Fast LSTD using stochastic approximation: Finite time analysis and application to traffic control

Prashanth L A [*1], Nathaniel Korda [†2] and Rémi Munos [‡1]

[1] INRIA Lille - Nord Europe, Team SequeL, FRANCE.
[2] Oxford University, UNITED KINGDOM.

Abstract

We propose a stochastic approximation based method with randomisation of samples for policy evaluation using the least squares temporal difference (LSTD) algorithm. Our method results in an $O(d)$ improvement in complexity in comparison to regular LSTD, where d is the dimension of the data. We provide convergence rate results for our proposed method, both in high probability and in expectation. Moreover, we also establish that using our scheme in place of LSTD does not impact the rate of convergence of the approximate value function to the true value function. This result coupled with the low complexity of our method makes it attractive for implementation in *big data* settings, where d is large. Further, we also analyse a similar low-complexity alternative for least squares regression and provide finite-time bounds there. We demonstrate the practicality of our method for LSTD empirically by combining it with the LSPI algorithm in a traffic signal control application.

Several machine learning problems involve solving a linear system of equations from a given set of training data. In this paper we consider the problem of policy evaluation in reinforcement learning (RL) using the method of temporal differences (TD). Given a fixed training data set, one popular temporal difference algorithm for policy evaluation is LSTD Bradtke and Barto [1996]. However, LSTD is computationally expensive as it requires $O(d^2)$ computations. We propose a stochastic approximation (SA) based algorithm that draws data samples from a uniform distribution on the training set. From the finite time analyses that we provide, we observe our algorithm converges at the optimal rate, in high probability as well as in expectation. Moreover, using our scheme in place of LSTD does not impact the rate of convergence of the approximate value function to the true value function. This finding coupled with the significant decrease in the computational cost of our algorithm, makes it appealing in the canonical *big data* settings.

The problem considered here is to estimate the value function V^π of a given policy π. Temporal difference (TD) methods are well-known in this context, and they are known to converge to the fixed point $V^\pi = \mathcal{T}^\pi(V^\pi)$, where $\mathcal{T}^\pi$ is the Bellman operator (see Section 2.1 for a precise definition). A popular approach to overcome the curse of dimensionality associated with large state spaces is to parameterize the value function using a linear function approximation architecture. For every s in the state space $\mathcal{S}$, we approximate $V^\pi(s) \approx \theta^\intercal \phi(s)$, where $\phi(\cdot)$ is a d-dimensional feature vector with $d << |\mathcal{S}|$, and θ is a tunable parameter. The function approximation variant of TD Tsitsiklis and Van Roy [1997] is known to converge to the fixed point of $\Phi\theta = \Pi\mathcal{T}^\pi(\Phi\theta)$, where Π is the orthogonal projection onto the space within which we approximate the value function, and Φ is the feature matrix that characterises this space.

LSTD estimates the fixed point of $\Pi\mathcal{T}^\pi$ using empirical data $\mathcal{D} := \{(s_i, r_i, s'_i), i = 1, \ldots, T)\}$ obtained by simulating the Markov decision process (MDP) with the underlying policy π. For every $i = 1, \ldots, T$, the 3-tuple (s_i, r_i, s'_i) corresponds to a transition from state s_i to s'_i under action $\pi(s_i)$ and the resulting reward is denoted by

[*] prashanth.la@inria.fr

[†] nathaniel.korda@eng.ox.ac.uk

[‡] remi.munos@inria.fr

r_i. The LSTD estimate is given as the solution to $\hat{\theta}_T = \bar{A}_T^{-1}\bar{b}_T$, where $\bar{A}_T = \frac{1}{T}\sum_{i=1}^{T}\phi(s_i)(\phi(s_i) - \beta\phi(s_i'))^\intercal$, and $\bar{b}_T = \frac{1}{T}\sum_{i=1}^{T} r_i\phi(s_i)$.

Computing the inverse of the matrix $\bar{A}_T$ is computationally expensive, especially when d is large. Indeed, assuming that the features $\phi(s_i)$ evolve in a compact subset of $\mathbb{R}^d$, the complexity of the above approach is $O(d^2T)$, where $\bar{A}_T^{-1}$ is computed iteratively using the Sherman-Morrison lemma. On the other hand, if we employ the Strassen algorithm or the Coppersmith-Winograd algorithm for computing $\bar{A}_T^{-1}$, the complexity is of the order $O(d^{2.807})$ and $O(d^{2.375})$, respectively, in addition to $O(d^2T)$ complexity for computing $\bar{A}_T$.

A common trick, in practice, to alleviate this problem in high dimensions, is to replace the inversion of the $\bar{A}_T$ matrix by an iterative procedure that performs a fixed point iteration. From a theoretical standpoint, this comes under the purview of stochastic approximation (SA), and one requires that the samples be chosen randomly to ensure convergence. In this paper, we analyse such an SA based scheme and show that it converges to the LSTD solution. The advantage is that the SA based scheme incurs lower computational cost in comparison to the approaches mentioned above. We also analyse a similar low-complexity alternative for the classic least squares parameter estimation problem.

We provide convergence rate results for our proposed method, both in high probability and in expectation. In particular, we show that, with probability $1 - \delta$, the SA based scheme constructs an ϵ-approximation of the corresponding LSTD solution with $O(d\ln(1/\delta)/\epsilon^2)$ complexity, irrespective of the number of samples T. Moreover, we also establish that using the SA based scheme in place of LSTD does not impact the rate of convergence of the approximate value function to the true value function (see Theorem 3).

The rate results coupled with the low complexity of our scheme make it more amenable to practical implementation in the canonical *big data* settings, where both d and T are large. Further, we provide explicit constants in the high probability bounds and we believe this opens several avenues for the use of SA based low complexity alternatives in higher level decision making procedures, for instance, least squares policy iteration (LSPI) and linear bandit algorithms. We demonstrate the practicality of our solution scheme for LSTD empirically by using it as a subroutine in the LSPI algorithm for adaptive traffic signal control[1]. In particular, for the experiments we employ step-sizes that were used to derive the finite-time bounds (see Corollary 2). We demonstrate that this choice results in rapid convergence of our SA based scheme in the experiments and also that the performance of the SA variant of LSPI is comparable to that of LSPI.

The rest of the paper is organized as follows: In Section 1, we review relevant previous works. In Section 2 we present the fast LSTD algorithm based on stochastic approximation and in Section 3 we provide the non-asymptotic bounds for this algorithm. In Section 4, we outline the variants of our algorithm to incorporate regularization and iterate averaging, while in Section 6, we provide extensions to solve the problem of least squares regression. Next, in Section 5, we provide outlines for the proof and derivation of rates. In Section 7, we provide experiments on a traffic signal control application. Finally, in Section 8 we provide the concluding remarks.

1 Related work

In the context of the problem of prediction in RL, temporal difference (TD) learning is a well-known algorithm. See Bertsekas and Tsitsiklis [1996], Sutton and Barto [1998] for a textbook introduction and Tsitsiklis and Van Roy [1997] for an asymptotic analysis. LSTD Bradtke and Barto [1996] is a popular batch algorithm that converges asymptotically to the TD solution. Finite time analysis of LSTD is provided by Lazaric et al. [2012] and we extend it to the case when LSTD solution is replaced by a SA iterate.

A popular line of research in RL is on improving the complexity of TD-like algorithms (cf. GTD Sutton et al. [2009b], GTD2 Sutton et al. [2009a], iLSTD Geramifard et al. [2007] and the references therein). The popular Computer Go with dimension $d = 10^6$ Silver et al. [2007] and several practical applications (e.g. transportation, networks) involve high-feature dimensions. Moreover, considering that linear function approximation is effective with a large number of features, our $O(d)$ improvement in complexity of LSTD by employing SA is meaningful.

[1] See Appendix D for another set of experiments that combines the SA based low-complexity variant for least squares regression with the LinUCB algorithm for contextual bandits, using the large scale news recommendation dataset from Yahoo Webscope [2011].

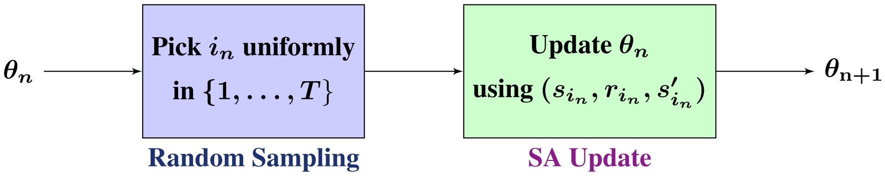

θ_n
Pick i_n uniformly in $\{1,\ldots,T\}$
Random Sampling
Update θ_n using $(s_{i_n}, r_{i_n}, s'_{i_n})$
SA Update
θ_{n+1}

Figure 1: Overall flow of the fLSTD-SA algorithm.

Our algorithms are based on the well-known stochastic approximation technique, originally proposed for finding zeroes of a nonlinear function by Robbins and Monro [1951]. The reader is referred to Kushner and Yin [2003] for a textbook introduction to SA. Iterate averaging is a standard approach to accelerate the convergence of SA schemes and was proposed independently by Ruppert [1991] and Polyak and Juditsky [1992]. Non asymptotic bounds for Robbins Monro schemes have been provided by Frikha and Menozzi [2012] and extended to incorporate iterate averaging by Fathi and Frikha [2013].

In comparison to previous work, we would like to point out that there is no finite time analysis of GTD-type algorithms. While iLSTD is an efficient approximation to LSTD, analysis by Geramifard et al. [2007] requires that the feature matrix be sparse. In contrast, we provide finite-time bounds and do not make any sparsity assumption. To the best of our knowledge, efficient SA algorithms that approximate LSTD without impacting its rate of convergence to true value function, have not been proposed before in the literature. The high probability bounds that we derive for the SA based scheme do not directly follow from earlier work on LSTD algorithms. Further, unlike Frikha and Menozzi [2012], we provide explicit constants in the bounds that we derive (see Corollary 2) and we employ these in our experiments as well.

Stochastic gradient descent (SGD) is a well-known method for optimising a function given only noisy observations. In the context of machine learning, finite time analysis of such methods have been provided by Bach and Moulines [2011]. While the bounds by Bach and Moulines [2011] are given in expectation, many machine learning applications require high probability bounds, which we provide for our case. Regret bounds for online SGD techniques have been given by Zinkevich [2003], Hazan and Kale [2011]: the gradient descent algorithm by Zinkevich [2003] is in the setting of optimising the average of convex loss functions whose gradients are available, while that by Hazan and Kale [2011] is for strongly convex loss functions.

In comparison to previous work w.r.t. least squares regression, we highlight the following differences: **(i)** Earlier works on least squares regression (cf. Hazan and Kale [2011]) require the knowledge of the strong convexity constant in deciding the step-size, while we average the iterates to get rid of this dependency. **(ii)** Our analysis is much simpler (since we work directly with least squares problems) and we make all the constants explicit for the problems considered.

2 Fast LSTD using Stochastic Approximation *(fLSTD-SA)*

We propose here a stochastic approximation variant of the least squares temporal difference (LSTD) algorithm, whose iterates converge to the same fixed point as the regular LSTD algorithm, while incurring much smaller overall computational cost.

The algorithm, which we call Stochastic Algorithm for LSTD Approximation (fLSTD-SA), is a simple stochastic approximation scheme with randomized samples. The results that we present establish that fLSTD-SA computes an ϵ-approximation to the LSTD solution $\hat{\theta}_T$ with probability $1-\delta$, while incurring a complexity of the order $O(d\ln(1/\delta)/\epsilon^2)$, irrespective of the number of samples T. In turn, this enables us to give a performance bound for the approximate value function computed by fLSTD-SA. A schema of fLSTD-SA is given in Figure 1.

Although our analysis for fLSTD-SA depends on a strong convexity assumption that may not hold in all situations, we present also a variant of fLSTD-SA employing iterate averaging for which error bounds can be given without resorting to a strong convexity assumption.

2.1 Background for LSTD

Consider an MDP with state space $\mathcal{S}$, action space $\mathcal{A}$ and transition probabilities $p(s,a,s')$, $s,s' \in \mathcal{S}, a \in \mathcal{A}$. The value function V^π for a given policy π (a mapping from states to actions) is the fixed point of the Bellman operator T^π defined as

$$\mathcal{T}^\pi(s) = r(s,\pi(s)) + \beta \sum_{s'} p(s,\pi(s),s')V^\pi(s'), \tag{1}$$

where $\beta \in (0,1)$ is the discount factor and $r(s,\pi(s'))$ denotes the instantaneous rewards obtained in state s with action $\pi(s)$. When the cardinality of $\mathcal{S}$ is huge and in the absence of knowledge of the transition dynamics, a popular approach is to parameterize the value function using a linear function approximation architecture, i.e., for every $s \in \mathcal{S}$, we approximate $V^\pi(s) \approx \theta^\intercal \phi(s)$, where $\phi(s)$ is a d-dimensional feature vector with $d << |\mathcal{S}|$, and θ is a tunable parameter. The well-known TD learning algorithm Bertsekas and Tsitsiklis [1996] attempts to find the fixed point of the operator ΠT given by

$$\Phi\theta = \Pi\mathcal{T}^\pi(\Phi\theta), \tag{2}$$

where $\mathcal{B} = \{\Phi\theta \mid \theta \in \mathbb{R}^d\}$ is the space within which we want to approximate the value function V^π, Π is the orthogonal projection onto $\mathcal{B}$, and Φ is the feature matrix with rows $\phi(s)^\intercal, \forall s \in \mathcal{S}$ denoting the features corresponding to state $s \in \mathcal{S}$. Let θ^* denote the solution to (2), P the transition probability matrix with components $p(s,\pi(s),s')$ and Ψ the stationary distribution (assuming it exists) of the Markov chain for the underlying policy π. Then, θ^* can be written as the solution to the following system of equations (cf. [Bertsekas, 2012, Section 6.3])

$$A\theta^* = b, \text{ where } A = \Phi^\intercal\Psi(I-\beta P)\Phi \text{ and } b = \Phi^\intercal\Psi r. \tag{3}$$

The LSTD approach is to approximate A and b using T samples $\{(s_i, r_i, s'_i), i = 1, \dots, T)\}$ obtained by simulating the MDP with the underlying policy π.

