Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Asynchronous Consensus Protocols: Preliminary Results, Simulations and Open Questions

Lei Fang, Panos J. Antsaklis, and Anastasis Tzimas

Abstract— **Consensus is well accepted as being a fundamental paradigm for coordination of groups of autonomous agents. Recently, we casted previous work on multi-agent consensus protocols into an asynchronous framework, where each agent updates on its own pace, and uses the most recently received information from other agents. Asynchronous consensus protocols encompass those synchronous ones with various communication patterns. In this paper, we study via simulation a number of open new problems introduced by the asynchronous operation of multi-agent systems. More interestingly, the existing consensus results are classified by their communication assumptions and future research directions are proposed. To facilitate our study, we develop a multi-vehicle simulator in Java, built on top of JProwler; JProwler is a discrete event simulator for prototyping, verifying and analyzing communication protocols of ad-hoc wireless networks. Implementation issues with their implications are also discussed.**

I. INTRODUCTION

Consensus has been recognized as being of great importance in coordinating groups of autonomous agents [10], [14], [20], [21]. A consensus protocol provides means through which all agents agree on some particular variable of interest. This shared variable can be given different interpretations depending on the application and is a necessary condition for cooperation in multi-agent systems [23]. The challenge here is for the group to have a consistent view of the coordination variable in the presence of unreliable, dynamically changing communication topology without global information exchange.

A number of researchers have addressed the multi-agent consensus problem under the information constraints in different settings. Recent results on consensus problems include [9], [10], [12], [14], [16], [19]–[21], [25], [28], to name a few. See also [24] for a review on consensus problems in multi-agent coordination. In the aforementioned papers, the consensus protocols operate in a synchronized fashion since each agent's decisions must be synchronized to a common clock shared by all other agents in the group. This synchronization requirement might not be natural in certain contexts. This entails the consideration of the asynchronous consensus problem, where each agent updates on its own pace, and uses the most recently received (but possibly outdated) information from other agents. We refer readers to [2], [8] for surveys on general theory of asynchronous systems.

L. Fang, P. J. Antsaklis and Anastasis Tzimas are with Dept. of Electrical Engineering, Univ. of Notre Dame, Notre Dame, IN 46556. E-mail: {lfang,antsaklis.1}@nd.edu. This work was partially supported by the Center for Applied Mathematics, Univ. of Notre Dame.

Work reported on the asynchronous consensus problem is relatively sparse compared to its synchronous counterparts. In [6], we introduced an asynchronous framework to study the consensus problems for discrete-time multi-agent systems with a fixed communication topology under the *spanning tree assumption*. A distributed iterative procedure under the *eventual update assumption* was developed in [16] for calculating averages on asynchronous communication networks. The asynchronous consensus problem with zero time delay was studied in [4] where the union of the communication graphs has *a common root spanning tree*. It was shown in [6] that once the stability result was established for asynchronous protocols, the stability for synchronous protocols under dynamically changing interaction topologies is immediate since it can be seen as a special case of the asynchronous protocol with zero communication delays. In this sense, the asynchronous results extends the existing (synchronous) results. For other related problems in asynchronous multi-agent systems, see [1], [13], [15], [26].

The asynchronism gives a new dimension to the consensus problems. In this work, we examine via simulation a number of open new problems that arise from the asynchronous operation of multi-agent systems. In particular, we examine how asynchronous consensus value, delay and updating sets, sensor noise and finite convergence, and communication topologies affect (or are affected by) the consensus convergence process. Furthermore, the existing consensus results are classified by their communication assumptions and future research directions are pointed out. To facilitate our study, we develop a multi-vehicle simulator in Java, built on top of JProwler; JProwler is a discrete event simulator for prototyping, verifying and analyzing communication protocols of ad-hoc wireless networks [11]. Implementation issues with their implications are also discussed. The main purpose of this paper is to provide an asynchronous view of consensus problems with the goal to facilitate research along this new direction.

