53rd IEEE Conference on Decision and Control
December 15-17, 2014. Los Angeles, California, USA

Differentiated Consensuses in Decentralized Load Balancing Problem with Randomized Topology, Noise, and Delays

Natalia Amelina, Oleg Granichin, Olga Granichina, and Yuming Jiang

Abstract— **In this paper, a new consensus problem, termed *differentiated consensuses*, is studied. This consensus problem is that, in a system with multiple classes, consensus is targeted for each class, which may be different among classes. Specifically, we investigate *differentiated consensuses* in a distributed stochastic network system of nodes (or agents), where tasks, classified with different priorities, are serviced. The network system is assumed to have switched topology, noise and delay in measurement, and cost on the topology. The goal is to reach/maintain balanced (equal) load, i.e. consensus, across the network and at the same time to meet the topology cost constraint, *both for every priority class*. A control protocol is proposed. We prove that the proposed control protocol is able to meet the topology cost constraint and achieve approximate consensus for each of the priority classes in the network.**

I. Introduction

In recent years distributed network systems have been increasingly used in different applications, such as those studied in [1]–[6]. A lot of these networks have their nodes (or agents) which work in parallel and operate collaboratively to achieve a common goal (e.g., a sensor network that collects the information about the environment, a computing network that executes incoming tasks, etc.). Many of such goals belong to or are closely related to the *consensus* problem. In these cases, consensus algorithms have been widely used (e.g., see [7]–[14]).

Consensus requires that there are control strategies or protocols in place that can drive the states of all agents in a system to the same steady-state values. Many computer and communication systems, such as distributed computing systems, sensor networks, and wireless mesh networks, also require distributed controls and may be run under stochastic environments. To achieve an optimal resource allocation in these systems, a consensus requirement is often implied. For example, a fundamental problem in distributed computing is how to distribute load to different servers. This problem is essentially a consensus problem; hence consensus control strategies can represent a solution.

In all the studied consensus problems in the literature, to the best of our knowledge, each only has one single consensus objective. Or, in other words, they all only have the objective of achieving the same *one consensus* across the network. However, in practice, many (network) systems support more than one class where service differentiation exists. For such a system, it is more natural to have consensus for each class across the system, than having the same consensus requirement for all classes. For example, in a system, the deadline requirement for a class of urgent tasks may be different than that for a class of normal tasks. This calls for *differentiated consensuses*, which we define as a consensus problem for systems with multiple classes, where a consensus is targeted for each class and may be different among classes.

In [15] *the group consensus* in multi-agent network is considered. Unlike group consensus, *differentiated consensuses* has to be achieved throughout the whole network, not just among certain group of agents and thus the consensus value for every priority class has to be the same for every agent.

In this work, we investigate *differentiated consensuses* in a distributed stochastic network with priorities and topology cost constrains. In this network, tasks are classified with priorities. At each node or agent, tasks with the same priority level share a first-come-first-served (FCFS) queue. The node is work-conserving, and tasks with a higher priority are served before those with a lower priority. (The randomized rule of a service for tasks with different priorities is considered in [16].) The network topology is assumed to possibly switch over time (e.g. due to the mobility of agents), and the information about the states of neighbor nodes is to be obtained with noise and delay. Depending on the priority level of tasks in a queue we choose a specific network topology that has to be used at the current time and satisfies the topology cost constraints. We propose a control protocol and prove its ability to achieve balanced (equal) load, i.e. consensus, *for every priority class* across the network (proofs for any specific classes are based on the previous result [17]) and at the same time to meet a required topology cost constraint.

It is important to say, that in many works (e.g. [18]) about achieving consensus with the control protocol authors consider a model of network connections where to reach a consensus they often do not use all the available communication links between agents, but only some subset of them. Basically, an important question is what is the optimal way to choose communication links in the protocol. One of the options is the minimization of transients. Certainly, if one uses all the connections, then ceteris paribus it will minimize the time. However, it turned out that in many cases it is sufficient to use a subset of links to remain

This work was financially supported by the Russian Ministry of Education and Science (agreement 14.604.21.0035, unique no. RFMEFI60414X0035).
N. Amelina, O. Granichin, and O. Granichina are with the Saint Petersburg State University (Faculty of Mathematics and Mechanics, and Research Laboratory for Analysis and Modeling of Social Processes), 198504, Universitetskii pr. 28, St. Petersburg, Russia, O. Granichin is also with the ITMO University, 197101, Kronwerkskii av. 49, St. Petersburg, Russia, Y. Jiang is with the Department of Telematics, Norwegian University of Science and Technology, NO-7491, Trondheim, Norway n.amelina@spbu.ru, oleg_granichin@mail.ru, jiang@item.ntnu.no

strong connectedness and maximize the Fiedler eigenvalue of the graph Laplacian matrix (the absolute value of the second eigenvalue of Laplacian matrix) [13]. Besides, from a practical point of view it is also important to consider the question about the cost of interactions. Suppose, that the values of elements of an adjacency matrix of a network graph correspond to the cost of using a various links. In this paper, we propose a solution that, with a strong network connectivity through the randomization use of links, allows to meet the cost constraints set in the problem statement. The results show that with a significant difference in the cost of communication between different segments of the network there is a "clustering of nodes": within the cluster consensus is achieved faster, but between clusters — with significant delay, which leads to the phenomenon of "differentiation" of consensus on clusters.

