Automatica 48 (2012) 2344–2351

Contents lists available at SciVerse ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

Convergence time analysis of quantized gossip consensus on digraphs[☆]

Kai Cai[a,1], Hideaki Ishii[b,2]

[a] Department of Electrical and Computer Engineering, University of Toronto, 10 King's College Rd, Toronto, M5S 3G4, Canada

[b] Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology, 4259-J2-54, Nagatsuta-cho, Midori-ku, Yokohama 226-8502, Japan

ARTICLE INFO

Article history:
Received 20 October 2010
Received in revised form
17 November 2011
Accepted 17 April 2012
Available online 11 July 2012

Keywords:
Quantized consensus
Average consensus
Convergence time
Randomized gossip algorithms
Directed graphs
Markov chains

ABSTRACT

We have recently proposed quantized gossip algorithms which solve the consensus and averaging problems on directed graphs with the least restrictive connectivity requirements. In this paper we study the convergence time of these algorithms. To this end, we investigate the shrinking time of the smallest interval that contains all states for the consensus algorithm, and the decay time of a suitable Lyapunov function for the averaging algorithm. The investigation leads us to characterize the convergence time by the hitting time in certain special Markov chains. We simplify the structures of state transition by considering the special case of complete graphs, where every edge can be activated with an equal probability, and derive polynomial upper bounds on convergence time.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

Inspired by aggregate behavior of animal groups and motion coordination of robotic networks, the *consensus* problem has been extensively studied in the recent literature of systems control (e.g., Jadbabaie, Lin, and Morse (2003), Olfati-Saber and Murray (2004) and Ren and Beard (2005)). The objective of consensus is to have a population of nodes, each possessing an initial state, agree eventually on *some* common value through only local information exchange. This problem is also related intimately to oscillator synchronization and load balancing. The *averaging* problem is of a special form, with the goal to decentrally compute the *average* of all initial states at every node.

We have recently proposed in Cai and Ishii (2011a,b) randomized gossip algorithms which solve the consensus and averaging problems on directed graphs (or digraphs), under a quantization constraint that each node has an integer-valued state. In particular, our derived connectivity condition ensuring average consensus is weaker than those in the literature (Olfati-Saber & Murray, 2004), in the sense that it does not postulate balanced topologies. Here the main difficulty is that the state sum of nodes cannot be preserved during algorithm iterations. This scenario was previously considered in Aysal, Coates, and Rabbat (2008); Aysal, Yildiz, Sarwate, and Scaglione (2009), where averaging is guaranteed in expectation but there is in general an error in mean square and with probability one. By contrast, we overcome this difficulty by augmenting the so-called "surplus" variables for individual nodes so as to maintain local records of state updates, thereby ensuring average consensus almost surely.

In this paper and its conference precursor (Cai & Ishii, 2010), we investigate the performance of our proposed algorithms by providing upper bounds on the *mean convergence time*. The state transition structures resulting from these algorithms turn out to be rather complicated. Hence in our analysis on convergence time, we focus on the special case of complete graphs whose topology is undirected. The analysis is still challenging, but we will also discuss that the general approach can be useful for other graph topologies. First, for the consensus algorithm, we find that the mean convergence time is $O(n^2)$. To derive this bound, we view reaching consensus as the smallest interval containing all states shrinking its length to zero. This perspective leads us to characterizing convergence time by the *hitting* time in a certain Markov chain, which yields the polynomial bound. Second, we obtain that the mean convergence time of the averaging algorithm is $O(n^3)$. As the original algorithm in Cai and Ishii (2011a,b) is found to induce complex state transition structures, we have suitably revised it to manage the complexity. In particular, a bidirectional communication protocol is introduced. For the modified algorithm, a Lyapunov function is proposed which measures the distance from average consensus. We then bound convergence time by way

☆ This work was supported in part by the Ministry of Education, Culture, Sports, Science and Technology in Japan under Grant-in-Aid for Scientific Research, No. 21760323. The material in this paper was partially presented at the 49th IEEE Conference on Decision and Control (CDC 2010), December 15–17, 2010, Atlanta, Georgia, USA. This paper was recommended for publication in revised form by Associate Editor Valery Ugrinovskii under the direction of Editor Ian R. Petersen.

E-mail addresses: kai.cai@scg.utoronto.ca (K. Cai), ishii@dis.titech.ac.jp (H. Ishii).

[1] Tel.: +1 416 978 6299; fax: +1 416 978 6299.

[2] Tel.: +81 45 924 5371.

0005-1098/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2012.06.048

of bounding the number of iterations required to decrease the Lyapunov function; the latter is again characterized by the hitting time in a special Markov chain.

Our work is related to Lavaei and Murray (2012), Kashyap, Başar, and Srikant (2007) and Zhu and Martínez (2008), who also tackle the convergence time of gossip algorithms with quantized states. In Kashyap et al. (2007), polynomial bounds on convergence time are obtained for fully connected and linear networks. The work (Zhu & Martínez, 2008) generalizes these bounds to arbitrarily connected networks (fixed or switching). Also, bounds for arbitrarily connected networks are provided in Lavaei and Murray (2012); these bounds are in terms of graph topology. In these references, a common feature is that the graphs are undirected. To bound the convergence time, a frequently employed approach is to bound the decay time of a Lyapunov function (Kashyap et al., 2007; Nedic, Olshevsky, Ozdaglar, & Tsitsiklis, 2009). In particular, Nedic et al. (2009) derives tight polynomial bounds on the convergence time of synchronized averaging algorithms, with either real or quantized states. In addition, Carli, Fagnani, Frasca, and Zampieri (2010) investigates a variety of quantization effects on averaging algorithms, and demonstrates favorable convergence properties by simulations. Our work adopts the Lyapunov method, as in Kashyap et al. (2007) and Nedic et al. (2009); the common function used in these papers turns out, however, not to be a valid Lyapunov function for our averaging algorithm. This is due again to that the state sum does not remain invariant, and the augmented surplus evolution must also be taken into account. According to these features, we establish an appropriate Lyapunov function, and prove that bounding its decay time can be reduced to finding the hitting time in a certain Markov chain.

1.1. Setup and organization

Consider a digraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \ldots, n\}$ is the node set, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ the edge set. Each directed edge (j, i) in $\mathcal{E}$, pointing from j to i, denotes that node j communicates to node i (namely, the information flow is from j to i). Communication among the nodes is by means of *gossip*: At each time instant, exactly one edge $(j, i) \in \mathcal{E}$ is activated independently from all earlier instants and with a time-invariant positive probability $p_{ji} \in (0, 1)$ such that $\sum_{(j,i)\in\mathcal{E}} p_{ji} = 1$.

To model the quantization effect in information flow, we assume that at time $k \in \mathbb{Z}_+$ (nonnegative integers), each node has an integer-valued state $x_i(k) \in \mathbb{Z},\ i \in \mathcal{V}$; the aggregate state is denoted by $x(k) = [x_1(k) \cdots x_n(k)]^T \in \mathbb{Z}^n$. Let

$$\mathcal{X} := \{x : m \le x_i \le M,\ i \in \mathcal{V}\}, \tag{1}$$

for some (finite) constants m, M. Suppose throughout the paper that the initial state satisfies $x(0) \in \mathcal{X}$. Also, let $\mathbf{1} = [1 \cdots 1]^T$ be the vector of all ones.

For the convergence time analysis, we will impose the following two assumptions on the graph topology and the probability distribution of activating edges. Let $|\cdot|$ denote the cardinality of a set.

Assumption 1. The digraph $\mathcal{G}$ is *complete* (i.e., every node is connected to every other node by a directed edge). It follows that there are $|\mathcal{E}| = n(n-1)$ edges.

Assumption 2. The probability distribution on edge activation is *uniform*; namely, each edge can be activated with the same probability $p := 1/|\mathcal{E}|$.

The rest of this paper is organized as follows. In Section 2, we address the convergence time analysis for the consensus algorithm. Then in Sections 3 and 4, we derive an upper bound for the convergence time of the averaging algorithm. Further, we compare convergence rates through a numerical example in Section 5, and finally, we state our conclusions in Section 6.