II. PRELIMINARIES AND BACKGROUND

A. Definitions and Notations

Let $G = \{V, E, A\}$ be a weighted digraph (or direct graph) of order n with the set of nodes $V = \{v_1, v_2, \dots, v_n\}$, set of edges $E \subseteq V \times V$, and a weighted adjacency matrix $A = [a_{ij}]$ with nonnegative adjacency elements a_{ij}. The node indices belong to a finite index set $\mathscr{I} = \{1, 2, \dots, n\}$. A directed edge of G is denoted by $e_{ij} = (v_i, v_j)$. For a digraph, $e_{ij} \in E$ does not imply $e_{ji} \in E$. The adjacency elements associated with the edges of the graph are positive, i.e., $a_{ij} > 0$ if and only if $e_{ji} \in E$. Moreover, we assume $a_{ii} \neq 0$ for all $i \in \mathscr{I}$. The set of neighbors of node v_i is the set of all nodes which point (communicate) to v_i, denoted by $N_i = \{v_j \in V : (v_j, v_i) \in E\}$.

A digraph G can be used to model the interaction topology among a group of agents, where every graph node corresponds to an agent and a directed edge e_{ij} represents a unidirectional information exchange link from v_i to v_j, that is, agent j can receive information from agent i. The interaction graph represents the communication pattern at certain time. The interaction graph is time-dependent since the information flow among agents may be dynamically changing. Let $\bar{G} = \{G_1, G_2, \ldots, G_M\}$ denote the set of all possible interaction graphs defined for a group of agents. The union of a collection of graphs $\{G_{i_1}, G_{i_2}, \ldots, G_{i_m}\}$, each with vertex set V, is a graph $\mathbb{G}$ with vertex set V and edge set equal to the union of the edge sets of G_{i_j}, $j = 1, \ldots, m$.

A directed path in graph G is a sequence of edges $e_{i_1 i_2}$, $e_{i_2 i_3}$, $e_{i_3 i_4}$, $\cdots$ in that graph. Graph G is called strongly connected if there is a directed path from v_i to v_j and v_j to v_i between any pair of distinct vertices v_i and v_j. Vertex v_i is said to be linked to vertex v_j across a time interval if there exists a directed path from v_i to v_j in the union of interaction graphs in that interval. A directed tree is a directed graph where every node except the root has exactly one parent. A *spanning tree* of a directed graph is a tree formed by graph edges that connect all the vertices of the graph.

Let $x_i \in \mathbb{R}$, $i \in \mathscr{I}$ represent the state associated with agent i. A group of agents is said to achieve global consensus asymptotically if for any $x_i(0)$, $i \in \mathscr{I}$, $\|x_i(t) - x_j(t)\| \to 0$ as $t \to \infty$ for each $(i, j) \in \mathscr{I}$.

Let $\mathbf{1}$ denote an $n \times 1$ column vector with all entries equal to 1. Let $M_n(\mathbb{R})$ represent the set of all $n \times n$ real matrices. A matrix $F \in M_n(\mathbb{R})$ is nonnegative, $F \geq 0$, if all its entries are nonnegative, and it is irreducible if and only if $(I + |F|)^{n-1} > 0$. Furthermore, if all its row sums are $+1$, F is said to be a (row) stochastic matrix.

B. Synchronous and Asynchronous Consensus Protocols

We consider the following (synchronous) discrete-time consensus protocol [17], [20], [21]

$$x_i(t+1) = \frac{1}{\sum_{j=1}^{n} a_{ij}(t)} \sum_{j=1}^{n} a_{ij}(t) x_j(t) \tag{1}$$

where $t \in \{0, 1, 2, \cdots\}$ is the discrete-time index, $(i, j) \in \mathscr{I}$ and $a_{ij}(t) > 0$ if information flows from v_j to v_i at time t and zero otherwise, $\forall j \neq i$. The magnitude of $a_{ij}(t)$ represents possibly time-varying relative confidence of agent i in the information state of agent j at time t or the relative reliabilities of information exchange links between them. We can rewrite (1) in a compact form

$$x(t+1) = F(t) x(t) \tag{2}$$

where $x = [x_1, \cdots, x_n]^T$, $F = F_{ij}$ with $F_{ij} = \frac{a_{ij}(t)}{\sum_{j=1}^{n} a_{ij}(t)}$, $(i, j) \in \mathscr{I}$. An immediate observation is that the matrix F is a nonnegative stochastic matrix, which has an eigenvalue at 1 with the corresponding eigenvalue vector equal to $\mathbf{1}$. Note that the protocol (1) or (2) is *synchronous* as all the agents update their states at the same time using the latest values of neighbors' states.

