PAPER • OPEN ACCESS

Distributed Time Synchronization Algorithms and Opinion Dynamics

To cite this article: Anatoly Manita and Larisa Manita 2018 J. Phys.: Conf. Ser. 955 012038

View the article online for updates and enhancements.

You may also like

- <u>The formation of continuous opinion</u> <u>dynamics based on a gambling</u> <u>mechanism and its sensitivity analysis</u> Yueying Zhu, Qiuping Alexandre Wang, Wei Li et al.
- <u>Cross-issue solidarity and truth</u> <u>convergence in opinion dynamics</u> Zhi Xuan Tan and Kang Hao Cheong
- <u>Sensitive agents under group pressure</u> form fewer opinion clusters Changwei Huang, Huasheng Cai and Wenchen Han





DISCOVER how sustainability intersects with electrochemistry & solid state science research



This content was downloaded from IP address 18.224.246.203 on 06/05/2024 at 07:48

Distributed Time Synchronization Algorithms and Opinion Dynamics

Anatoly Manita^{1,2} and Larisa Manita²

¹ Lomonosov Moscow State University, Faculty of Mechanics and Mathematics, 119991, GSP-1, Moscow, Russia

² National Research University Higher School of Economics, Moscow Institute of Electronics and Mathematics, Russia

E-mail: manita@mech.math.msu.su, lmanita@hse.ru, amanita@hse.ru

Abstract. We propose new deterministic and stochastic models for synchronization of clocks in nodes of distributed networks. An external accurate time server is used to ensure convergence of the node clocks to the exact time. These systems have much in common with mathematical models of opinion formation in multiagent systems. There is a direct analogy between the *time server/node clocks* pair in asynchronous networks and the *leader/follower* pair in the context of social network models.

1. Agreement algorithms for distributed networks

Distributed algorithms have a variety of applications in computer science [1], mobile multiagent systems [2, 3], biology [4, 5], social network dynamics [6, 7] and are also important for physicists [8, 9] interested in modeling of coordinated behavior. The present study is devoted to a special class of distributed algorithms adapted to the local clocks synchronization in asynchronous networks and to convergence to a leader opinion in large social networks. From the mathematical point of view, as we will soon show, the both problems lead to the same type of linear iterative algorithms

$$\boldsymbol{x}(s) = W(s)\,\boldsymbol{x}(s-1) + D(s)\boldsymbol{\delta}(s), \qquad s = 1, 2, \dots, \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^N$ is a state of an *N*-node network, $\boldsymbol{\delta}(s) \in \mathbb{R}^N$ is a random noise vector added on step *s* of the algorithm, W(s) and D(s) are nonnegative $N \times N$ -matrices to be precised later. Proposed models belong to (or, it would be better to say, are modifications of) famous agreement algorithms [10] known also as network consensus (NC) algorithms. One says that the network $\boldsymbol{x}(s)$ reaches a consensus if for any initial state $\boldsymbol{x}(0)$ there is $c \in \mathbb{R}^1$ such that $\boldsymbol{x}(s) \to c\mathbf{1}$ as $s \to \infty$. Here and henceforth $\mathbf{1}$ denote a column vector of all ones. Let M_N be the set of all real $N \times N$ -matrices. We write $A = (a_{jk})_{j,k=1}^N \in M_N^{(+)}$ and call *A nonnegative* if $A \in M_N$ and all $a_{jk} \ge 0$. We call *A* a stochastic matrix and write $A \in M_N^{\text{st}}$ if $A \in M_N^{(+)}$ and $A\mathbf{1} = \mathbf{1}$.

DeGroot [6] proposed the algorithm $\boldsymbol{x}(s) = A \, \boldsymbol{x}(s-1)$ where $A \in M_N^{\text{st}}$. It can be considered as a deterministic time invariant version of (1) and thus this is the simplest agreement algorithm. DeGroot interpreted it as a model of reaching a common decision in a group of experts by pooling their individual opinions. Technically the analysis of the consensus problem can be based on the discrete Markov chains theory (as in [6]) or on the direct study of powers A^n in the framework of the Perron-Frobenius theory. A condition for a consensus to be reached was given in [6] in terms the matrix powers A^n .