2. Convergence time of consensus algorithm

2.1. Algorithm and problem formulation

First we recall the quantized consensus (**QC**) algorithm from Cai and Ishii (2011b). Suppose that the edge $(j, i) \in \mathcal{E}$ is randomly activated at time k. Along the edge node j sends to i its state information, $x_j(k)$, but does not perform any update, i.e., $x_j(k+1) = x_j(k)$. On the other hand, node i receives j's state $x_j(k)$ and updates its own as follows:

(R1) If $x_i(k) = x_j(k)$, then $x_i(k+1) = x_i(k)$;
(R2) if $x_i(k) < x_j(k)$, then $x_i(k+1) \in (x_i(k), x_j(k)]$;
(R3) if $x_i(k) > x_j(k)$, then $x_i(k+1) \in [x_j(k), x_i(k))$.

Let the subset $\mathscr{C}$ of $\mathbb{Z}^n$ be the set of general consensus states given by $\mathscr{C} := \{x : x_1 = \cdots = x_n\}$. If $x(k) = x^* \in \mathscr{C}$, then $x(k') = x^* \in \mathscr{C}$ for all $k' > k$ because only (R1) applies. We say that the nodes achieve general consensus almost surely if for each $x(0)$, $\Pr[(\exists T < \infty, x^* \in \mathscr{C})(\forall k \ge T) x(k) = x^*] = 1$, where the probability is with respect to the sequence of edges chosen in the algorithm. Under **QC** algorithm, a necessary and sufficient graphical condition that ensures almost sure general consensus is that the digraph $\mathcal{G}$ contains a *globally reachable node* (i.e., a node that is connected to every other node via a directed path) (Cai & Ishii, 2011b). Clearly if $\mathcal{G}$ is complete, then every node is globally reachable.

The *convergence time* of **QC** algorithm is the random variable T_{qc} defined by $T_{qc} := \inf\{k \ge 0 : x(k) \in \mathscr{C}\}$. The mean convergence time (with respect to the probability distribution on edge activation), starting from a state $x_0 \in \mathcal{X}$, is then given by

$$E_{qc}(x_0) := \mathbb{E}\left[T_{qc} | x(0) = x_0\right]. \tag{2}$$

Problem 1. Let Assumptions 1 and 2 hold. Find an upper bound of the mean convergence time $E_{qc}(x_0)$ of **QC** algorithm with respect to all initial states $x_0 \in \mathcal{X}$.

We now present the main result of this section.

Theorem 3. *Let Assumptions 1 and 2 hold. Then* $\max_{x_0\in\mathcal{X}} E_{qc}(x_0) < n(n-1)(M-m) = O(n^2)$.

To derive this bound, we first provide preliminaries on the *hitting time* in finite Markov chains.

2.2. Preliminaries on hitting time

Let $\{X_k\}_{k\ge0}$ be a Markov chain with a finite state space $\mathcal{S}$ and a transition probability matrix $P = (P_{ij})$ (e.g., Norris (1997)). The entry P_{ij} denotes the one-step transition probability from state i to state j. In particular, the diagonal entry P_{ii} denotes the *selfloop* transition probability. A state $i \in \mathcal{S}$ is said to be *absorbing* if $P_{ii} = 1$. For a given $\{X_k\}_{k\ge0}$, the *hitting time* of a subset $\mathcal{T}$ of $\mathcal{S}$ is the random variable $H_{\mathcal{T}}\left(\{X_k\}_{k\ge0}\right)$ defined by $H_{\mathcal{T}}\left(\{X_k\}_{k\ge0}\right) := \inf\{l \ge 0 : X_l \in \mathcal{T}\}$. The mean time (with respect to the probability distribution specified by P) taken for the chain, starting from a state $i \in \mathcal{S}$, to hit $\mathcal{T}$ is given by

$$\begin{aligned} E_i &:= \mathbb{E}\left[H_{\mathcal{T}}\left(\{X_k\}_{k\ge0}\right) | X_0 = i\right] \\ &= \sum_{l=0}^{\infty} l \cdot \mathbb{P}\left[H_{\mathcal{T}}\left(\{X_k\}_{k\ge0}\right) = l | X_0 = i\right], \end{aligned} \tag{3}$$

where $\mathbb{E}[\cdot|\cdot]$ and $\mathbb{P}[\cdot|\cdot]$ denote the conditional expectation and conditional probability operators, respectively.

Using a standard fact on mean hitting time (Norris, 1997, Theorem 1.3.5), we derive a closed-form expression of the mean hitting time for a specific Markov chain; this chain will be shown to characterize the state transition structure under **QC** algorithm. For the proof, refer to Cai and Ishii (2010).

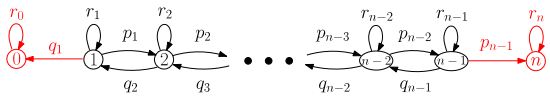

Fig. 1. Markov chain I: states 0 and n are absorbing. Here $r_0, \ldots, r_n$ are selfloop transition probabilities.

Lemma 4. *Consider the Markov chain in Fig. 1 with transition probabilities $r_0 = 1, r_n = 1, p_z + r_z + q_z = 1, p_z = q_z$ $(z = 1, \ldots, n-1)$. Then the mean hitting time of the state 0 or n starting from state z is*

$$E_z = \left(1 - \frac{z}{n}\right) \sum_{i=1}^{z-1} \frac{i}{p_i} + \frac{z}{n} \sum_{j=z}^{n-1} \frac{n-j}{p_j} \quad (z = 1, \ldots, n-1).$$

2.3. Analysis of convergence time

We now proceed as follows. For an arbitrary $x(k)$ define the minimum and maximum states by

$$m(k) := \min_{i \in \mathcal{V}} x_i(k), \qquad M(k) := \max_{i \in \mathcal{V}} x_i(k). \tag{4}$$

We view the state $x(k)$ converging to $\mathscr{C}$ as the interval $[m(k), M(k)]$ shrinking to length 0. Let the random variable T^1_{qc} be the time when one interval shrinkage occurs; then the corresponding mean time, starting from a state x, is $E^1_{qc}(x) := \mathbb{E}\left[T^1_{qc} | x \in \mathcal{X}\right]$. Since one shrinkage decreases the interval length by at least 1, there can be at most $M - m$ shrinkages for $x_0 \in \mathcal{X}$. It then follows that

$$\max_{x_0 \in \mathcal{X}} E_{qc}(x_0) \le \max_{x \in \mathcal{X}} E^1_{qc}(x) \cdot (M - m). \tag{5}$$

Consider a subset $\mathcal{X}_1$ of $\mathcal{X}$ defined by

$$\mathcal{X}_1 := \{x : x_1 = \cdots = x_z = 1 \;\&\; x_{z+1} = \cdots = x_n = 0, z \in [1, n-1]\}. \tag{6}$$

Thus the interval has length 1 for all $x \in \mathcal{X}_1$. It is not difficult to see that $\max_{x_0 \in \mathcal{X}_1} E_{qc}(x_0) = \max_{x \in \mathcal{X}} E^1_{qc}(x)$. The following lemma states an upper bound of $E_{qc}(x_0)$ for $x_0 \in \mathcal{X}_1$.

Lemma 5. *Let Assumptions 1 and 2 hold. Then* $\max_{x_0 \in \mathcal{X}_1} E_{qc}(x_0) < n(n-1) = O(n^2)$.