II. Problem Statement

Consider a dynamic network system of n agents, which collaborate with each other, and a set of tasks with different priorities, which have to be executed in the system. Tasks came to possibly different agents of the system in different discrete time instants $t = 0, 1, \ldots$. Agents process incoming tasks in parallel. Tasks can be redistributed among agents based on a feedback. Note that a task cannot be interrupted after it is being processed by an agent, i.e. the system is non-preemptive.

Without loss of generality, agents in the system are numbered and let i be the number of an agent, $i = 1, \ldots, n$. $N = \{1, \ldots, n\}$ denotes the set of agents in the network system. The network topology switches over time (e.g., due to the mobility of agents). Let the dynamic network topology be modeled by a sequence of digraphs $\{(N, E_t)\}_{t\geq 0}$, where E_t denotes the set of edges at time t. We associate a weight $a_t^{i,j} > 0$ at time t with each edge $(j,i) \in E$, $a_t^{i,j} > 0$ if agent j is connected with agent i and $a_t^{i,j} = 0$ otherwise. $a_t^{i,j} > 0$ is the cost of maintaining communication between agent j and agent i. Here and below, an upper index of agent i is used as the corresponding number of an agent (while not as an exponent). Matrix $A_t = [a_t^{i,j}]$ is an adjacency matrix of the graph at time t. Denote $\mathscr{G}_{A_t}$ as the corresponding graph.

To introduce some properties of the network topology, the following definitions from the graph theory will be used. Define *the weighted in-degree* of node i as the sum of i-th row of matrix A: $d^i(A) = \sum_{j=1}^n a^{i,j}$; $D(A) = \text{diag}\{d^i(A)\}$ is the corresponding diagonal matrix; $d_{\max}(A)$ is the maximum in-degree of graph $\mathscr{G}_A$; $\mathscr{L}(A) = D(A) - A$ is the *Laplacian* of graph $\mathscr{G}_A$. Let $\cdot^{\mathrm{T}}$ stands for a vector or matrix transpose operation; $||A||$ is the Euclidian norm: $||A|| = \sqrt{\sum_i \sum_j (a^{i,j})^2}$; $Re(\lambda_2(A))$ is the real part of the second eigenvalue of matrix A ordered by absolute magnitude; $\lambda_{\max}(A)$ is the maximum eigenvalue of matrix A. Digraph $\mathscr{G}_B$ is said to be a subgraph of the digraph $\mathscr{G}_A$ if $b^{i,j} \leq a^{i,j}$ for all $i, j \in N$. Digraph $\mathscr{G}_A$ is said to contain a *spanning tree* if there exists a directed tree $\mathscr{G}_{tr} = (N, E_{tr})$ as a subgraph of $\mathscr{G}_A$.

We suppose that tasks (jobs) are classified with different priorities $k = 1, \ldots, m$. A smaller priority number means higher priority such that $k = 1$ is the highest priority and $k = m$ is the lowest priority of task. When two tasks with different priorities should be served by the agent, the one with high priority will be served first. The execution time of a task varies from one agent to another and depends on the productivity of an agent. We assume, at each time, the same agent has the same productivity capability for each class.

Particularly, the behavior of an agent $i \in N$ is described by two characteristics:

- the m-vector of queue lengths of tasks $\mathbf{q}_t^i = [q_t^{i,k}]$ at time t whose kth element is defined by the amount of tasks with priority $k = 1, \ldots, m$;
- the productivity $p^{i,k}$.

In addition, for all $i \in N$, $t = 0, 1, \ldots, T$, the dynamics of the network system is as follows

$$\mathbf{q}_{t+1}^i = \mathbf{q}_t^i - p^{i,k(t,i)} \mathbf{e}_{k(t,i)} + \mathbf{z}_t^i + \mathbf{u}_t^i \tag{1}$$

where

$$k(t,i) = \min_k \{k : q_t^{i,k} > 0\}^1 \tag{2}$$

and $\mathbf{e}_k$ are unit basis vectors in $\mathbb{R}^m$ (kth element equals 1 and others equal 0); $\mathbf{z}_t^i$ is a vector whose kth element is the amount of new system tasks received through agent i at time instant t with priority k; $\mathbf{u}_t^i \in \mathbb{R}^m$ is a vector of control actions consisting of redistributed tasks $u_t^{i,k}$ (parts of system tasks previously received through other agents) with priority k to agent i at time instant t, which could (and should) be chosen based on some information about queue lengths $\mathbf{q}_t^j$ and productivities $\mathbf{p}^j$ of neighbors $j \in N_t^i$, where N_t^i is a part of set $\{j \in N : a_t^{i,j} > 0\}$.