While in [6] there was no word *network*, it is straightforward to introduce a digraph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ with the set of vertices $\mathcal{N} = \{1, \ldots, N\}$ and the set \mathcal{E} of directed edges (m_1, m_2) such that $a_{m_1m_2} > 0, m_1 \neq m_2$. Denote by $\mathcal{O}_j := \{k : a_{jk} > 0\}$ the set of neighbours of a node j in the graph \mathcal{G} . Hence the DeGroot algorithm takes the form $x_j(s) = \sum_{k \in \mathcal{O}_j} a_{jk} x_k(s-1)$ and has the meaning of *averaging* of the values $x_k(s-1)$. We may say that the entry a_{jk} quantifies the importance of the value of $x_k(s-1)$ for evaluation of $x_j(s)$. For this reason A is called the interconnection matrix. If $|\mathcal{O}_j| < |\mathcal{N}|$ for all j then the network is called distributed [11].

Cybenko [12] considered a multiprocessor network and studied a dynamics load balancing model $\boldsymbol{x}(s) = A \boldsymbol{x}(s-1) + \boldsymbol{\delta}(s)$ with symmetric $A \in M_N^{\text{st}}$ and independent identically distributed (i.i.d.) noise vectors $\boldsymbol{\delta}(s)$. Some optimization problem for a similar model was considered in [13].

Vicsek *et al.* [8] introduced and *numerically* studied a model of moving particles where the velocity of the particles is determined by a simple rule and random fluctuations. The simple rule is that at each time step a given particle driven with a constant absolute velocity assumes the *average direction* of motion of the particles in its neighborhood of radius r. Depending on the density and a noise strength parameter the model exhibits such rich behavior as clustering, transport and phase transition. The Vicsek model (VM) immediately became very popular. In [2] the authors *analytically* studied a *linearized* deterministic version of the VM and found conditions when all particles reach an agreement on direction of motion. An agreement algorithms of [2] has the form $\theta(s) = A(s) \theta(s-1)$ where θ_j is the direction of the particle j and the sequence of matrices $\{A(s)\}_{s\geq 1}$ reflects a changing topology of the network. The spirit of the VM suggests

$$a_{jk}(s) = \frac{\mathbf{1}(|q_k(s) - q_j(s)| \le r)}{\sum_{m=1}^N \mathbf{1}(|q_m(s) - q_j(s)| \le r)}$$

where q_j is the position of the particle j. However, the paper [2] deals with abstract sequences $\{A(s)\}_{s\geq 1}$ taken from a finite set of *symmetric* stochastic matrices $\{A_1, \ldots, A_M\}$. The above mentioned sufficient conditions were given purely in terms of this time-varying topology, i.e., in terms of connectivity properties of the sequence of graphs $\{\mathcal{G}(s)\}$. This approach was developed by other authors [14, 15]. Another result of the paper [2] is related to the "leader-followers" assumption and is important for the opinion dynamics modeling. According to this assumption the particle 1 is a leader and does not change its direction: $\theta_1(s) \equiv \theta_1$. It was proved that under suitable conditions the direction of motion of any other particle $j \geq 2$ tends to the direction of the leader: $\theta_j(s) \to \theta_1$ as $s \to \infty$.

Hegselmann and Krause [7] introduced an opinion dynamics model $\boldsymbol{x}(s) = A(s) \boldsymbol{x}(s-1)$ with

$$a_{jk}(s) = \frac{\mathbf{1}\left(|x_k(s) - x_j(s)| \le \varepsilon_j\right)}{\sum_{m=1}^N \mathbf{1}\left(|x_m(s) - x_j(s)| \le \varepsilon_j\right)}$$
(2)

where $\varepsilon_j > 0$ are *confidence* levels. So, generally speaking, the matrices A(s) are not symmetric. It was shown in [7] that sometimes, depending on the choice of $\{\varepsilon_j\}$ and an initial configuration $\mathbf{x}(0)$, the model can exhibit an opinion fragmentation phenomenon.

Other opinion dynamics models (Abelson, Friedkin and Johnsen etc.) are out of the present discussion because they are not so closely related to the network consensus problem as (2).

2. Clock synchronization problem for WSNs

Wireless sensor networks (WSNs) are typical examples of distributed networks. The sensor nodes $\mathcal{N} = \{1, \ldots, N\}$ communicate via radio waves. Their interconnection matrices are naturally determined by range and power characteristics of the radio communication (see [16]).

