AN UPPER BOUND ON THE CONVERGENCE TIME FOR DISTRIBUTED BINARY VOTING ON AN ARBITRARY GRAPH

SHANG SHANG, PAUL W. CUFF, SANJEEV R. KULKARNI
THE DEPARTMENT OF ELECTRICAL ENGINEERING, PRINCETON UNIVERSITY

PROBLEM STATEMENT

The problem of interest in this work is the convergence time of the distributed binary consensus algorithm proposed by Benezit et al [1], which the nodes can update their states by communicating with neighbors via a 2-bit message in an asynchronous setting, as shown in Fig. 1.

MAIN RESULT

We bound the convergence time of the binary voting algorithm to $O(N \log N)$, for an arbitrary connected graph.

BACKGROUND

Opinions of individuals may be collected distributedly due to some restrictions, and a global consensus of a majority opinion is desired, such as in sensor detection, human decision-making, etc. This work is motivated by the distributed algorithm proposed by Florence Benezit et al [1]. The algorithm suggested in [1] reaches consensus on the quantized interval that contains the average almost surely. However, the convergence time is unknown. Moez Draief and Milan Vojnovic provided expected convergence time bound depending on network topologies and voting margin [2]. In this work, we are interested in an upper bound for the convergence speed of the distributed algorithm on an arbitrary graph.

PROBLEM FORMULATION

• Network Model: connected graph $G = (V, E)$.
  Nodes: $V = \{1, 2, ..., N\}$
  Edges: $E \subseteq V \times V$, $i \neq j$ can communicate with each other.

• Asynchronous Communication Model: each node asynchronously chooses one pairwise comparison at a time to exchange opinion with its neighbor.

PROBLEM FORMULATION CONT.

• Asynchronous Time Model:
  Local: rate 1 Poisson clock.
  Global: rate $N$ Poisson clock, ticking at $\{i\}_{i=0}^{\infty}$

When node $i$'s clock ticks, it randomly chooses one neighbor, $j$, to update their states according to the rules described in Fig. 2. We study the expected time this distributed algorithm takes for all the nodes to converge to the same sign of states.

DEFINITIONS

• Define Biased Random Walk $X_i$ with transition matrix $P_j = (P_{ij})$.

$$P_j^1 = 1 - \frac{1}{N} + \sum_{k \neq j} \frac{1}{N} \quad \text{for} \quad i \in V$$

$$P_j^2 = \left( \frac{1}{N} \right)^{-1} \left( \frac{1}{N} \right) + \sum_{k \neq j} \left( \frac{1}{N} \right)^{-1} \quad \text{for} \quad (i, j) \in E$$

• Define Simple Random Walk $X_i$ with transition matrix $P_j = (P_{ij})$.

$$P_j^0 = 0 \quad \text{for} \quad i \in V$$

• Define Natural Random Walk $X_i$ with transition matrix $P_j = (P_{ij})$.

$$P_j^1 = \frac{1}{N} \quad \text{for} \quad (i, j) \in E$$

METHODS

Suppose $|S.P| > |S.N.|$. Similar as in [2], we consider the system evolution over two phases: depletion of S.N. and depletion of W.N. Instead of using differential equation to solve the problem in [2], we model the random process as random walks on a weighted graph.

Figure 1. Network setup

S.P.: +2
W.P.: +1
W.N.: -1
S.N.: -2

Figure 2. Updating rules

CENTER FOR SCIENCE OF INFORMATION

A National Science Foundation Science and Technology Center

soihub.org

This research was supported in part by the Center for Science of Information (CSoI), an NSF Science and Technology Center, under grant agreement CCF-1053975, by NSF through the grant CCF-1110113, by the U.S. Army Research Office under grant number W911NF-07-1-0686, and by a research grant from Nokia Bell Labs, Inc.