# Finding the Average number of Jobs in a Random Queue

**Date – 5/2/2016**

**Ankit Vijay**
M.tech. 2nd Yr.
Control and Computing
EE Dept.
IIT Bombay

# Outline

- Basics of Markov chains
- Stationary average
- Monte Carlo methods
- Value iteration with Stochastic approximations
- Function approximation
- Simulation Results

# Markov chain

$\{S_t\}$ is a sequence of random variable.

$S_1, S_2, ..., S_n$ are markov if

$$P(S_{n+1} = j | S_1 = i_1, ...., S_n = i_n) = P(S_{n+1} = j | S_n = i_n)$$

$$\forall i_1, i_2, i_n, j \in \mathcal{S}$$

where , $\mathcal{S}$ is the set of states.

# Designing a Markov chain / Queue

- Let,

  $X_n$ : representing the amount of work seen by n'th arrival in a Queue

  $B_n$ : denotes the amount of work bring by the n'th arrival in a Queue

  $A_n$ : denotes the inter arrival time between the $X_n$ and $X_{n+1}$ arrival.

- This Queue can be modeled as:

$$X_{n+1} = X_n + B_n - A_n$$

where, X denotes the state of the markov chain

B, A denotes the random variable, generated by some known distribution.

# Estimation of the Stationary average

$$\beta = \sum_{i \in \mathcal{S}} f(i)\eta(i)$$

where, f is a prescribed function.

η is the stationary distribution of the chain.

- To calculate the stationary average of the markov chain, we take f(X) = X

# Monte Carlo method

The standard Monte Carlo approach is to simulate the Markov chain as per the given distributions and then take the sample average

$$\frac{1}{N}\sum_{m=1}^{N} f(X_m)$$

where, N denotes the number of samples.

By strong law of large no., as N tends to infinity, the function will converge to the stationary average.

# Analysis of Monte Carlo method

- Although the chain is assumed to be irreducible, there might be exist some almost invariant sets of the state space

- Convergence does not depends upon the initial state we choose but convergence time does depend.

# Stochastic Approximation

Stochastic approximation algorithms are recursive update rules that can be used, to solve optimization problems and fixed point equations.

Suppose we wish to find the root $\bar{\theta}$ of the function $f : \mathbb{R} \rightarrow \mathbb{R}$.

<u>Newton iteration method</u>

$$\theta_{n+1} = \theta_n - \frac{f(\theta_n)}{f'(\theta_n)}$$

**What if we do not have the mathematical model of ‘f’ ?**

An alternative approach by **Robbins and Monro**, is to simply use directly the noisy version of ‘f‘ in a slightly modified version of algorithm.

$$\theta_{n+1} = \theta_n - \gamma_n y_n$$

where $\gamma_n$ is a sequence of positive numbers converging to 0

$$\sum_n \gamma_n = \infty$$

$$\text{and } y_n = f(\theta_n) + d_n$$

is the noisy version of $f(\theta_n)$

The intuition of decreasing step size is that it provides a sort of averaging of the observation.

# ODE Approach

- Consider a function we want to minimize.
- The Gradient descent algorithm for this is

$$x_{n+1} = x_n + \gamma_n[-\nabla f(x_n) + D_{n+1}]$$

- The limiting ODE is then

$$\dot{x}(t) = -\nabla f(x(t))$$

- If the noise is martingale, i.e. $\mathbb{E}[D_{n+1}|D_n] = 0$

and $$\sum_n \gamma_n = \infty, \ \sum_n \gamma_n^2 < \infty$$

Then, iteration converges almost surely to the fixed point or we can say, the optimal point of the function.

# Value iteration with Stochastic approx.

- Let V(i) denotes the Value function/ Cost function correspond to state i.
- Considering the asynchronous Poisson equation, the cost function for each state is updated as[4]

$$V(i) = f(i) - \beta' + \sum_{j \in \mathcal{S}} p(j|i)V(j), \quad j \in \mathcal{S}$$

- The iteration for solving the above equation is

$$V_{n+1}(i) = f(i) - V_n(i_0) + \sum_{j \in \mathcal{S}} p(j|i)V_n(j)$$

Refer: Section 6.7, Applied Probability Models with Optimization Application Sheldon M. Ross

# Value iteration with Stochastic approx.

- In the above iteration:

$$V_n \to V \text{ and } V_n(i_0) \to \beta$$

- The Value iteration incremental update

$$V_{n+1}(i) = V_n(i) + a(n)[f(i) - V_n(i_0) + V_n(X_{n+1}) - V_n(i)]$$

- Step size a(n) is chosen in such a way that it satisfies the stochastic approximation properties.

