

## Review Article

# Principles, Applications, and Challenges of Synchronization in Nature for Future Mobile Communication Systems

Hyun-Ho Choi<sup>1</sup> and Jung-Ryun Lee<sup>2</sup>

<sup>1</sup>*Department of Electrical, Electronic and Control Engineering and the Institute for Information Technology Convergence, Hankyong National University, Anseong 17579, Republic of Korea*

<sup>2</sup>*School of the Electrical Engineering, Chung-Ang University, Seoul 06974, Republic of Korea*

Correspondence should be addressed to Jung-Ryun Lee; [jrlee@cau.ac.kr](mailto:jrlee@cau.ac.kr)

Received 18 November 2016; Revised 13 December 2016; Accepted 22 December 2016; Published 22 January 2017

Academic Editor: Yujin Lim

Copyright © 2017 Hyun-Ho Choi and Jung-Ryun Lee. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In dynamic and complex natural environments, a number of individuals such as fireflies, flocks of birds, schools of fish, and pacemaker cells show various synchronization phenomena, with the purpose of achieving their certain goals in an efficient and distributed manner. So far, there has been considerable research attention on the working principles behind synchronization phenomena in nature and, as a result, various models and theoretical investigation have been developed to apply synchronization principles to various mobile communication systems. In this article, we present an exploration of synchronization phenomena in nature. Some representative models on synchronization are investigated and its working principles are analyzed. In addition, we survey some key applications inspired by synchronization principles for the future mobile communication systems. The characteristics and limitations of the applications inspired by synchronization in nature are evaluated in the context of the use of nature-inspired technologies. Finally, we provide the discussion of further research challenges for developing the advanced application of natural synchronization phenomena in the future mobile communication systems.

## 1. Introduction

The inclination of living entities, ranging from animals to humans, to synchronize with each other is the most common tendency in the universe [1]. Thousands of fireflies synchronously illuminate, while geese fly at the same speed in formation. Applause at concert halls merges to produce a harmonized sound in time, and the menstrual periods of women who closely interact for a long time also synchronize. Thousands of cardiac pacemaker cells in the heart fire in synchronization to sustain life. Inanimate objects, such as particles and planets, synchronize as well. Laser beams are created when trillions of atoms oscillating in sync emit photons of the same phase and frequency. Moreover, only one side of the moon can be viewed because the orbital and rotational periods of the moon are synchronized by the gravitational pull between the earth and moon.

As shown in the above examples, synchronization occurs with both conscious living organisms and inanimate objects

without consciousness. A key aspect of this phenomenon is that it does not involve a leader who directs the behavior, nor does it require the obtaining of clues from the surrounding environment. Rather, the entities synchronize in a certain rhythm. In other words, this order of synchronization emerges—it naturally occurs out of “nothing”—in all cases [2].

Fireflies, flocks of birds, schools of fish, and pacemaker cells are all modeled as populations of oscillators [3]. An oscillator is an entity that continues to repeat itself in a regular time interval. If more than one oscillator influences another oscillator through a physical or chemical process, they are considered to be connected. Fireflies communicate through light, birds in a flock identify their respective position using their sight, and pacemaker cells transmit electric currents to each other. Nature utilizes all available communication channels for the connecting of oscillators. This kind of communication often results in synchronization.

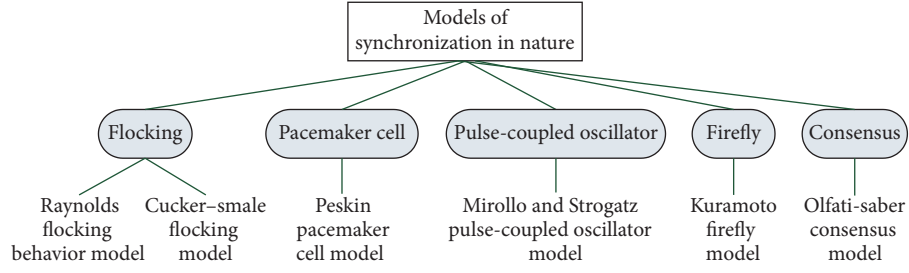


FIGURE 1: A taxonomy of models of synchronization in nature.