An approximate solution to (3) is constructed as follows:

$$\hat{\theta}_T = \bar{A}_T^{-1}\bar{b}_T \tag{4}$$

where $\bar{A}_T = \dfrac{1}{T}\sum\limits_{i=1}^{T} \phi(s_i)(\phi(s_i) - \beta\phi(s'_i))^\intercal$, and $\bar{b}_T = \dfrac{1}{T}\sum\limits_{i=1}^{T} r_i\phi(s_i)$. Here $\phi(s_i)$ is a d-dimensional feature vector corresponding to state s_i, for all $i = 1, \dots, T$. By invoking the strong law of large numbers, one can show that $\bar{A}_T \to A$ and $\bar{b}_T \to b$ as the number of samples T tends to infinity.

2.2 Update rule for fLSTD-SA

Starting with an arbitrary θ_0, update the parameter θ_n as follows:

$$\theta_n = \theta_{n-1} + \gamma_n \left(r_{i_n} + \beta\theta_{n-1}^\intercal\phi(s'_{i_n}) - \theta_{n-1}^\intercal\phi(s_{i_n})\right)\phi(s_{i_n}), \tag{5}$$

where each i_n is chosen uniformly randomly from the set $\{1, \dots, T\}$. In other words, we pick a sample with uniform probability $1/T$ from the set $\mathcal{D} := \{(s_i, r_i, s'_i), i = 1, \dots, T)\}$ and use it to perform a fixed point iteration in (5). The quantities γ_n above are *step sizes* that are chosen in advance and satisfy standard stochastic approximation conditions (see (A4) below). Notice that the above update is the usual TD update, except that the samples are drawn uniformly randomly from the sample set $\mathcal{D}$.

3 Main Results

3.1 Error bounds

We make the following assumptions for the analysis fLSTD-SA:
(A1) Bounded features, i.e., $\|\phi(s_i)\|_2 \leq 1$, for $i = 1, \dots, T$.

(A2) Bounded rewards, i.e., $|r_i| \leq R_{\max} < \infty$ for $i = 1, \ldots, T$ and bounded linear space, i.e., $-V_{\max} \leq \Phi\theta \leq V_{\max} < \infty$.

(A3) Writing $\Phi_T \triangleq (\phi(s_1)^\intercal; \ldots; \phi(s_T)^\intercal)$, the covariance matrix $\frac{1}{T}\Phi_T^\intercal\Phi_T$ is positive definite and its smallest (positive) eigenvalue is at least μ.

(A4) The step sizes γ_n satisfy $\sum_n \gamma_n = \infty$, and $\sum_n \gamma_n^2 < \infty$.

By working in a bounded linear space along with bounded rewards and features, along with step sizes that satisfy standard stochastic approximation conditions, we ensure that the parameter θ remains stable, and hence that (5) converges.

Let $z_n := \theta_n - \hat{\theta}_T$ denote the *approximation error* for the algorithm (5), i.e. the error incurred by the n^{th} iterate of our optimization procedure. To obtain high probability bounds on the error we consider separately the deviation of z_n from its mean (see (6) in Theorem 1), and the size of its mean itself (see (7) in Theorem 1). In this way the first quantity can be directly decomposed as a sum of martingale differences, and then a standard martingale concentration argument applied, while the second quantity can be analyzed by directly unrolling iteration (5) (a proof outline is provided in Section 5, while the detailed proofs are available in Appendix A).

Theorem 1. *Under (A1)-(A4), we have* $\forall \epsilon > 0$,

$$P(\left\|\theta_n - \hat{\theta}_T\right\|_2 - \mathbb{E}\left\|\theta_n - \hat{\theta}_T\right\|_2 \geq \epsilon) \leq \exp\left(-\epsilon^2/(2\sum_{i=1}^{n} L_i^2)\right), \tag{6}$$

$$\mathbb{E}\left\|\theta_n - \hat{\theta}_T\right\|_2 \leq \underbrace{\exp(-(1-\beta)\mu\Gamma_n)\left\|\theta_0 - \hat{\theta}_T\right\|_2}_{\textit{initial error}} + \underbrace{\left(\sum_{k=1}^{n-1} H_\beta^2 \gamma_{k+1}^2 \exp(-2(1-\beta)\mu(\Gamma_n - \Gamma_{k+1})\right)^{\frac{1}{2}}}_{\textit{sampling error}}, \tag{7}$$

where $L_i := \gamma_i \prod_{j=i}^{n-1}(1 - 2\gamma_{j+1}\mu((1-\beta) - \beta(2-\beta)\gamma_{j+1}))^{1/2}$, $\Gamma_n := \sum_{i=1}^n \gamma_i$ *and* $H_\beta^2 := R_{\max}(R_{\max}+2) + (1+\beta)^2 V_{\max}^2$.

The initial error depends on the initial point θ_0 of the algorithm. The sampling error arises out of a martingale difference sequence and is the dominant term in (7). Under a suitable choice of step-sizes (see Corollary 2), it can be shown that the initial error is forgotten faster than the sampling error.

The above theorem assumes no specific form for the step-sizes γ_n. Specifying the step-size sequence, we can merge the two claims above to deduce the following bounds on the approximation error z_n with explicit constants:

Corollary 2 (Error Bound for iterates of fLSTD-SA)**.** *Under (A1)-(A4), choosing* $\gamma_n = \frac{(1-\beta)c}{2(c+n)}$ *and c such that* $(1-\beta)^2\mu c \in (1.33, 2)$*, we have, for any* $\delta > 0$*,*

$$\mathbb{E}\left\|\theta_n - \hat{\theta}_T\right\|_2 \leq \frac{K_1}{\sqrt{n+c}} \textit{ and } P\left(\left\|\theta_n - \hat{\theta}_T\right\|_2 \leq \frac{K_2}{\sqrt{n+c}}\right) \geq 1 - \delta, \tag{8}$$

where

$$K_1 := \frac{\sqrt{c}\left\|\theta_0 - \hat{\theta}_T\right\|_2}{n^{((1-\beta)^2\mu c - 1)/2}} + \frac{(1-\beta)cH_\beta}{2} \textit{ and } K_2 := \frac{(1-\beta)c}{2\sqrt{\left(\frac{4}{3}(1-\beta)^2\mu c - 1\right)}}\sqrt{\log\frac{1}{\delta}} + K_1.$$

Remark 1. *We note that setting c such that* $(1-\beta)^2\mu c = \eta \in (1,2)$ *we can rewrite the constants in Corollary 2 as:*

$$K_1 := \frac{\left\|\theta_0 - \hat{\theta}_T\right\|_2}{(1-\beta)\sqrt{\mu n^{(\eta-1)}}} + \frac{H_\beta}{2(1-\beta)\mu} \textit{ and } K_2 := \frac{\sqrt{\log\delta^{-1}}}{2(1-\beta)\mu\sqrt{\left(\frac{4}{3}\eta - 1\right)}} + K_1.$$

So both the bounds in expectation and high probability have a linear dependence on the inverse of $(1-\beta)\mu$.

3.2 Performance Bound

Let $\tilde{v}_T := \Phi\theta_T$ denote the approximate value function and v denote the true value function, evaluated at the states $s_1, \ldots, s_T$. Then the following lower bound on the performance of $\tilde{v}_T$ can be deduced from Corollary 2 in conjunction with Theorem 1 of Lazaric et al. [2012]:

Theorem 3. *Under conditions of Corollary 2, for any $\delta > 0$, with probability $1-\delta$ we have*

$$\|v - \tilde{v}_T\|_T \leq \underbrace{\frac{\|v - \Pi v\|_T}{\sqrt{1-\beta^2}}}_{\textit{residual error}} + \underbrace{O\left(\sqrt{\frac{d}{(1-\beta)^2\mu T}}\right)}_{\textit{estimation error}} + \underbrace{O\left(\sqrt{\frac{1}{(1-\beta)\mu T}\ln\frac{1}{\delta}}\right)}_{\textit{approximation error}},$$

where $\|f\|_T^2 := \frac{1}{T}\sum_{i=1}^{T} f(s_i)^2$, for any function f.

The residual and estimation errors (first and second terms in the RHS above) are artifacts of function approximation and least squares methods, respectively. The third term, of order $O\left(\frac{1}{\sqrt{T}}\right)$, is a consequence of using fLSTD-SA in place of the LSTD. From the above theorem, we observe that using our scheme in place of LSTD does not impact the rate of convergence of the approximate value function $\tilde{v}_T$ to the true value function v. This finding coupled with the fact that our scheme is of low complexity makes it attractive for implementation in *big data* settings, where the feature dimension d is large.

4 Variants

To obtain the best performance from fLSTD-SA we need to know the value of μ. However with minor adjustments to the analysis we can provide two variants of fLSTD-SA for which it is not necessary to know the value of μ to obtain the (optimal) approximation error of order $O(n^{-1/2})$ and explicit constants.

4.1 Regularization.

A popular approach is to search not for the LSTD solution, but instead for a regularized LSTD solution defined as follows:

$$\hat{\theta}_T^{reg} = (\bar{A}_T + \mu I)^{-1}\bar{b}_T \tag{9}$$

where μ is now a constant set in advance. The update rule for this variant is

$$\theta_n^{reg} = (1-\gamma_n\mu)\theta_{n-1} + \gamma_n\left(r_{i_n} + \beta\theta_{n-1}^{\mathsf{T}}\phi(s'_{i_n}) - \theta_{n-1}^{\mathsf{T}}\phi(s_{i_n})\right)\phi(s_{i_n}). \tag{10}$$

This algorithm retains all the properties of the non-regularized fLSTD-SA algorithm, except that it converges to the solution of (9) rather than to that of (4). In particular the conclusions of Theorem 1, and of Corollary 2 hold without requiring assumption (A3), but where $z_n = \theta_n - \hat{\theta}_T^{reg}$ measures the error to the regularized fixed point $\hat{\theta}_T^{reg}$.

4.2 Iterate Averaging.

Here we employ the well-known Polyak-Ruppert scheme of averaging the iterates and coupling it with larger step-sizes. In particular, we fix the step-size $\gamma_n := \frac{(1-\beta)}{2}\left(\frac{c}{c+n}\right)^{\alpha}$, and then use the averaged iterate $\bar{\theta}_{n+1} := (\theta_1 + \ldots + \theta_n)/n$ to approximate the LSTD solution. Here the quantities θ_n are just the iterates of the fLSTD-SA presented earlier. An analogue of Corollary 2 for iterate averaging is as follows (see Appendix B for a detailed proof):

Corollary 4. *Under (A1)-(A2), choosing* $\gamma_n = \frac{(1-\beta)}{2}\left(\frac{c}{c+n}\right)^{\alpha}$*, with* $\alpha \in (1/2, 1)$ *and* $c \in (1.33, 2)$*, we have, for any* $\delta > 0$*,*

$$\mathbb{E}\left\|\bar{\theta}_n - \hat{\theta}_T\right\|_2 \leq \frac{K_1^{IA}}{(n+c)^{\alpha/2}} \text{ and } P\left(\left\|\bar{\theta}_n - \hat{\theta}_T\right\|_2 \leq \frac{K_2^{IA}}{(n+c)^{\alpha/2}}\right) \geq 1-\delta, \tag{11}$$

where, writing $C = \sum_{n=1}^{\infty} \exp(-\mu c n^{1-\alpha}) (< \infty)$*,*

$$K_1^{IA} := \frac{C\left\|\theta_0 - \hat{\theta}_T\right\|_2}{(n+c)^{(1-\alpha)/2}} + \frac{H_\beta c^\alpha (1-\beta)}{(\mu c^\alpha (1-\beta)^2)^{\alpha \frac{1+2\alpha}{2(1-\alpha)}}}, \text{ and}$$

$$K_2^{IA} := \frac{\sqrt{\log \delta^{-1}}}{\mu(1-\beta)}\left[3^\alpha + \left[\frac{2\alpha}{\mu c^\alpha (1-\beta)^2} + \frac{2^\alpha}{\alpha}\right]^2\right]\frac{1}{(n+c)^{(1-\alpha)/2}} + K_1^{IA}.$$

Thus, it is possible to remove the dependency on the knowledge of μ for the choice of c through averaging of the iterates, at the cost of $(1-\alpha)/2$ in the rate. However, choosing α close to 1 causes a sampling error blowup. As suggested by earlier works on stochastic approximation, it is preferred to average after a few iterations since the initial error $\|\theta_0 - \theta_T\|_2$ is not forgotten exponentially fast with averaging.

5 Outline of analysis

In this section we give outline proofs of the main results concerning the fLSTD-SA algorithm. We split these into two sections: first, we sketch the martingale analysis that leads to the proof of Theorem 1 and which forms the template for the proof for extension to least squares regression (see Theorem 10 in Appendix C) and the regularized and iterate averaged variants of fLSTD-SA (see Corollary 4); second, we give the derivation of the rates when the step sizes a chosen in specific forms.