On the other hand, in the *asynchronous* setting the order in which states of agents are updated is not fixed and the selection of previous values of the states used in the updates is also not fixed. Now let $t_0 < t_1 < \cdots < t_n < \cdots$ be the time instants when the state of the multi-agent system undergoes change. Let $x_i(k)$ denote the state of agent i at time t_k. The index k is also called the event-based discrete time index in the literature. The dynamics of asynchronous systems can be written as

$$x_i(k+1) = \begin{cases} \sum_{j=1}^{n} F_{ij}(k) x_j(k - d(i,j,k)) & \text{if } i \in S(k), \\ x_i(k) & \text{otherwise}, \end{cases} \tag{3}$$

where $d(i, j, k) \geq 0$ are nonnegative integers, $S(k)$ are nonempty subsets of $\{1, \cdots, n\}$, the initial states are specified by $x(0) = x(-1) = \cdots$ Henceforth, we write the initial vector $x(0)$ to abbreviate reference to this set of equal initial states. We refer to the $d(i, j, k)$ as *iteration delays* and $S(k)$ as *updating sets*. The following assumption (called *regularity assumption*) is usually made in the study of asynchronous systems.

- There exists a nonnegative integer D such that
$$0 \leq d(i, j, k) \leq D < \infty, \ \forall\, (i, j, k). \tag{4}$$
Condition (4) indicates that only a finite number of updating instants can occur within any time interval of finite length.
- There exists a nonnegative integer B such that the updating sets $S(k)$ satisfy
$$\bigcup_{k=K}^{K+B} S(k) = \{1, \cdots, n\}, \text{ for any } K. \tag{5}$$
Condition (5) says that every agent is updated infinitely often as time goes on and, in particular, no agent fails to be updated within a time interval of length B.

C. Convergence Results

In this section, we review convergence results for synchronous and asynchronous consensus protocols under a (structurally) fixed topology. Both synchronous and asynchronous cases requires that the interaction matrix F has a directed spanning tree in the associated interaction graph G.

Theorem 1 ([22]) *Given the synchronous protocol (1) with $F(k) = F$, $\forall k \in \mathbb{N}$, the consensus is asymptotically reachable if and only if the associated interaction graph G has a spanning tree. That is, global consensus is asymptotically reachable.*

Theorem 2 ([6]) *Consider the asynchronous protocol (3) with structurally fixed topology $F(k) = F$, $\forall k \in \mathbb{N}$. Assume that all the agents can access their own states (i.e., F has positive diagonal entries) and at least one of the agents can* *access its own state without delay. Then the consensus is asymptotically reachable if the associated (directed) graph G has a spanning tree. That is, global consensus is asymptotically reachable under the asynchronous mode.*

Note that the time-varying version of Theorem 2 is also provided in the full version of [6]. The following theorem is obtained by several authors [6], [17], [21] via different approaches.

Theorem 3 *Let $G(t) \in \bar{G}$ be a time-varying interaction graph at time t, with the weights selected from a finite set of arbitrary positive numbers. The protocol (3) achieves global consensus asymptotically if and only if there exists an infinite sequence of contiguous, nonempty, bounded time intervals $[t_l, t_{l+1})$, $l \geq 0$, starting at $t_0 = 0$, with the property that across each such interval, the union of the interaction graphs has a spanning tree.*

III. Simulation Results and Open Questions

In the asynchronous framework, we examine how asynchronous consensus value, delay and updating sets, communication noise, and communication topology affect (or are affected by) the consensus convergence process. To be more concrete, we use as a baseline example a multi-agent system with six agents and a (structurally fixed and equally weighted) interaction topology as shown in Fig. 1. A number of simulation results are presented and open questions are raised.

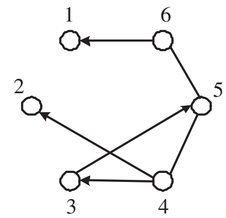


Fig. 1. The interaction topology of an asynchronous multi-agent system.