Now we define the cost of a chosen topology $\{N_t^i, i \in N\}$

$$C(\{N_t^i, i \in N\}) = \max_{i \in N} \sum_{j \in N_t^i} a_t^{i,j}. \tag{3}$$

We will consider control protocols that satisfy some specific constraint on the cost of the topology for each task priority class.

Assume, that $p^{i,k} \neq 0$, $\forall i \in N$, $k \in \{1, \ldots, m\}$. It is not so hard to prove that from all possible options for all tasks redistribution the minimum operation time of the system is achieved when the load (defined as the ratio of the queue length over the productivity) is equalized throughout the network (see, e.g. [14]).

The goal is to maintain balanced (equal) load across the network for every priority class, and, at the same time, to meet the cost constraint requirement.

At this setting we can consider the consensus problem for states $\mathbf{x}_t^i$ of agents, where $x_t^{i,k} = \frac{q_t^{i,k}}{p^{i,k}}$. *We emphasize, that* $\mathbf{x}_t^i$ *is a state vector, consisting of states of m classes.*

When there is no consensus between agents' states, it is naturally to use a redistribution protocol over time to ensure balanced load across the network (e.g., in order to increase the overall throughput of the system and to reduce the execution time).

[1] This implicitly specifies the rule of priority.

We assume, that to form the control strategy $\mathbf{u}_t^i$ each agent $i \in N$, if the set N_t^i is not empty, has an information about its own productivity $\mathbf{p}^i$ and queue state $\mathbf{q}_t^i$, about neighbors' productivities $\mathbf{p}_t^j$, $j \in \{i\} \cup N_t^i$, and noisy and possibly delayed observations about neighbors' states

$$\mathbf{y}_t^{i,j} = \mathbf{x}_{t-s_t^{i,j}}^j + \mathbf{w}_t^{i,j},\ j \in N_t^i, \tag{4}$$

where $\mathbf{w}_t^{i,j}$ is a noise vector, $0 \le s_t^{i,j} \le \bar{s}$ are integer-valued delays, and $\bar{s}$ is a maximum of possible delays.

III. Topology Cost Constraints and Randomized Topology Decomposition

Let $(\Omega, \mathscr{F}, P)$ be the underlying probability space corresponding to the sample space, the collection of all events, and the probability measure respectively, E be a mathematical expectation symbol.

We assume, that graphs $\mathscr{G}_{A_t}$, $t = 1, \dots$ are i.i.d. (independent identically distributed), i.e. the random events of appearance of edge (j,i) are independent and identically distributed for the fixed (j,i). Let $a_{av}^{i,j}$ define mean values (mathematical expectations) of $a_t^{i,j}$, and A_{av} is the correspondence adjacency matrix.

Assume that the following condition is satisfied:

- **A1**. Graph $\mathscr{G}_{A_{av}}$ has a spanning tree.

Tasks has different priorities and, for each priority, the maximum cost of the network graphs that could be used is defined. For each time instant t, consider m ways (which may be different and each corresponds to one class) to select the topology subgraphs $\mathscr{G}_t^k$: $\mathscr{G}_t^m \subset \mathscr{G}_t^{m-1} \subset \ldots \subset \mathscr{G}_t^1$ of the graph $\mathscr{G}_{A_t}$, which allows to use redistribution protocols for tasks with priority k, $k = 1, \dots, m$. Let B_t^k be the corresponding adjacency matrices. Note that one of the possible ways of choosing $\mathscr{G}_t^k$ is to use $\mathscr{G}_{A_t}$ for all k.

A. Topology Cost Constraints and Randomized Topology

Let c_k, $k = 1, \dots, m$, stand for the maximum average cost of the network links for a task with kth priority. Assume $c_1 \ge c_2 \ge \dots c_m > 0$.

Definition *1:* We will say that network topology decomposition $\{\mathscr{G}_t^k\}$ satisfies average cost constraints $\{c_k\}$ if for every priority class k

$$d_{\max}(B_{av}^k) = \mathrm{E} d_{\max}(B_t^k) = \mathrm{E} \max_{i \in N} \sum_{j \in N_t^{i,k}} b_t^{i,j,k} \le c_k, \tag{5}$$

where $N_t^{i,k}$ is the neighbors set of agent i at time t formed in accordance with the topology $\mathscr{G}_t^k$.

Theorem *1:* **If** Assumption **A1** holds **then** for any average cost constraints $\{c_k\}$, $c_k > 0$, there exists network topology decomposition $\{\mathscr{G}_t^k\}$ that satisfies the averaged cost constraints $\{c_k\}$ and for which all averaged graphs $\mathscr{G}_{av}^k$ have spanning trees.

Proof: We fix average cost constraints $\{c_k\}$, $c_k > 0$. Now let us give a constructive proof.