Computer Simulations in Physics and beyond (CSP2017)

IOP Conf. Series: Journal of Physics: Conf. Series **955** (2018) 012038 doi:10.1088/1742-6596/955/1/012038

From now and onwards, the possibility of direct communications between pairs of nodes will be described by an $N \times N$ stochastic matrix $W = (w_{jk})_{j,k \in \mathcal{N}}$. Elements of \mathcal{N} will be called client nodes. There is also a stand-alone time server formally not belonging to the network but being reffered in the sequel as the node 0. The system is observed at discrete times. Any client node $j \in \mathcal{N}$ is equipped with an unperfect clock which current value is $\tau_j \in \mathbb{R}$. The time server (the node 0) has a perfect clock providing exact time $\tau_0 \in \mathbb{R}$. Let v > 0 be the rate of the clocks with respect to t. Being isolated these clocks evolve as

$$\tau_0(t) = \tau_0(t-1) + v, \qquad \tau_j^{\circ}(t) = \tau_j^{\circ}(t-1) + v + \delta_j(t), \quad j \in \mathcal{N},$$
(3)

where $t \in \mathbb{T} := \{1, 2, ...\}$ and $\delta(t) = (\delta_j(t), j \in \mathcal{N}) \in \mathbb{R}^N$ are i.i.d. random vectors representing random noise related to unperfect clocks of client nodes. We assume that $\mathsf{E}\,\delta_j(t) = 0$.

The evolutions (3) are discrete time versions of the clock models used in [17–20]. Since rates of clients are equal to the server rate v the above system of clocks is drift-free [21].

To perform a common task the WSN needs a common time. However, the client clocks $\tau_j^{\circ}(t)$ are desynchronized for two reasons: 1) the presense of random noise $\delta(t) \neq 0$; 2) initial values $\tau_j^{\circ}(0), j \in \mathcal{N}$, are different and unknown. The second reason is present even in the deterministic model ($\delta(t) = 0$). A natural idea is to run the following agreement algorithm: for any $j \in \mathcal{N}$

$$\tau_j(t) = \tau_j(t-1) + v + \delta_j(t) + \sum_{k \neq j} w_{jk} \left(\tau_k(t-1) - \tau_j(t-1) \right).$$
(4)

The sum in (4) is in fact taken over $k \in \mathcal{O}_j = \{k : w_{jk} > 0\}$ and so (4) is a distributed algorithm. The last summand in (4) is a correction made by the node j by using information on local time values $\tau_k(t-1)$ obtained from the neighbourhood \mathcal{O}_j . Equally, one can say that the sum in (4) introduces a special interaction between client clocks. Since $w_{jj} = 1 - \sum_{k \neq j} w_{jk}$ the matrix form of (4) is $\tau(t) = W \tau(t-1) + v \mathbf{1} + \delta(t)$ where τ and δ are column vectors of length N.

To eliminate the influence of a constant drift we introduce a new vector $\boldsymbol{\tau}'(t) = \boldsymbol{\tau}(t) - vt\mathbf{1}$. The iterative scheme (4) turns into $\boldsymbol{\tau}'(t) = W\boldsymbol{\tau}'(t-1) + \boldsymbol{\delta}(t)$ which is a stochastic version of the network consensus (NC) algorithm. Consider first the *deterministic case* $\boldsymbol{\delta}(t) = 0$. If the matrix W satisfies any known conditions ([1,6]) sufficient for reaching concensus then $\boldsymbol{\tau}'(t) \to c\mathbf{1}$ as $t \to \infty$. This means that the algorithm (4) provides an *internal synchronization* of the clients clocks, i.e., $\tau_{j_0}(t) - \tau_{k_0}(t) \to 0$ for any nodes j_0 and k_0 . For the stochastic model with $\boldsymbol{\delta}(t) \neq 0$ the local times $\tau_j(t)$ are random variables. So the best we can expect is that under suitable conditions (see [12, 13]) the differences $\tau_{j_0}(t) - \tau_{k_0}(t)$ converge in distribution to some limit probability laws. Nevertheless, the algorithm (4) keeps the client clocks far from the *accurate time* of the time server 0 and does not provide a global synchronization. To see this consider deviations $x_j = \tau_j - \tau_0$. The vector $\boldsymbol{x}(t) = (x_1(t), \ldots, x_N(t))$ evolves as $\boldsymbol{x}(t) = W\boldsymbol{x}(t-1) + \boldsymbol{\delta}(t)$ so all above arguments are applicable. For example, in the deterministic case $\boldsymbol{\delta} = 0$ under suitable assumptions one has $\tau_j(t) - \tau_0(t) \to c_0 \mathbf{1}$ as $t \to \infty$ but $c_0 \neq 0$ depends on initial values $\boldsymbol{\tau}(0)$ of the clients clocks which are unknown. More details can be found in [22].