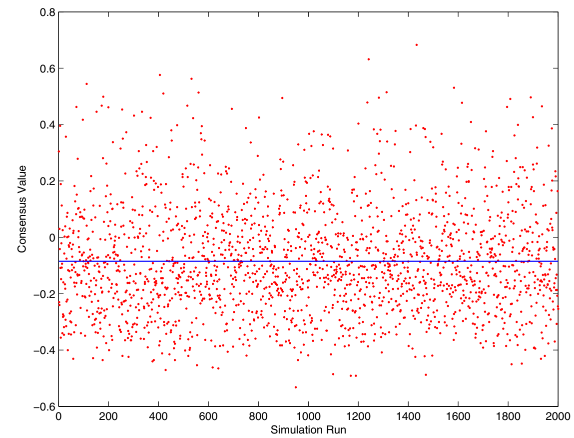


Fig. 2. Asynchronous consensus value ($p = 1/2$ and $D = 0$) depends on various topology selections made in the course of the computations (Horizontal line shows the synchronous consensus value).

A. Asynchronous Consensus Value

This part shows that the asynchronous consensus value generally depends on the course of the computations. The system is always simulated with the same initial condition $x(0) = [0.94 \ \ -0.90 \ \ -0.68 \ 0.85 \ \ -0.41 \ \ -0.71]^T$ for easy comparison. At every iteration, a node is chosen to update its state randomly and independently of other nodes with probability p. The delay $d(i, j, k)$ in (3) is a discrete random variable taking an integer value between 0 and D with an equal probability.

Fig. 2 shows different consensus values for 2000 Monte Carlo runs with node selection probability $p = 1/2$ and the delay bound $D = 0$ (zero-asynchronism). From this figure, while the synchronous consensus value depends only on the initial condition once the interaction topology is given, it is clear that the asynchronous consensus value can take any value in a bounded range (a rough estimate is given by $[\min_i x_i(0), \max_i x_i(0)]$). Interestingly, the mean of the asynchronous consensus values is very close to the synchronous consensus value of -0.085.

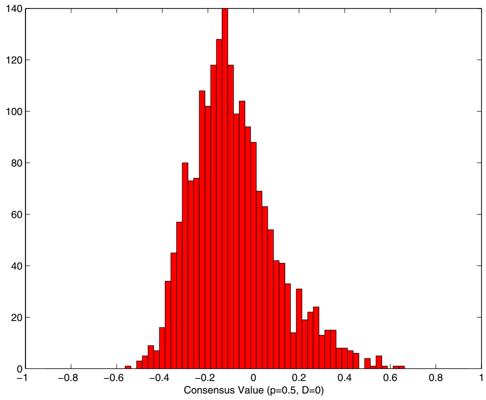
B. Delay and Updating Sets

The effects of the magnitudes of the node selection probability p and the delay bound D on the consensus dynamics are explored. On one hand, the asynchronous systems tend to behave more like synchronous systems with increasing p (for fixed topology F and delay bound D). As expected, this effect is less noticeable when D becomes large. On the other hand, one may expect that decreasing D (with fixed p) has a similar effect on asynchronous systems as increasing p. However, this turns out to be false. The histogram of consensus value with a large delay bound $D = 8$ (2000 simulation runs) is shown in Fig. 3(b). It can be seen that the consensus region instead shrinks by more than 40% compared to the zero asynchronism case Fig. 3(a).

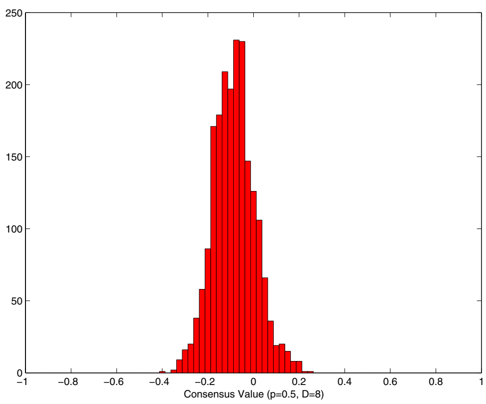
Simulation results also suggest that the time to reach consensus is shorter under zero asynchronism compared to that of the long delay case. Since a slow convergence process may result in a tighter region for the final consensus value, a fast consensus convergence process is not always desirable. This leads to the first open question:

Q1: What is the effect of p and D on the consensus convergence process? This problem is meaningful since we may have some prior knowledge about p and D. Does a deterministic updating scheme exist that would allow the control of the consensus value?