5.1 Outline of Theorem 1 proof

Recall that Theorem 1 decomposes the problem of bounding the approximation error $z_n := \theta_n - \hat{\theta}_T$ into bounding the deviation of z_n from its mean in high probability and then bounds the mean of z_n itself. In the following, we first provide a sketch of the proof of high probability bound and later outline the proof for the bound in expectation. For the former, we employ a proof technique similar to that used in Frikha and Menozzi [2012]. However, our analysis is much simpler and we make all the constants explicit for the problem at hand. Moreover, in order to eliminate a possible exponential dependence of the constants in the resulting bound on the inverse of $(1-\beta)\mu$, we depart from the argument in Frikha and Menozzi [2012].

High probability bound. *(Sketch)* Recall that $z_n := \theta_n - \hat{\theta}_T$. We rewrite $\|z_n\|_2^2 - E\|z_n\|_2^2$ as a telescoping sum of martingale differences:

$$\|z_n\|_2 - \mathbb{E}\|z_n\|_2 = \sum_{i=1}^{n} g_i - \mathbb{E}[g_i \,|\mathcal{F}_{i-1}] = \sum_{i=1}^{n} D_i,$$

where $D_i \triangleq g_i - \mathbb{E}[g_i \,|\mathcal{F}_{i-1}]$, $g_i = \mathbb{E}[\|z_n\|_2 \,|\theta_i]$, and $\mathcal{F}_i$ denotes the sigma algebra generated by the random variables $\{i_1, \ldots, i_n\}$.

The next step is to show that the functions g_i are Lipschitz continuous in the rewards, with Lipschitz constants L_i. In order to obtain constants with no exponential dependence on the inverse of $(1-\beta)\mu$ we depart from the general scheme of Frikha and Menozzi [2012], and use our knowledge of the form of the update function f_i to eliminate the noise due to the rewards between time $i+1$ and time n. Specifically, letting $\Theta^i_j(\theta)$ denote the mapping that returns the value of the iterate θ_j at instant j, given that $\theta_i = \theta$, we show that

$$\begin{aligned}\mathbb{E}\left[\left\|\Theta^i_n(\theta) - \Theta^i_n(\theta')\right\|_2^2\right] &= \mathbb{E}\left[\mathbb{E}\left([I - \gamma_n[\phi(s_{i_n})\phi(s_{i_n})^\intercal - \beta\phi(s_{i_n})\phi(s'_{i_n})^\intercal]]\right.\right.\\ &\qquad\qquad \left.\left..(\Theta^i_{n-1}(\theta) - \Theta^i_{n-1}(\theta')) \mid \Theta^i_{n-1}(\theta), \Theta^i_{n-1}(\theta')\right)\right]\\ &\le (1 - \gamma_n\mu(1 - \beta - \gamma_n\beta(2-\beta)))\mathbb{E}\left[\left\|\Theta^i_{n-1}(\theta) - \Theta^i_{n-1}(\theta')\right\|_2^2\right],\end{aligned}$$

where we used the specific form of f_i in obtaining the equality, and have applied assumption (A3) to obtain the inequality. Unrolling this iteration then yields the new Lipschitz constants.

Now we can invoke a standard martingale concentration bound: Using the L_i-Lipschitz property of the g_i functions and the assumption (A2) we find that

$$P(\left\|z_n\right\|_2 - \mathbb{E}\left\|z_n\right\|_2 \ge \epsilon) = P\left(\sum_{i=1}^n D_i \ge \epsilon\right) \le \exp(-\lambda\epsilon)\exp\left(\frac{\alpha\lambda^2}{2}\sum_{i=1}^n L_i^2\right).$$

The claim follows by optimizing the above over λ. The full proof is available in Appendix A.1. □

Bound in expectation. (*Sketch*) First we extract a martingale difference from the update rule (5): Recall that $z_n := \theta_n - \hat\theta_T$. Let $f_n(\theta) := (\theta^\intercal x_{i_n} - (r_{i_n} + \beta\theta^\intercal x'_{i_n}))x_{i_n}$ and let $F(\theta) := \mathbb{E}_{i_n}(f_n(\theta))$. Then, we have

$$z_n = \theta_n - \hat\theta_T = \theta_{n-1} - \hat\theta_T - \gamma_n\left(F(\theta_{n-1}) - \Delta M_n\right),$$

where $\Delta M_{n+1}(\theta) = F_n(\theta) - f_n(\theta)$ is a martingale difference. Now since $\hat\theta_T$ is the LSTD solution, $F(\hat\theta_T)) = 0$. Moreover, $F(\cdot)$ is linear, and so we obtain

$$z_n = z_{n-1} - \gamma_n\left(z_{n-1}\bar A_n - \Delta M_n\right) = \Pi_n z_0 - \sum_{k=1}^n \gamma_k \Pi_n \Pi_k^{-1}\Delta M_k,$$

where $\bar A_n = \dfrac{1}{n}\sum\limits_{i=1}^n x_i(x_i - \beta x'_i)^\intercal$ and $\Pi_n := \prod_{k=1}^n\left(I - \gamma_k\bar A_k\right)$.
By Jensen's inequality, we obtain

$$\mathbb{E}(\|z_n\|_2) \le (\mathbb{E}(\langle z_n, z_n\rangle))^{\frac{1}{2}} = \left(\mathbb{E}\left\|\Pi_n z_0\right\|_2^2 + \sum_{k=1}^n \gamma_k^2\mathbb{E}\left\|\Pi_n\Pi_k^{-1}\Delta M_k\right\|_2^2\right)^{\frac{1}{2}} \qquad (12)$$

The rest of the proof amounts to bounding the martingale difference ΔM_n as follows:

$$\mathbb{E}[\|\Delta M_n\|_2^2] \le \mathbb{E}_{i_t}\langle f_{i_t}(\theta_{t-1}), f_{i_t}(\theta_{t-1})\rangle \le R_{\max}(R_{\max}+2) + (1+\beta)^2\left\|\theta_{t-1}\right\|_2^2 \le H_\beta^2.$$

□

5.2 Derivation of rates

Now we give the proof of Corollary 2, which gives explicitly the rate of convergence of the approximation error in high probability for the specific choice of step sizes $\gamma_n = \frac{1-\beta}{2}\frac{c}{c+n}$:

Proof of Corollary 2: Note that when $\gamma_n = \frac{(1-\beta)c}{2(c+n)}$,

$$
\begin{aligned}
\sum_{i=1}^{n} L_i^2 &= \sum_{i=1}^{n} \frac{(1-\beta)^2 c^2}{4(c+i)^2} \prod_{j=i}^{n} \left(1 - 2\mu \frac{(1-\beta)c}{2(c+n)}((1-\beta) - \beta(2-\beta)\frac{(1-\beta)c}{2(c+n)})\right) \\
&\leq \sum_{i=1}^{n} \frac{(1-\beta)^2 c^2}{4(c+i)^2} \exp\left(-\frac{3}{4}(1-\beta)^2 \mu c \sum_{j=i}^{n} \frac{1}{(c+n)}\right) \\
&\leq \frac{(1-\beta)^2 c^2}{4(n+c)^{\frac{3}{4}(1-\beta)^2\mu c}} \sum_{i=1}^{n} (i+c)^{-(2-\frac{3}{4}(1-\beta)^2\mu c)}.
\end{aligned}
$$

We now find three regimes for the rate of convergence, based on the choice of c:
(i) $\sum_{i=1}^{n} L_i^2 = O\left((n+c)^{\frac{3}{4}(1-\beta)^2\mu c}\right)$ when $\frac{3}{4}(1-\beta)^2\mu c \in (0,1)$,
(ii) $\sum_{i=1}^{n} L_i^2 = O\left(n^{-1}\ln n\right)$ when $\frac{3}{4}(1-\beta)^2\mu c = 1$, and
(iii) $\sum_{i=1}^{n} L_i^2 = \frac{(1-\beta)^2 c^2}{4(\frac{3}{4}(1-\beta)^2\mu c - 1)}(n+c)^{-1}$ when $\frac{3}{4}(1-\beta)^2\mu c \in (1,2)$.
(We have used comparisons with integrals to bound the summations.) Thus, setting $2/((1-\beta)^2\mu) > c > 1/((1-\beta)^2\mu)$, the high probability bound from Theorem 1 gives

$$
P(\left\|\theta_n - \hat{\theta}_T\right\|_2 - \mathbb{E}\left\|\theta_n - \hat{\theta}_T\right\|_2 \geq \epsilon) \leq \exp\left(-\frac{\epsilon^2(n+c)}{2K_{\mu,c,\beta}}\right) \tag{13}
$$

where $K_{\mu,c,\beta} := \frac{(1-\beta)^2c^2}{4((1-\beta)^2\mu c - 1)}$.

Under the same choice of step-size, the bound in expectation in Theorem 1 we have:

$$
\begin{aligned}
&\sum_{k=1}^{n-1} H_\beta^2 \gamma_{k+1}^2 \exp(-2(1-\beta)\mu(\Gamma_n - \Gamma_{k+1})) \\
&\qquad \leq \frac{(1-\beta)^2 c^2 H_\beta^2}{4}(n+c)^{-(1-\beta)^2\mu c} \sum_{k=1}^{n} (c+k)^{-(2-(1-\beta)^2\mu c)} \\
&\qquad \leq \frac{(1-\beta)^2 c^2 H_\beta^2}{4}(n+c)^{-1}
\end{aligned}
$$

we in the last inequality we have again compared the sum with an integral. Similarly

$$
\exp(-(1-\beta)\mu\Gamma_n) \leq \left(\frac{c}{n+c}\right)^{\frac{(1-\beta)^2\mu c}{2}} \leq \left(\frac{c}{n+c}\right)^{\frac{1}{2}}.
$$

So we have

$$
\mathbb{E}\left\|\theta_n - \hat{\theta}_T\right\|_2 \leq \left(\sqrt{c}\left\|\theta_0 - \theta^*\right\|_2 + \frac{(1-\beta)cH_\beta}{2}\right) n^{-\frac{1}{2}}, \tag{14}
$$

and the result now follows. □

6 Extension to Least Squares Regression

In this section, we describe the classic parameter estimation problem using the method of least squares, the standard approach to solve this problem and a low-complexity alternative using stochastic approximation.

In this setting, we are given a set of samples $\mathcal{D} := \{(x_i, y_i), i = 1, \ldots, T\}$ with the underlying observation model $y_i = x_i^\intercal \theta^* + \xi_i$ (ξ_i is zero mean and bounded noise, and θ^* is an unknown parameter). The least squares estimate $\hat{\theta}_T$ minimizes $\frac{1}{2}\sum\limits_{i=1}^{T}(y_i - \theta^\intercal x_i)^2$. It can be shown that $\hat{\theta}_T = \bar{A}_T^{-1} b_T$, where $\bar{A}_T = \dfrac{1}{T}\sum\limits_{i=1}^{T} x_i x_i^\intercal$ and $\bar{b}_T = \dfrac{1}{T}\sum\limits_{i=1}^{T} x_i y_i$.

Notice that, unlike the RL setting, $\hat{\theta}_T$ here is the minimizer of an empirical loss function. However, as in the case of LSTD, the computational cost for a Sherman-Morrison lemma based approach for solving the above would be of the order $O(d^2T)$. Similarly to the case of the fLSTD-SA algorithm, we update the iterate θ_n using a SA scheme as follows (starting with an arbitrary θ_0),

$$\theta_n = \theta_{n-1} + \gamma_n (y_{i_n} - \theta_{n-1}^\intercal x_{i_n}) x_{i_n}, \tag{15}$$

where, as before, each i_n is chosen uniformly randomly from the sample set $\mathcal{D}$ and γ_n are step-sizes.

Unlike fLSTD-SA which is a fixed point iteration, the above is a stochastic gradient descent procedure. Nevertheless, using the same proof template as for fLSTD-SA earlier, we can derive bounds on the approximation error, i.e., the distance between θ_n and least squares solution $\hat{\theta}_T$, both in high probability as well as expectation.

Results. As in the case of fLSTD-SA, we assume that the features are bounded, the noise is i.i.d, zero-mean and bounded and the matrix $\bar{A}_T$ is positive definite, with smallest eigenvalue at least $\mu > 0$. An analogue of Corollary 2 for this setting is as follows (See Appendix C for a detailed proof.):

Corollary 5. *Choosing $\gamma_n = \frac{c}{2(c+n)}$ and c such that $\mu c \in (1.33, 2)$, we have, for any $\delta > 0$,*

$$\mathbb{E}\left\|\theta_n - \hat{\theta}_T\right\|_2 \leq \frac{K_1^{LS}}{\sqrt{n+c}} \text{ and } P\left(\left\|\theta_n - \hat{\theta}_T\right\|_2 \leq \frac{K_2^{LS}}{\sqrt{n+c}}\right) \geq 1 - \delta,$$

$$\text{with } K_1^{LS} := \frac{\sqrt{c}\left\|\theta_0 - \hat{\theta}_T\right\|_2}{(n+c)^{(\mu c - 1)/2}} + \frac{h(n)}{2},\ K_2^{LS} := \frac{\sqrt{c}}{\sqrt{((\mu c)/2 - 1)}}\sqrt{\log\frac{1}{\delta}} + K_1,$$

$$\text{and } h(n) := c\left[\left(Var(\xi_{i_n}) + 2\left\|\theta_0 - \hat{\theta}_T\right\|_2^2\right) + 4\left\|\theta_0 - \hat{\theta}_T\right\|_2 \ln n + 2\ln^2 n\right].$$

7 Traffic Control Application

LSPI Lagoudakis and Parr [2003] is a well-known algorithm for control and is based on the policy iteration procedure for MDPs. It performs policy evaluation and policy improvement in tandem. For the purpose of policy evaluation, LSPI uses a LSTD-like algorithm called LSTDQ, which learns the state-action value function. In contrast, LSTD learns the state value function. We now briefly describe LSTDQ and its fast SA variant fLSTDQ-SA: We are given a set of samples $\mathcal{D} := \{(s_i, a_i, r_i, s_i'), i = 1, \ldots, T)\}$, where each sample i denotes a one-step transition of the MDP from state s_i to s_i' under action a_i, while resulting in a reward r_i. LSTDQ attempts to approximate the Q-value function for any policy π by solving the linear system $\hat{\theta}_T = \bar{A}_T^{-1}\bar{b}_T$, where $\bar{A}_T = \frac{1}{T}\sum_{i=1}^{T}\phi(s_i, a_i)(\phi(s_i, a_i) - \beta\phi(s_i', \pi(s_i')))^\intercal$, and $\bar{b}_T = \frac{1}{T}\sum_{i=1}^{T} r_i \phi(s_i, a_i)$.
fLSTDQ-SA approximates LSTDQ by an iterative update scheme as follows (starting with an arbitrary θ_0):

$$\theta_k = \theta_{k-1} + \gamma_k \left(r_{i_k} + \beta\theta_{k-1}^\intercal \phi(s_{i_k}', \pi_n(s_{i_k}')) - \theta_{k-1}^\intercal \phi(s_{i_k}, a_{i_k})\right) \phi(s_{i_k}, a_{i_k}) \tag{16}$$

From Section 2, it is evident that the claims in Theorem 1 and Corollary 2 hold for the above scheme as well.