For all $i \in N$ we have

$$d^i(A_{av}) = \mathrm{E} \sum_{j \in N_t^i} a_t^{i,j} > 0,$$

since graph $\mathscr{G}_{A_{av}}$ has a spanning tree.

Consider the following method of subgraph $\mathscr{G}_t^k$ construction.

1. Let $k = 0$. Choose $\mathscr{G}_t^0 = \mathscr{G}_{A_t}$ and $r_0 = \max_{i \in N} d^i(A_{av})$.
2. Take $k := k + 1$.
3. If $c_k \ge r_{k-1}$ then we choose all edges (j,i) from $\mathscr{G}_t^k$ to form a topology graph $\mathscr{G}_t^k$. Else we choose edges (j,i) from $\mathscr{G}_t^k$ with probability $\frac{c_k}{r_{k-1}}$ to form a topology graph $\mathscr{G}_t^k$. Compute
$$r_k = \mathrm{E} \max_{i \in N} \sum_{j \in N_t^{i,k}} b_t^{i,j,k}.$$
4. If $k < m$ repeat Steps 2–4.

By virtue the construction method for subgraphs $\mathscr{G}_t^k = \mathscr{G}_{B_t^k}$ we have

$$\mathrm{E} \max_{i \in N} \sum_{j \in N_t^{i,k}} b_t^{i,j,k} \le c_k.$$

Hence constraints (5) satisfy.

For any $k \in \{1, \dots, m\}$ and for all $i, j \in N$ if $a_{av}^{i,j} > 0$ then $b_{av}^{i,j,k} > 0$ by virtue the construction method. Hence the averaged graph $\mathscr{G}_{av}^k$ has spanning tree when graph $\mathscr{G}_{A_{av}}$ has a spanning tree. This completes the proof. ■

Essentially, the considered in the proof approach for generating the network topology decomposition could be called randomized topology since random links arise by our will.

In practice, a possibly more useful way for network topology decomposition is to cluster agents (nodes) into groups with "cheap" internal communications and to randomize interaction rules between groups.

B. Example

Consider an example of network of 5 agents which are divided into two clusters: {1, 2} {3, 4, 5} (Fig. 1). Assume, that the cost of maintaining communication between agents equals 9 for communications 2-4 and 4-2, and equals 1 for other communications. We suppose that tasks have two priorities, i.e. $k = 2$.

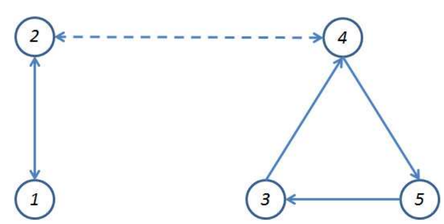

Fig. 1. The network topology.

The adjacency matrix of the network is as follows

$$A_t = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 9 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 9 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}. \tag{6}$$

In this case $C(\{N_t^i, i \in N\}) = 10$. We set $\{c_k\}_{k=1,2} = \{10, 1.5\}$. For tasks with the higher priority we can use the original topology $B_{av}^1 = A_{av} = A_t$

For tasks with the priority 2 we can use the following randomization of interaction rules between agents: links 2-4 and 4-2 appear with probability $\frac{1}{18}$. Thus, the corresponding adjacency matrix is

$$B_{av}^2 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{1}{2} & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}, \qquad (7)$$

where the sum of elements in rows is no more than $c_2 = 1.5$.

IV. Control Protocol

In [14], [17] properties of the control algorithm, called local voting protocol, for load balancing problem of a stochastic network were studied. The control value of the local voting protocol for each agent was determined by the weighted sum of differences between the information about the state of the agent and the information about its neighbors' states.

Let's consider a similar family of protocols as follows. For each $k = 1, \ldots, m$ with a network topology decomposition $\{\mathscr{G}_t^k\}$ for cost constraints $\{c_k\}$, $c_k > 0$:

$$u_t^{i,k} = \gamma p^{i,k} \sum_{j \in N_t^{i,k}} b_t^{i,j,k}(y_t^{i,j,k} - x_t^{i,k}), \qquad (8)$$

where $\gamma > 0$ is a step-size of the control protocol, $N_t^{i,k} \subset N_t^i$ is the neighbor set of agent i at time t formed in accordance with the topology $\mathscr{G}_t^k$. The system works in such a way that within each priority tasks are distributed evenly.

The dynamics of the closed loop system with protocol (8) is as follows

$$\mathbf{x}_{t+1}^i = \mathbf{x}_t^i - \mathbf{e}_{k(t,i)} + \tilde{\mathbf{z}}_t^i + \gamma \sum_{j \in N_t^i} (\mathbf{y}_t^{i,j} - \mathbf{x}_t^i)^{\mathrm{T}} \mathbf{b}_t^{i,j}, \ i \in N, \qquad (9)$$

where $\tilde{\mathbf{z}}_t^i$ and $\mathbf{b}_t^{i,j}$ are vectors of $\tilde{z}_t^{i,k} = z_t^{i,k}/p^{i,k}$ and $b_t^{i,j,k}$, $k = 1, \ldots, m$.