Proof. By Assumptions 1 and 2, every directed edge in $\mathcal{G}$ can be activated with the uniform probability $p = 1/(n(n-1))$. Starting from an arbitrary state in the set $\mathcal{X}_1$, the transition structure under **QC** algorithm is the Markov chain displayed in Fig. 1; in the diagram,

$$\begin{cases} \text{state } 0: & \text{the vector } \mathbf{0} = [0 \cdots 0]^T \text{ of all zeros,} \\ \text{state } n: & \text{the vector } \mathbf{1} = [1 \cdots 1]^T \text{ of all ones,} \\ \text{state } z: & \text{the vector } [\overbrace{1 \cdots 1}^{z}\ 0 \cdots 0]^T \text{ in } \mathcal{X}_1, \end{cases}$$

and the transition probabilities are $p_z = q_z = z(n-z)p, z \in [1, n-1]$. To see this, consider the transition from state z to state $z+1$; this occurs when an edge (j, i) is activated, with $x_j = 1$ and $x_i = 0$, so that (R2) of **QC** algorithm applies. Since there are $z(n-z)$ such edges, the transition probability $p_z = z(n-z)p$. Likewise, one may derive that the transition from state z to state $z-1$ is with probability $q_z = z(n-z)p$, which occurs when (R3) of **QC** algorithm applies. Now observe in Fig. 1 that the states $0, n \in \mathscr{C}$ and $1, \ldots, n-1 \in \mathcal{X}_1$; hence $\max_{z \in [1,n-1]} E_z = \max_{x_0 \in \mathcal{X}_1} E_{qc}(x_0)$, with E_z in (3). Invoking the formula of E_z in Lemma 4 for the transition probabilities, we obtain $E_z < n(n-1)$ for all $z \in [1, n-1]$. Therefore $\max_{x_0 \in \mathcal{X}_1} E_{qc}(x_0) < n(n-1) = O(n^2)$. □

Finally, our main result Theorem 3 follows immediately from Lemma 5 and (5).

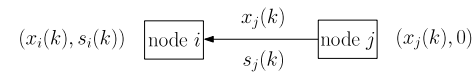

Fig. 2. Stage (I): node j sends to i its state and surplus through the edge (j, i).

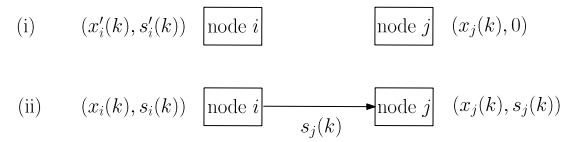

Fig. 3. Stage (II): either (i) node i updates its state and surplus, or (ii) it sends $s_j(k)$ back to node j through edge (i, j).

Remark 6. We discuss the idea of how this result for complete graphs might be extended to handle more general topologies. We still view reaching consensus as the interval $[m(k), M(k)]$ shrinking to length 0; thereby the inequality (5) holds. We then again consider the subset $\mathcal{X}_1$ given in (6), and as long as the digraph is strongly connected (i.e., every node is connected to every other node) one can verify that the state transition structure under **QC** algorithm is still the one in Fig. 1. The associated transition probabilities, however, depend crucially on topologies. In order to apply again Lemma 4 to derive bounds, it would be important to establish the relation between transition probabilities and graph topologies; this will be targeted in our future work.

3. Averaging algorithm and Lyapunov function

In this and next sections, we address the convergence time analysis for the quantized averaging (**QA**) algorithm, which is a modification of the one in Cai and Ishii (2011a,b). We start by presenting the modified algorithm, and formulate the corresponding time analysis problem. We then propose a Lyapunov function, which turns out to be a suitable measure for the average consensus error. In Section 4, we will derive an upper bound on the mean convergence time by means of bounding the decay time of the proposed Lyapunov function.

3.1. Algorithm and problem formulation

First we present **QA** algorithm. As in Cai and Ishii (2011a,b), since the state sum cannot be preserved at each time instant, we associate each node $i \in \mathcal{V}$ with an additional *surplus* variable, $s_i(k) \in \mathbb{Z}$, to locally record the changes of $x_i(k)$. The aggregate surplus is denoted by $s(k) = [s_1(k) \cdots s_n(k)]^T \in \mathbb{Z}^n$, whose initial value is set to be $s(0) = [0 \cdots 0]^T$. Now suppose that the edge $(j, i) \in \mathcal{E}$ is activated at time k. There are two stages: (I) Along the edge, node j sends to i its state $x_j(k)$ and surplus $s_j(k)$. Node j does not update its state, but sets its surplus to be 0 after transmission (see Fig. 2). (II) Based on the information received, node i determines either to update its state and surplus, or to send back to j the surplus $s_j(k)$ by activating the opposite edge (i, j) (see Fig. 3). Notice that the latter operation in (II) requires bidirectional communication between two nodes at a single time instant; this is possible in complete digraphs, but not in general strongly connected digraphs. Formally, **QA** algorithm is described as follows.

(R1) If $x_i(k) = x_j(k)$, then there are two cases:

(i) If $s_i(k) > 0 \;\&\; s_j(k) > 0$, then

$$x_i(k+1) = x_i(k), \qquad s_i(k+1) = s_i(k);$$
$$x_j(k+1) = x_j(k), \qquad s_j(k+1) = s_j(k).$$

(ii) Otherwise (i.e., either surplus equals zero),

$$x_i(k+1) = x_i(k), \qquad s_i(k+1) = s_i(k) + s_j(k) \in \{0, 1\};$$
$$x_j(k+1) = x_j(k), \qquad s_j(k+1) = 0.$$

(R2) If $x_i(k) < x_j(k)$, then there are two cases:

(i) If $s_i(k) + s_j(k) > 0$, then

$$x_i(k+1) = x_i(k) + 1,$$
$$s_i(k+1) = s_i(k) + s_j(k) - 1 \in \{0, 1\};$$
$$x_j(k+1) = x_j(k), \qquad s_j(k+1) = 0.$$

(ii) Otherwise (i.e., $s_i(k) + s_j(k) = 0$),

$$x_i(k+1) = x_i(k), \qquad s_i(k+1) = s_i(k) + s_j(k) = 0;$$
$$x_j(k+1) = x_j(k), \qquad s_j(k+1) = 0.$$

(R3) If $x_i(k) > x_j(k)$, then there are two cases:

(i) If $s_i(k) + s_j(k) = 0$, then

$$x_i(k+1) = x_i(k) - 1,$$
$$s_i(k+1) = s_i(k) + s_j(k) + 1 = 1;$$
$$x_j(k+1) = x_j(k), \qquad s_j(k+1) = 0.$$

(ii) Otherwise (i.e., $s_i(k) + s_j(k) > 0$),

$$x_i(k+1) = x_i(k), \qquad s_i(k+1) = s_i(k);$$
$$x_j(k+1) = x_j(k), \qquad s_j(k+1) = s_j(k).$$

In the algorithm, observe that (1) (R1)(i) and (R3)(ii) are where node i sends $s_j(k)$ back to node j in stage (II), which requires bidirectional communication; (2) only (R3)(i) 'generates' one surplus, and only (R2)(i) 'consumes' one surplus; (3) the quantity $(x+s)^T\mathbf{1}$ stays invariant, i.e., for every $k \geq 0$, $(x(k+1) + s(k+1))^T\mathbf{1} = (x(k) + s(k))^T\mathbf{1} = x(0)^T\mathbf{1}$. Distinct from the algorithm in Cai and Ishii (2011a,b), this **QA** algorithm does not involve the threshold constant and the local extrema variables, thus reducing individual computation effort. Also each surplus variable is indeed binary-valued, and therefore requires merely one bit for both storage and transmission. A further difference between the two algorithms lies in the use of surplus variables: The algorithm in Cai and Ishii (2011a,b) allows surpluses to pile up, which is indeed required to achieve average consensus for arbitrary strongly connected digraphs. By contrast, our **QA** algorithm here prevents surpluses from piling up, and meanwhile simplifies the transition structure.

Now let the subset $\mathscr{A}$ of $\mathbb{Z}^n \times \mathbb{Z}^n$ be the set of average consensus states given by $\mathscr{A} := \{(x, s) : x_i = \lfloor x(0)^T\mathbf{1}/n \rfloor \text{ or } \lceil x(0)^T\mathbf{1}/n \rceil, i \in \mathcal{V}\}$. If $(x(k), s(k)) \in \mathscr{A}$, we have $(x(k'), s(k')) \in \mathscr{A}$ for all $k' > k$ by inspecting the rules of **QA** algorithm (also see Cai and Ishii (2011b) for detailed justification). We say that the nodes achieve average consensus almost surely if for each $(x(0), s(0))$, $\Pr[(\exists T < \infty)(\forall k \geq T)(x(k), s(k)) \in \mathscr{A}] = 1$, where the probability is with respect to the sequence of edges chosen in the algorithm. Here is the convergence result of **QA** algorithm for complete digraphs.