The idea behind the experimental setup is to study both LSPI and a variant of LSPI, referred to as fLSPI-SA, where we use fLSTDQ-SA as a subroutine to approximate the LSTDQ solution. Algorithm 1 provides the pseudo-code for the latter algorithm.

We consider a traffic signal control application for conducting the experiments. The problem here is to adaptively choose the sign configurations for the signalized intersections in the road network considered, in order to

Algorithm 1 fLSPI-SA

```
**Input:** Sample set $D := \{s_i, a_i, r_i, s'_i\}_{i=1}^{T}$, obtained from an initial (arbitrary) policy
**Initialisation:** $\epsilon$, $\tau$, step-sizes $\{\gamma_k\}_{k=1}^{\tau}$, initial policy $\pi_0$ (given as $\theta_0$)
$\pi \leftarrow \pi_0$, $\theta \leftarrow \theta_0$
repeat
    Policy Evaluation
        Approximate LSTDQ($D, \pi$) using fLSTDQ-SA($D, \pi$) as follows:
        for $k = 1 \ldots \tau$ do
            Get random sample index: $i_k \sim U(\{1, \ldots, T\})$
            Update fLSTD-SA iterate $\theta_k$ using (16)
        end for
    $\theta' \leftarrow \theta_\tau$, $\Delta = \|\theta - \theta'\|_2$
    Policy Improvement
        Obtain a greedy policy $\pi'$ as follows: $\pi'(s) = \arg\max_{a \in \mathcal{A}} \theta'^{\mathsf{T}} \phi(s, a)$
    $\theta \leftarrow \theta'$, $\pi \leftarrow \pi'$
until $\Delta < \epsilon$
```

maximize the traffic flow in the long run. Let L be the total number of lanes in the road network considered. Further, let $q_i(t), i = 1, \ldots, L$ denote the queue lengths and $t_i(t), i = 1, \ldots, L$ the elapsed time (since signal turned to red) on the individual lanes of the road network. Following Prashanth and Bhatnagar [2011], the traffic signal control MDP is formulated as follows:

State $x_t = \big(q_1(t), \ldots, q_L(t), t_1(t), \ldots, t_L(t)\big)$,

Action a_t belongs to the set of feasible sign configurations,

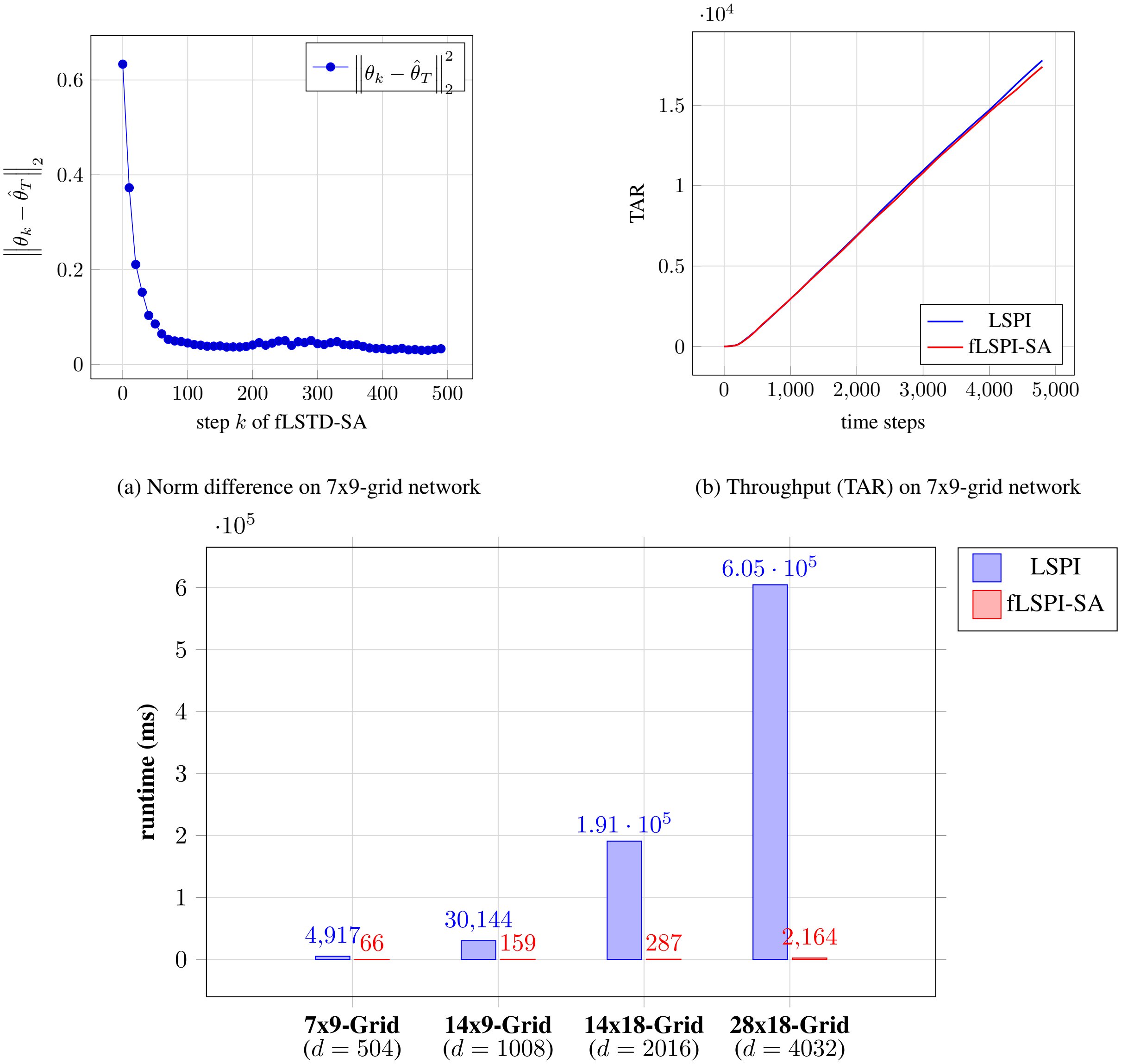
Single-stage cost $h(x_t) = u_1 \left[\sum_{i \in I_p} u_2 \cdot q_i(t) + \sum_{i \notin I_p} w_2 \cdot q_i(t)\right] + w_1 \left[\sum_{i \in I_p} u_2 \cdot t_i(t) + \sum_{i \notin I_p} w_2 \cdot t_i(t)\right]$, where $u_i, w_i \geq 0$ such that $u_i + w_i = 1$ for $i = 1, 2$ and $u_2 > w_2$. Here, the set I_p is the set of prioritized lanes.

Function approximation is a standard technique employed to handle high-dimensional state spaces (as is the case with the traffic signal control MDP on large road networks). We employ the feature selection scheme from Prashanth and Bhatnagar [2012], which is briefly described in the following: The features $\phi(s, a)$ corresponding to any state-action tuple (s, a) is a L-dimensional vector, with one bit for each line in the road network. The feature value $\phi_i(s, a), i = 1, \ldots, L$ corresponding to lane i is chosen as described in Table. 1, with q_i and t_i denoting the queue length and elapsed times for lane i. Thus, as the size of the network increases, the feature dimension scales in a linear fashion.

Note that the above feature selection scheme depends on certain thresholds $\mathcal{L}_1$ and $\mathcal{L}_2$ on the queue length and $\mathcal{T}_1$ on the elapsed times. The motivation for using such graded thresholds is owing to the fact that queue lengths are difficult to measure precisely in practice. We set $(\mathcal{L}_1, \mathcal{L}_2, \mathcal{T}_1) = (6, 14, 130)$ in all our experiments and this choice has been used, for instance, in Prashanth and Bhatnagar [2012].

We implement both LSPI as well as fLSPI-SA for the above problem. We collect $T = 10000$ samples from a exploratory policy that picks the actions in a uniformly random manner. For both LSPI and fLSPI-SA, we set $\beta = 0.9$ and $\epsilon = 0.1$. For fLSPI-SA, we set $\tau = 500$ steps. This choice is motivated by an experiment where we observed that at 500 steps, fLSTD-SA is already very close to LSTDQ and taking more steps did not result in any significant improvements for fLSPI-SA. We implement the regularized variant of LSTDQ, with regularization constant μ set to 1. Motivated by Corollary 2, we set the step-size $\gamma_k = \dfrac{(1-\beta)c}{2(c+k)}$, with $c = \dfrac{1.33}{(1-\beta)^2}$.

Results. We report the norm differences, total arrived road users (TAR) and run-times obtained from our experimental runs in Figs. 2a–2c. Norm difference measures the distance in ℓ^2 norm between the fLSTD-SA iterate θ_k, $k = 1, \ldots, \tau$ and LSTDQ solution $\hat{\theta}_T$ in iteration 1 of fLSPI-SA. TAR is a throughput metric that denotes the total



(a) Norm difference on 7x9-grid network

(b) Throughput (TAR) on 7x9-grid network

(c) Run-times on four road networks

Figure 2: Norm difference, throughput and runtime performance of LSPI and fLSPI-SA

Table 1: Feature selection

State	**Action**	**Feature** $\phi_i(s, a)$
$q_i < \mathcal{L}_1$ and $t_i < \mathcal{T}_1$	RED GREEN	0.01 0.06
$q_i < \mathcal{L}_1$ and $t_i \geq \mathcal{T}_1$	RED GREEN	0.02 0.05
$\mathcal{L}_1 \leq q_i < \mathcal{L}_2$ and $t_i < \mathcal{T}_1$	RED GREEN	0.03 0.04
$\mathcal{L}_1 \leq q_i < \mathcal{L}_2$ and $t_i \geq \mathcal{T}_1$	RED GREEN	0.04 0.03
$q_i \geq \mathcal{L}_2$ and $t_i < \mathcal{T}_1$	RED GREEN	0.05 0.02
$q_i \geq \mathcal{L}_2$ and $t_i \geq \mathcal{T}_1$	RED GREEN	0.06 0.01

number of road users who have reached their destination. The choice 1 of the iteration in Fig 2a is arbitrary, as we observed that fLSTD-SA iterate θ_τ is close to the corresponding LSTDQ solution in each iteration of fLSPI-SA. The runtime reports in Fig. 2c are for four different road networks of increasing size and hence, increasing feature dimension.

From Fig. 2a, we observe that fLSTD-SA algorithm converges rapidly to the corresponding LSTDQ solution. Further, from the runtime plots (see Fig. 2c), we notice that fLSPI-SA is several orders of magnitude faster than regular LSPI. From a traffic application standpoint, we observe in Fig. 2b that fLSPI-SA results in a throughput (TAR) performance that is on par with LSPI.

8 Conclusions

We analysed a stochastic approximation based algorithm with randomised samples for policy evaluation by the method of LSTD. We provided convergence rate results for this algorithm, both in high probability and in expectation. Further, we also established that using this scheme in place of LSTD does not impact the rate of convergence of the approximate value function to the true value function. This result coupled with the fact that the SA based scheme possesses lower computational complexity in comparison to traditional techniques, makes it attractive for implementation in *big data* settings, where the feature dimension is large. On a traffic signal control application, we demonstrated the practicality of a low-complexity alternative to LSPI that uses our SA based scheme in place of LSTDQ for policy evaluation.

A Full Proofs for fLSTD-SA

Recall that fLSTD-SA is a stochastic approximation scheme with randomized samples, using the following update rule (starting with an arbitrary θ_0):

$$\theta_n = \theta_{n-1} + \gamma_n \left(r_{i_n} + \beta \theta_{n-1}^\intercal \phi(s'_{i_n}) - \theta_{n-1}^\intercal \phi(s_{i_n}) \right) \phi(s_{i_n}), \tag{17}$$

where each i_n is chosen uniformly randomly from the set $\{1, \dots, T\}$.

In the following, we present the proof of the Theorem 1 that bounds the approximation error $z_n := \theta_n - \hat{\theta}_T$ in high probability as well as in expectation. Recall that $\hat{\theta}_T$ denotes the LSTD solution.

A.1 Proof of Theorem 1: High probability bound

Proof. Recall that $z_n := \theta_n - \hat{\theta}_T$. First, we rewrite $\|z_n\|_2^2 - E\|z_n\|_2^2$ as a telescoping sum of martingale differences:

$$\|z_n\|_2 - \mathbb{E}\|z_n\|_2 = \sum_{i=1}^{n} g_i - \mathbb{E}[g_i \,|\mathcal{F}_{i-1}] = \sum_{i=1}^{n} D_i, \tag{18}$$

where $D_i \triangleq g_i - \mathbb{E}[g_i \,|\mathcal{F}_{i-1}]$, $g_i = \mathbb{E}[\|z_n\|_2 \,|\theta_i]$, and $\mathcal{F}_i$ denotes the sigma algebra generated by the random variables $\{i_1, \dots, i_n\}$.