Let $\mathbf{x}_t^i \equiv 0, i \in N$ for $-\bar{s} \le t < 0$. If $\bar{s} > 0$ we "artificially" add $n\bar{s}$ new agents to the current network topology. At each time instant t the new "fictitious" agents have states which are equal to the corresponding states of the "real" agents at previous time instants: $t-1, t-2, \ldots, t-\bar{s}$.

For each $k = 1, \ldots, m$ denote $\bar{\mathbf{x}}_t^k \in \mathbb{R}^{\tilde{n}}$, $\tilde{n} = n(\bar{s}+1)$, as an extended state vector for $t = 0, 1, \ldots$ which consists of $x_t^{1,k}, \ldots, x_t^{n,k}, x_{t-1}^{1,k}, \ldots, x_{t-1}^{n,k}, \ldots, \ldots, x_{t-\bar{s}}^{n,k}$, i.e. it includes all the components with all kinds of delays not exceeding $\bar{s}$. Introduce the extended $\tilde{n} \times \tilde{n}$ matrix $\bar{B}_t^k$ of the control protocol (8) which consists of zeros at all places except entries $\bar{b}_t^{i,j+ns_t^{i,j},k}$, $i \in N$, $j \in N_t^{i,k}$ in n first lines which are equal to $b_t^{i,j,k}$ and $\bar{b}_t^{i,i-n,k} = 1/\gamma$ in next $n\bar{s}$ lines, $i = n+1, \ldots, \tilde{n}$. Due to the view of the Laplacian matrix $\mathscr{L}(\bar{B}_t^k)$ we can rewrite the dynamics of the system in the following vector-matrix form:

$$\bar{\mathbf{x}}_{t+1}^k = \bar{\mathbf{x}}_t^k - \gamma \mathscr{L}(\bar{B}_t^k)\bar{\mathbf{x}}_t^k + \gamma \begin{pmatrix} \mathbf{w}_t^k \\ 0 \end{pmatrix} + \begin{pmatrix} \mathbf{f}_t^k \\ 0 \end{pmatrix}, \qquad (10)$$

where m-vectors $\mathbf{w}_t^k$ and $\mathbf{f}_t^k$ consist of elements of corresponding vectors $\sum_{j \in N_t^1} (\mathbf{w}_t^{1,j})^{\mathrm{T}} \mathbf{b}_t^{1,j}, \ldots, \sum_{j \in N_t^n} (\mathbf{w}_t^{n,j})^{\mathrm{T}} \mathbf{b}_t^{n,j}$, and $-\mathbf{e}_{k(t,1)} + \tilde{\mathbf{z}}_t^1, \ldots, -\mathbf{e}_{k(t,n)} + \tilde{\mathbf{z}}_t^n$.

V. Approximate Consensus

A. Assumptions

Assume that the following additional conditions are satisfied:

- **A2**. **a)** For all $i \in N$, $j \in N_t^i$, observation noise vectors $\mathbf{w}_t^{i,j}$ are zero-mean, independent identically distributed (i.i.d.) random vectors with bounded variances: $\mathrm{E}(\mathbf{w}_t^{i,j})^2 \le \sigma_w^2$.
  b) For all $i \in N, j \in N_{\max}^i = \cup_t N_t^i, k = 1, \ldots, m$ the appearance of "variables" edges (j,i) in graph $\mathscr{G}_t^k$ is independent random event (i.e. matrices B_t^k are i.i.d. random matrices). For all $i \in N$, $j \in N_t^i, k = 1, \ldots, m$ weights $b_t^{i,j,k}$ in the control protocol are independent random variables with expectations: $\mathrm{E}b_t^{i,j,k} = b^{i,j,k}$, and bounded variances: $\mathrm{E}(b_t^{i,j,k} - b^{i,j,k})^2 \le \sigma_{b,k}^2$.
  c) For all $i \in N, j \in N^i$ there exists a finite value $\bar{s} \in \mathbb{N}$: $s_t^{i,j} \le \bar{s}$ with probability 1, and integer-valued delays $s_t^{i,j}$ are i.i.d. random variables taking value $l = 0, \ldots, \bar{s}$ with probability $p_l^{i,j}$.
  d) For all $k = 1, \ldots, m$, $i \in N$, $t = 0, 1, \ldots$ vectors $\mathbf{f}_t^k$ from (10) are i.i.d. random vectors with expectations: $\mathrm{E}f_t^{i,k} = \bar{f}^k$, and variances: $\mathrm{E}(f_t^{i,k} - \bar{f}^k)^2 = \sigma_{f,k}^2$.
  Additionally, all these random variables and vectors are mutually independent.
  In general, if Assumptions **A2.b** and **A2.c** hold, the averaged matrixes $\bar{B}_{av}^k = \mathrm{E}\bar{B}_t^k$, $k = 1, \ldots, m$, consist of elements
  $$\bar{b}_{av}^{i,j,k} = \begin{cases} p_{j \div \bar{s}}^{i,j \bmod \bar{s}} b^{i,j \bmod \bar{s},k}, \text{ if } i \in N, \ j = 1, \ldots, \tilde{n}, \\ 1/\gamma, \text{ if } i = n+1, \ldots, \tilde{n}, \ j = i-n, \\ 0, \text{ otherwise.} \end{cases} \qquad (11)$$
  Here, the operation mod is a remainder of division, and $\div$ is a division without remainder.
  Note, that if $\bar{s} = 0$, then $\bar{B}_{av}^k = B_{av}^k$.
- **A3.** For the step-size of the control protocol $\gamma > 0$ the following conditions are satisfied:
  $$\gamma \le \min_{1 \le k \le m} \frac{1}{d_{\max}(\bar{B}_{av}^k)} \qquad (12)$$
  and for any $k = 1, \ldots, m$
  $$\delta_k = Re(\lambda_2(\bar{B}_{av}^k)) - \gamma \lambda_{\max}(Q^k) > 0 \qquad (13)$$
  where $Q^k = \mathrm{E}(\mathscr{L}(\bar{B}_{av}^k) - \mathscr{L}(\bar{B}_t^k))^{\mathrm{T}}(\mathscr{L}(\bar{B}_{av}^k) - \mathscr{L}(\bar{B}_t^k))$.