The known results in asynchronous theory suggest that the final consensus value is difficult to control [27]. To attack the problem, we should explore those special structures induced by the multi-agent systems. If agent A did not receive information from agent B but expects that information, A should retry the link to get information from B before updating its own state (as often used in the medium access control in wireless communication)? In certain case, it may be beneficial for an agent to put the time-varying weights



(a) Histogram of consensus value under zero asynchronism



(b) Histogram of consensus value with a large delay bound

Fig. 3. The effects of delay and updating sets in asynchronous consensus process.

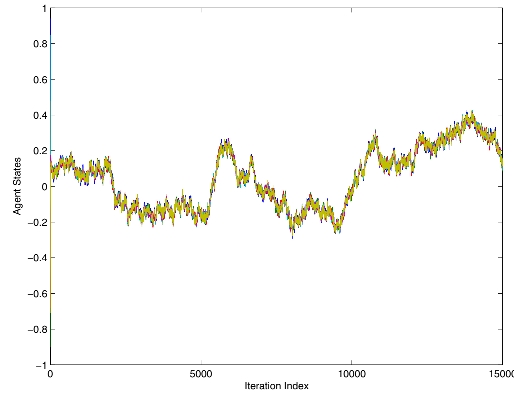


Fig. 4. Asynchronous consensus value driven by bounded noise.

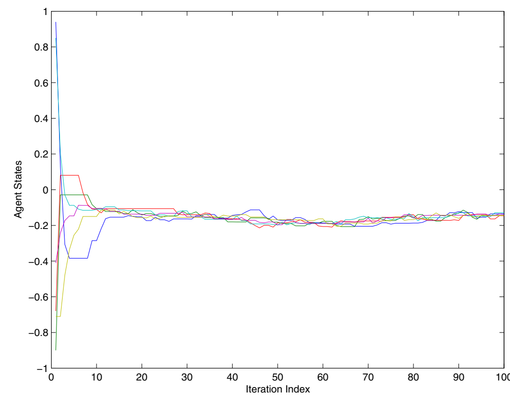


Fig. 5. Asynchronous consensus process can be safely terminated after certain time has passed.

(instead of the equal weights) on the state values received in computation. In short, we need to impose additional restrictions to the consensus process so to get the "desirable" final consensus value.

C. Sensor Noise and Finite Time Convergence

In practice, the agent state may be corrupted by noise due to the defective sensors or unreliable information exchange. For synchronous consensus protocols, explicit upper bounds for the inconsistency (absolute difference) between agent state when there exist bounded noise are given via an input-to-state analysis [22]. This analysis, however, cannot be directly adapted to the asynchronous case due to (computational) the path-dependency of the asynchronous consensus value. Fig. 4 shows one realization of the asynchronous consensus process for 15,000 updating events. Agent states become unbounded as $k \longrightarrow \infty$ when driven by the white noise, but the inconsistency among them is bounded. Fig. 5 provides a zoomed-in look at the same consensus process. Indeed, it is safe to terminate the consensus process after 100 iterations. A consensus termination condition should be identified so to terminate the consensus process earlier as long as agent state inconsistency is acceptable.

Q2: When does the consensus process need to be terminated under noisy communication? Will an adaptive updating step-size help the convergence of asynchronous consensus protocols?

One of the less pleasant facts of existing consensus protocols is that they only guarantee the consensus in the asymptotic manner.

Q3: How to construct communication protocols that guarantee the consensus in a finite number of iterations?

The above question is related to Q2 and was partly solved provided that the global information on the communication topologies is available to the agents [5]. But the difficulty here is how to solve the problem in a decentralized setting.

D. Assumptions on Communication

There are numerous consensus results in the literature. It is of great importance to classify them so to guide the future research directions. In the following, we re-examine several important results and focus on three classes of communication assumptions being made, i.e.,

- Direction: Bidirectional vs. Unidirectional;
- Connectivity: Uniform vs. Non-uniform;
- Synchronism: Synchronous vs. Asynchronous.