This can be rewritten as

$$V_{n+1}(i) = V_n(i) + a(n)I\{X_n = i\}[T_i(V_n) - V_n(i_0) + M_{n+1}(i) - V_n(i)],$$

where $T(\cdot) = [T_1(\cdot), \ldots, T_s(\cdot)]^T$ is given by

$$T_k(x) \stackrel{def}{=} f(k) + \sum_j p(j|k)x_j$$

for $x = [x_1, \ldots, x_s]^T \in \mathcal{R}^s$, and for $n \geq 0$,

$$M_{n+1}(j) \stackrel{def}{=} f(j) + V_n(X_{n+1}) - T_j(V_n),\ n \geq 0, 1 \leq j \leq s,$$

# Analysis of Value iteration method

- Unlike the Monte Carlo method, this iteration uses the incremental mean towards the next update, but still have the same problems as in the case of previous method.

- The variance in the final converged value is less than the Monte Carlo method.

- This method can only be applied to the finite state space Markov chains.

# Function Approximation

- We approximates the value function in terms of basis function and then calculates the weights correspond to the basis.

$$V(i) \approx \phi(i)^T r = \sum_{j=1}^{M} r_j \phi_j(i) \qquad \forall i \in \mathcal{S}$$

- Let the Basis function matrix be

$$\Phi \;=\; [[\varphi_{ij}]]_{1\leq i\leq s, 1\leq j\leq M}$$

where, each basis vector is $[\phi_j]_{1\leq j\leq M}$

- Defining $\phi(i) \;=\; [\varphi_{i1}, \varphi_{i2}, \ldots, \varphi_{iM}]^T$

# Function Approximation iterations

- The iteration is given as

$$r_{n+1} = r_n(1 - a(n)) + a(n)[B_n^{-1}\phi(X_n)(\phi^T(X_{n+1})r_n - \phi^T(i_0)r_n + f(X_n))]$$

- where, $B_n = \frac{1}{N+1}\sum_{m=0}^{n}\phi(X_m)\phi^T(X_m)$

- Stationary average $V_{i_0} = \phi(i_0)^T r^* = \sum_{j=1}^{M} r_j\phi_j(i_0)$

# Analysis of Function approx. method

- The convergence of this scheme is dependent on the choice of the basis function. Only the correct set of basis function will lead to convergence.

- The convergence time of this algorithm is independent of the initial state we choose.

- Variance is quite less than the previous method discussed.

# Basis Functions

- M/M/1 and M/G/1 queues are been modeled and the basis functions for them is been found.

- There were many basis and all the permutations of those were checked and the correct set was found.