B. Averaged Model

Let $\mathbf{x}_0^\star$ be the weighted average vector of the initial states $\mathbf{x}_0^\star = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_0^i$ and $\{\mathbf{x}_t^\star\}$ is the trajectory of the averaged system

$$\mathbf{x}_{t+1}^\star = \mathbf{x}_t^\star + \bar{\mathbf{f}}, \qquad (14)$$

where $\bar{\mathbf{f}}$ is the vector of mean values from the Assumption **A2.d**.

C. Differentiated Consensuses

Theorem *2:* Let average cost constraints $\{c_k\}$, $c_k > 0$, be satisfied for some topology subgraphs $\mathscr{G}_t^k : \mathscr{G}_t^m \subset \mathscr{G}_t^{m-1} \subset \ldots \subset \mathscr{G}_t^1$ of graphs $\mathscr{G}_{A_t}$, with corresponding adjacency matrices B_t^k.

If $\mathrm{E}||\mathbf{x}_0^i||^2 < \infty$, $i \in N$, Assumption **A1** holds for graphs $\mathscr{G}_{av}^k$, Assumption **A2** holds for above-described vectors and matrixes, and Assumption **A3** holds for step-size γ of control protocol (8) **then** for trajectories of closed-loop systems (9) and (14) the following inequality asymptotically holds in mean square sense:

$$\overline{\lim_{t\to\infty}}\mathrm{E}||\mathbf{x}_t^i - \mathbf{x}_t^\star||^2 \leq \sum_{k=1}^{m} \frac{1}{\gamma}\frac{n\sigma_{f,k}^2}{\delta_k} + \gamma\frac{2\sigma_w^2(n^2\sigma_{b,k}^2 + ||\bar{B}_{av}^k||^2)}{\delta_k}, \tag{15}$$

i.e. states of all agents approximately synchronize with trajectory $\mathbf{x}_t^\star$.

Basically, an approximate synchronization of agents' states means an approximate load balancing. Note, if the system does not receive new tasks of priority k, then $\mathbf{f}_t^k = 0$, $\sigma_{f,k}^2 = 0$, and we can get a semi-consensus for the distribution of tasks of priority k when γ is chosen sufficiently small.

Proof: Note, that due to the definition and Assumption **A2b,c** we have: $\mathscr{L}(\bar{B}_{av}^k) = \mathrm{E}\mathscr{L}(\bar{B}_t)$. Moreover, if Assumption **A1** is satisfied then $0 < Re(\lambda_2(B_{av}^k))$ and bound $Re(\lambda_2(B_{av}^k)) < 1$ follows by the conditions (12) (see, e.g. [13]). All conditions of Theorem 1 from [17] satisfy under conditions of the Theorem 2. Hence for any $k = 1, \ldots, m$, when we consider the sub-statespace including only tasks with priority k, we have

$$\overline{\lim_{t\to\infty}}\mathrm{E}(x_t^{i,k} - x_t^{\star,k})^2 \leq \frac{\Delta_k}{\gamma\delta_k}, \tag{16}$$

where $\Delta_k = n\sigma_{f,k}^2 + 2\sigma_w^2\gamma^2(n^2\sigma_{b,k}^2 + ||\bar{B}_{av}^k||^2)$. Summing (16) for all $k = 1, \ldots, m$ we derive the inequality (15) which is the mail result of Theorem 2. ■

At this point, we highlight that, the proof of Theorem 2 shows that queues with different priorities achieve m different consensus levels separately. This behavior is termed as *differentiated consensus*.

VI. Simulations

We return to the example in Section III.