The proof is given through three lemmas. The first lemma is a technical requirement to establish a monotonicity property of the limit function $F(\cdot)$ (see Lemma 6 for a precise definition). The next lemma establishes that the functions g_i are Lipschitz continuous with Lipschitz constants L_i. This is a crucial ingredient to invoke the concentration bound in Lemma 7.

Lemma 6. *Conditioned on $\mathcal{F}_{i-1}$, the functions g_i are Lipschitz continuous in the random innovation $f_i(\theta_{i-1})$, with constants*

$$L_i := \gamma_i \left[\prod_{j=i+1}^{n} (1 - 2\gamma_j \mu (1 - \beta - \frac{\gamma_j}{2} \beta (2 - \beta))) \right]^{1/2}.$$

Proof. Denote $f_m(\theta) := (\theta^\intercal \phi(s_{i_m}) - (r_{i_m} + \beta \theta^\intercal \phi(s_{i_{m+1}})))\phi(s_{i_m})$. Let $\Theta_j^i(\theta)$ denote the mapping that returns the value of the iterate θ_j at instant j, given that $\theta_i = \theta$.

$$\begin{aligned}
\Theta_{j+1}^i(\theta) - \Theta_{j+1}^i(\theta') &= \Theta_j^i(\theta) - \Theta_j^i(\theta') - \gamma_{j+1}[f_{j+1}(\Theta_j^i(\theta)) - f_{j+1}(\Theta_j^i(\theta'))] \\
&= \Theta_j^i(\theta) - \Theta_j^i(\theta') - \gamma_{j+1}[\phi(s_{i_{j+1}})\phi(s_{i_{j+1}})^\intercal - \beta \phi(s_{i_{j+1}})\phi(s'_{i_{j+1}})^\intercal](\Theta_j^i(\theta) - \Theta_j^i(\theta')) \\
&= [I - \gamma_{j+1}[\phi(s_{i_{j+1}})\phi(s_{i_{j+1}})^\intercal - \beta \phi(s_{i_{j+1}})\phi(s'_{i_{j+1}})^\intercal]](\Theta_j^i(\theta) - \Theta_j^i(\theta')) \qquad (19)
\end{aligned}$$

The second equality follows from the definition of f_j. Let $a_{j+1} := [\phi(s_{i_{j+1}})\phi(s_{i_{j+1}})^\intercal - \beta \phi(s_{i_{j+1}})\phi(s'_{i_{j+1}})^\intercal]$. Then note that

$$\begin{aligned}
a_{j+1}^\intercal a_{j+1} &= \phi(s_{i_j+1})\phi(s_{i_j+1})^\intercal \phi(s_{i_j+1})\phi(s_{i_j+1})^\intercal \\
&\quad - 2\beta \phi(s_{i_j+1})\phi(s_{i_j+1})^\intercal \phi(s_{i_j+1})\phi(s'_{i_j+1})^\intercal + \beta^2 \phi(s'_{i_j+1})\phi(s_{i_j+1})^\intercal \phi(s_{i_j+1})\phi(s'_{i_j+1})^\intercal \\
&= \left\|\phi(s_{i_j+1})\right\|_2^2 \phi(s_{i_{j+1}})\phi(s_{i_{j+1}})^\intercal - \beta(2 - \left\|\phi(s_{i_{j+1}})\right\|_2^2 \beta)\phi(s'_{i_{j+1}})\phi(s'_{i_{j+1}})^\intercal,
\end{aligned}$$

where the in the first inequality we have used that for two column vectors of equal dimension, x and y, $(xy^\intercal)^\intercal =$

$yx^\intercal$, and $(xx^\intercal)^\intercal = xx^\intercal$. Setting $\Delta_j = diag(\|\phi(s_1)\|_2^2, \dots, \|\phi(s_j)\|_2^2)$ we find that for any vector θ:

$$
\begin{aligned}
&\theta^\intercal \mathbb{E}_{i_{j+1}}(I - 2\gamma_{j+1}[a_{j+1} - \frac{\gamma_{j+1}}{2} a_{j+1}^\intercal a_{j+1}])\theta && (20)\\
&= \|\theta\|_2^2 - 2\gamma_{j+1}\frac{1}{T}\theta^\intercal \Phi_j^\intercal (I - \beta \hat{P} - \frac{\gamma_{j+1}}{2}(\Delta_j - \beta \hat{P}_j^\intercal (2I_j - \beta\Delta_j)\hat{P}_j))\Phi_j \theta && (21)\\
&= \|\theta\|_2^2 - 2\gamma_{j+1}\frac{1}{T}\theta^\intercal \Phi_j^\intercal (I - \beta \Pi \hat{P} - \frac{\gamma_{j+1}}{2}(\Delta_j - \beta \hat{P}_j^\intercal \Pi_j^\intercal (2I_j - \beta\Delta_j)\Pi_j \hat{P}_j))\Phi_j \theta && (22)\\
&\le \|\theta\|_2^2 - 2\gamma_{j+1}\frac{1}{T}(\|\Phi\theta\|_2^2 - \beta \|\Phi\theta\|_2 \left\|\Pi\hat{P}\Phi\theta\right\|_2 - \frac{\gamma_{j+1}}{2}\beta(2-\beta)\left\|\Pi\hat{P}\Phi\theta\right\|_2^2) && (23)\\
&\le \|\theta\|_2^2 - 2\frac{(\gamma_{j+1}(1 - \beta - \frac{\gamma_{j+1}}{2}\beta(2-\beta)))}{T}\|\Phi\theta\|_2^2 && (24)\\
&\le (1 - 2\gamma_{j+1}\mu(1 - \beta - \frac{\gamma_{j+1}}{2}\beta(2-\beta)))\|\theta\|_2^2, && (25)
\end{aligned}
$$

where (22) follows from the fact that $\theta^\intercal \Phi^\intercal D(I - \Pi)x = 0$ since Π is a projection, (23) by an application of Cauchy-Schwarz inequality and (24) from the non-expansiveness property of Π and $\hat{P}$. The final inequality (25) follows from (A3). Hence, from the tower property of conditional expectations, it follows that:

$$
\begin{aligned}
\mathbb{E}\left[\left\|\Theta_n^i(\theta) - \Theta_n^i(\theta')\right\|_2^2\right] &= \mathbb{E}\left[\mathbb{E}\left(\left\|\Theta_n^i(\theta) - \Theta_n^i(\theta')\right\|_2^2 \mid \Theta_{n-1}^i(\theta), \Theta_{n-1}^i(\theta')\right)\right]\\
&\le (1 - 2\gamma_n\mu(1 - \beta - \frac{\gamma_n}{2}\beta(2-\beta)))\mathbb{E}\left[\left\|\Theta_{n-1}^i(\theta) - \Theta_{n-1}^i(\theta')\right\|_2^2\right]\\
&\le \left[\prod_{j=i+1}^{n}(1 - 2\gamma_j\mu(1 - \beta - \frac{\gamma_j}{2}\beta(2-\beta)))\right]\|\theta - \theta'\|_2^2
\end{aligned}
$$

Finally we have

$$
\begin{aligned}
&\left\|\mathbb{E}\left[\left\|\theta_n - \hat{\theta}_T\right\|_2 |\theta_{i-1}, f_{i_i} = f\right] - \mathbb{E}\left[\left\|\theta_n - \hat{\theta}_T\right\|_2 |\theta_{i-1}, f_{i_i} = f'\right]\right\|_2\\
&\quad\le \mathbb{E}\left[\left\|\Theta_n^i(\theta) - \Theta_n^i(\theta')\right\|_2\right] \le \left[\prod_{j=i+1}^{n}(1 - 2\gamma_j\mu(1 - \beta - \frac{\gamma_j}{2}\beta(2-\beta)))\right]^{\frac{1}{2}}\gamma_i\|f - f'\|_2 = L_i\|f - f'\|_2.
\end{aligned}
$$

□

In the following lemma, we invoke a standard martingale concentration bound using the L_i-Lipschitz property of the g_i functions and the assumption (A2).

Lemma 7. *Under the conditions of Theorem 1, we have*

$$
P(\|z_n\|_2 - \mathbb{E}\|z_n\|_2 \ge \epsilon) \le \exp(-\lambda\epsilon)\exp\left(\frac{\alpha\lambda^2}{2}\sum_{i=1}^{n} L_i^2\right). \tag{26}
$$

Proof.

$$
\begin{aligned}
P(\|z_n\|_2 - \mathbb{E}\|z_n\|_2 \ge \epsilon) &= P\left(\sum_{i=1}^{n} D_i \ge \epsilon\right) && (27)\\
&\le \exp(-\lambda\epsilon)\mathbb{E}\left(\exp\left(\lambda\sum_{i=1}^{n} D_i\right)\right) && (28)\\
&= \exp(-\lambda\epsilon)\mathbb{E}\left(\exp\left(\lambda\sum_{i=1}^{n-1} D_i\right)\mathbb{E}\Big(\exp(\lambda D_n)\,|\mathcal{F}_{n-1}\Big)\right). && (29)
\end{aligned}
$$

The first equality above follows from (18), while the inequality follows from Markov inequality. Now for any bounded random variable f, and L-Lipschitz function g we have

$$\mathbb{E}\left(\exp(\lambda g(f))\right) \leq \exp\left(\lambda^2 L^2/2\right).$$

Note that each $f_i(\theta_{i-1})$ is a bounded random variable by (A2), and, conditioned on $\mathcal{F}_{i-1}$, g_i is Lipschitz in $f_i(\theta_{i-1})$ with constant L_i (Lemma 6). So we obtain

$$\mathbb{E}\left(\exp(\lambda D_n)\left|\mathcal{F}_{n-1}\right.\right) \leq \exp\left(\frac{\lambda^2 L_n^2}{2}\right), \tag{30}$$

and so

$$P(\|z_n\|_2 - \mathbb{E}\,\|z_n\|_2 \geq \epsilon) \leq \exp(-\lambda\epsilon)\exp\left(\frac{\alpha\lambda^2}{2}\sum_{i=1}^{n} L_i^2\right). \tag{31}$$

□

The proof of Theorem 1 follows by optimizing over λ in (26). □

A.2 Proof of Theorem 1: Bound in expectation

Proof. First we extract a martingale difference from the update rule (5): Recall that $z_n := \theta_n - \hat{\theta}_T$. Let $f_n(\theta) := (\theta^\intercal\phi(s_{i_n}) - (r_{i_n} + \beta\theta^\intercal\phi(s'_{i_n})))\phi(s_{i_n})$ and let $F(\theta) := \mathbb{E}_{i_n}(f_n(\theta))$. Then

$$z_n = \theta_n - \hat{\theta}_T = \theta_{n-1} - \hat{\theta}_T - \gamma_n\left(F(\theta_{n-1}) - \Delta M_n\right),$$

where $\Delta M_{n+1}(\theta) = F_n(\theta) - f_n(\theta)$ is a martingale difference. Now since $\hat{\theta}_T$ is the LSTD solution, $F(\hat{\theta}_T)) = 0$. Moreover $F(\cdot)$ is linear, and so we obtain a recursive procedure:

$$\begin{aligned} z_n =& z_{n-1} - \gamma_n\left(z_{n-1}A_n - \Delta M_n\right) \\ =& \Pi_n z_0 - \sum_{k=1}^{n}\gamma_k \Pi_n \Pi_k^{-1}\Delta M_k, \end{aligned}$$

where $\bar{A}_n = \dfrac{1}{n}\sum\limits_{i=1}^{n}\phi(s_i)(\phi(s_i) - \beta\phi(s'_i))^\intercal$ and $\Pi_n := \prod_{k=1}^{n}\left(I - \gamma_k\bar{A}_k\right)$.

By Jensen's inequality, we obtain

$$\begin{aligned} \mathbb{E}(\|z_n\|_2) &\leq \left(\mathbb{E}(\langle z_n, z_n\rangle)\right)^{\frac{1}{2}} \\ &= \left(\mathbb{E}\,\|\Pi_n z_0\|_2^2 + \sum_{k=1}^{n}\gamma_k^2\mathbb{E}\left\|\Pi_n\Pi_k^{-1}\Delta M_k\right\|_2^2\right)^{\frac{1}{2}} \end{aligned} \tag{32}$$

Notice that $\bar{A}_n - (1-\beta)\mu I$ is positive definite by (A3) and hence

$$\begin{aligned} \left\|\Pi_n\Pi_k^{-1}\right\|_2 =& \left\|\prod_{j=k+1}^{n}\left(I - \gamma_j\bar{A}_j\right)\right\|_2 \leq \prod_{j=k+1}^{n}\left\|(1-\gamma_j(1-\beta)\mu)I - \gamma_j(\bar{A}_j - (1-\beta)\mu I)\right\|_2 \\ \leq& \prod_{j=k+1}^{n}\|(1-\gamma_j(1-\beta)\mu)I\|_2 \leq \prod_{j=k+1}^{n}(1-\gamma_j(1-\beta)\mu) \leq \exp\left(-(1-\beta)\mu(\Gamma_n - \Gamma_k)\right), \end{aligned} \tag{33}$$

We now bound the martingale difference ΔM_n as follows:

$$\begin{aligned}\mathbb{E}[\|\Delta M_n\|_2^2] =&\mathbb{E}_{i_t}\langle f_{i_t}(\theta_{t-1}), f_{i_t}(\theta_{t-1})\rangle - \mathbb{E}_{i_t}\langle F(\theta_{t-1}), F(\theta_{t-1}\rangle\\ \leq&\mathbb{E}_{i_t}\langle f_{i_t}(\theta_{t-1}), f_{i_t}(\theta_{t-1})\rangle\\ \leq&R_{\max}(R_{\max}+2) + (1+\beta)^2 \left\|\theta_{t-1}\right\|_2^2 \qquad (34)\\ \leq&H_\beta^2, \qquad (35)\end{aligned}$$

where (34) follows from (A1) and (A3). The claim now follows by plugging (33) and (35) into (32). □

B Incorporating Iterate Averaging

Here we incorporate the well-known Polyak-Ruppert scheme to average the iterates θ_n. As mentioned earlier, averaging coupled with larger step-sizes $\gamma_n = \frac{(1-\beta)}{2}\left(\frac{c}{c+n}\right)^{-\alpha}$ with $\alpha \in (1/2, 1)$ leads to a convergence rate of the order $O(n^{-\alpha/2})$ irrespective of the choice of c in the step-size.

