Random Walk Gradient Descent for Decentralized Learning on Graphs

by Ghadir Ayache, Salim El Rouayheb

2019 IEEE IPDPSW

Group 8

Jihun Shin

Random Walk

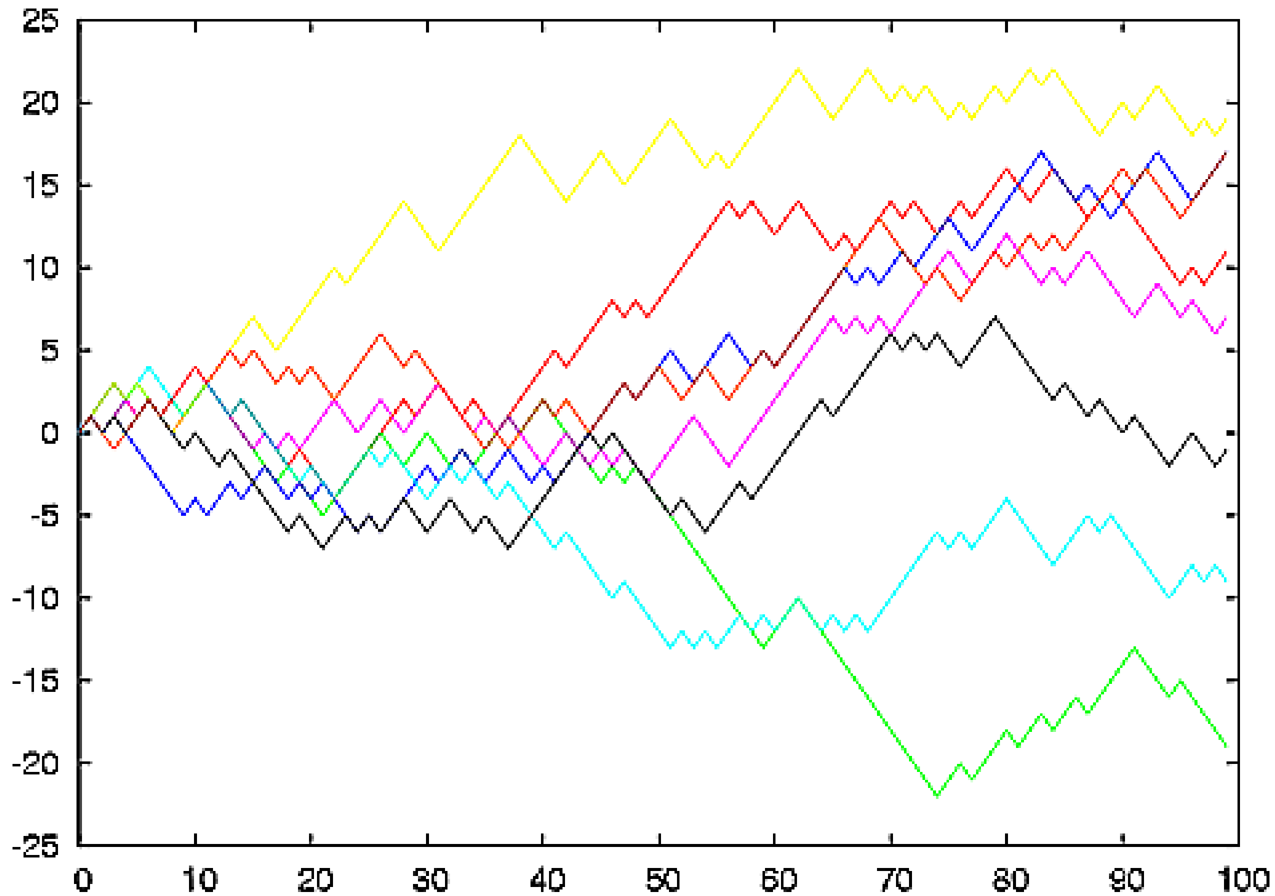
- Representation of random movements at every moment on a mathematical space

- Discrete random walks are widely used in situations where discrete mathematics is applied