Synchronization in nature has been attracting considerable research attention and has been applied to theoretical investigations and a variety of mobile communication and networking systems [4–6]. Some examples of its applications include time synchronization and scheduling in large-scale distributed networks; distributed fusion in sensor networks; rapid consensus in various network topologies; distributed formation control of multiple vehicles; and solutions of nonlinear optimization problems. Due to the advancement of communication technologies in the last few years, the number of nodes available for networking is dramatically increasing. Consequently, applications that are intended to achieve collective goals through cooperation among nodes are increasing. However, the conventional approach of controlling numerous nodes in a centralized system has limitations in terms of cost, complexity, resource availability, and performance. Therefore, approaches inspired by the phenomena of natural orders, such as synchronization—which efficiently achieve certain goals in complex environments with various constraints—are being investigated to address complex engineering problems.

In this article, we examine synchronization phenomena in nature, elucidate representative mathematical models concerning synchronization, and investigate the principles of synchronization. We then introduce some applications that apply synchronization principles to communication network systems. We thereby analyze the attributes and limitations of nature-inspired synchronization methods. Finally, we discuss research challenges for developing the advanced applications based on synchronization phenomena in the future mobile communication systems.

The remainder of this article is organized as follows. Section 2 overviews representative theoretical studies on synchronization, and the principles of synchronization are investigated. Section 3 examines major applications based on these principles. Section 4 describes challenges involved in applying synchronization phenomena. Section 5 discusses the similarities between synchronization in nature and certain engineering problems, while highlighting the use of nature-inspired technologies. Section 6 presents our conclusions.

## 2. Principles of Synchronization

In this section, representative theoretical synchronization models are outlined to elucidate synchronization principles and related conditions. The taxonomy of synchronization

model is provided in Figure 1, which is classified by the objects in nature that inspire to make each synchronization model.

**2.1. Reynolds Flocking Behavior Model.** Reynolds created the first behavioral model of animal groups, such as those of birds and fish, and he demonstrated their behaviors through computer simulation [7]. In this “flocking” model, animal group behavior aligns with three rules—separation, alignment, and cohesion—as shown in Figure 2. Each animal in a group independently controls its own position and speed based on these rules. According to the separation rule, each animal tends to maintain a certain distance from its neighboring group members. In the alignment rule, each animal tends to determine its heading direction in accordance with the average direction of its neighboring group members. According to the cohesion rule, each animal tends to steer itself toward the average position of its neighboring group members to avoid being separated from them. An animal’s neighbors in this context are determined according to the physical distance between the respective animal and the surrounding animals, as well as their respective visual fields. The other group members that are not recognized in this way are excluded from the calculation. Accordingly, each animal in the group adjusts its position and speed based on the average position and speed of its neighbors in accordance with the three rules. After a period of time, the animals move at the same speed in the same distance. That is, synchronization occurs in terms of distance and speed.

**2.2. Peskin Pacemaker Cell Model.** Peskin modeled cardiac pacemaker cells as a parallel circuit with capacitance and resistance [8]. This electrical circuit is charged along a gradually increasing curve and fires when the voltage reaches a certain threshold, as shown in Figure 3. Thereafter, the voltage resets and a new cycle begins. That is, this behaves like an oscillator. The pacemaker cell model mimics the periodic firing of cardiac cells and decreasing of voltage to the bottom of a curve.

