Available online at www.sciencedirect.com
ScienceDirect
automatica
Automatica 43 (2007) 1192–1203
www.elsevier.com/locate/automatica

Quantized consensus☆

Akshay Kashyap*, Tamer Başar, R. Srikant

Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA

Received 22 August 2006; received in revised form 10 November 2006; accepted 3 January 2007
Available online 16 May 2007

Abstract

We study the distributed averaging problem on arbitrary connected graphs, with the additional constraint that the value at each node is an integer. This discretized distributed averaging problem models several problems of interest, such as averaging in a network with finite capacity channels and load balancing in a processor network.

We describe simple randomized distributed algorithms which achieve consensus to the extent that the discrete nature of the problem permits. We give bounds on the convergence time of these algorithms for fully connected networks and linear networks.

Keywords: Sensor fusion; Quantization; Distributed detection; Estimation algorithms; Stochastic systems; Random processes; Clock synchronization

1. Introduction

Consider a distributed network of agents, each of which initially has a numerical value—for example a sensor network in which each sensor has a measurement taken from the environment. A *distributed averaging algorithm* is a procedure using which the agents can exchange messages and update their values iteratively, so that eventually, each agent is able to compute the average of all initial values.

The computation of the average is important in many different contexts, such as information fusion in sensor networks (Boyd, Ghosh, Prabhakar, & Shah, 2005; Xiao, Boyd, & Lall, 2005), load balancing in processor networks (Bertsekas & Tsitsiklis, 1997; Ghosh & Muthukrishnan, 1996; Ghosh et al., 1999; Rabani, Sinclair, & Wanka, 1998; Subramanian & Scherson, 1994), clock synchronization (Akar & Shorten, 2006; Giridhar & Kumar, 2006), and multi-agent coordination and flocking (Bertsekas & Tsitsiklis, 1997; Blondel, Hendrickx, Olshevsky, & Tsitsiklis, 2005; Jadbabaie, Lin, & Morse, 2003; Moreau, 2005; Olfati-Saber & Murray, 2004; Savkin, 2004; Tsitsiklis, 1984).

Constraints on communication resources are a key factor in the design of a distributed averaging algorithm. Each agent may be able to communicate with only a small subset of all agents. The communication links between agents may not be reliable and may fail over the time-scale of the computation. It is therefore of interest to design distributed averaging algorithms in which each agent needs to communicate only with its immediate neighbors, and does not need to know any further information about the global structure of the network.

Several such algorithms have been studied in the papers cited above. In this paper, we address another communication constraint—that on the bit rates of the communication links in the network. Finite rate communication links require us to quantize the numerical values being exchanged and stored. In particular, it is not possible to exchange real values over finite rate links. We study a discrete version of the distributed averaging problem that models such quantization, and also has applications to load balancing in processor networks.

1.1. Outline

A brief outline of this paper is as follows: we start with a precise description of the discrete averaging problem in Section 1.2 and a summary of our results in Section 1.3.

☆ This paper was not presented at any IFAC meeting. This paper was recommended for publication in revised form by Associate Editor George Yin under the direction of Editor Ian Petersen.

* Corresponding author. Tel.: +1 217 333 7916; fax +1 217 244 1642.
E-mail addresses: kashyap@uiuc.edu (A. Kashyap), tbasar@control.csl.uiuc.edu (T. Başar), rsrikant@uiuc.edu (R. Srikant).

0005-1098/$ - see front matter
doi:10.1016/j.automatica.2007.01.002

We discuss some applications in Section 2 and review prior work in Section 3. We present our main results in Section 4, where we describe a class of discrete averaging algorithms which we call *quantized gossip* algorithms, and in Section 6, where we derive bounds on the convergence time of these algorithms. We give examples of quantized gossip algorithms in Section 5. We make some concluding remarks and discuss future work in Section 7.

1.2. Problem statement

We consider a network of N nodes, numbered 1 through N, the connections between which are specified by an undirected connected graph $\mathscr{G} = (V, E)$, where $V = \{1, \ldots, N\}$. There is an integer value associated with each node. Time is assumed to be discrete. We denote the value at node i at time t by $x[t]_i$, and the vector of values in the network by $x[t]=(x[t]_1, \ldots, x[t]_N)$. Let $S = \sum_i x[0]_i$, where $x[0]$ is the vector of initial values.

We describe algorithms in which nodes update their values using the values of their neighbors in $\mathscr{G}$ in such a way that eventually, the value of each node converges to an integer approximation of the average of the initial values, $(1/N)\sum_{i=0}^{N} x[0]_i$, under the further constraints that:

(1) The value at each node is always an integer.
(2) The sum of values in the network does not change with time: $\sum_i x[t]_i = S$ for all t.