3. Interaction with the time server

To synchronize client clocks $\tau_j(t)$ with the accurate time $\tau_0(t)$ we modify the model (4) by adding the time server (TS) to the system. The time server 0 can send messages to some (but not to all) client nodes. A message $m^{0\to j'}$ sent on some step t = t' from 0 to j' contains the value $\tau_0(t')$. The message $m^{0\to j'}$ instantly reaches the destination node j' which immediately adjusts its clock to the value recorded in $m^{0\to j'}$: $\tau_{j'}(t') = \tau_0(t')$. This is the usual message passing mechanism with zero delays. If on step t there is no message from 0 to j then the clock value $\tau_j(t)$ is adjusted according to the jth row of (4). A node just received a message from the time server is aware IOP Conf. Series: Journal of Physics: Conf. Series **955** (2018) 012038 doi:10.1088/1742-6596/955/1/012038

that its newly adjusted clock value is more precious than ones of its neighbours. If on step t_1 a client clock τ_{j_1} was set to the value $\tau_0(t_1)$ then during the time interval $t_1 + 1, \ldots, t_1 + \Delta_{j_1}$ the node j_1 decides to ignore opinions of its neighbours. Let $T^{0,j} = \left\{ t_n^{(j)} \right\}_{n \in \mathbb{N}}$ denote the sequence of steps t when j receives messages from 0. Hence to define the modified model one needs to specify $T := (T^{0,j}, j \in \mathcal{N})$ and $\Delta := (\Delta_j, j \in \mathcal{N})$. Now the variables $x_j = \tau_j - \tau_0$ evolve as in (1) where $W(s) = W(s; T, \Delta)$ is some $N \times N$ -matrix with entries $w_{jk}(s) \in \{0, w_{jk}, 1\}$ uniquely determined by the above algorithm. Notation D(s) = D(s; T) stands for the diagonal matrix of 0s and 1s, $D(s) := \text{diag}(\mathbb{I}_{\{s \notin T^{0,j}\}}, j \in \mathcal{N})$, indicating that random noise terms $\delta_j(s)$ are not added to $\tau_j(s)$ on steps $s = t_n^{(j)}$.

Define $\mathcal{R}(s) := \{j \mid T^{0,j} \ni s\} \subset \mathcal{N}$, the subset of client nodes receiving messages from the time server on the step s. From the viewpoint of the node 0 the set \mathcal{R} defines a prescribed sequence of recipients $\mathcal{R}(s)$ to whom it should consequently send messages. We assume henceforth that the *scheduling sequence* \mathcal{R} is not empty and *periodic* with period d, i.e., $\mathcal{R}(s+d) = \mathcal{R}(s), s \in \mathbb{T}$. It is easy to see that $\{D(s)\}_{s \in \mathbb{T}}$ and $\{W(s)\}_{s \ge s_0}$ are d-periodic too where $s_0 = s_0(\mathcal{R}, \Delta)$ is sufficiently large.

It is convenient to consider separately *two variants* of the above defined model: TS+DN (deterministic model with $\delta(t) = 0$) and TS+SN (stochastic model with $\delta(t) \neq 0$). It is readily seen that the mean $m(s) = \mathsf{E} x(s)$ of the TS+SN behaves as the state x(s) of the TS+DN.

An asymptotic behavior of the both TS+DN and TS+SN models was studied in [22]. Under some *additional* conditions on W and $\delta(t)$ we have the following results as $s \to \infty$.

1) In TS+DN model all clients synchronize with the time server, i.e., $x_i(s) = \tau_i(s) - \tau_0(s) \rightarrow 0$.

2) Var x(s), the covariance matrix of synchronization errors of the TS+SN, is bounded in s.