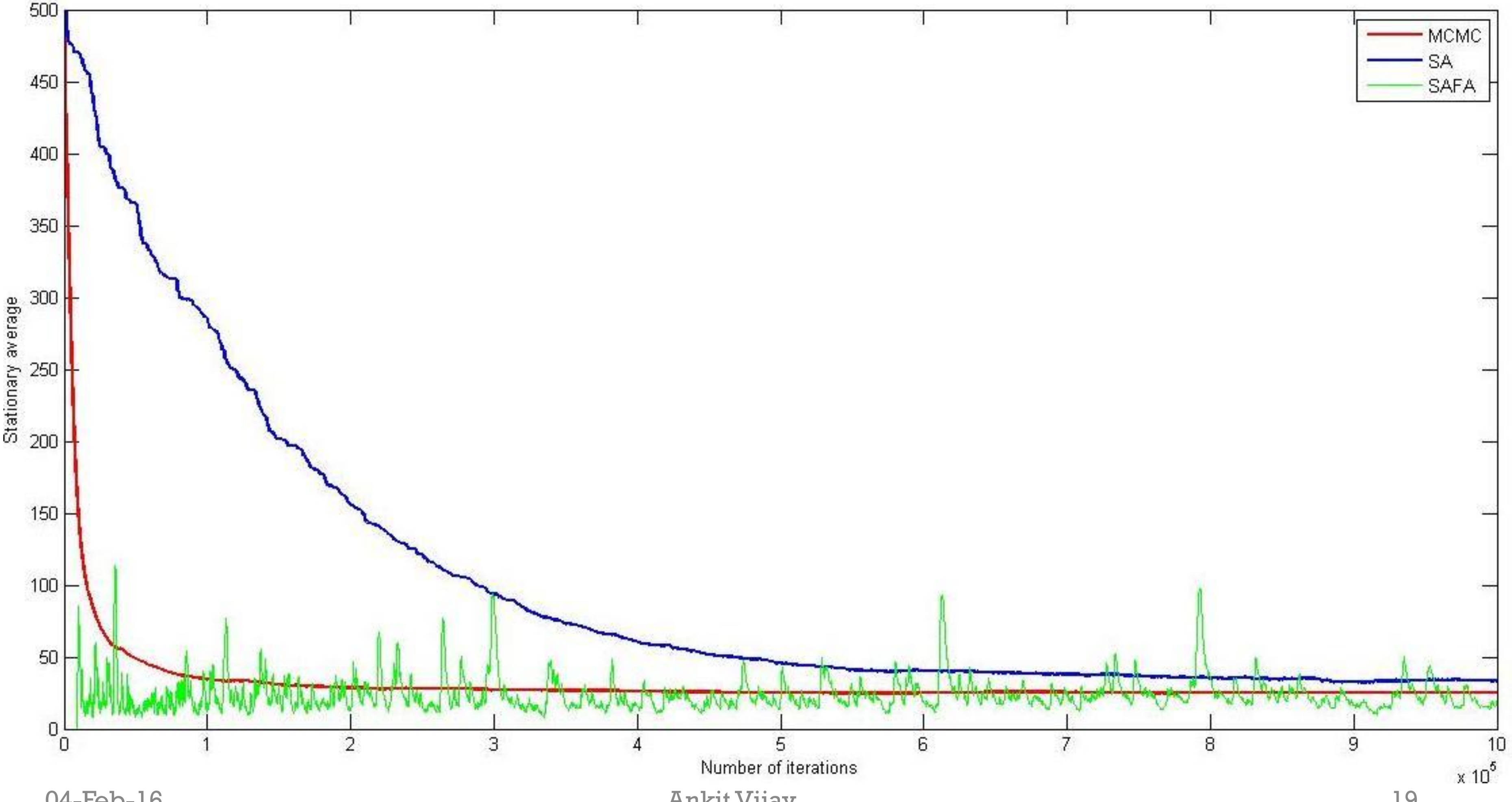
- There can be more than one set of correct basis functions exist for which the iteration will converge.