Random Walk

Example: Random walk on the 1-dimensional integer line

- starts at 0
- moves $+1$ or 1 with same probability at each steps

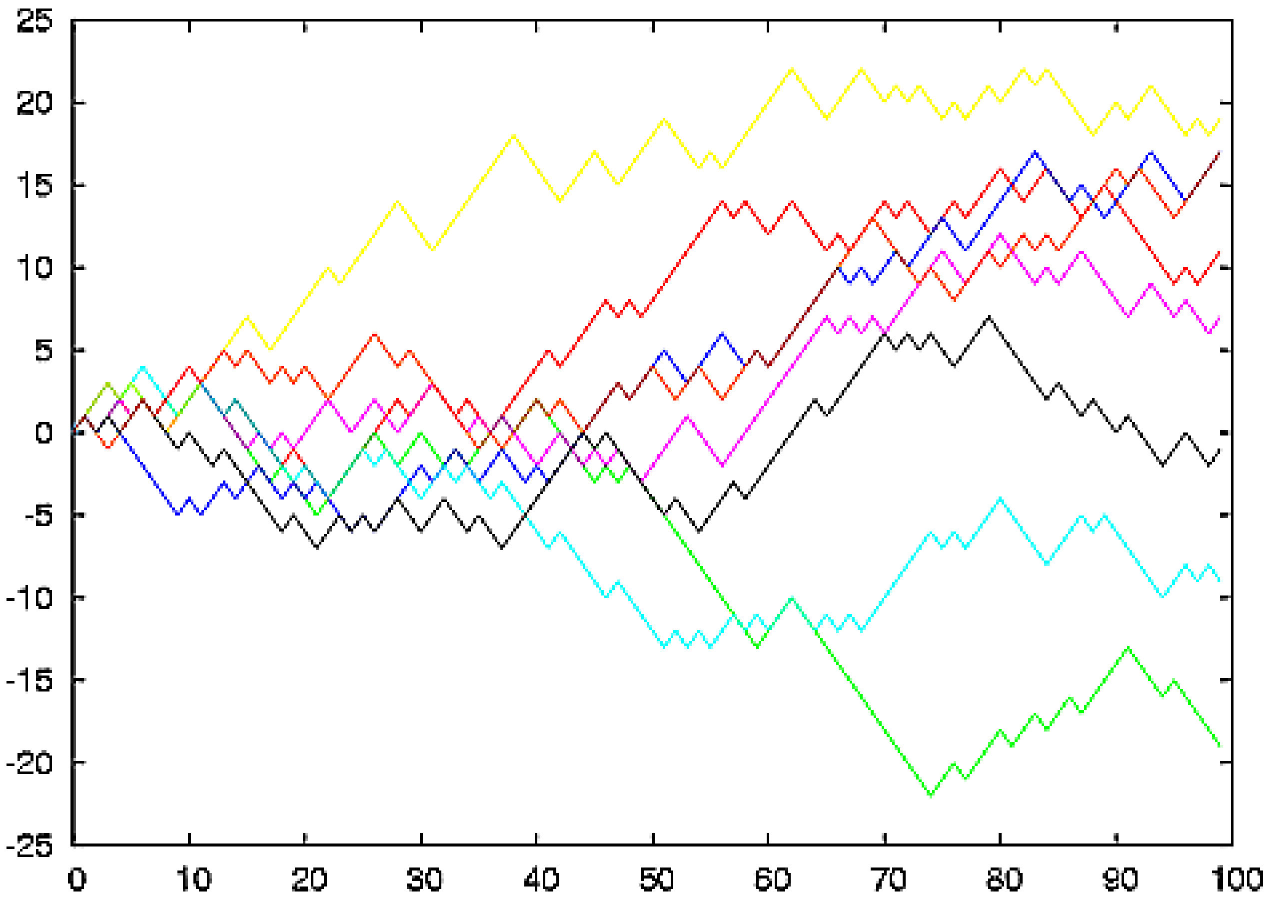


Random Walk

Each random step Z_i is either 1 or -1

For the series of random steps S_n,

- $E[S_n] = 0$
- $E[{S_n}^2] = n$



Model

We consider a network of N interacting nodes represented by an undirected connected graph $G(V, E)$.

Interested in learning a global model $w^* \in \mathcal{W}$ that minimizes an average loss function $f(w) = \frac{1}{N}\sum_{i=1}^{N} f_i(w)$, subject to $w \in \mathcal{W}$.

$f_i(\cdot)$ is the local loss function at node i and $\mathcal{W}$ is a convex compact set.

The goal is to find $w^* \in \mathcal{W}$ satisfying $w^* = \arg\min_{w \in \mathcal{W}} \frac{1}{N}\sum_{i=1}^{N} f_i(w)$.

Algorithms

Probability for moving from a node v to a node u in V:
$Q(v, u) = \frac{1}{deg(v)}$

Acceptance probability of a proposed jump from node v to node u:
$a(v, u) = \min\left(1, \frac{\pi(u)}{\pi(v)}\frac{Q(u,v)}{Q(v,u)}\right)$, while desired stationary distribution is π

Transition matrix P:

$$\begin{aligned} P(v, u) &= Q(v, u)a(v, u) \\ &= \min\left(Q(v, u), Q(u, v)\frac{\pi(u)}{\pi(v)}\right) \end{aligned}$$

Algorithms

Algorithm 1 Uniform Random Walk GD

Initialization: Initial node v_0, Initial model w_0

for $t = 0$ **to** T **do**

Choose node u uniformly at random from $\mathcal{N}(v_t)$.