Then, Peskin modeled the cardiac cellular phenomenon as a network of electrical oscillators. Among  $N$  oscillators, the voltage of the  $i$ th oscillator,  $x_i$ , is modeled as

$$\frac{dx_i}{dt} = S_0 - \gamma x_i, \quad 0 \leq x_i \leq 1, \quad i = 1, 2, \dots, N, \quad (1)$$

where  $S_0$  is the initial value and  $\gamma$  indicates the rate of dissipation. The  $i$ th oscillator fires when  $x_i = 1$  and it regresses

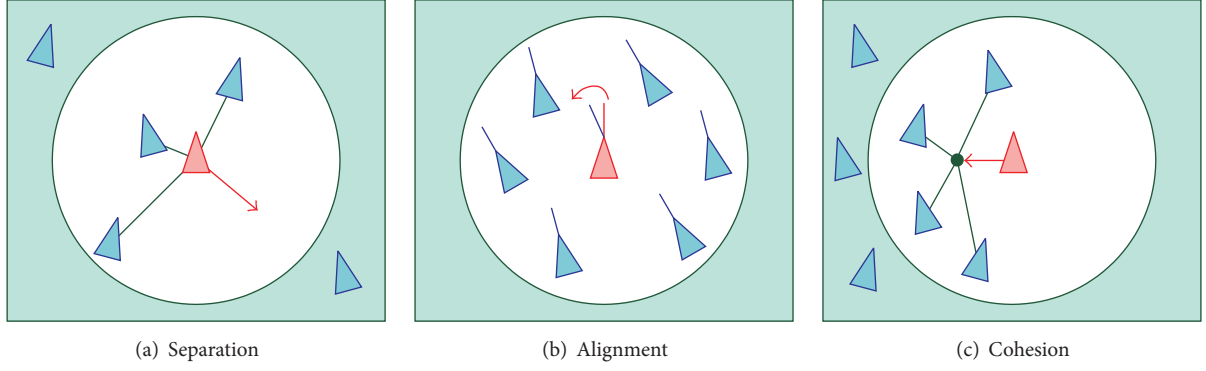


FIGURE 2: Three rules of the flocking behavior model.

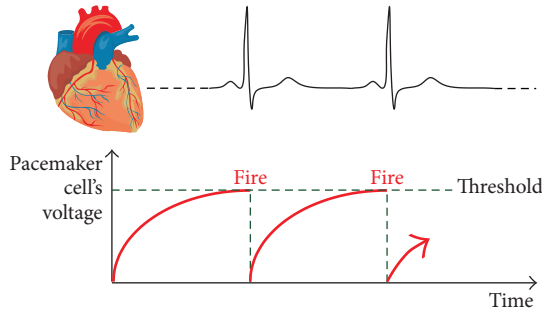


FIGURE 3: Cardiac pacemaker cell model.

to  $x_i = 0$ . Oscillators influence each other through firing. If an oscillator fires, either the voltages of the other oscillators that detect it increase by coupling strength  $\epsilon$  or they fire when their voltages increase beyond the given threshold. This rule is expressed as

$$x_i(t) = 1 \rightarrow x_j(t^+) = \min(1, x_j(t) + \epsilon), \quad \forall j \neq i. \quad (2)$$

Assuming that all oscillators are identical (i.e., they have the same charge curve) and are coupled by the same strength, Peskin proved that two oscillators always simultaneously converge and fire, even when starting activity in random initial conditions.

**2.3. Strogatz Pulse-Coupled Oscillator Model.** Mirollo and Strogatz modeled the synchronization phenomenon of entities into a pulse-coupled oscillator (PCO) [9]. PCO has an integrated clock. It emits pulses along a certain defined cycle and receives the pulses of other oscillators through the media existing between them. When receiving these pulses, each oscillator adjusts its internal clock according to appropriate clock adjustment rules. The synchronization then occurs after a period of time.

Figure 4 illustrates the PCO phase synchronization process. In the PCO model, each node acts as an oscillator with a fixed time cycle,  $T$ . The oscillator has an internal time phase,  $t$ , and increases at a certain rate from  $t = 0$  to  $t = T$ . When  $t = T$ , the nodes fire and the phase is reset to  $t = 0$ . At that point, the surrounding nodes that detect the firing signal