3) Any subsequence $\{x(nd+i)\}_{n\in\mathbb{N}}$ has a limit in distribution as $n\to\infty$.

The first item is similar to results on the NC problem in the presence of leaders [2,23]. Additional sufficient conditions assumed in [22] for proving 1–3 are easy to verify. We omit details.

4. Performance of the modified algorithm and network design problem

4.1. Spectral radius. Let $\mathcal{U}(s)$ be the set of clients which are receiving messages from the server node 0 on the step s or are ignoring opinions of neighbors on the step s. As it follows from [22] the rate of convergence of the modified algorithms depends of the spectral radius ρ of the matrix $\Pi = \overset{\circ}{W}(s_0 + d - 1)\overset{\circ}{W}(md + i - 1)\cdots\overset{\circ}{W}(s_0)$ where $\overset{\circ}{W}(s) = Z(s)W$ and $Z(s) = \text{diag}(\mathbb{I}_{\{j\notin\mathcal{U}(s)\}})$. Note that $\rho = \rho(W, \mathcal{R})$. It is clear that Π is substochastic and $0 \leq \rho < 1$. Below we describe some important cases when effective evaluation of ρ is possible.

4.2. Boundary and internal nodes. Denote by $\mathcal{B} = \{j : T^{0,j} \neq \emptyset\}$ the set of nodes receiving messages from the time server node 0. Evidently, $\mathcal{B} = \bigcup \mathcal{U}(s)$. Nodes of \mathcal{B} will be called

a boundary and the set of nodes $\mathcal{I} = \mathcal{N} \setminus \mathcal{B}$ will be called an *internal network*. The most interesting situation is that where the set \mathcal{B} is only a small part the whole distributed network. For WSNs it means that the internal network is large but the time server communicates only with relatively small number of boundary nodes. Let the nodes of \mathcal{N} be renumbered in such a way that $\mathcal{B} = \{1, 2, \ldots, k\}$ and $\mathcal{I} = \{k + 1, \ldots, k + n\}$. Thus N = k + n.

4.3. Design of the network. Assume that the internal network $\mathcal{I} = \{k + 1, \ldots, k + n\}$ and its interconnection (stochastic) matrix $W^{(\mathcal{I})} = (q_{j_1,j_2})_{j_1,j_2=k+1}^{k+n}$ are given. The task is to organize communication of \mathcal{I} with the time server. For WSNs it is easy imagine situations when the direct radio connection between 0 and \mathcal{I} is not possible due to natural obstacles (hills, walls etc.). An opinion dynamics interpretation of this task may be the following one. The opinion leader (node 0) and the client nodes speak different languages. It is necessary to add "translators" \mathcal{B} to organize transmission of information from the node 0 to client nodes \mathcal{I} .

doi:10.1088/1742-6596/955/1/012038

To build an accurate time synchronization algorithm the *network designer* needs

• to introduce a set of "boundary" nodes $\mathcal{B} = \{1, 2, \dots, k\},\$

IOP Conf. Series: Journal of Physics: Conf. Series 955 (2018) 012038

- to fix some periodic scheduling sequence $\mathcal{R} = \{\mathcal{R}(s), s \in \mathbb{T}\},\$
- to choose topology of connection links between \mathcal{I} and \mathcal{B} .

The choice of topology here is the choice of $W = (w_{i_1,i_2})_{i_1,i_2=1}^{k+n} \in M_{k+n}^{\text{st}}$ subject to the condition $w_{j_1,j_2} = r q_{j_1,j_2}, j_1, j_2 = k+1, \ldots, k+n$, where $r \in (0,1)$ is some coefficient.

4.4. A special class of synchronization algorithms. We say that a matrix $W \in M_{k+n}$ belongs to the class $\mathrm{HCRS}_{k,n}$ if

1)
$$w_{i_1j} = w_{i_2j}$$
 $\forall i_1, i_2 \in \{k+1, \dots, k+n\}, \quad j \in \{1, \dots, k\}.$

2) $\sum_{i=k+1}^{k+n} w_{ij} = \text{const} \quad \forall i \in \{k+1, \dots, k+n\}.$

In terms of the WSNs the item 1 means that all communications in the direction $\mathcal{B} \to \mathcal{I}$ (transmission of clock values information from nodes of the boundary set to nodes of the internal network) pass through a common gate. Note that there is no restriction on communications in direction $\mathcal{I} \to \mathcal{B}$ and communications between nodes of \mathcal{B} are also arbitrary. The item 2 evidently holds for the network designs discussed above. For opinion dynamics models the item 1 means that while different nodes of \mathcal{B} may have different information on opinion of the leader node 0, any boundary node distributes such information between nodes of \mathcal{I} uniformly.