By uniform connectivity, we mean that a uniform (with respect to initial time) bound T is involved in the communication assumption on graph connectivity, c.f. Theorem 3.

Let us review two additional synchronous consensus theorems both using protocols (2) but with different communication assumptions.

Theorem 4 ([18]) *Consider a sequence of directed interaction graphs $(V, G(t))$ with common vertex set V. Assume that*

1) *For all $t_0 \in \mathbb{N}$ each agent is linked to each other agent across $[t_0, \infty)$ (i.e., each agent is the root of a spanning tree of $\bigcup_{t=t_0}^{\infty} G(t)$);*
2) *There is $T \in \mathbb{N}$ such that for all $t \in \mathbb{N}$ and all $v_1, v_2 \in V$ we have that if $(v_1, v_2) \in G(t)$ then v_2 is linked to v_1 across $[t, t+T]$ (i.e., there is a directed path from v_2 to v_1 in $\bigcup_{t=t}^{t+T} G(t)$).*

Then consensus is asymptotically reachable.

In the degenerate case $T = 0$, 2) is equivalent to the condition that there is a bidirectional link between v_1 and v_2.

Theorem 5 ([12], [17]) *Assume the interaction graphs are bidirectional for all $t \in \mathbb{N}$. If for all $t_0 \in \mathbb{N}$ the sequence of graphs $(V, G(t))$ has a spanning tree in $\bigcup_{t=t_0}^{\infty} G(t)$), then consensus is asymptotically reachable.*

We now categorize in Table I all the aforementioned theorems according to their communication assumptions. This way of categorizing consensus results is motivated by a private communication with L. Moreau. Since the spanning tree requirement plays a role in all the theorems, we do not explicitly consider this assumption in the table.

TABLE I

A CATEGORIZATION OF EXISTING CONSENSUS RESULTS

Theorems	Direction	Connectivity	Synchronism
1	Unidirect.	Fixed	Sync.
2	Unidirect.	Fixed	Async.
3	Unidirect.	Uniform	Sync.
4	Unidirect.	Non-uniform + 2)	Sync.
5	Bidirect.	Non-uniform	Sync.

We show the relationship between the theorems in the following diagram, where the arrows show that one theorem implies the other.

$$
\begin{array}{ccccccc}
\text{Th. 2} & \Rightarrow & \text{Th. 3} & \not\Leftrightarrow & \text{Th. 4} & \Rightarrow & \text{Th. 5} \\
\Downarrow & & & & & & \\
\text{Th. 1} & & & & & &
\end{array}
$$

There are a number of open questions that concern convergence under different assumptions, e.g.,

Q4: What are other sets of assumptions needed to guarantee that the synchronous/asynchronous consensus is reachable? What are the relationships among them?

A detail discussion of Q4 is provided in [7]; see also [3].

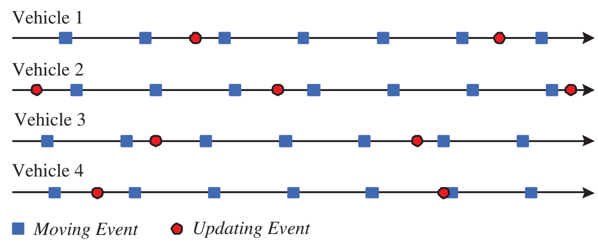


Fig. 6. Vehicles operate under an asynchronous mode.

IV. A MULTI-VEHICLE SIMULATOR

In order to implement and evaluate consensus protocols, a simulation environment is needed. We developed a Java-based multi-vehicle simulator (MultiVeh) on top of JProwler; JProwler is a discrete event simulator for prototyping, verifying and analyzing communication protocols of ad-hoc wireless networks. JProwler was selected because it has been widely used in the simulation study of wireless sensor and actuator applications. Each vehicle is modeled as a point mass with fully actuated dynamics and has the ability to move in two dimensions with the constant speed but varying headings (orientations). In particular the consensus parameter is the heading of the moving vehicles. In addition, each vehicle is able to communicate with others using wireless (Gaussian or Rayleigh) channels with the MAC protocol for MICA2 sensor nodes (a variant of CSMA). The collision avoidance mechanism is not considered at this time. The vehicle mobility creates a dynamic interaction graph and the MAC protocol imposes an asynchronous and uncertain message exchange among vehicles.