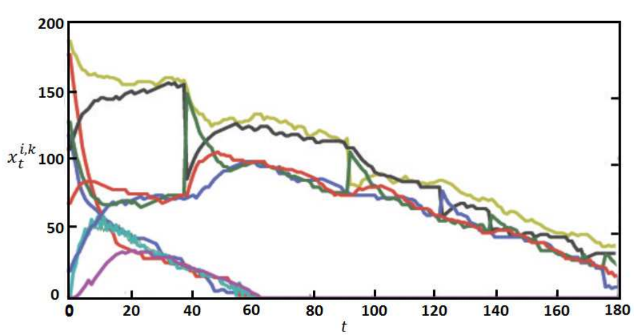

Fig. 2. Evaluation of queue lengths in the example for $z_t^{ik} = 0, t > 1, i = 1, \ldots, 5, k = 1, 2$

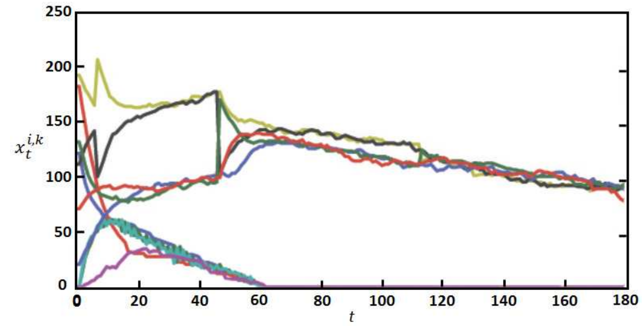

Fig. 3. Evaluation of queue lengths in the example for $z_t^{ik} = 0, t > 0, i > 1, k = 1, 2$ and $z_t^{11} = 0$, z_t^{12} are random values from set {0, 1, 2, 3}

Let's assume that $\bar{s} = 1$ and each nonzero link with weight equals 1 appears in A_t with probability $\frac{2}{3}$ without delay, and with probability $\frac{1}{3}$ with delay.

In this case, we have

$$\bar{B}_{av}^1 = \begin{pmatrix} 0 & \frac{2}{3} & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 \\ \frac{2}{3} & 0 & 0 & 9 & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & \frac{1}{3} \\ 0 & 9 & \frac{2}{3} & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & 0 & \frac{2}{3} & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

$$\bar{B}_{av}^2 = \begin{pmatrix} 0 & \frac{2}{3} & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 \\ \frac{2}{3} & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & \frac{1}{3} \\ 0 & \frac{1}{2} & \frac{2}{3} & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & 0 & \frac{2}{3} & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

and we can compute

$$Re(\lambda_2(\bar{B}_{av}^1)) = 0.2258,\ \lambda_{\max}(Q^1) = 1.2222,$$

$$Re(\lambda_2(\bar{B}_{av}^2)) = 0.4031,\ \lambda_{\max}(Q^2) = 2.3341.$$

For $\gamma = 0.1$ conditions (12), (13) are satisfied.

We suppose, that $p^{i,k} = 1$ for all $i \in N$, $k = 1, 2$, and new tasks are feeded into the system only at the beginning so that

$$\mathbf{z}_1^1 = \begin{pmatrix} 120 \\ 190 \end{pmatrix}, \mathbf{z}_1^2 = \begin{pmatrix} 0 \\ 110 \end{pmatrix}, \mathbf{z}_1^3 = \begin{pmatrix} 180 \\ 20 \end{pmatrix},$$

$$\mathbf{z}_1^4 = \begin{pmatrix} 0 \\ 130 \end{pmatrix}, \mathbf{z}_1^5 = \begin{pmatrix} 0 \\ 70 \end{pmatrix}.$$

Note, that measurements $y_t^{i,j,k}$, $i, j \in N$, $k = 1, 2$, were made with Gaussian zero-mean noise with variance 1.

The typical result of behaviors of queue lengths of tasks with different priorities is shown in Fig. 2. At the beginning when $t \leq 60$ all agents proceed only tasks with higher priorities (in the example with priority $k = 1$). Tasks with second priority redistribute into two separate clusters since communication links $2-4$ and $4-2$ is "expensive" and turn on rarely (one times per 18 iterations in average). However, the queue lengths of tasks with second priority do not decrease while $t < 60$. After $t > 60$, when tasks with first priority are already done, tasks with second priority begin processing.

Fig. 3 illustrates the case, when new tasks with priority $k = 2$ come into the system over time. Specifically, we assume that z_t^{i2} takes value 0 with probability 1/2 and, if $z_t^{i2} \neq 0$, then z_t^{i2} equals 1, 2 or 3 with equal probability.

VII. Discussion

To this point, it is worth highlighting that the above-described and analyzed consensus problem, i.e. differentiated consensuses, arises naturally in many practical applications. One application area is wireless sensor networks.