- Some examples of the basis functions are

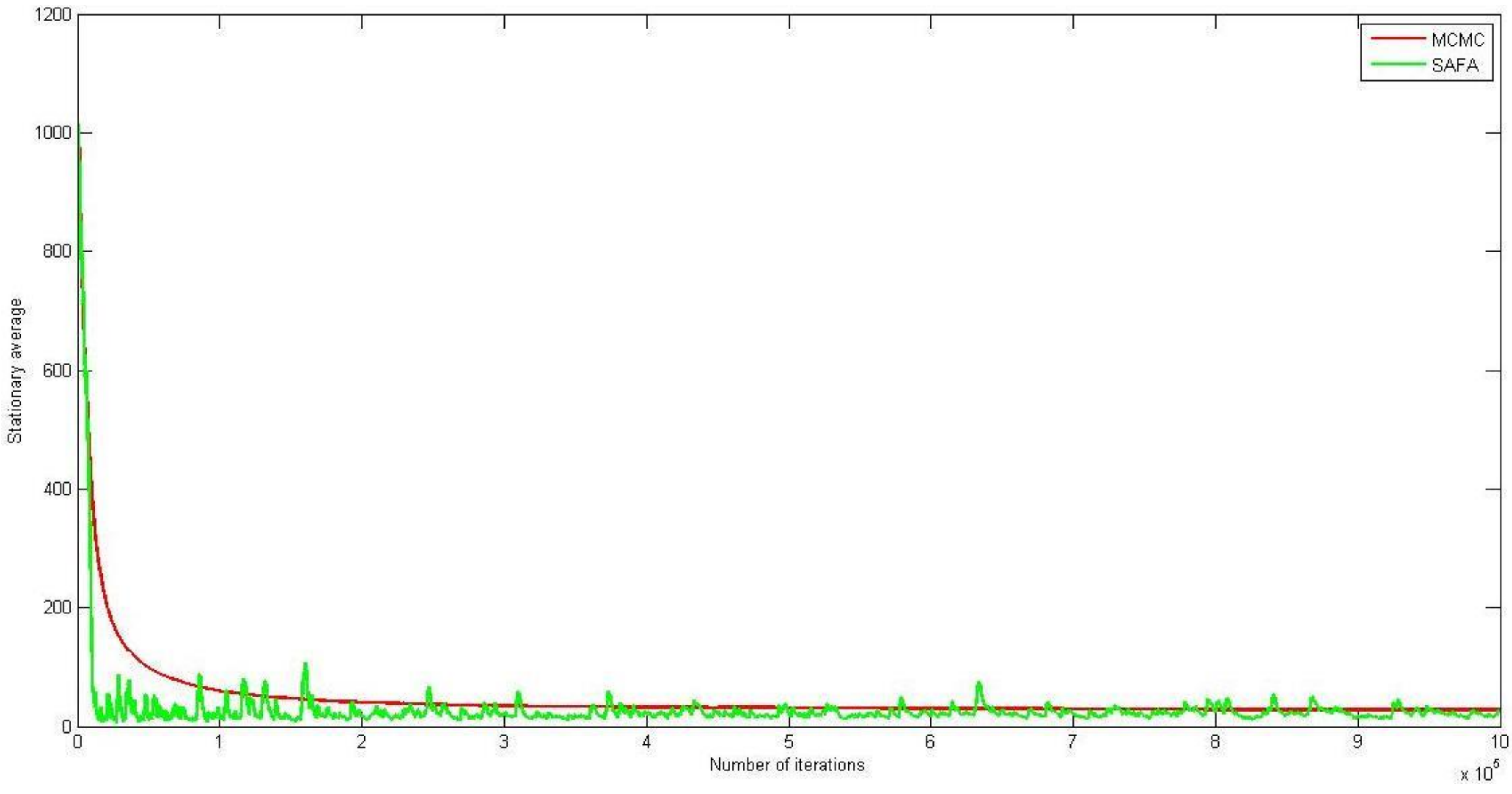
$$\phi = \begin{bmatrix} 1^n & 2^n & \dots & N^n \end{bmatrix}^T, \begin{bmatrix} 0 & \dots & 0, & 1 & \dots & 1 \end{bmatrix}^T$$

$$, \begin{bmatrix} 0 & \dots & 0, & 1 & \dots & 1, & 0 & \dots & 0 \end{bmatrix}^T, \text{etc}$$