Lemma 1. The class $HCRS_{k,n}$ is closed with respect to multiplication.

Definition 1. We write $W \in SD(\mathcal{B}, \mathcal{I}, W^{(\mathcal{I})})$ and call W a specially designed matrix if W is designed with \mathcal{B}, \mathcal{I} and $W^{(\mathcal{I})}$ as specified above and $W \in \mathrm{HCRS}_{k,n}$.

Lemma 2. If $W \in SD(\mathcal{B}, \mathcal{I}, W^{(\mathcal{I})})$ then any $\overset{\circ}{W}(s) = Z(s)W \in HCRS_{k,n}$.

Definition 2. Given $A \in M_{k+n}$ define $B^{(A)} = (b_{ij})_{i,j=1}^{k+1} \in M_{k+1}$ as follows $b_{ij} = a_{ij}, \quad (i,j) \in \{1, \dots, k+1\} \times \{1, \dots, k\}, \\ b_{i,k+1} = \sum_{m=k+1}^{k+n} a_{im}, \quad i \in \{1, \dots, k+1\}.$ **Lemma 3.** If $A^{(1)}, \ldots, A^{(m)} \in \mathrm{HCRS}_{k,n}$ then $B^{(A^{(1)} \cdots A^{(m)})} = B^{(A^{(1)})} \cdots B^{(A^{(m)})}$

4.5. Spectral problem in the case of specially designed W. Let $W \in SD(\mathcal{B}, \mathcal{I}, W^{(\mathcal{I})})$. Then from Lemmas 1 and 2 we see that $\Pi \in \text{HCRS}_{k,n}$. Consider the $(k+1) \times (k+1)$ matrix $B^{(\Pi)}$. By Lemma 3 $B^{(\Pi)} = B^{\left(\overset{\circ}{W}(s_0+d-1)\right)} B^{\left(\overset{\circ}{W}(s_0+d-2)\right)} \cdots B^{\left(\overset{\circ}{W}(s_0)\right)}$. Hence while Π is a large-size matrix, $B^{(\Pi)}$ can be obtained as a product of small-size matrices.

Denote by Λ_B and Λ_Q complete lists of eigenvalues of $B^{(\Pi)}$ and $W^{(\mathcal{I})}$, respectively, including all repeated eigenvalues. Clearly, $|\Lambda_B| = k + 1$. Let Λ'_Q be obtained from Λ_Q by excluding only one element 1. Denote by $(\Lambda'_Q)^d$ the list of *d*-powers of elements of Λ'_Q . Note that $\left|\Lambda_Q'\right| = \left|(\Lambda_Q')^d\right| = n - 1.$

Theorem 1. The complete list of eigenvalues of the matrix Π is $\Lambda_B \bigcup (r \Lambda'_Q)^d$.

This theorem is useful when some information on the spectrum of $W^{(\mathcal{I})}$ is a priori known and \mathcal{B} is relatively small. Proofs of the above statements are purely algebraic and are omitted.

5. Conclusions

We have analytically studied new algorithms for synchronization of local clocks in nodes of distributed networks. The proposed algorithms can also be applied to studying social networks dynamics in the presence of opinion leaders. We show that they fit to the general scheme (1)

Computer Simulations in Physics and beyond (CSP2017)

IOP Conf. Series: Journal of Physics: Conf. Series **955** (2018) 012038 doi:10.1088/1742-6596/955/1/012038

of distributed iterative algorithms with time-dependent topologies and a time-nonhomogeneous random noise. However, structures of matrices W(s) and D(s) in our algorithms are quite original. Our approach is to combine agreement algorithms with the message passing mechanism. The latter is widely used in various synchronization models [24–27]. We have discussed some network design problem naturally issued from our approach. We have described a class of algorithms that admit an evaluation of their spectrum, and, hence, a quantitative estimation of their performance.

