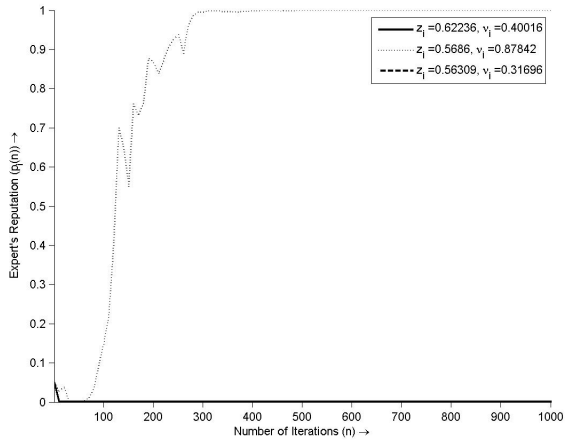
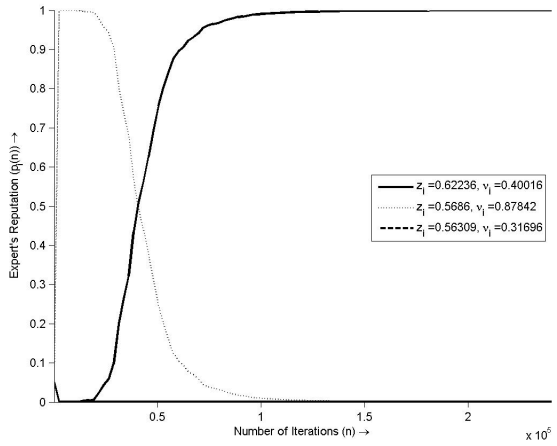


Fig. 2. Case 1: Steady State Behavior of Best 3 Experts' Reputation



V. Remarks

1) The i.i.d. assumption on $\{E_t\}$ may be relaxed to stationarity using the results on 'stochastic approximation with stationary noise' from [10]. In fact, one can drop even stationarity as long as (7) holds (cf. [2]).
2) Algorithms based on replicator dynamics have been used extensively in communication networks (see, e.g., [5]). A closer connection here is with the celebrated 'multiplicative rule' algorithm in machine learning [1], whose normalized version leads to a replicator dynamics as argued in [9].
3) Several different cases were simulated and all gave consistent results as per Theorem 2. In Case 1 as well as Case 2, different number of experts such as 50, 100 etc. were considered. Furthermore, many randomly generated tuples of z and ν were also considered among variants of Case 2. In all cases, given sufficient number of iterations, $p_{i^*}(n)$ converged to 1. Obviously if the best expert has a high ν_i compared to other experts, the convergence is very fast.

Fig. 3. Case 2: Transient Behavior of Best 3 Experts' Reputation

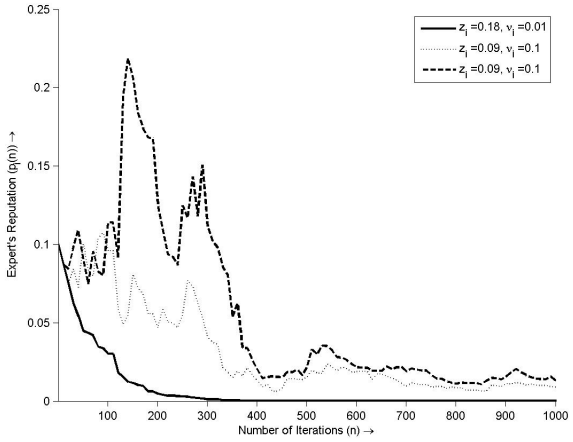
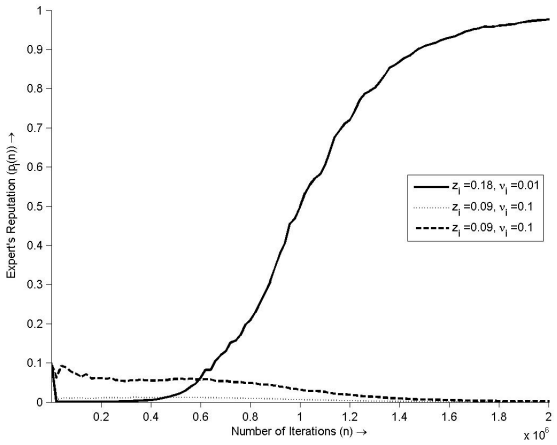


Fig. 4. Case 2: Steady State Behavior of Best 3 Experts' Reputation



VI. Conclusions

Online recommendation systems use votes from experts (read other users) to recommend objects to a particular customer (say, Anju) . They rely on matching taste, likes and dislikes of different experts with Anju and updating their reputation (read rating) depending on the extent of matching. Several existing schemes depend on different assumptions such as equally likely availability of experts at all times, presence of a good number of objects, voting by different users on a large number of objects, etc. Our scheme presented in Section III does not need such assumptions and successfully identifies the best expert. With a judicious modification in the step size schedule, we are able to handle infrequent opining of a good expert. With an update rule as given by (14), if an expert is better than any other expert, his

reputation will eventually take over everyone else's reputation, and so for further advice, Anju can rely on this best expert alone. All we need is the experts to opine 'comparably often' in the sense of (7), which is a reasonable assumption.