B.1 High probability bound

Define $\bar\theta_{n+1} \triangleq \dfrac{\theta_1 + \ldots + \theta_n}{n}$ and let $z_n = \bar\theta_{n+1} - \hat\theta_T$ denote the distance of the averaged iterate to the LSTD solution.

First we directly give a bound on the error in high probability for the averaged iterates:

Theorem 8. *Under (A1)-(A2) we have, for all $\epsilon \geq 0$ and $\forall n \geq 1$,*

$$P(\left\|z_n\right\|_2 - \mathbb{E}\left\|z_n\right\|_2 \geq \epsilon) \leq \exp\left(-\epsilon^2/2\sum_{i=1}^n L_i^2\right), \quad \textit{where}$$

$$L_i := \frac{\gamma_i}{n}\left(1 + \sum_{l=i+1}^{n-1}\prod_{j=i}^{l}(1 - 2\gamma_{j+1}\mu((1-\beta) - \beta(2-\beta)\gamma_{j+1}))^{1/2}\right).$$

Proof. As in Theorem 1, we first decompose $\left\|z_n\right\|_2^2 - E\left\|z_n\right\|_2^2$ into a sum of martingale differences as follows:

$$\left\|z_n\right\|_2 - \mathbb{E}\left\|z_n\right\|_2 = \sum_{i=1}^n D_i, \qquad (36)$$

where $D_i \triangleq g_i - \mathbb{E}[g_i \left| \mathcal{F}_{i-1}\right.]$ and $g_i = \sum\limits_{i=1}^n \mathbb{E}[\left\|z_n\right\|_2 \left|\zeta_i = (\zeta_i^1, \zeta_i^2)\right.]$. Here ζ_i^1 is the value of the averaged iterate $\bar\theta_{i+1}$ at instant i and ζ_i^2 is the value of the iterate θ_i at instant i.

The next step is to prove that the functions g_i are Lipschitz continuous with constants

$$L_i := \frac{\gamma_i}{n}\left(1 + \sum_{l=i+1}^{n-1}\prod_{j=i}^{l}\left(1 - 2\gamma_{j+1}\mu((1-\beta) - \beta(2-\beta)\frac{\gamma_{j+1}}{2})\right)^{1/2}\right).$$

Let $\bar\Theta_j^i(\zeta)$ denote the mapping that returns the value of the averaged iterate at instant j, $\bar\theta_j$, given that $\bar\theta_{i-1} = \zeta^1$ and $\theta_i = \zeta^2$. Then, we have

$$\begin{aligned}\mathbb{E}\left[\left\|\bar\Theta_n^i(\zeta) - \bar\Theta_n^i(\zeta')\right\|_2\right] \leq&\frac{i+1}{n}\left\|\zeta^1 - \zeta'^1\right\|_2\\ &+ \frac{1}{n}\sum_{l=i+1}^{n-1}\prod_{j=i}^{l}\left(1 - 2\gamma_{j+1}\mu((1-\beta) - \beta(2-\beta)\frac{\gamma_{j+1}}{2})\right)^{1/2}\left\|\zeta^2 - \zeta'^2\right\|_2 \qquad (37)\end{aligned}$$

Note that since we consider only the smoothness with respect to ξ_i, the value of the averaged iterate at time $i-1$ is irrelevant. Hence, similarly to the proof of Lemma 6, we find that g_i is L_i-Lipschitz in ξ_i.

The rest of the proof follows in a similar manner to the proof of Theorem 1. □

B.2 Proof of Corollary 4

The proof involves the following steps:

Step 1. We derive the bounds for the Lipschitz constants L_i when the iterates are averaged and the step-sizes are chosen to be $\gamma_n = \frac{(1-\beta)}{2}\left(\frac{c}{c+n}\right)^{-\alpha}$ for some $\alpha \in \left(\frac{1}{2}, 1\right)$. This is a crucial step that helps in establishing the order $O(n^{-1/2})$ rate for the high-probability bound in Theorem 1, independent of the choice of c. Recall that in order to obtain this rate for the algorithm without averaging one had to choose $(1-\beta)^2\mu c \in (1,2)$. The main ingredients of this derivation can be found in the argument of pp. 15 in Fathi and Frikha [2013], however here we manage to give all the constants explicitly.

Step 2. We bound the expected error by directly averaging the errors of the non-averaged iterates:

$$\mathbb{E}\left\|\bar{\theta}_{n+1} - \hat{\theta}_T\right\|_2 \le \frac{1}{n}\sum_{k=1}^{n}\mathbb{E}\left\|\theta_k - \hat{\theta}_T\right\|_2,$$

and directly applying the bounds in expectation given in Theorem 1. This involves specializing the bounds for the bound in expectation in Theorem 1 for the new choice of step-size sequence.

Step 1: Bounding the Lipschitz constants.

Lemma 9. *Under conditions of Corollary 4, we have*

$$\sum_{i=1}^{n} L_i^2 \le \frac{1}{\mu(1-\beta)}\left\{3^\alpha + \left[\frac{2\alpha}{\mu c^\alpha(1-\beta)^2} + \frac{2^\alpha}{\alpha}\right]^2\right\}^2 \frac{1}{n} \tag{38}$$

Proof. We perform the calculation:

$$\begin{aligned}
\sum_{i=1}^{n} L_i^2 &= \sum_{i=1}^{n}\left[\frac{\gamma_i}{n}\left(1 + \sum_{l=i+1}^{n-1}\prod_{j=i}^{l}\left(1 - 2\mu\gamma_{j+1}((1-\beta) - \beta(2-\beta)\frac{\gamma_{j+1}}{2}))\right)^{1/2}\right)\right]^2 \\
&\le \frac{1}{n^2}\sum_{i=1}^{n}\left[\gamma_i\left(1 + \sum_{l=i+1}^{n-1}\exp\left(-\sum_{j=i}^{l}\mu\gamma_{j+1}((1-\beta) - \beta(2-\beta)\frac{\gamma_{j+1}}{2}))\right)\right)\right]^2 \\
&= \frac{1}{n^2}\sum_{i=1}^{n}\left[\frac{1-\beta}{2}\left(\frac{c}{c+i}\right)^\alpha\left(1 + \sum_{l=i+1}^{n-1}\exp\left(-\frac{(1-\beta)^2\mu}{4}\sum_{j=i}^{l}\left(\frac{c}{c+i}\right)^\alpha\right)\right)\right]^2 \\
&\le \frac{1}{\mu(1-\beta)n^2}\sum_{i=1}^{n}\Bigg[\left(\frac{c+i+2}{c+i}\right)^\alpha + \frac{1}{(c+i)^\alpha}\sum_{l=i}^{n-1}\exp\left(-(1-\beta)^2\mu c^\alpha\frac{((c+l)^{1-\alpha} - (c+i)^{1-\alpha})}{1-\alpha}\right) \\
&\qquad .((c+l+2)^\alpha - (l+1)^\alpha)\Bigg]^2 \\
&\le \frac{1}{\mu(1-\beta)n^2}\left\{3^\alpha + \sum_{i=1}^{n}\left[\frac{2\alpha}{\mu c^\alpha(1-\beta)^2} + \frac{2^\alpha}{\alpha}\right]^2\right\}
\end{aligned}$$

In the second equality we have substituted $\gamma_i = \frac{(1-\beta)}{2}\left(\frac{c}{c+n}\right)^{\alpha}$. For the second inequality we have used an Abel transform (see page 15 in Fathi and Frikha [2013], display (2.2), for details). For the last inequality we have noted, as in page 15 in Fathi and Frikha [2013], that

$$\begin{aligned}(A) :=& \sum_{l=i+1}^{n-1} \exp\left(-\frac{\mu c^{\alpha}(1-\beta)^2((c+l)^{1-\alpha}-(c+i)^{1-\alpha})}{1-\alpha}\right)((c+l+2)^{\alpha}-(c+l+1)^{\alpha})\\ \leq& \frac{1}{1-\alpha}\exp\left(\frac{\mu c^{\alpha}(1-\beta)^2(c+i)^{1-\alpha}}{1-\alpha}\right)\int_{(c+i+1)^{1-\alpha}}^{(c+n)^{1-\alpha}}\exp\left(\frac{\mu c^{\alpha}(1-\beta)^2 l}{1-\alpha}\right) l^{\frac{2\alpha-1}{1-\alpha}}\,dl.\end{aligned}$$

Now, by taking the derivative and setting it to zero, we find that $l \mapsto \exp\left(\frac{\mu c(1-\beta)l}{1-\alpha}\right) l^{\frac{2\alpha}{1-\alpha}}$ is decreasing on $[2\alpha/\mu c^{\alpha}(1-\beta)^2, \infty)$, and so we deduce that $(A) \leq (c+i+1)^{\alpha}/\alpha$ when $c+i \geq 2\alpha/\mu c^{\alpha}(1-\beta)^2$. When $c+i < 2\alpha/\mu c^{\alpha}(1-\beta)^2$ we use that the summand is bounded by 1. □

Bounding the error in expectation.

Substituting $\gamma_n = \frac{(1-\beta)}{2}\left(\frac{c}{c+n}\right)^{-\alpha}$ for some $\alpha \in \left(\frac{1}{2}, 1\right)$ gives

$$\begin{aligned}\mathbb{E}\left\|\theta_n - \hat{\theta}_T\right\|_2 \leq& \exp\left(-\frac{\mu c^{\alpha}(1-\beta)^2}{2}(n+c)^{1-\alpha}\right)\left\|\theta_0-\hat{\theta}_T\right\|_2\\ &+\left(\sum_{k=1}^{n} H_{\beta}^2 c^2 \frac{(1-\beta)^2}{4}\left(\frac{c}{k+c+1}\right)^{2\alpha}\exp(-\mu(1-\beta)^2c^{\alpha}((n+c)^{1-\alpha}-(k+1+c)^{1-\alpha})\right)^{\frac{1}{2}}\\ \leq& \exp\left(-\frac{\mu c^{\alpha}(1-\beta)^2}{2}(n+c)^{1-\alpha}\right)\left[\left\|\theta_0-\hat{\theta}_T\right\|_2 + H_{\beta}c^{\alpha}(1-\beta)\left\{\int_0^n x^{-2\alpha}\exp(\mu(1-\beta)^2c^{\alpha}x^{1-\alpha})dx\right\}^{\frac{1}{2}}\right]\\ \leq& \exp\left(-\frac{\mu c^{\alpha}(1-\beta)^2}{2}(n+c)^{1-\alpha}\right)\Big[\left\|\theta_0-\hat{\theta}_T\right\|_2\\ &+H_{\beta}c^{\alpha}(1-\beta)\left\{\left(\mu c^{\alpha}(1-\beta)^2\right)^{-2\alpha}\int_0^{(\mu c^{\alpha}(1-\beta)^2)^{1/(1-\alpha)}n} y^{-2\alpha}\exp(y^{1-\alpha})dy\right\}^{\frac{1}{2}}\Big]\\ \leq& \exp\left(-\frac{\mu c^{\alpha}(1-\beta)^2}{2}(n+c)^{1-\alpha}\right)\Big[\left\|\theta_0-\hat{\theta}_T\right\|_2\\ &+H_{\beta}c^{\alpha}(1-\beta)\left\{\left(\mu c^{\alpha}(1-\beta)^2\right)^{-2\alpha}\int_0^{(\mu c^{\alpha}(1-\beta)^2)^{1/(1-\alpha)}n}((1-\alpha)y^{-2\alpha}-\alpha y^{-(1+\alpha)})\exp(y^{1-\alpha})dy\right\}^{\frac{1}{2}}\Big]\\ \leq& \exp(-\mu c n^{1-\alpha})\left\|\theta_0-\theta_T\right\|_2 + H_{\beta}c^{\alpha}(1-\beta)\left(\mu c^{\alpha}(1-\beta)^2\right)^{-\alpha\frac{1+2\alpha}{2(1-\alpha)}}(n+c)^{-\frac{\alpha}{2}}\end{aligned}$$

So we have

$$\mathbb{E}\left\|\bar{\theta}_n - \theta_T\right\|_2 \leq \sum_{n=1}^{\infty}\exp(-\mu c(n+c)^{1-\alpha})\left\|\theta_0-\theta_T\right\|_2 n^{-1} + H_{\beta}c^{\alpha}(1-\beta)\left(\mu c^{\alpha}(1-\beta)^2\right)^{-\alpha\frac{1+2\alpha}{2(1-\alpha)}}(n+c)^{-\frac{\alpha}{2}}.$$

The proof of Corollary 4 follows from the above and Lemma 9.