In the following, we concentrate on the inherent asynchronism in the system operation, which also justifies the usefulness of an asynchronous framework in studying consensus problems. Individual vehicle behavior could be described as a sequence of periodically (or quasi periodically) occurring events. Two events actually govern the vehicles' behavior, the *moving* event and the *updating* event. The moving event is translated into the movement of the vehicle to a new position according to its speed and heading. The updating event includes two steps that a vehicle has to complete. First, a vehicle computes its new heading using the heading information gathered from its neighbors. This information is collected by the vehicle since the last updating event. The second step is to broadcast its new heading to all of its neighbors, where neighbors are considered as those vehicles that can receive the message.

Assume that all the vehicles are initially synchronized and use the same channel to transmit their heading information. However, all the vehicles would fail to receive the messages due to packet collisions. It is known that the performance of the wireless protocols (CSMA/ALOHA) improves when vehicles are trying to access the medium at different time instants, with sufficient time interval between two successive tries. Successful transmission heavily depends on the number of vehicles and the rate at which new messages are generated.

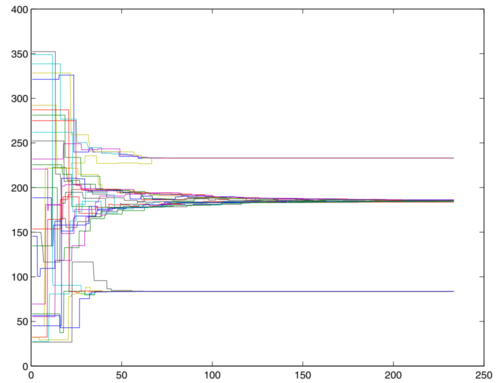


Fig. 7. Asynchronous updating of vehicles' heading information.

Fig. 6 illustrates the possible timing relationship between different events and different vehicles. A simulation example in Fig. 7 clearly shows the asynchronous updating of the heading of a group of vehicles. Another drawback of the wireless ad-hoc communication is the lack of guarantee of delivery. The time that a vehicle might wait to successfully transmit a message is unbounded. Such long delays for a vehicle to inform its neighbor could result in disconnected graph components. To avoid packet collisions and long time delays, the vehicles should spread medium access over the available time slots. If the available time slots are not enough then the rate of updating messages should be reduced. But, limited communication in the group is equivalent to "small neighborhoods," leading to weakly connected interaction graph and in turn affect vehicles' dynamics. As demonstrated in Fig. 7, two subgroups of vehicles may move toward different directions and it is impossible for them to reach consensus in the future due to the communication range limitation. It will be interesting to develop a scheme to prevent this from happening.

V. Discussions and Conclusions

By introducing asynchronism, we add a new dimension to consensus problems. The framework of asynchronous consensus is founded in the solid theoretical foundations laid out in previous work. A number of recent results were classified and their relationships with asynchronous consensus protocols are pointed out. Via simulations, the scheme has been demonstrated as feasible with rich and complex behaviors, which have also provided insights into future research directions. Current research focuses are shifting from the analysis of convergence properties of various consensus algorithms to design new algorithms to meet additional performance requirements (e.g., collision avoidance, obstacle avoidance, cohesion) based on some form of local coupling for the agents.

References

[1] G. Beni and P. Liang, "Pattern reconfiguration in swarms - convergence of a distributed asynchronous and bounded iterative algorithm," *IEEE Trans. R&A*, vol. 12, no. 3, pp. 485-490, 1996.

[2] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Prentice-Hall, Englewood Cliffs, NJ 1989.

[3] V. D. Blondel, J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis, "Convergence in multiagent coordination, consensus, and flocking," in *Proc. 44th CDC and ECC 2005*, Seville, Spain, 2005.

[4] M. Cao, A. S. Morse, and B. D. O. Anderson, "Coordination of an asynchronous multi-agent system via averaging," in *Proc. 16th IFAC World Congress*, Prague, 2005.

[5] G. Crombez, "Finding common fixed points of a class of paracontractions," *Acta Math. Hungar.*, vol. 103, no. 3, pp. 233-241, 2004.