As already mentioned, we picked an off-the-shelf problem from [11] for illustrative purposes. More generally we can consider a network of agents collectively rating each other, a framework that could be of relevance in, e.g., a 'social networking' environment.

Another problem that naturally suggests itself is that of *tracking* the 'current best' expert when the competence of experts is time varying. Unlike the regime switching models in econometrics, we do not expect such a change to be abrupt (unless we allow for arrivals and departures of experts), so it falls within the ambit of the classical tracking problem for which there is a classical fix. One simply replaces the slowly decreasing stepsize $a(n)$ by a constant stpsize $a > 0$. The interpretation of stochastic approximation as a noisy discretization of an o.d.e. suggests that the stepsize should be viewed as a discrete time step that determines the time scale of the iteration. Thus the algorithm with a decreasing stepsize becomes slower with time, eventually becoming too slow to track the slowly varying environment. This prompts the use of a constant stepsize for such applications. One loses, however, the advantages of a decreasing stepsize, viz., asymptotic suppression of errors and fluctuations due to discretization and noise. Thus a higher constant stepsize a would mean faster algorithm but with greater fluctuations. Therefore the stepsize has to be chosen judiciously: large enough to ensure a time scale faster than that of the environmental drift, small enough that the fluctuations do not become overwhelming. This fix, however, does not work in the present case because constant stepsize will not satisfy the additional hypotheses we imposed for removing the sampling bias, so the bias is back in. One way out is to keep tab on relative frequencies of $i-j$ interactions and compensate accordingly, which becomes cumbersome. A better way seems to be do what we have done above with decreasing stepsize and reset occasionally, e.g., when the stepsize drops below a threshold.

An important potential application, right now still on the distant horizon, is networked control. Stochastic approximation and its variants have been extensively used in adaptive control [6], a networked variant of which would immediately land us with the kind of issues considered above.

On the theoretical side, there are some interesting issues. Since we are only interested in relative ranking of the experts, this is an *ordinal* optimization problem [7]. As is characteristic of such problems, while the convergence of the algorithm may be slow, the relative ranking of its components becomes apparent much faster. This calls for a finite time analysis to establish the 'probably approximately correct' behavior of the scheme. Also, there may be constraints on how often a given expert can be polled or costs associated with such an act. This makes polling strategic, turning it into a stochastic control problem.

ACKNOWLEDGMENT

Research of the second author was supported in part by a J. C. Bose Fellowship, a grant for "Distributed Computation for Optimization over large network and high dimensional data analysis" from the Department of Science and Technology and a Sponsored University Research Award from IBM.

REFERENCES

[1] Arora, S.; Hazan, E. and Kale, S.; "The multiplicative weights update method: a meta-algorithm and applications", *Theory of Computing* 8 (2012), 121-164.

[2] Borkar, V. S.; "Asynchronous stochastic approximation", *SIAM J. Control and Optim.* 36 (1998), 840-851 (correction note in *ibid.*, 38 (2000), 662-663).

[3] Borkar, V. S.; *Stochastic Approximation: A Dynamical Systems Viewpoint*, Hindustan Publ. Agency, New Delhi, and Cambridge Uni. Pres, Cambridge, UK, 2008.

[4] Borkar, V. S. and Das, D. J.; "A novel ACO algorithm for optimization via reinforcement and initial bias", *Swarm Intelligence* 3 (2009), 3-34.

[5] Borkar, V. S. and Kumar, P. R.; "Dynamic Cesaro-Wardrop equilibration in networks", *IEEE Trans. on Automatic Control* 48 (2003), 382-396.

[6] Chen, H.-F. and Guo, L.; *Identification and Stochastic Adaptive Control*, Birkhäuser, Boston, 1991.

[7] Ho, Y.-C.; "An explanation of ordinal optimization: Soft computing for hard problems", *Information Sciences* 113(3-4), 1999, 169-192.

[8] Hofbauer, J. and Sigmund, K.; "*Evolutionary Games and Population Dynamics*", Cambridge Uni. Press, Cambridge, UK, 1998.

[9] Kleinberg, R.; Piliouras, G. and Tardos, E.; "Multiplicative updates outperform generic no-regret learning in congestion games: extended abstract", *Proceedings of the 41st annual ACM symposium on Theory of computing (STOC'09)*, ACM, New York, 2009, 533-542.

[10] Kushner, H. J. and Yin, G. G.; *Stochastic Approximation and Recursive Algorithms and Applications* (2nd ed.), Springer Verlag, New York, 2003.

[11] Truong, A.; Kiyavash, N. and Borkar, V. S.; "Convergence analysis for an online reommendation system", *Proceedings of the 50th IEEE Conf. on Decision and Control and European Control Conf.*, Orlando, Florida, Dec. 12-15, 2011, 3889-3894.