C Stochastic Approximation for Least Squares Regression

Recall that this algorithm stochastic approximation scheme that updates the parameter θ_n according to the update rule (starting with an arbitrary $\theta_0 \in \mathbb{R}^d$),

$$\theta_n = \theta_{n-1} + \gamma_n (y_{i_n} - \theta_{n-1}^{\mathsf{T}} x_{i_n}) x_{i_n}, \tag{39}$$

where each i_n is chosen uniformly randomly from the set $\{1, \dots, T\}$ (i.e., the samples (x_{i_n}, y_{i_n}) passed to (39) are picked randomly with uniform probability $1/T$ from the set $\{(x_1, y_1), \dots, (x_T, y_T)\}$), and the quantities γ_n are *step sizes*.

We make the following assumptions for the analysis:

(A1) Boundedness of x_i, i.e., $\|x_i\|_2 \leq 1$, for $i = 1, \dots, T$.

(A2) The noise $\{\xi_i\}$ is i.i.d., zero mean and $|\xi_i| \leq 1$, for $i = 1, \dots, T$.

(A3) The matrix $\bar{A}_T$ is positive definite, and its smallest eigenvalue is at least $\mu > 0$.

(A4) The step sizes γ_n satisfy $\sum_n \gamma_n = \infty$, and $\sum_n \gamma_n^2 < \infty$.

Our first two assumptions are standard in the context of least squares minimization. As for fLSTD-SA, in cases when the third assumption is not satisfied we can employ one of the variants described in Section 4 of the main paper to produce similar results.

In the following, we present an analogue of Theorem 1 in this setting (Recall that $\hat{\theta}_T$ is the least squares solution):

Theorem 10. *Under (A1)-(A4), we have* $\forall \epsilon > 0$,

$$P(\left\|\theta_n - \hat{\theta}_T\right\|_2 - \mathbb{E}\left\|\theta_n - \hat{\theta}_T\right\|_2 \geq \epsilon) \leq \exp\left(-\epsilon^2 / (2 \sum_{i=1}^{n} L_i^2)\right), \tag{40}$$

$$\mathbb{E}\left\|\theta_n - \hat{\theta}_T\right\|_2 \leq \underbrace{\exp(-(1-\beta)\mu\Gamma_n)\left\|\theta_0 - \hat{\theta}_T\right\|_2}_{\textit{initial error}} + \underbrace{\left(\sum_{k=1}^{n-1} 2h(k)\gamma_{k+1}^2 \exp(-2(1-\beta)\mu(\Gamma_n - \Gamma_{k+1})\right)^{\frac{1}{2}}}_{\textit{sampling error}}, \tag{41}$$

where $L_i := \gamma_i \prod_{j=i}^{n-1} (1 - 2\gamma_{j+1}\mu(1-\gamma_{j+1}))^{1/2}$, $\Gamma_n := \sum_{i=1}^{n} \gamma_i$, $h(k) := (\sigma_\xi^2 + 2\|z_0\|_2^2) + 4\|z_0\|_2 \ln k + 2\ln^2 k$, *and* $\sigma_\xi^2 := Var(\xi) < \infty$.

The proof of the above theorem has the same scheme as the proof of Theorem 1. The major difference is that the update rule is no longer the update rule of a fixed point iteration, but of a gradient descent scheme. Therefore we see differences in the proof wherever the update rule is unrolled and bounds on the various quantities in the resulting expansion need to be obtained.

C.1 Proof of Theorem 10: High probability bound

This theorem follows the proof of high probability bound in Theorem 1, except in the derivation of the Lipschitz constants (Lemma 6 in the proof of Theorem 1). This is the only part we prove here:

Proof. Denote $f_n(\theta) := \frac{1}{2}(\xi_{i_n} - (\theta - \theta^*)^{\mathsf{T}} x_{i_n})^2$. The update (39) can be re-written as

$$\theta_n = \theta_{n-1} - \gamma_n (F'(\theta_{n-1}) - \Delta M_n),$$

where $F(\theta) \triangleq \mathbb{E}_{i_n}[f_n(\theta)]$ and ΔM_{n+1} is the associated martingale difference sequence defined by $\Delta M_{n+1}(\theta) = F'(\theta) - f_n'(\theta)$.

Let $\Theta^i_j(\theta)$ denote the mapping that returns the value of the iterate updated according to (39) at instant j, given that $\theta_i = \theta$. Now we note that

$$\Theta^i_n(\theta) - \Theta^i_n(\theta') = \left(I - \gamma_n x_{i_n} x_{i_n}^T\right)\left[\Theta^i_{n-1}(\theta) - \Theta^i_{n-1}(\theta')\right]$$

and

$$\left(I - \gamma_n x_{i_n} x_{i_n}^T\right)^T \left(I - \gamma_n x_{i_n} x_{i_n}^T\right) = \left(I - 2\gamma_n(1 - \|x_{i_n}\|_2^2 \gamma_n) x_{i_n} x_{i_n}^T\right).$$

So using Jensen's inequality, the Tower property of conditional expectations, and Cauchy-Schwarz, we can deduce that

$$\begin{aligned} &E\left[\|\Theta^i_n(\theta) - \Theta^i_n(\theta')\|_2 \mid \Theta^i_{n-1}(\theta), \Theta^i_{n-1}(\theta')\right] \\ &\quad \leq \left[\|I - 2\gamma_n(1-\gamma_n)\bar{A}_{n-1}\|_2^2 \|\Theta^i_{n-1}(\theta) - \Theta^i_{n-1}(\theta')\|_2^2\right]^{1/2} \end{aligned}$$

A repeated application of this inequality yields the following

$$\mathbb{E}\left[\left\|\Theta^i_n(\theta) - \Theta^i_n(\theta')\right\|_2^2\right] \leq \|\theta - \theta'\|_2^2 \prod_{j=i}^{n-1}(1 - 2\mu\gamma_{j+1}(1 - \gamma_{j+1})).$$

Finally putting all this together we have

$$\begin{aligned} &\left\|\mathbb{E}\left[\left\|\theta_n - \hat{\theta}_T\right\|_2 |\theta_{i-1}, f_{i_i} = f\right] - \mathbb{E}\left[\left\|\theta_n - \hat{\theta}_T\right\|_2 |\theta_{i-1}, f_{i_i} = f'\right]\right\|_2 \\ &\quad \leq \mathbb{E}\left[\left\|\Theta^i_n(\theta) - \Theta^i_n(\theta')\right\|_2\right] \leq \left(\prod_{j=i}^{n-1}(1 - 2\mu\gamma_{j+1}(1 - \gamma_{j+1}))\right)^{\frac{1}{2}} \gamma_i \left\|f - f'\right\|_2 = L_i \left\|f - f'\right\|_2. \end{aligned}$$

□

C.2 Proof of Theorem 10: Bound in expectation

Proof. First we extract a martingale difference from the update rule (39): Let $f_n(\theta) := \frac{1}{2}(\xi_{i_n} - (\theta - \hat{\theta}_T)^\intercal x_{i_n})^2$ and let $F(\theta) := \mathbb{E}_{i_n}(f_n(\theta))$. Then

$$z_n = \theta_n - \hat{\theta}_T = \theta_{n-1} - \hat{\theta}_T - \gamma_n\left(F'(\theta_{n-1}) - \Delta M_n\right),$$

the $\Delta M_{n+1}(\theta) = F'_n(\theta) - f_n(\theta)$ is a martingale difference.

Now since $\hat{\theta}_T$ is the least squares solution, $F'(\hat{\theta}_T) = 0$. Moreover $F'(\cdot)$ is linear, and so we obtain a recursive procedure:

$$z_n = z_{n-1} - \gamma_n\left(z_{n-1}A_n - \Delta M_n\right) = \Pi_n z_0 - \sum_{k=1}^{n} \gamma_k \Pi_n \Pi_k^{-1} \Delta M_k,$$

where $\bar{A}_n \triangleq \dfrac{1}{2n}\sum\limits_{i=1}^{n} x_i x_i^\intercal$ and $\Pi_n := \prod_{k=1}^{n}\left(I - \gamma_k \bar{A}_k\right)$. By Jensen's inequality

$$\mathbb{E}(\|z_n\|_2) \leq (\mathbb{E}(\langle z_n, z_n\rangle))^{\frac{1}{2}} = \left(\mathbb{E}\left\|\Pi_n z_0\right\|_2^2 + \sum_{k=1}^{n}\gamma_k^2 \mathbb{E}\left\|\Pi_n \Pi_k^{-1}\Delta M_k\right\|_2^2\right)^{\frac{1}{2}} \tag{42}$$

Notice that $\bar{A}_n - \mu I$ is positive definite by (A3) and hence

$$\begin{aligned}\left\|\Pi_n\Pi_k^{-1}\right\|_2 =& \left\|\prod_{j=k+1}^{n}\left(I - \gamma_j \bar{A}_j\right)\right\|_2 \leq \prod_{j=k+1}^{n}\left\|(1-\gamma_j\mu)I - \gamma_j(\bar{A}_j - \mu I)\right\|_2 \\ \leq& \prod_{j=k+1}^{n}\left\|(1-\gamma_j\mu)I\right\|_2 \leq \prod_{j=k+1}^{n}(1-\gamma_j\mu) \leq \exp\left(-\mu(\Gamma_n - \Gamma_k)\right), \end{aligned} \tag{43}$$

Finally we need to bound the squared martingale difference:

$$\mathbb{E}[\|\Delta M_n\|_2^2] = \mathbb{E}_{\xi,i_t}\langle f'_{i_t}(\theta_{t-1}), f'_{i_t}(\theta_{t-1})\rangle - \mathbb{E}_{\xi,i_t}\langle F'(\theta_{t-1}), F'(\theta_{t-1})\rangle$$

Using (A1) and (A3), a calculation shows that

$$\mathbb{E}_{\xi,i_t}\langle f'_{i_t}(\theta_{t-1}), f'_{i_t}(\theta_{t-1})\rangle \leq \sigma_\xi^2 - 2\mathbb{E}_\xi \left\|z_t\right\|_2 + \mathbb{E}_\xi \left\|z_t\right\|_2^2 \text{ and } \mathbb{E}_\xi\langle F'(\theta_{t-1}, F'(\theta_{t-1}\rangle \leq \mathbb{E}_\xi \left\|z_t\right\|_2^2$$

where $\sigma_\xi^2 := Var(\xi) < \infty$ (ξ is distributed according to the noise distribution). Now

$$\begin{aligned}\left\|z_t\right\|_2 =& \left\|\left[\prod_{k=1}^{t}(I - \gamma_k x_{i_k} x_k^\intercal)\right] z_0 \quad + \sum_{k=1}^{t}\gamma_k \left[\prod_{j=k}^{t}(I - \gamma_j x_{i_j} x_j^\intercal)\right] \xi_k x_k\right\|_2 \\ \leq& \left\|z_0\right\|_2 + \sum_{k=1}^{t}\gamma_k \leq \left\|z_0\right\|_2 + \ln t. \end{aligned}$$

and so $\mathbb{E}[\|\Delta M_t\|_2^2] \leq h(t)$.

The result now follows from (42) and (43). □

D Simulation Experiments for Fast Least Squares Variant

Setup The idea behind the experimental setup here is to involve a higher level machine learning algorithm that requires to compute least squares solution at each iteration and then use the fast stochastic approximation variant (henceforth referred to as fLS-SA) to replace traditional least square solution schemes in the higher level algorithm. We choose LinUCB, a well known contextual bandit algorithm proposed in Li et al. [2010] for this purpose. At each iteration n, LinUCB computes a least squares estimate based on the arms x_i and rewards y_i seen so far, $i = 1, \ldots, n-1$. Note that $\{x_i, y_i\}$ do not come from a distribution. Instead, at every iteration n, the arm x_n chosen by LinUCB is based on the least squares estimate $\hat{\theta}_n$. We implement a variant of LinUCB, where we use fLS-SA as a subroutine to approximate $\hat{\theta}_n$. In particular, at any instant n of the LinUCB algorithm, we run the update (15) for 20 steps and use the resulting θ_{20} to derive the UCB values for each arm. Pseudocode for this algorithm, henceforth referred to as ***fLinUCB-SA***, is presented in Algorithm 2.

For conducting the experiments, we use the framework provided by the ICML exploration and exploitation challenge Mary et al. [2012], based on the user click log dataset Webscope [2011] for the Yahoo! front page today module (see Fig. 3). We run each algorithm on several data files corresponding to different days in October, 2011.

The choice of the number of iterations of fLS-SA to make in fLinUCB-SA is an arbitrary one. Our aim is simply to show that using a stochastic approximation iterate in place of an exact solution to the least squares problem does not significantly decrease performance of a higher level algorithm while it does drastically decreasing complexity.

Each data file has an average of nearly two million records of user click information. Each record in the data file contains various information obtained from a user visit. These include the displayed article, whether the user clicked on it or not, user features and a list of available articles that could be recommended. The precise format is described in Mary et al. [2012]. The evaluation of the algorithms in this framework is done in a off-line manner using a procedure described in Li et al. [2011].

For fLinUCB-SA, we set μ to 1, α to 0.1, τ to 20 and θ_0 to the $d = 136$ dimensional $\mathbf{1}$ vector. Further, the step-sizes γ_k are chosen as $1/k$. The constant κ used in second term of the UCB value for each arm in Algorithm 2, is set to 0.1. This choice is motivated by a cross-validation experiment, the results of which are provided Table 2.