Proposition 7. *Let Assumption 1 hold. Then, under* **QA** *algorithm, the nodes achieve average consensus almost surely.*

This convergence result may be justified by a similar argument as given in Cai and Ishii (2011a,b); some care, however, has to be taken for the operations on surplus variables, as pointed out above. In addition, we note that the convergence can also be implied by the time analysis using Lyapunov approach in Section 4 below.

The *convergence time* of **QA** algorithm is the random variable T_{qa} defined by $T_{qa} := \inf\{k \geq 0 : (x(k), s(k)) \in \mathscr{A}\}$. The mean time taken for this convergence (according again to the probability distribution on edge activation), starting from $(x_0, 0)$ with $x_0 \in \mathcal{X}$, is then given by

$$E_{qa}(x_0) := \mathbb{E}\left[T_{qa} | (x(0), s(0)) = (x_0, 0)\right]. \tag{7}$$

Problem 2. Let Assumptions 1 and 2 hold. Find an upper bound of the mean convergence time $E_{qa}(x_0)$ of **QA** algorithm with respect to all initial states $x_0 \in \mathcal{X}$.

Our main result is the following upper bound of $E_{qa}(x_0)$ with respect to all possible initial states $x_0 \in \mathcal{X}$.

Theorem 8. *Let Assumptions 1 and 2 hold. Then*

$$\max_{x_0 \in \mathcal{X}} E_{qa}(x_0) < n^2(n-1)\frac{3(M-m)}{2} + n(n-1)\frac{R(R-1)}{n-(R/2)} = O(n^3),$$

where $R \in [0, n-1]$ is an integer, as in (8) below.

We note that the order of this polynomial bound is the same as that in Kashyap et al. (2007) for undirected, complete graphs. To derive this bound, we will first propose a valid Lyapunov function for **QA** algorithm. Then we will upper bound the mean convergence time by way of upper bounding the mean decay time of the Lyapunov function.

3.2. Lyapunov function

We start by introducing two variables, called positive surplus S_+ and negative surplus S_-; they are global variables, but are needed only for the convergence time analysis. Write the initial state sum

$$x(0)^T\mathbf{1} = nL + R, \tag{8}$$

where $L := \lfloor x(0)^T\mathbf{1}/n \rfloor$ is one of the possible values for average consensus, and $0 \leq R < n$. Observe that when a surplus is generated/consumed, the corresponding state moves one-step either closer to or farther from the value L. Positive and negative surplus variables are used to identify these two directions. Concretely, when a surplus is generated, we increase S_+ (resp. S_-) if the corresponding state moves towards (resp. away from) L. On the other hand, when a surplus is consumed, we distinguish the following two situations: In one case where the state moves closer to L, we decrease S_- if it is nonzero, and S_+ otherwise; in the other case where the state moves away from L, we decrease only S_+.

We now formalize the updating rules of S_+ and S_-. Let $D(k) := \sum_{i=1}^n |x_i(k) - L|$ be the sum of average consensus errors, and suppose that the edge $(j, i) \in \mathcal{E}$ is activated at time k.

(S1) If (R3)(i) generates a surplus, there are two cases:

(i) If $D(k+1) = D(k) - 1$ (i.e., $x_i(k) > L$), then

$$S_+(k+1) = S_+(k) + 1.$$

(ii) If $D(k+1) = D(k) + 1$ (i.e., $x_i(k) \leq L$), then

$$S_-(k+1) = S_-(k) + 1.$$

(S2) If (R2)(i) consumes a surplus, there are two cases:

(i) If $D(k+1) = D(k) + 1$ (i.e., $x_i(k) \geq L$), then

$$S_+(k+1) = S_+(k) - 1.$$

(ii) If $D(k+1) = D(k) - 1$ (i.e., $x_i(k) < L$), then

$$S_-(k) = 0 \Rightarrow S_+(k+1) = S_+(k) - 1;$$
$$S_-(k) > 0 \Rightarrow S_-(k+1) = S_-(k) - 1.$$

(S3) Otherwise

$$S_+(k+1) = S_+(k); \qquad S_-(k+1) = S_-(k).$$

The case (S3) above includes (R1), (R2)(ii), and (R3)(ii) of **QA** algorithm; note that, in these cases, there is no state update. Since initially there is no surplus in the system (i.e., $s(0) = 0$), we set $S_+(0) = S_-(0) = 0$. Also, one may readily see that $S_+(k) + S_-(k) = s(k)^T\mathbf{1}$, which relates the global surpluses to the local ones.

We are ready to define the Lyapunov function $V(k)$, $k \geq 0$, which is given by

$$V(k) := D(k) + S_+(k) - S_-(k). \tag{9}$$

It is not difficult to see from (S1) to (S3) that $V(k)$ is non-increasing. Indeed, $V(k)$ stays put except for only one case – (S2)(ii) and negative surplus $S_-(k) = 0$ – where it decreases by 2, i.e., $V(k+1) = V(k) - 2$. Notice that after this decrement, $S_+(k+1) \geq 0$ and $S_-(k+1) = 0$.

Remark 9. We emphasize that the validity of $V(k)$ as a Lyapunov function is not restricted only to undirected graphs, since the updating rules (S2) and (S3) do not involve (R1)(i) and (R3)(ii) where bidirectional communication is required. Indeed, $V(k)$ is a suitable Lyapunov function for the original **QA** algorithm in Cai and Ishii (2011a,b), which can achieve average consensus on arbitrary strongly connected digraphs. This is one contribution of our work, which might also provide a preliminary to attack convergence time on more general topologies.

In the following lemma, we collect several useful implications from the definition of function $V(k)$.

Lemma 10. (1) *A lower bound of $V(k)$ is R, i.e., $V(k) \geq R$ for all k.*

(2) *If $V(k) = R$, then $S_-(k) = 0$, $S_+(k) \geq 0$, and $(\forall i \in [1, n])$ $x_i(k) \geq L$.*

(3) *If $D(k) = 0$, then $S_-(k) = 0$ and $V(k) = S_+(k) = R$.*

(4) *Suppose $R = 0$. Then $D(k) = 0$ if and only if $V(k) = 0$, and in both cases $S_-(k) = S_+(k) = 0$.*

Proof. Our proof is in this order: (2), (1), (3), and (4).

(2) Let $V(k) = R$. Then there must exist $k_0 \leq k$ such that $V(k_0 - 1) = R + 2$ and $V(k_0) = R$. Also we have $S_+(k_0) \geq 0$ and $S_-(k_0) = 0$. Now assume $x_1(k_0) < L$. Then $x_1(k_0) + \sum_{i=2}^n x_i(k_0) + s(k_0)^T\mathbf{1} = nL + R$. Rearranging the terms and by $s(k_0)^T\mathbf{1} = S_+(k_0) + S_-(k_0)$, we obtain $\sum_{i=2}^n x_i(k_0) - (n-1)L = (L - x_1(k_0)) + R - S_+(k_0)$. Thus $V(k_0) = (L - x_1(k_0)) + \sum_{i=2}^n x_i(k_0) + S_+(k_0) - S_-(k_0) = 2(L - x_1(k_0)) + R > R$. This contradicts $V(k_0) = R$, and hence $x_i(k_0) \geq L$ for all i. The latter holds also for time k because the minimum states are non-decreasing by **QA** algorithm. Finally, according to the updating rules of S_+ and S_-, one may easily see that $S_-(k) = 0$ and $S_+(k) \geq 0$.

(1) When $V(k) = R$, every state $x_i(k) \geq L$ and consequently (S3)(ii) cannot occur. As $V(k)$ is non-increasing, it is lower bounded by R.

(3) Let $D(k) = 0$. Then $x(k)^T\mathbf{1} = nL$, and thus $S_+(k) + S_-(k) = s(k)^T\mathbf{1} = R$. It follows that $V(k) = S_+(k) - S_-(k) \leq R$. But $V(k) \geq R$, so that necessarily $V(k) = S_+(k) - S_-(k) = R$, which also implies that $S_-(k) = 0$ and $S_+(k) = R$.