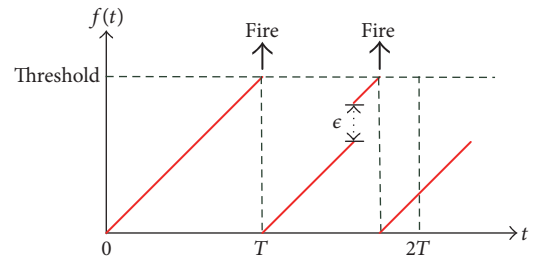


FIGURE 4: PCO phase synchronization process.

adjust their phases of  $t$  to the upper point to shorten the time until firing. The new phase value,  $t_{\text{update}}$ , is determined as

$$t_{\text{update}} = f^{-1}(f(t) + \epsilon), \quad (3)$$

where firing function  $f$  should be an asymptotically increasing concave function, and the increased amount  $\epsilon$  is a real number smaller than 1. When the value of  $t_{\text{update}}$  is greater than  $T$ , the node immediately fires and  $t_{\text{update}}$  is reset to 0. The nodes initiate activity with different  $t$  values; nonetheless, in accordance with the same rules, their phases of  $t$ , respectively, converge to the same value in the course of time.

Strogatz generalized the coupled oscillator model of Peskin into  $N$  oscillators. According to Strogatz model, under the assumption that all oscillators are identical and connected, it is proved that  $N$  oscillators with random initial values always synchronize. However, it was found that PCO model synchronization is not realized in actuality when noise or a pulse transmission delay exists [10].

**2.4. Kuramoto Firefly Model.** Assuming that identical oscillators are weakly coupled and the interactions between the oscillators are dependent on the sine function of the phase difference, Kuramoto modeled the synchronization of fireflies as

$$\frac{d\theta_i}{dt} = \omega_i - \frac{K}{N} \sum_{j=1}^N \sin(\theta_i - \theta_j), \quad i = 1, 2, \dots, N, \quad (4)$$

where  $\theta_i$  is the phase of each individual,  $\omega_i$  is the frequency of each individual, and  $K$  is the coupling strength [11–13].

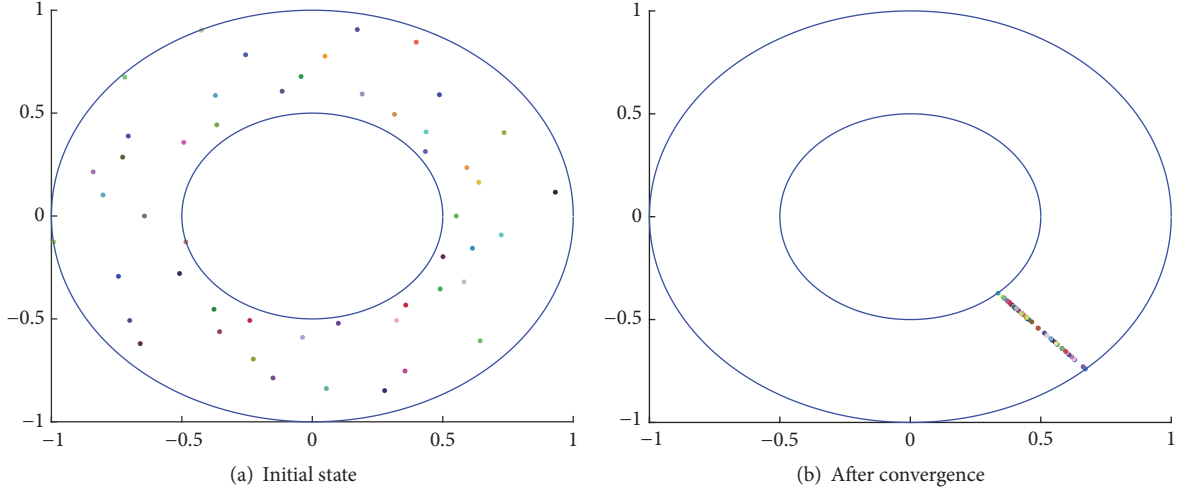


FIGURE 5: Phase synchronization phenomenon according to the Kuramoto model.

In the initial state before synchronization, each individual oscillates at its own frequency, and its phase differs from those of the others. However, the phase of each individual changes in time through interactions by  $\sin(\theta_i - \theta_j)$ . Ultimately, synchronization occurs when the phases of all the individuals converge, as shown in Figure 5.