Let S be written as $NL + R$, where L and R are integers with $0 \leqslant R < N$. We accept both L and $L + 1$ as integer approximations of the true average S/N. We define the *distribution* of a vector x as the list $\{(v_1, n_1), (v_2, n_2), \ldots\}$ in which n_i is the number of entries of x which have value v_i. We say that a vector x has a *quantized consensus distribution* if $x \in \mathscr{S}$ where

$$\mathscr{S} = \left\{ x | x_i \in \{L, L+1\}, i = 1, \ldots, N, \sum_{i=1}^{N} x_i = S \right\}. \qquad (1)$$

Similarly, we say that the network has reached *quantized consensus* when the vector of values $x[t]$ lies in the set $\mathscr{S}$.

For example, in a three node network, in which $x[0] = (x[0]_1, x[0]_2, x[0]_3) = (2, 3, 5)$, the vectors which have quantized consensus distributions are given by $(3, 3, 4)$, $(3, 4, 3)$ and $(4, 3, 3)$. For $x[0] = (2, 3, 4)$, the only such vector is $(3, 3, 3)$.

1.3. Contribution

The main contribution of this paper is the design of a class of simple distributed algorithms, which we call quantized gossip algorithms, that converge to the set of quantized consensus distributions for an arbitrary initial vector $x[0]$ and arbitrary connected graph $\mathscr{G}$. More generally, we point out certain mild conditions under which convergence to quantized consensus holds (Theorem 2 of Section 4), and describe some variations of quantized gossip algorithms that satisfy these conditions. We also derive bounds on the convergence time of quantized gossip algorithms.

2. Applications

2.1. Capacity and memory constrained sensor networks

Let the graph $\mathscr{G}$ model a network of N sensors, with each node representing a sensor. Sensor i can communicate with sensor j if $\{i, j\} \in \mathscr{G}$. Sensor i makes a measurement q_i, for $i = 1, \ldots, N$. We are interested in updating sensor values distributedly so that the value at each sensor converges to the average of the measurements, $(1/N)\sum_i q_i$.

The average of sensor measurements is a sufficient statistic for many problems of interest in sensor networks. The following are two examples:

Estimation: Assume that we are interested in estimating some parameter θ, and the sensor measurements are noisy versions of this parameter, $q_i = \theta + z_i$, where z_i are independent identically distributed zero mean Gaussian random variables. Then $(1/N)\sum_i q_i$ is the minimum variance unbiased estimator for θ (see Poor, 1994).

Detection: Assume that the nodes make measurements Y_i, which are independent and identically distributed conditioned on some state of nature H. H can take one of two values, H_0 and H_1, each with equal probability. The probability density of Y_1 (and therefore also of Y_i for any i) conditioned on the event $H = H_j$ is denoted by $p_j(y)$ for $j = 0, 1$. Let $q_i = \log L(Y_i)$, where $L(y) = p_1(y)/p_0(y)$ is the likelihood ratio of y. Then, it is well known (Poor, 1994) that the optimal decision is to detect H_0 if $(1/N)\sum_i q_i \leqslant 0$, and H_1 otherwise.

However, since both the capacity of the communication channels between sensors and the memory capacity of sensors are finite, it is not possible to exchange real numbers and arrive at the real valued average $(1/N)\sum_i q_i$. We assume that the sensors quantize their measurements and let $x[0]_i = Q(q_i)$, where

$$Q(s) = n \quad \text{if } s \in [(n - \tfrac{1}{2})\delta, (n + \tfrac{1}{2})\delta], \quad n \in \mathbb{Z}, \qquad (2)$$

denotes the quantization *level* of the measurement at sensor i.[1] Then, in a quantized consensus distribution of node values, each node has a quantizer-precision estimate of the sufficient statistic.

2.2. Load balancing

Let the nodes represent processors, connected as described by the graph $\mathscr{G}$, and let $x[0]_i$ be the number of tasks queued for processing at processor i, for $i = 1, \ldots, N$. The problem of load-balancing is one of equalizing the distribution of tasks over the processors. If the tasks are indivisible and of equal size, then a quantized consensus distribution represents such an equalized distribution of tasks.

[1] As such, this represents an infinite rate (uniform) quantizer. However, if for some $\Delta \in \mathbb{N}$, the measurements q_i always lie in the bounded set, $|q_i| \leqslant \Delta\delta$ for each $i = 1, \ldots, N$, then we can truncate $Q(.)$ as $Q(s) = \Delta$ if $s \geqslant (\Delta - \frac{1}{2})\delta$ and similarly on the negative half of the real line. The communication rate required then is $\log_2 \Delta + 1$ bits per channel use.