(4) Assume $R = 0$. (Only if) The conclusion follows immediately from (3). (If) Let $V(k) = 0$. Then there must exist $k_0 \leq k$ such that $V(k_0 - 1) = 2$ and $V(k_0) = 0$. Also we have $S_+(k_0) \geq 0$ and $S_-(k_0) = 0$. Hence $D(k_0) + S_+(k_0) = 0$, which results in $D(k_0) = S_+(k_0) = 0$. As average consensus is achieved at k_0, the conclusion for time k follows. □

Next, we find an upper bound for the function $V(k)$.

Proposition 11. *Let $x(0) \in \mathcal{X}$ in (1). Then for every $k \geq 0$, $V(k) \leq ((M - m)n/2) + R$.*

Proof. Since the function $V(k)$ is non-increasing, it suffices to find an upper bound for $V(0) = \sum_{i=1}^n |x_i(0) - L|$. Consider the function $V(0) - R$; it is convex in $x(0)$, and $\mathcal{X}$ is a convex set. Hence, one of the extreme points of $\mathcal{X}$ is a maximizer. Fix $r \in [1, n]$, and let $x(0) \in \mathcal{X}$ be such that $x_1(0) = \cdots = x_r(0) = m$ and $x_{r+1}(0) = \cdots = x_n(0) = M$. Then $V(0) - R = r(L - m) + (n - r)(M - L) - R$. Also we have $L = (\mathbf{1}^T x(0) - R)/n = (rm + (n - r)M - R)/n$. Substituting this into the above equation and rearranging the terms, we derive

$$\begin{aligned} V(0) - R &= -\frac{2(M-m)}{n} r^2 + \left(2(M-m) - 2\frac{R}{n}\right) r \\ &\leq \frac{2(M-m)}{n} \cdot \frac{1}{4}\left(n - \frac{R}{M-m}\right)^2 \\ &\leq \frac{1}{2}\frac{(n(M-m))^2}{n(M-m)} = \frac{(M-m)n}{2}. \end{aligned}$$

Thus $V(k) - R$ is upper bounded by $(M - m)n/2$, which is achievable if and only if $R = 0$ and $r = n/2$. □

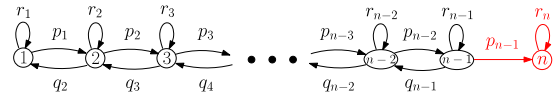

Fig. 4. Markov chain II: state n is absorbing.

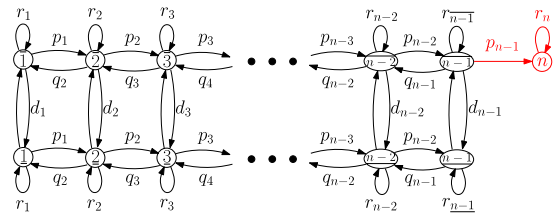

Fig. 5. Markov chain III: state n is absorbing.

4. Convergence time analysis of QA algorithm

We turn now to analyzing the mean convergence time of **QA** algorithm, by way of upper bounding the mean decay time of the Lyapunov function $V(\cdot)$ in (9). This Lyapunov approach is also adopted in Kashyap et al. (2007) and Nedic et al. (2009); the common function used is $V'(k) = \sum_{i=1}^n (x_i(k) - x(0)^T\mathbf{1}/n)^2$. It can be verified that $V'(k)$ is, however, not a valid Lyapunov function with respect to our **QA** algorithm. This is due to that the state sum is not preserved in each iteration and the surplus evolution must also be taken into account, as in our function $V(k)$.

4.1. Preliminaries on hitting time

As in Section 2.2, we provide preliminaries on the hitting time in finite Markov chains, specific for the time analysis of **QA** algorithm. For the proofs, see Cai and Ishii (2010).

Lemma 12. *Consider the Markov chain in Fig. 4 with transition probabilities $p_1 + r_1 = 1$, $p_z + r_z + q_z = 1$ $(z = 2, \ldots, n-1)$, $r_n = 1$. Then the mean hitting times of the state n starting from state 1 and z are respectively*

$$E_1 = \sum_{l=2}^{n-1}\left[\left(\prod_{i=2}^{l}\frac{q_i}{p_i}\right)\cdot\frac{1}{p_1} + \sum_{j=2}^{l}\left(\prod_{i=j+1}^{l}\frac{q_i}{p_i}\right)\cdot\frac{1}{p_j}\right] + \frac{1}{p_1},$$

$$E_z = \sum_{l=z}^{n-1}\left[\left(\prod_{i=2}^{l}\frac{q_i}{p_i}\right)\cdot\frac{1}{p_1} + \sum_{j=2}^{l}\left(\prod_{i=j+1}^{l}\frac{q_i}{p_i}\right)\cdot\frac{1}{p_j}\right],$$

where $z = 2, \ldots, n-1$.

Lemma 13. *Consider the Markov chain in Fig. 5 with transition probabilities $p_1 + r_1 + d_1 = 1$, $p_z + r_z + q_z + d_z = 1$ $(z = 2, \ldots, n-2)$, $r_{\underline{n-1}} + q_{n-1} + d_{n-1} = 1$, $p_{n-1} + r_{\overline{n-1}} + q_{n-1} + d_{n-1} = 1$, $r_n = 1$. Here $\underline{\cdot}$ and $\overline{\cdot}$ denote the states of the lower and upper rows, respectively.*

Then for states $\underline{n-1}$ and $\overline{n-1}$, their mean hitting times of the absorbing state n are

$$E_{\overline{n-1}} = \left(\prod_{i=2}^{n-1}\frac{q_i}{p_i}\right)\cdot\frac{2}{p_1} + \sum_{j=2}^{n-1}\left(\prod_{i=j+1}^{n-1}\frac{q_i}{p_i}\right)\cdot\frac{2}{p_j},$$

$$E_{\underline{n-1}} < \left(1+\frac{p_{n-1}}{d_{n-1}}\right)E_{\overline{n-1}}.$$

In the rest of this section, the proof of our main result Theorem 8 is given. We will need the following notation. Define the random variable $T_V := \inf\{k \geq 0 : V(k) = R\}$; thus T_V is the time when $V(\cdot)$ decreases to R. The mean decay time, starting from $(x_0, 0)$, is then given by

$$E_V(x_0) := \mathbb{E}\left[T_V | (x(0), s(0)) = (x_0, 0)\right]. \tag{10}$$

Now recall R from Eq. (8); we proceed with two cases in this order: $R = 0$ and $R > 0$. When $R = 0$ the mean convergence time $E_{qa}(x_0)$ is found to satisfy $E_{qa}(x_0) = E_V(x_0)$, whereas when $R > 0$ we have $E_{qa}(x_0) \geq E_V(x_0)$ in general and the corresponding analysis turns out to be based on the former case.

4.2. Proof of the case $R = 0$

In this case, the mean convergence time $E_{qa}(x_0)$ is characterized by the mean time that the function $V(k)$ decays to 0; that is, $E_{qa}(x_0) = E_V(x_0)$ in (10). This is because by Lemma 10(4), $V(k) = 0$ if and only if $D(k) = 0$, and the latter implies $(x(k), s(k)) \in \mathscr{A}$. As each decrement reduces $V(k)$ by 2, the initial value $V(0)$ is necessarily even, and there need in total $V(0)/2$ decrements.

To upper bound $E_V(x_0)$, we view the decay of $V(k)$ as the descent of *level sets* in the $(n+2)$-dimensional space of the triples $u := (x, S_+, S_-)$ (see Fig. 6). In this space, the average consensus state is simply the point $(L\mathbf{1}, 0, 0)$. Define the level sets $\mathcal{U}_l := \{u : V = \sum_{i=1}^n |x_i - L| + S_+ - S_- = 2\cdot l\}$, $l = 1, \ldots, V(0)/2$. Thus when $u(k) \in \mathcal{U}_l$, we interpret that $(x(k), s(k))$ is l-step away from $\mathscr{A}$ (i.e., $V(k)$ requires l decrements to reach 0). Also, it is important to note that on every level set $\mathcal{U}_l$, the triple evolution may start, and may descend to the next level, only from a strict subset $\mathcal{U}_l^0$ defined by $\mathcal{U}_l^0 := \{u \in \mathcal{U}_l : S_- = 0 \,\&\, S_+ \geq 0\}$. To see this, first recall that the decrement of $V(\cdot)$ (i.e., level set descent) requires $S_- = 0$ and $S_+ > 0$. Moreover, for the outmost level $\mathcal{U}_{V(0)/2}$, the initial triple is of the form $(x_0, 0, 0)$; and for each subsequent level, the triple evolution starts right after descending from the preceding level, where we have $S_- = 0$ and $S_+ \geq 0$.