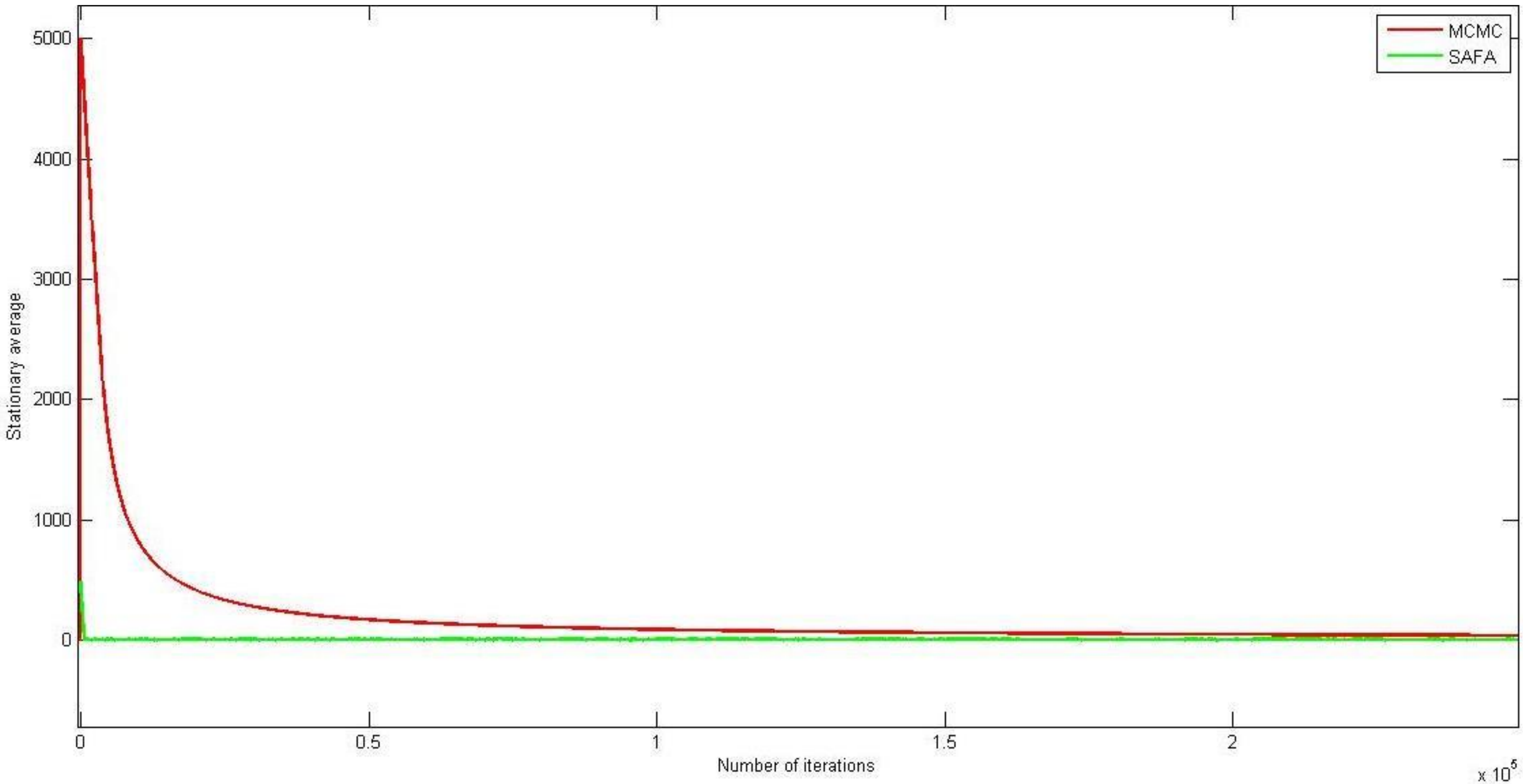
- Experiment 1 : Comparing the SA, MC, SAFA methods for finite state space Markov chain describing M/M/1 queue