Consider a wireless sensor network. The overall goal of the network is to collect information about the environment and send it to the sink. In this sensor network, sensors are clustered. In each cluster, there is one cluster head sensor whose main responsibility is to forward the information collected by other sensors in the cluster to the sink. In this way, we can divide sensors in the network into two groups. The first group is formed by cluster heads. These head sensors may not collect information directly, but they are typically "connected with" (or are one communication hop away from) the sink and forward information from other sensors in the cluster to the sink. Head sensors may communicate with each other. One cluster head may use other head sensors as relays to communication with the sink, if the communication cost using other head sensors is "cheaper" then direct communication with the sink. The second group of sensors are those remaining sensors that collect information about the environment. A sensor in the second group can communicate only with the corresponding cluster head sensor. In other words, sensor in the second group send the collected information to the first group sensors, and then the first group sensors forward the information to the sink.

In this network, there are two types of communication tasks. One is for a second group sensor to send the collected information to the corresponding head sensor. Another is for the head sensor to forward the received information to the sink. In this scenario, depending on the chosen policy, the first type may be given a lower priority, while the second type is given a higher priority. In addition, if different types of information are collected by sensors, for the corresponding transmission / communication task of each information type, a different priority level may be given.

VIII. Conclusion

In this paper we introduced a new consensus problem, called *differentiated consensuses*. Specifically, we considered a distributed stochastic network with priorities and topology cost constraints. For this network, we presented the conditions of achieving m (possibly different) consensus levels separately under switched topology and noise and delay in measurement. There were two simultaneous goals in the network: to maintain the balanced load across the network for every priority class and to meet the topology cost constraint. A control protocol was proposed and it was proved with the ability in achieving differentiated consensuses under topology cost constraints.

References

[1] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[2] D. Armbruster, *Networks of Interacting Machines: Production Organization in Complex Industrial Systems and Biological Cells*. World Scientific Publishing Company Incorporated, 2005, vol. 3.

[3] W. Ren, R. Beard, and E. Atkins, "Information consensus in multivehicle cooperative control," *Control Systems, IEEE*, vol. 27, no. 2, pp. 71–82, 2007.

[4] O. Granichin, P. Skobelev, A. Lada, I. Mayorov, and A. Tsarev, "Cargo transportation models analysis using multi-agent adaptive real-time truck scheduling system," in *Proc. of the 5th International Conference on Agents and Artificial Intelligence. (ICAART2013)*, vol. 2, Barcelona, Spain, 2013, pp. 244–249.

[5] S. Kar, J. M. Moura, and K. Ramanan, "Distributed parameter estimation in sensor networks: Nonlinear observation models and imperfect communication," *Information Theory, IEEE Transactions on*, vol. 58, no. 6, pp. 3575–3605, 2012.

[6] S. Barbarossa, G. Scutari, and A. Swami, "Achieving consensus in self-organizing wireless sensor networks: The impact of network topology on energy consumption," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 2. IEEE, 2007, pp. II–841.

[7] J. Tsitsiklis, D. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *Automatic Control, IEEE Transactions on*, vol. 31, no. 9, pp. 803–812, 1986.

[8] J. Turek and D. Shasha, "The many faces of consensus in distributed systems," *Computer*, vol. 25, no. 6, pp. 8–17, 1992.

[9] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *Automatic Control, IEEE Transactions on*, vol. 49, no. 9, pp. 1520–1533, 2004.

[10] W. Ren and R. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *Automatic Control, IEEE Transactions on*, vol. 50, no. 5, pp. 655–661, 2005.

[11] A. Kashyap, T. Başar, and R. Srikant, "Quantized consensus," *Automatica*, vol. 43, no. 7, pp. 1192–1203, 2007.

[12] S. Kar and J. M. Moura, "Sensor networks with random links: Topology design for distributed consensus," *Signal Processing, IEEE Transactions on*, vol. 56, no. 7, pp. 3315–3326, 2008.

[13] F. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative Control of Multi-Agent Systems*, 2014.

[14] N. Amelina and A. Fradkov, "Approximate consensus in the dynamic stochastic network with incomplete information and measurement delays," *Automation and Remote Control*, vol. 73, no. 11, pp. 1765–1783, 2012.

[15] J. Yu and L. Wang, "Group consensus in multi-agent systems with switching topologies and communication delays," *Systems & Control Letters*, vol. 59, pp. 340–348, 2010.

[16] N. Amelina, O. Granichin, O. Granichina, Y. Ivanskiy, and Y. Jiang, "Differentiated consensuses in a stochastic network with priorities," in *Proc. of 2014 IEEE Multi-conference on Systems and Control, October 8-10, 2014, Antibes/Nice, France*.

[17] N. Amelina, O. Granichin, and A. Kornivetc, "Local voting protocol in decentralized load balancing problem with switched topology, noise, and delays," *Proc. of 52nd IEEE Conference on Decision and Control (CDC 2013)*, pp. 4613–4618, 2013.

[18] M. Huang, "Stochastic approximation for consensus: a new approach via ergodic backward products," *IEEE Transactions on Automatic Control*, vol. 57, no. 12, pp. 2994–3008, 2012.