Now let the random variable T_1 be the time of *one decrement* of $V(\cdot)$. The corresponding mean time, starting from a triple $u \in \mathcal{U}_l^0$, is then given by $E_1^l(u) := \mathbb{E}\left[T_1 | u \in \mathcal{U}_l^0\right]$, $l \in [1, V(0)/2]$. Since the initial value $V(0)$ is upper bounded by $(M-m)n/2$ (Proposition 11), the function $V(\cdot)$ requires at most $(M-m)n/4$ decrements to reach 0. Hence, an upper bound of its mean decay time is the following:

$$\max_{x_0\in\mathcal{X}} E_V(x_0) \leq \max_{l\in[1,V(0)/2], u\in\mathcal{U}_l^0} E_1^l(u)\cdot\frac{(M-m)n}{4}. \tag{11}$$

Proposition 14. *Let Assumptions 1 and 2 hold. Then* $\max_{l\in[1,V(0)/2], u\in\mathcal{U}_l^0} E_1^l(u) < 6n(n-1) = O(n^2)$.

To prove Proposition 14, it suffices to establish

$$\max_{u\in\mathcal{U}_l^0} E_1^l(u) < 6n(n-1) = O(n^2), \tag{12}$$

for every $l \in [1, V(0)/2]$. In the sequel we will provide the proof for the case $l = 1$ (i.e., one step away from average consensus), which contains the essential idea of our argument. Specifically, we first exhaust the possible triple evolution under **QA** algorithm, second derive the evolution structure and transition probabilities, and third calculate the corresponding mean hitting time. The analysis of the case $l \geq 2$ follows in a similar fashion but is more involved; we refer to Cai and Ishii (2010) for the proof.

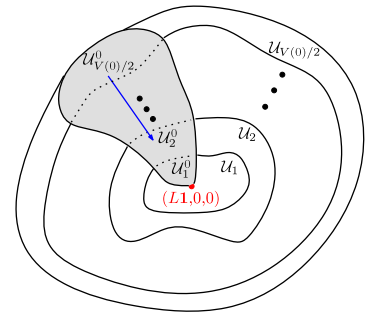

Fig. 6. Decay of $V(k)$ viewed as level set descent in the $(n+2)$ dimensions of (x, S_+, S_-). Descending is possible only from the shaded area and through the dotted curves.

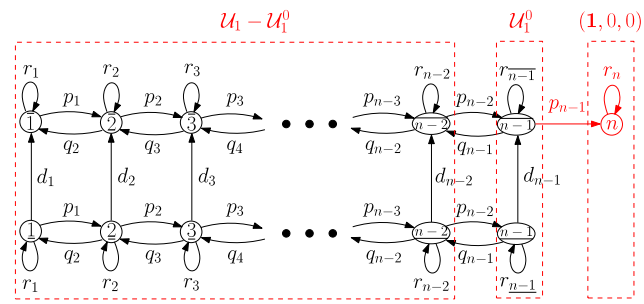

Fig. 7. One step away: from $\mathcal{U}_1$ to $(\mathbf{1}, 0, 0)$.

Proof for the case $l = 1$. Without loss of generality let $L = 1$. We investigate the triple evolution from the level set $\mathcal{U}_1$, starting in $\mathcal{U}_1^0$, to the average consensus state $(\mathbf{1}, 0, 0)$. By Assumptions 1 and 2, every directed edge in $\mathcal{G}$ can be activated with the uniform probability $p = 1/(n(n-1))$. Consider the triple $([2\ \overbrace{1\cdots1}^{n-2}\ 0]^T,\ 0,\ 0) \in \mathcal{U}_1^0$; we show that either S_- or S_+ can be generated. Case 1: an edge (j, i) is activated, with $x_j = 0$ and $x_i = 1$. In this case, (R3)(i) of **QA** algorithm applies, and the resulting triple is $([2\ \overbrace{1\cdots1}^{n-3}\ 0\ 0]^T,\ 0,\ 1) \in \mathcal{U}_1 - \mathcal{U}_1^0$. There are $n-2$ such edges; so the probability of this transition is $(n-2)p$. In fact, such transitions can continue until all the ones become zeros, generating in total $S_- = n-2$. Case 2: an edge (j, i) is activated, with $x_j = 0$ or 1 and $x_i = 2$. Again (R3)(i) of **QA** algorithm applies, the resulting triple being $([\overbrace{1\ 1\cdots1}^{n-1}\ 0]^T,\ 1,\ 0) \in \mathcal{U}_1^0$. This transition is with probability $(n-1)p$, since there are $n-1$ such edges. Now starting from the triple $([\overbrace{1\ 1\cdots1}^{n-1}\ 0]^T,\ 1,\ 0)$, on one hand, we can have a similar process, as from $([2\ \overbrace{1\cdots1}^{n-2}\ 0]^T,\ 0,\ 0)$ described above, generating in total $S_- = n-2$. On the other hand, observe that there is only one edge (j, i) such that $x_j = 1$, $s_j = 1$, and $x_i = 0$, $s_i = 0$. If this edge is activated (with probability p), then (R2)(i) of **QA** algorithm applies, which results in the average consensus state $(\mathbf{1}, 0, 0)$.

Based on the above descriptions, we derive that the transition structure from $\mathcal{U}_1$ to $(\mathbf{1}, 0, 0)$ under **QA** algorithm is the one displayed in Fig. 7. In this diagram, the state n is the average

consensus state $(\mathbf{1}, 0, 0)$, and the other states belong to $\mathcal{U}_1$, listed below:

$$\begin{cases} \underline{n-1}: & ([2\ 1\ 1\ \cdots\ 1\ 1\ 0]^T,\ 0,\ 0) \\ \underline{n-2}: & ([2\ 1\ 1\ \cdots\ 1\ 0\ 0]^T,\ 0,\ 1) \\ \vdots & \\ \underline{2}: & ([2\ 1\ 0\ \cdots\ 0\ 0\ 0]^T,\ 0,\ n-3) \\ \underline{1}: & ([2\ 0\ 0\ \cdots\ 0\ 0\ 0]^T,\ 0,\ n-2) \end{cases}$$

$$\begin{cases} \overline{n-1}: & ([1\ 1\ 1\ \cdots\ 1\ 1\ 0]^T,\ 1,\ 0) \\ \overline{n-2}: & ([1\ 1\ 1\ \cdots\ 1\ 0\ 0]^T,\ 1,\ 1) \\ \vdots & \\ \overline{2}: & ([1\ 1\ 0\ \cdots\ 0\ 0\ 0]^T,\ 1,\ n-3) \\ \overline{1}: & ([1\ 0\ 0\ \cdots\ 0\ 0\ 0]^T,\ 1,\ n-2). \end{cases}$$

Note that negative surplus is zero ($S_- = 0$) only in the states $\underline{n-1}$ and $\overline{n-1}$; hence these two triples are in $\mathcal{U}_1^0$. Also, one may verify that the transition probabilities are as follows: $p_1 = (n-2)p$, $d_1 = p$, $p_{n-1} = p$, $q_{n-1} = (n-2)p$, $d_{n-1} = (n-1)p$, $p_z = (n-1-z)zp$, $q_z = (z-1)p$, $d_z = zp (z = 2, \ldots, n-2)$. To upper bound $E_1^1(u)$ for $u \in \mathcal{U}_1^0$, in Fig. 7 we add transitions from the state $\overline{z}$ to $\underline{z}$ with the probability d_z, $z \in [1, n-1]$, thereby increasing the probabilities of moving away from the average consensus state n. This modification leads us to the same structure displayed in Fig. 5; thus, we have $\max_{u\in\mathcal{U}_1^0} E_1^1(u) \le E_{\underline{n-1}}$, where $E_{\underline{n-1}}$ is given in (3).