Unlike the Strogatz model, the Kuramoto model considers oscillators that have different natural frequencies and assumes that the interactions of the oscillators are limited to the sine function. Therefore, the Kuramoto model can be exactly solved in mathematical terms, even though it is a nonlinear model. This model achieves synchronization with a random initial value on the condition that  $K$  value is sufficiently large in a fully connected network [14]. However, synchronization is not always guaranteed when the differences of the natural frequencies of the oscillators transcend a specific range or when a transmission delay exists [15].

**2.5. Cucker–Smale Flocking Model.** Cucker and Smale mathematically modeled the flocking behavior of birds in flight as

$$v_i(t+1) - v_i(t) = \frac{\lambda}{N} \sum_{j=1}^N \Psi(|x_j(t) - x_i(t)|) (v_j(t) - v_i(t)), \quad (5)$$

where  $x_i(t)$  and  $v_i(t)$  are the position and velocity of the  $i$ th bird at time  $t$ , respectively [16].  $N$  is the total number of birds,  $\lambda$  is the coupling strength among the birds, and  $\Psi$  is the communication range function with input of the distance between two birds. For example, if each bird in a flock has only information of the adjacent birds within a certain distance,  $r$ , then  $\Psi = 1$  when  $|x_j - x_i| \leq r$  and  $\Psi = 0$  otherwise. As expressed in (5), the Cucker–Smale flocking model shows that each bird adjusts its velocity by averaging the velocity of its perceived neighbors. Therefore, this model can be a mathematical representation of the Reynolds flocking behavior model.

Cucker and Smale considered various  $\Psi$  functions and proved the synchronization conditions for the general  $\Psi$  functions in which path loss is proportional to distance [16, 17]. They proved that the convergence value by (5) becomes the average of the initial velocities of the birds. Additionally, they showed that the position of each bird does not diverge; rather, it remains within a certain range of space when converging. The convergence characteristics of the Cucker–Smale model are expressed as

$$v_c := \lim_{t \rightarrow \infty} v_i(t) = \frac{1}{N} \sum_{i=1}^N v_i(0), \quad \text{for } \forall i, \quad (6)$$

$$\sup_{0 \leq t < \infty} |x_i(t) - x_j(t)| < \infty, \quad \text{for } \forall i \neq j.$$

Figure 6 shows the simulated synchronization phenomenon of bird flocking according to this model.

**2.6. Olfati-Saber Consensus Model.** In dynamic systems, such as multiagent networks, the term “consensus” is used to mean “synchronization” [18]. Consensus connotes the reaching of an agreement regarding a certain quantity of interest that depends on the state of all agents. The consensus algorithm refers to an interaction rule that specifies the information exchange between an agent and all of its neighbors in the network in order to reach a consensus.

In accordance with continuous time and discrete time intervals in a linear system, a generalized consensus algorithm is, respectively, expressed as

$$\begin{aligned} x'_i(t) &= \sum_{j \in N_i} a_{ij} (x_j(t) - x_i(t)), \\ x_i(k+1) &= x_i(k) + \epsilon \sum_{j \in N_i} a_{ij} (x_j(k) - x_i(k)), \end{aligned} \quad (7)$$

where  $x_i(t)$  is the status value of node  $i$  at time  $t$ ,  $N_i$  is the set of neighboring nodes of node  $i$ ,  $a_{ij}$  is the coupling strength