Generate $p \sim U(0, 1)$ where U is the uniform distribution.

if $p \leq \min\left\{1, \frac{deg(v_t)}{deg(u)}\right\}$ **then**

$v_{t+1} \leftarrow u$

else

$v_{t+1} \leftarrow v_t$

end if

$w_{t+1} = \Pi_{\mathcal{W}}\left(w_t - \gamma_t \nabla f_{v_{t+1}}(w_t)\right)$

end for

Return: w_T and $\bar{w}_T = \frac{\sum_{i=1}^{T}(\gamma_i w_i)}{\sum_{j=1}^{T}\gamma_j}$. {returned to node 1}

Algorithms

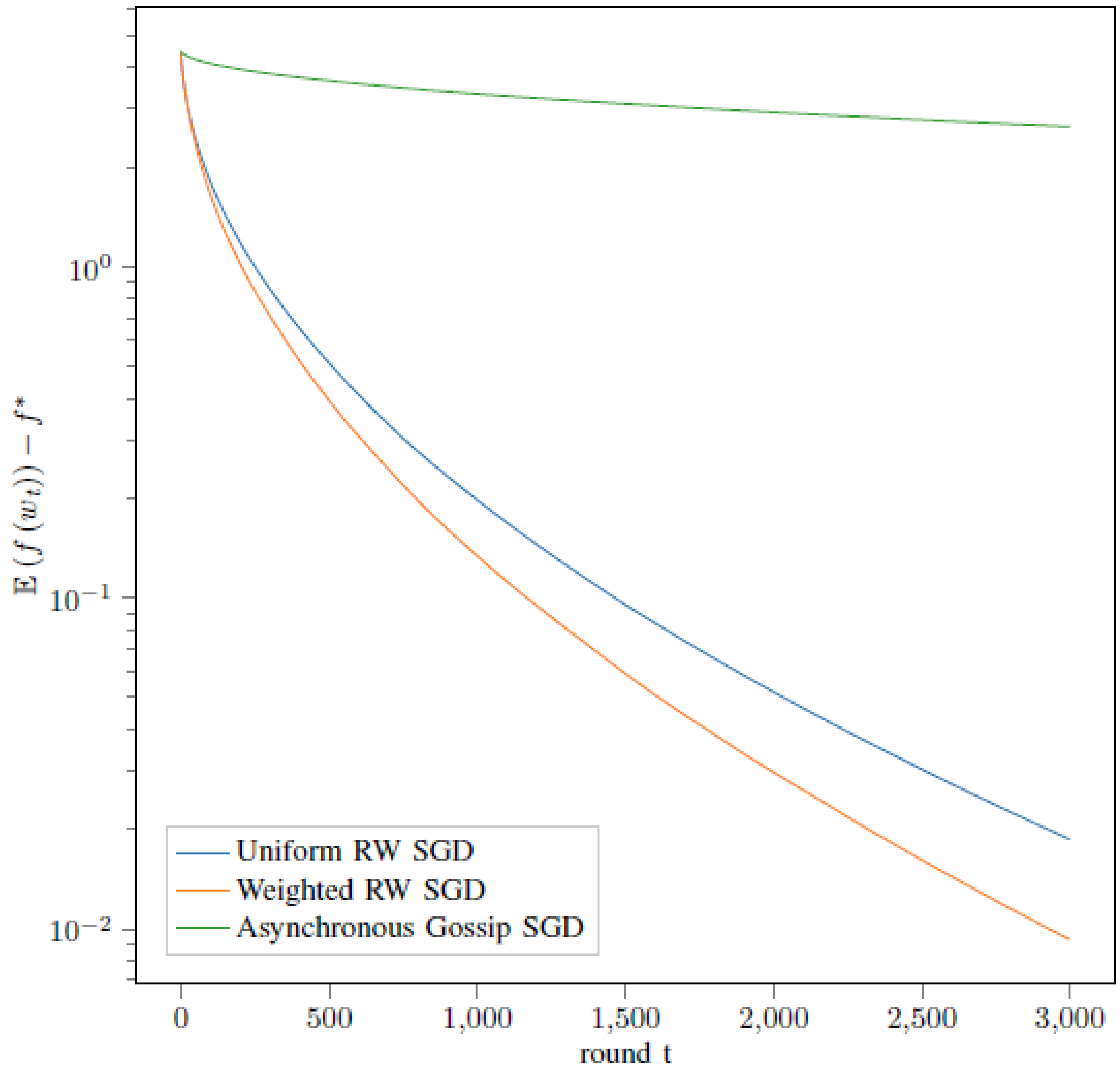


Fig. 2. Comparison of the Uniform RW SGD, Weighted RW SGD and the Gossip SGD on a chordal cycle graph of 20 nodes and 60 edges.

Relating to Our Project

Topic: Study and Application of Random Walk in Machine Learning

In this paper, I mainly checked the followings

- How the random walking can be applied to gradient descent algorithm
- Can random walk give us better performance

The way how to apply random walk outside of the graph must be considered.