It is left to calculate $E_{\underline{n-1}}$ with respect to the current transition probabilities; for this the formulas in Lemma 13 are invoked. We refer to Cai and Ishii (2010) for the calculation process, which gives $E_{\underline{n-1}} < (1 + (p_{n-1}/d_{n-1}))\, E_{\overline{n-1}} < 6n(n-1) = O(n^2)$. □

Therefore, it follows from Proposition 14 and Eq. (11) that the upper bound of $E_{qa}(x_0)$ in Theorem 8 holds for the case $R = 0$.

4.3. Proof of the case $R \in [1, n-1]$

When $R \neq 0$, we have $E_{qa}(x_0) \ge E_V(x_0)$ in general. This is because $V(k) = R$ does not generally imply $(x(k), s(k)) \in \mathscr{A}$, and even after $V(k)$ reaches its lower bound R (Lemma 10(1) and (2)), the pair $(x(k), s(k))$ may require extra time to reach $\mathscr{A}$. Define the level set $\mathcal{U}_R := \{u : V = \sum_{i=1}^n |x_i - L| + S_+ - S_- = R\}$; then the mean convergence time starting from a triple $u \in \mathcal{U}_R$ is given by $E_{qa}(u) := \mathbb{E}\left[T_{qa} | u \in \mathcal{U}_R\right]$. Also recall from (10) that $E_V(x_0)$, with $x_0 \in \mathcal{X}$ in (1), denotes the mean decay time of $V(k)$ to the lower bound R. From these we obtain the mean convergence time of **QA** algorithm

$$\max_{x_0\in\mathcal{X}} E_{qa}(x_0) \le \max_{x_0\in\mathcal{X}} E_V(x_0) + \max_{u\in\mathcal{U}_R} E_{qa}(u). \tag{13}$$

In the sequel, we find upper bounds for $E_V(x_0)$ and $E_{qa}(u)$, respectively. First, as in the case $R = 0$, we have

$$\max_{x_0\in\mathcal{X}} E_V(x_0) < n^2(n-1)\frac{3(M-m)}{2} = O(n^3). \tag{14}$$

This is due to the following reason. The function $V(k)$ decays from its initial value $V(0)$ to R, and each decrement reduces $V(k)$ by 2. It follows that $V(0) - R$ is necessarily even and there need in total $(V(0)-R)/2$ decrements. For $l \in [1, (V(0)-R)/2]$ recall that $E_1^l(u)$ denotes the mean time spent for one decrement of $V(k)$, starting from a triple $u \in \mathcal{U}_l^0$. Following Proposition 14, one may similarly derive that $\max_{l\in[1,(V(0)-R)/2], u\in\mathcal{U}_l^0} E_1^l(u) < 6n(n-1)$. Moreover, $V(0) - R \le (M-m)n/2$ by Proposition 11; thus $V(k)$ requires at most $(M-m)n/4$ decrements to reach R. Therefore, $\max_{x_0\in\mathcal{X}} E_V(x_0) \le \max_{l\in[1,(V(0)-R)/2], u\in\mathcal{U}_l^0} E_1^l(u) \cdot (M-m)n/4 < n^2(n-1)\, 3(M-m)/2 = O(n^3)$.

Next, we find an upper bound for $\max_{u\in\mathcal{U}_R} E_{qa}(u)$. By Lemma 10 (2) we have $(\forall i \in \mathcal{V})\ x_i \ge L$; so the maximum state $M(k)$ in (4) satisfies $M(k) \in [L, L+R]$. If $R = 1$, then in fact $(x(k), s(k)) \in \mathscr{A}$; thus in this case $E_{qa}(u_R) = 0$, and we have from (13) and (14) that $\max_{x_0\in\mathcal{X}} E_{qa}(x_0) = O(n^3)$. It is left to consider $R \in [2, n-1]$. Since $M(k) = L$ or $L+1$ implies $(x(k), s(k)) \in \mathscr{A}$, the mean convergence time $E_{qa}(u)$ can be characterized by the mean time that $M(k)$ decays to $L+1$. Clearly $M(k)$ requires at most $R-1$ decrements to reach $L+1$. Let $E_M(u)$ denote the mean time taken for one decrement of $M(k)$, starting from a triple $u \in \mathcal{U}_R$. Then an upper bound for $E_{qa}(u)$ is as follows:

$$\max_{u\in\mathcal{U}_R} E_{qa}(u) \le \max_{u\in\mathcal{U}_R} E_M(u) \cdot (R-1). \tag{15}$$

Proposition 15. *Let Assumptions 1 and 2 hold.*
Then $\max_{u\in\mathcal{U}_R} E_M(u) < n(n-1)R/(n-(R/2)) = O(n^2)$.

To prove Proposition 15, we first find the subset in which one decay of $M(k)$ takes the longest time, second derive the transition structure and probabilities under **QA** algorithm, and third compute the mean hitting time.

Proof of Proposition 15. Suppose that R is even, and let $\mathcal{U}_e$ be a subset of $\mathcal{U}_R$ given by $\mathcal{U}_e := \{u = (x, S_+, S_-) : x \in \mathcal{X}_e,\ S_+ = S_- = 0\}$, where $\mathcal{X}_e := \{x : x_1 = \cdots = x_{R/2} = L+2,\ \ x_{(R/2)+1} = \cdots = x_n = L\}$. For a state in $\mathcal{X}_e$, one decrement of its maximum value $L+2$ occurs only when all the $R/2$ state components having that value decrease; thus it is not hard to see $\max_{u\in\mathcal{U}_R} E_M(u) = \max_{u\in\mathcal{U}_e} E_M(u)$.

Now pick an arbitrary triple u in $\mathcal{U}_e$; we investigate its evolution under **QA** algorithm. If an edge (j, i) is activated, with $x_j = L$ and $x_i = L+2$, then (R3)(i) of **QA** algorithm applies, and the resulting triple is $([\overbrace{L+2\ \cdots L+2}^{(R/2)-1}\ L+1\ L\ \cdots L]^T,\ 1,\ 0)$. Namely, one maximum state decreases. Also observe that there are $(R/2)\,(n-(R/2))$ such edges; so the probability of this transition is $(R/2)\,(n-(R/2))\,p$, where $p = 1/(n(n-1))$ by Assumptions 1 and 2. Indeed, this process can continue until all the $R/2$ maximum states decrease to the value $L+1$, and we derive that the corresponding transition structure under **QA** algorithm is the one displayed in Fig. 4 with the length $n = (R/2)+1$. In the diagram,

$$\begin{cases} \text{state } 1: & ([\overbrace{L+2\ \cdots\ L+2}^{R/2}\ L\ \cdots\ L]^T, 0, 0) \\ \text{state } 2: & ([L+2\ \cdots\ L+1\ L\ \cdots\ L]^T, 1, 0) \\ \vdots & \\ \text{state } \dfrac{R}{2}: & \left([L+2\ \cdots\ L+1\ L\ \cdots\ L]^T, \dfrac{R}{2}-1, 0\right) \\ \text{state } \dfrac{R}{2}+1: & \left([L+1\ \cdots\ L+1\ L\ \cdots\ L]^T, \dfrac{R}{2}, 0\right) \end{cases}$$

and transition probabilities are $p_1 = (R/2)\,(n-(R/2))\,p$, $p_z = ((R/2)-z+1)(n-(R/2))p$, $q_z = (z-1)((R/2)-z+1)p$, $z \in [2, R/2]$. Observe that the state $1 \in \tilde{\mathcal{U}}_R$ and the state $(R/2)+1 \in \mathcal{A}$; so $\max_{u\in\mathcal{U}_e} E_M(u) = E_1$, where E_1 is from (3). It remains to invoke the formulas in Lemma 12 to calculate E_1. We refer to Cai and Ishii (2010) for the calculation process, which yields $\max_{u\in\mathcal{U}_R} E_M(u) < n(n-1)R/(n-(R/2)) = O(n^2)$.