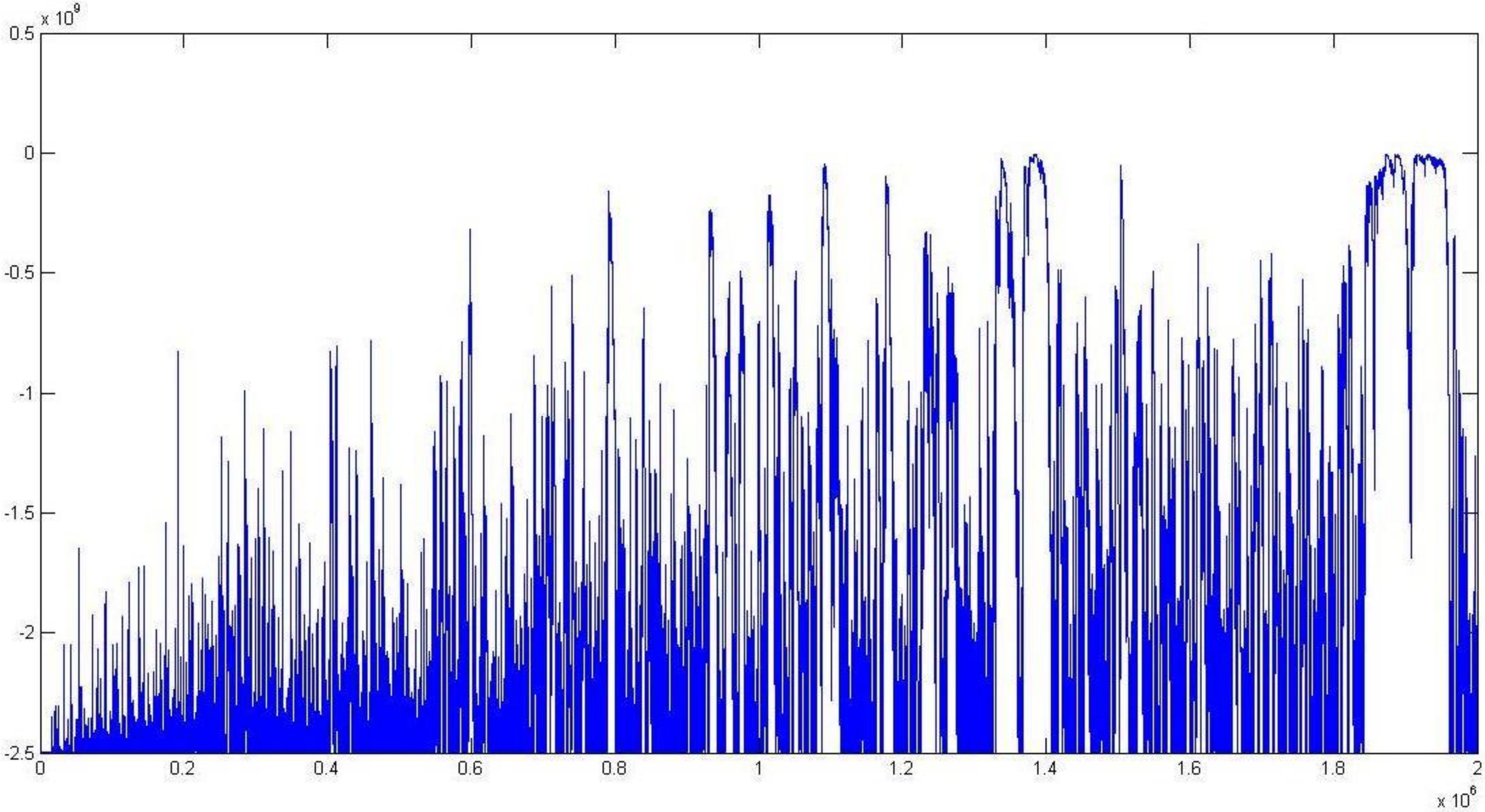
- Experiment 2: Comparing the MC, SAFA methods for infinite state space Markov chain describing M/M/1 queue



• Experiment 3 : Comparing the MC, SAFA methods for infinite state space Markov chain describing M/G/1 queue



- Experiment 4 : Simulate the M/G/1 queue with same parameters as earlier but with wrong set of basis functions.



# Split Sampling

- Instead of generating a random variable in the state space, we now generate two random variable in the same state space, keeping the transition probabilities same.

$$X_{n+1} = Y_n + B_n - A_n$$

where $Y_n$ is any other independent distribution.

say, $Y_n \sim \text{Uniform}(S)$

- This will improve the convergence, but only for the small state space. For the large state space, it is difficult to use this method and will not give good results.

# References

[1] Vivek S. Borkar, Reinforcement Learning - A Bridge Between Numerical Methods and Monte Carlo, Worls scientific review Volume 9in x 6in , May 7, 2009.

[2] Vivek S. Borkar, Stochastic Approximation- A Dynamical Systems Viewpoint, Hindustan book agency.

[3] Bertsekas, D. P. (2007). Dynamic Programming and Optimal Control, Vol. 2(3rd edition). Athena Scientific, Belmont, Mass.

[4] Sheldon M. Ross, Applied Probability Models with Optimization Applications,

# Thank You