[6] L. Fang and P. J. Antsaklis, "Information consensus of asynchronous discrete-time multi-agent systems," *American Contr. Conf.*, 2005, (Full version submitted for journal publication).

[7] L. Fang and P. J. Antsaklis, "On communication requirements for multi-agent consensus seeking," Workshop on Networked Embedded Sensing and Control, Notre Dame, Oct. 2005.

[8] A. Frommer and D. B. Szyld, "On asynchronous iterations," *J. Computational and Applied Math.*, vol. 123, pp. 201-216, 2000.

[9] Y. Hatano and M. Mesbahi, "Agreement over random networks," *Proc. 43rd IEEE CDC*, Atlantis, Bahamas, 2004.

[10] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Automat. Contr.*, vol. 48, no. 6, pp. 988-1001, 2003.

[11] JProwler, Available at `http://www.isis.vanderbilt.edu/projects/nest/jprowler`

[12] S. Li and H. Wang, "Multi-agent coordination using nearest neighbor rules: Revisiting the Vicsek model," `http://arxiv.org/abs/cs/0407021`

[13] J. Lin, A. S. Morse, and B. D. O. Anderson, "The multi-agent rendezvous problem - The asynchronous case," *Proc. 43rd IEEE CDC*, Atlantis, Bahamas, 2004.

[14] Z. Lin, M. Broucke, and B. Francis, "Local control strategies for groups of mobile autonomous agents," *IEEE Trans. Automat. Contr.*, vol. 49, no. 4, pp. 622-629, 2004.

[15] Y. Liu, K. Passino, and M. Polycarpou, "Stability analysis of M-dimensional asynchronous swarms with a fixed communication topology," *Proc. 2002 ACC*, Anchorage, AK, 2002, pp. 1278-1283.

[16] M. Mehyar, D. Spanos, J. Pongsajapan, S. H. Low, and R. M. Murray, "Distributed Averaging on Asynchronous Communication Networks," in *Proc. CDC-ECC'05*, Sevilla, Spain, 2005.

[17] L. Moreau, "Leaderless coordination via bidirectional and unidirectional time-dependent communication," *Proc. 42nd IEEE CDC*, Maui, HI, 2003, pp. 3070-3075.

[18] L. Moreau, "A note on leadless coordination via bidirectional and unidirectional time-dependent communication," *MTNS'04*, Belgium, 2004.

[19] L. Moreau, "Stability of multi-agent systems with time-dependent communication links," *IEEE Trans. Automat. Contr.*, vol. 50, no. 2, pp. 169-182, 2005.

[20] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Automat. Contr.*, vol. 49, no. 9, pp. 1520-1533, 2004.

[21] W. Ren and R. W. Beard, "Consensus of information under dynamically changing interaction topologies," *Proc. 2004 ACC*, Boston, MA, 2004, pp. 4939-4944.

[22] W. Ren and R. W. Beard, "Dynamic consensus seeking in distributed multi-agent coordinated control," submitted to *IEEE 2004 Int. Conf. R&A*, 2004.

[23] W. Ren, R. W. Beard, and T. W. McLain,"Coordination variables and consensus building in multiple vehicle systems," *Lecture Notes in Control and Information Systems*, vol. 309, Springer-Verlag, pp. 171-188, 2004.

[24] W. Ren, R. W. Beard, and E. M. Atkins, "A survey of consensus problems in multi-agent coordination," *American Control Conf.*, 2005.

[25] J. J. E. Slotine and W. Wang, "A study of synchronization and group cooperation using partial contraction theory," *Block Island Workshop on Cooperat. Contr.*, V.J. Kumar, N.E. Leonard, A.S. Morse Eds., Springer-Verlag, 2003.

[26] F. J. Vazquez-Abad, C. G. Cassandras, and V. Julka, "Centralized and decentralized asynchronous optimization of stochastic discrete-event systems," *IEEE Trans. AC*, vol. 43, no. 5, pp. 631-655, 1998.

[27] M. Verkama, "Random relaxation of fixed-point iteration," *SIAM J. Scientific Computing*, vol. 17, no. 4, pp. 906-912, 1996.

[28] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Proc. 42nd IEEE CDC*, Maui, HI, 2003, pp. 4997-5002.