The analysis of the other case when R is odd is analogous, by considering another subset $\mathcal{U}_o := \{u = (x, S_+, S_-) : x \in \mathcal{X}_o,\ S_+ = S_- = 0\}$, where $\mathcal{X}_o := \{x : x_1 = \cdots = x_{(R-1)/2} = L+2,\ \ x_{(R+1)/2} = L+1,\ \ x_{(R+1)/2+1} = \cdots = x_n = L\}$. □

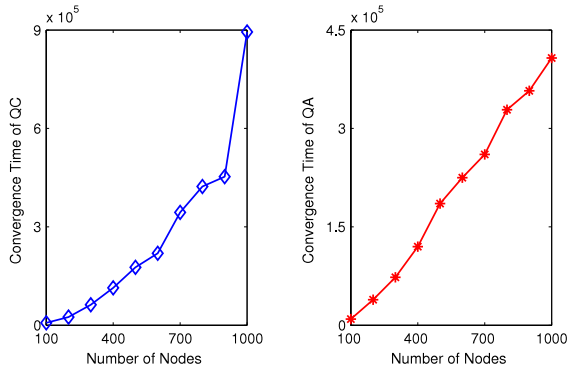

Fig. 8. Convergence time of **QC** and **QA**.

Finally, it follows from Eqs. (13) to (15) and Proposition 15 that an upper bound of the mean convergence time $E_{qa}(x_0)$ of **QA** algorithm is $E_{qa}(x_0) < 3n^2(n-1)(M-m)/2 + n(n-1)R(R-1)/(n-(R/2)) = O(n^3)$ when $R > 0$. This completes the proof of Theorem 8.

Remark 16. We have derived an upper bound for the convergence time of **QA** algorithm on complete graphs, by proposing a suitable Lyapunov function for the algorithm and characterizing a Markov chain for the state-surplus transition structure. To extend this result to more general topologies is, however, rather difficult. While the Lyapunov function is still valid (see Remark 9) which in turn validates inequalities (11) and (13), it is challenging to establish the relation between graph topologies and the transition structure with associated probabilities, as done in the proofs of Propositions 14 and 15 for complete graphs.

5. Numerical example

We have proved polynomial upper bounds on the convergence time of **QC** and **QA** algorithms for complete digraphs. Now we compare these theoretic bounds with numerical simulations, so as to illustrate the tightness of our derived results. For this purpose, we use the initial states $x(0)$ which have been shown in the technical proofs to give the worst case convergence times: $x(0) = [\overbrace{1\cdots1}^{\lfloor n/2\rfloor}\ 0\cdots0]^T$ for **QC** algorithm from Lemma 5, and $x(0) = [2\ \overbrace{1\cdots1}^{n-2}\ 0]^T$ for **QA** algorithm from Proposition 14. The simulation results are displayed in Fig. 8, each plotted value being the mean convergence time of 100 runs of the corresponding algorithms.

It is observed that the convergence rate of **QC** algorithm is approximately quadratic, which demonstrates that the derived theoretic bound is relatively tight. On the other hand, the convergence rate of **QA** algorithm appears to be at most quadratic, if not linear. This indicates that the cubic theoretic bound may not be tight, though it is in the same order as the one in Kashyap et al. (2007) also for complete graphs. Thus, deriving tighter bounds for the convergence time of **QA** algorithm awaits further effort. In addition, more extensive numerical studies are aimed at in our future work.

6. Conclusions

We have studied convergence time of the quantized gossip algorithms in Cai and Ishii (2011a,b) which solve the consensus and averaging problems. Polynomial upper bounds on the mean convergence time have been derived for the special case of complete digraphs, where the problem becomes tractable. In future work, it is of interest to analyze convergence time of our gossip algorithms on more general topologies, similar to the work of Nedic et al. (2009) and Zhu and Martínez (2008).

References

Aysal, T. C., Coates, M. J., & Rabbat, M. G. (2008). Distributed average consensus with dithered quantization. *IEEE Transactions on Signal Processing*, *56*(10), 4905–4918.

Aysal, T. C., Yildiz, M. E., Sarwate, A. D., & Scaglione, A. (2009). Broadcast gossip algorithms for consensus. *IEEE Transactions on Signal Processing*, *57*(7), 2748–2761.

Cai, K., & Ishii, H. (2010). Convergence time analysis of quantized gossip algorithms on digraphs. In *Proceedings of the 49th IEEE conference on decision and control*, (pp. 7669–7674). Atlanta, GA. Also Technical Report (available at http://arxiv.org/abs/1105.1668).

Cai, K., & Ishii, H. (2011a). Further results on randomized quantized averaging: a surplus-based approach. In *Proceedings of the IEEE multi-conference on systems and control*. (pp. 1558–1563). Yokohama, Japan. 2010. Also *SICE Journal of Control, Measurement, and System Integration*, 4(3). (pp. 236–242).

Cai, K., & Ishii, H. (2011b). Quantized consensus and averaging on gossip digraphs. *IEEE Transactions on Automatic Control*, *56*(9), 2087–2100.

Carli, R., Fagnani, F., Frasca, P., & Zampieri, S. (2010). Gossip consensus algorithms via quantized communication. *Automatica*, *46*(1), 70–80.

Jadbabaie, A., Lin, J., & Morse, A. S. (2003). Coordination of groups of mobile autonomous agents sing nearest neighbor rules. *IEEE Transactions on Automatic Control*, *48*(6), 988–1001.

Kashyap, A., Başar, T., & Srikant, R. (2007). Quantized consensus. *Automatica*, *43*(7), 1192–1203.

Lavaei, J., & Murray, R. M. (2012). Quantized consensus by means of gossip algorithm. *IEEE Transactions on Automatic Control*, *57*(1), 19–32.

Nedic, A., Olshevsky, A., Ozdaglar, A., & Tsitsiklis, J. N. (2009). On distributed averaging algorithms and quantization effects. *IEEE Transactions on Automatic Control*, *54*(11), 2506–2517.

Norris, J. R. (1997). *Markov chains*. Cambridge University Press.

Olfati-Saber, R., & Murray, R. M. (2004). Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, *49*(9), 1520–1533.

Ren, W., & Beard, R. W. (2005). Consensus seeking in multiagent systems under dynamically changing interaction topologies. *IEEE Transactions on Automatic Control*, *50*(5), 655–661.

Zhu, M., & Martínez, S. (2008). On the convergence time of distributed quantized averaging algorithms. In *Proceedings of the 47th IEEE conference on decision and control*. (pp. 3971–3976). Cancun, Mexico.

Kai Cai received the B.Eng. degree in Electrical Engineering from Zhejiang University (China) in 2006, the M.A.Sc. degree in Electrical and Computer Engineering from the University of Toronto (Canada) in 2008, and the Ph.D. degree in Systems Science from Tokyo Institute of Technology (Japan) in 2011.

He is currently a Postdoctoral Fellow in Systems Control, with the Department of Electrical and Computer Engineering of the University of Toronto. His research interests are distributed control of multi-agent systems, distributed control of discrete-event systems, and theory of control architecture.

Hideaki Ishii received the M.Eng. degree in Applied Systems Science from Kyoto University in 1998, and the Ph.D. degree in Electrical and Computer Engineering from the University of Toronto in 2002.

He was a Postdoctoral Research Associate of the Coordinated Science Laboratory, the University of Illinois at Urbana-Champaign from 2001 to 2004, and a Research Associate of the Department of Information Physics and Computing, the University of Tokyo from 2004 to 2007. He is currently an Associate Professor of the Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology. He is a co-author of the book *Limited Data Rate in Control Systems with Networks*, Lecture Notes in Control and Information Sciences, vol. 275, Springer, Berlin, 2002. He is the Chair of the IFAC Technical Committee on Networked Systems. He is on the editorial board of *Automatica* and previously of the *IEEE Transactions on Automatic Control*. His research interests are in networked control systems, multi-agent systems, hybrid systems, and probabilistic algorithms.