References

- Olshevsky A and Tsitsiklis J N 2011 Convergence speed in distributed consensus and averaging SIAM Review 53(4) 747–772
- Jadbabaie A, Lin J and Morse A S 2003 Coordination of groups of mobile autonomous agents using nearest neighbor rules *IEEE Transactions on automatic control* 48(6) 988–1001
- [3] Ren W and Beard R W 2008 Distributed consensus in multi-vehicle cooperative control (Springer)
- [4] Ben-Jacob E, Schochet O, Tenenbaum A, Cohen I, Czirók A and Vicsek T 1994 Generic modelling of cooperative growth patterns in bacterial colonies NATURE 368
- [5] Cucker F and Smale S 2007 Emergent behavior in flocks IEEE Transactions on automatic control 52(5) 852–862
- [6] DeGroot M H 1974 Reaching a consensus Journal of the American Statistical Association 69(345) 118–121
- [7] Hegselmann R, Krause U et al. 2002 Opinion dynamics and bounded confidence models, analysis and simulation Journal of Artificial Societies and Social Simulation 5(3) 1–33
- [8] Vicsek T, Czirók A, Ben-Jacob E, Cohen I and Shochet O 1995 Novel type of phase transition in a system of self-driven particles *Phys. Rev. Lett.* **75** 1226–1229
- [9] Vicsek T and Zafeiris A 2012 Collective motion Physics Reports 517(3) 71–140
- [10] Kshemkalyani A D and Singhal M 2008 Distributed computing: principles, algorithms, and systems (Cambridge University Press)
- [11] Olfati-Saber R and 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
- [12] Cybenko G 1989 Dynamic load balancing for distributed memory multiprocessors J. Parallel Distrib. Comput. 7(2) 279–301
- [13] Xiao L, Boyd S P and Kim S J 2007 Distributed average consensus with least-mean-square deviation J. Parallel Distrib. Comput. 67(1) 33–46
- [14] Hendrickx J M and Blondel V D 2006 Convergence of linear and non-linear versions of Vicsek's model Proceedings of the 17th Symposium on Mathematical Theory of Networks (Kyoto) pp. 1229–1240
- [15] Shi G and Johansson K H 2012 Persistent graphs and consensus convergence Proceedings of the 51st Annual Conference on Decision and Control (IEEE) pp. 2046–2051
- [16] Simeone O, Spagnolini U, Bar-Ness Y and Strogatz S H 2008 Distributed synchronization in wireless networks IEEE Signal Processing Magazine 25(5) 81–97
- [17] Manita A 2014 Clock synchronization in symmetric stochastic networks Queueing Systems 76(2) 149–180
- [18] Manita L 2014 Controlling of clock synchronization in WSNs: structure of optimal solutions arXiv:1408.1434 [math.OC]
- [19] Manita L 2015 Singular extremals in control problems for wireless sensor networks Stochastic Modeling, Data Analysis and Statistical Applications (ISAST) pp. 461–474
- [20] Manita L 2016 Optimization problems for wsns: trade-off between synchronization errors and energy consumption Journal of Physics: Conference Series 681(1) 012009
- [21] Patt-Shamir B and Rajsbaum S 1994 A theory of clock synchronization Proceedings of the 26th Annual ACM Symposium on Theory of Computing (ACM) pp. 810–819
- [22] Manita L and Manita A 2017 Stochastic time synchronization models based on agreement algorithms Lecture notes in computer science 10684 361–375
- [23] Lin F, Fardad M and Jovanovic M R 2014 Algorithms for leader selection in stochastically forced consensus networks *IEEE Transactions on Automatic Control* 59(7) 1789–1802
- [24] Fujimoto R 2000 Parallel and Distributed Simulation Systems (Wiley)
- [25] Shchur L N and Shchur L 2015 Parallel discrete event simulation as a paradigm for large scale modeling experiments. Data Analytics and Management in Data Intensive Domains (Obninsk, Russia) pp. 107–113
- [26] Manita A 2017 Probabilistic issues in the node synchronization problem for large distributed systems Lobachevskii Journal of Mathematics $\mathbf{38}(5)$ 948–953
- [27] Manita A 2016 On behavior of stochastic synchronization models Journal of Physics: Conference Series 681(1) 012024