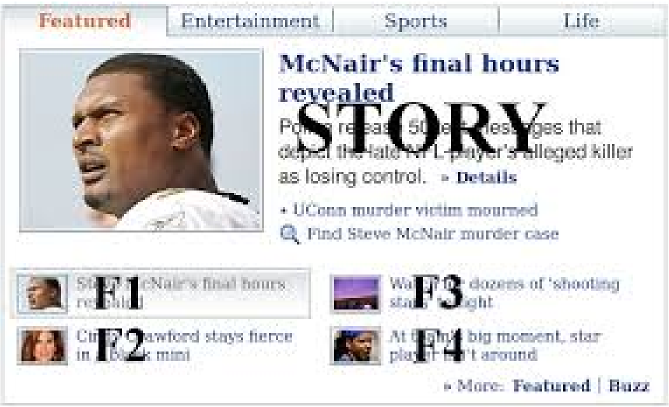

Featured
Entertainment
Sports
Life
McNair's final hours revealed
STORY
as losing control. » Details
• UConn murder victim mourned
Find Steve McNair murder case
F1
F2
F3
F4
» More: Featured | Buzz

Figure 3: The *Featured* tab in Yahoo! Today module (src: Li et al. [2010])

Algorithm 2 fLinUCB-SA

```
Initialisation: Set $\theta_0$, $\mu > 0$ - the regularization parameter, $\gamma_k$ - the step-size sequence.
for $n = 1, 2, \ldots$ to do
    Approximate least squares solution $\hat{\theta}_n$ based on data $\{x_i, y_i\}_{i=1}^{n-1}$ using fLS-SA as follows:
    for $k = 1 \ldots \tau$ do
        Get random sample index: $i_k \sim U(\{1, \ldots, n-1\})$
        Update fLS-SA iterate $\theta_k(n)$ as follows:
        $\theta_k(n) = \theta_{k-1}(n) + \gamma_k(y_{i_k} - \theta_{k-1}^{\mathsf{T}} x_{i_k})x_{i_k} - \gamma_k \mu \theta_{k-1}$
    end for
    Choose arm $a_n = \arg\max_a \left(\theta_\tau^{\mathsf{T}} x_{n,a} + \alpha \frac{\kappa}{\sqrt{n}}\right)$
    Observe $y_n$.
end for
```

Results We report the norm difference and CTR score value obtained from our experimental runs in Figs. 4 and 6, respectively. The norm difference we report is the distance in ℓ^2 norm between the fLS-SA iterate θ_n and the least squares solution $\hat{\theta}_n$ at each instant n of the LinUCB algorithm. The CTR score value here is the ratio of the number of clicks that an algorithm gets to the total number of iterations it completes, multiplied by 10000 for ease of visualization purposes.

From Fig. 4, we observe that, at every instant n of the LinUCB algorithm, fLS-SA algorithm iterate θ_{20} tracks the corresponding least squares solution $\hat{\theta}_n$. Further, the observed difference in ℓ^2-norm is negligible, reinforcing the usefulness of fLS-SA in a higher level machine learning algorithm such as LinUCB.

Figs. 5 and 6 present the CTR scores and runtimes observed by running LinUCB and fLinUCB-SA on five different data files corresponding to five days in October, 2009 of the dataset Webscope [2011]. We observe that the CTR scores observed when fLS-SA is used are not significantly worse than the vanilla LinUCB algorithm. On the other hand, fLinUCB-SA resulted in a runtime gain of approximately 25% when the input data was a period of 5 days. While these experiments correspond to a feature dimension of $d = 136$, one can expect the gains to amplify when settings with larger feature dimensions are considered.

In addition to the above experiments, we also tested a variant of fLinUCB-SA in which at each iteration, n,

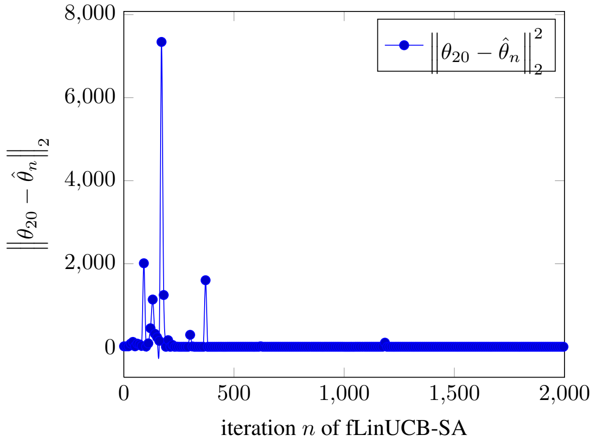


Figure 4: Distance between fLS-SA iterate θ_{20} and $\hat{\theta}_n$ in ℓ^2 norm with day 1's data file as input to fLinUCB-SA

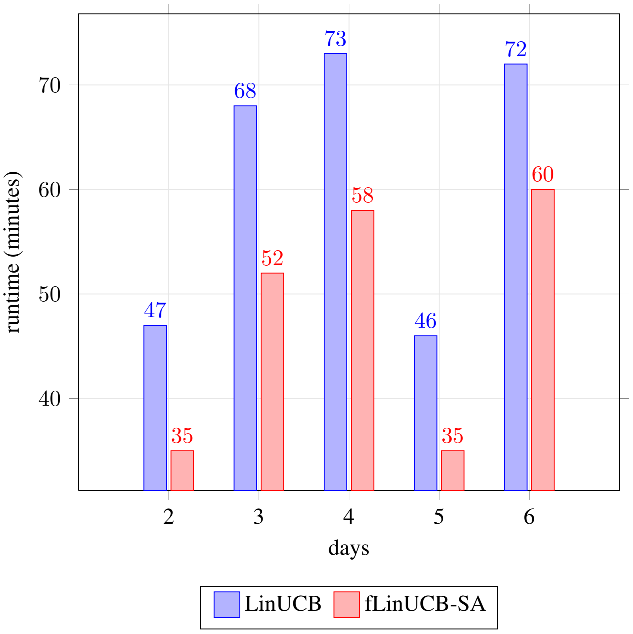


Figure 5: Performance comparison of the algorithms using runtimes on various days of the dataset

	κ value				
day	**0.01**	**0.1**	**1**	**3**	**5**
0.5	372.83	359.57	377.99	390.04	389.48
2	469.96	464.95	475.84	450.85	487.11
3	550.16	564.23	543.99	541.16	570.62
4	502.19	524.02	507.28	507.01	539.62
5	624.72	613.35	648.17	700.20	700.80
6	781.40	823.85	710.15	734.63	733.40

Table 2: CTR scores for different values of kappa on various days of the dataset

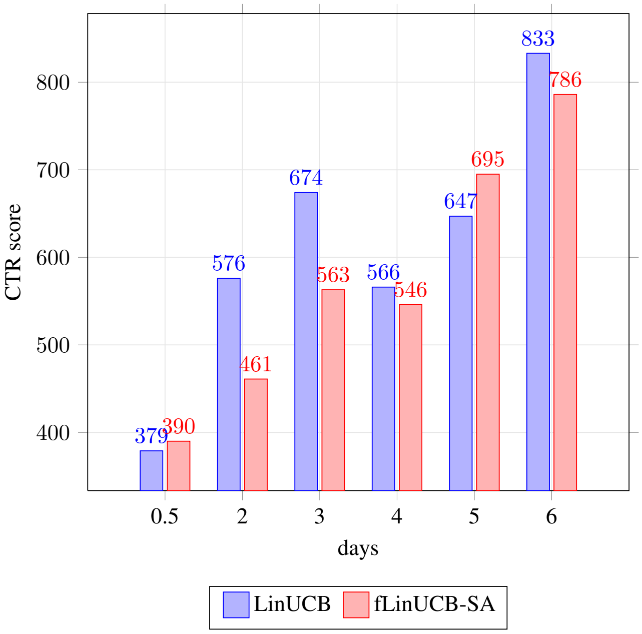
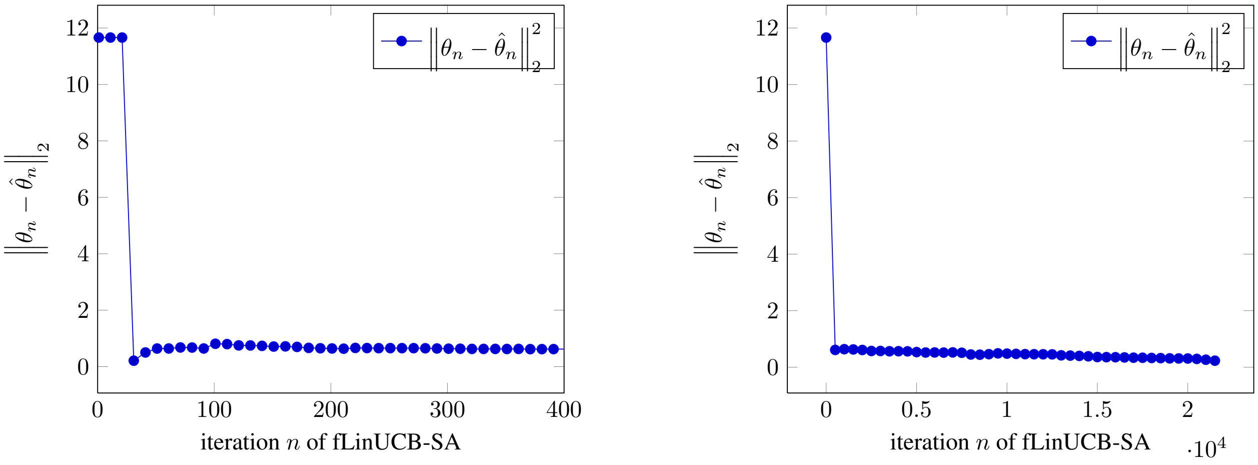


Figure 6: Performance comparison of the algorithms using CTR scores on various days of the dataset (Note: 0.5 refers to 50% of day 2's records, while the rest correspond to days 2 to 6 of October, 2011.)



(a) norm difference in the initial phase

(b) norm difference over a long run

Figure 7: Distance between fLS-SA iterate θ_n and $\hat{\theta}_n$ in ℓ^2 norm with day 1's data file as input to fLinUCB-SA

of the LinUCB algorithm, we perform n iterations (rather than 20) of fLS-SA. Fig. 7 reports results on the norm difference from this experiments and it can be seen that that after 500 iterations fLS-SA is already very accurate.

References

Francis Bach and Eric Moulines. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In *NIPS*, 2011.

Dimitri P Bertsekas. Dynamic Programming and Optimal Control, Vol. II, 4th Edition: Approximate Dynamic Programming. 2012.

Dimitri P Bertsekas and John N Tsitsiklis. *Neuro-Dynamic Programming (Optimization and Neural Computation Series, 3)*, volume 7. Athena Scientific, 1996.

S.J. Bradtke and A.G. Barto. Linear least-squares algorithms for temporal difference learning. *Machine Learning*, 22:33–57, 1996.

Max Fathi and Noufel Frikha. Transport-entropy inequalities and deviation estimates for stochastic approximation schemes. *arXiv preprint arXiv:1301.7740*, 2013.

Noufel Frikha and Stéphane Menozzi. Concentration Bounds for Stochastic Approximations. *Electron. Commun. Probab.*, 17:no. 47, 1–15, 2012.

Alborz Geramifard, Michael Bowling, Martin Zinkevich, and Richard S Sutton. iLSTD: Eligibility traces and convergence analysis. In *NIPS*, volume 19, page 441, 2007.

Elad Hazan and Satyen Kale. Beyond the regret minimization barrier: an optimal algorithm for stochastic strongly-convex optimization. *Journal of Machine Learning Research-Proceedings Track*, 19:421–436, 2011.

Harold J Kushner and George Yin. *Stochastic approximation and recursive algorithms and applications*, volume 35. Springer Verlag, 2003.

Michail G Lagoudakis and Ronald Parr. Least-squares policy iteration. *The Journal of Machine Learning Research*, 4:1107–1149, 2003.

Alessandro Lazaric, Mohammad Ghavamzadeh, and Rémi Munos. Finite-sample analysis of least-squares policy iteration. *Journal of Machine Learning Research*, 13:3041–3074, 2012.

Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670. ACM, 2010.

Lihong Li, Wei Chu, John Langford, and Xuanhui Wang. Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 297–306. ACM, 2011.

J. Mary, Aurlien Garivier, L. Li, R. Munos, O. Nicol, R. Ortner, and P. Preux. Icml exploration and exploitation 3 - new challenges, 2012.

Boris T Polyak and Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992.

L.A. Prashanth and S. Bhatnagar. Reinforcement Learning with Function Approximation for Traffic Signal Control. *IEEE Transactions on Intelligent Transportation Systems*, 12(2):412–421, 2011.

L.A. Prashanth and S. Bhatnagar. Threshold Tuning using Stochastic Optimization for Graded Signal Control. *IEEE Transactions on Vehicular Technology*, 61(9):3865–3880, 2012.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

David Ruppert. Stochastic approximation. *Handbook of Sequential Analysis*, pages 503–529, 1991.

David Silver, Richard S Sutton, and Martin Müller. Reinforcement Learning of Local Shape in the Game of Go. In *IJCAI*, volume 7, pages 1053–1058, 2007.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, volume 1. Cambridge Univ Press, 1998.

Richard S Sutton, Hamid Reza Maei, Doina Precup, Shalabh Bhatnagar, David Silver, Csaba Szepesvári, and Eric Wiewiora. Fast gradient-descent methods for temporal-difference learning with linear function approximation. In *ICML*, pages 993–1000. ACM, 2009a.

Richard S Sutton, Csaba Szepesvári, and Hamid Reza Maei. A convergent O(n) algorithm for off-policy temporal-difference learning with linear function approximation. *NIPS*, 21:1609–1616, 2009b.

John N Tsitsiklis and Benjamin Van Roy. An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control*, 42(5):674–690, 1997.

Yahoo! Webscope. Yahoo! Webscope dataset ydata-frontpage-todaymodule-clicks-v2_0, 2011. URL `"http://research.yahoo.com/Academic_Relations"`.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *ICML*, pages 928–925, 2003.