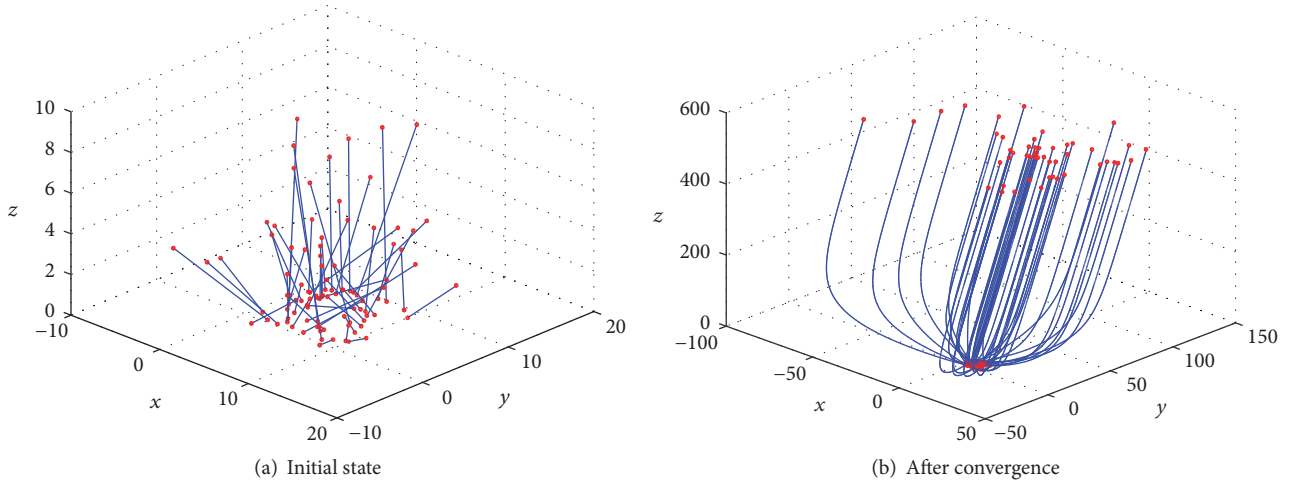


FIGURE 6: Synchronization phenomenon of bird flocking in three-dimensional space according to the Cucker–Smale flocking model.

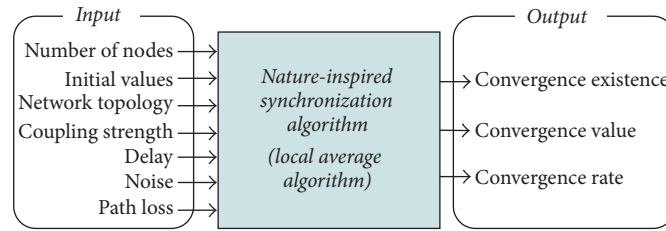


FIGURE 7: System model of synchronization.

between the two nodes,  $i$  and  $j$ , and  $\epsilon > 0$  is the step size. According to (7), the change of the status value of each node is determined by the sum of the differences of the neighboring nodes' state values. Based on these algorithms, the status value of each node converges in time to the average of the initial values of all the nodes. It is given as

$$\alpha = \frac{1}{N} \sum_{i=1}^N x_i(0), \quad (8)$$

where  $N$  is the total number of nodes. The expressions and convergence result of this kind of consensus algorithm correspond to those of the Cucker–Smale model.

Various theoretical studies have been conducted on the consensus model and their convergence rate has been analyzed. Equations (7) can be simply converted into the matrix forms as follows:

$$\begin{aligned} \mathbf{x}' &= -\mathbf{L}\mathbf{x}, \\ \mathbf{x}(k+1) &= \mathbf{P}\mathbf{x}(k), \end{aligned} \quad (9)$$

where  $\mathbf{L}$  is a Laplacian matrix and  $\mathbf{P}$  is a Perron matrix. These matrices are determined according to the given network topology. The convergence rate of each consensus algorithm is determined depending on the second smallest eigenvalue of Laplacian  $\mathbf{L}$  and the second largest eigenvalue of Perron  $\mathbf{P}$ , respectively [18].

**2.7. Synchronization System Model.** In synchronization studies to date, the action of each individual is mathematically modeled. Then, the existence of convergence, the convergence value, and the convergence rate are theoretically analyzed based on the various conditions, such as the number of participating nodes, initial values, network topology, coupling strength between nodes, delay, noise, and path loss. The principle of synchronization algorithms is that each node repeats the process of updating its own information using only the information of the neighboring nodes. This update approach can thus be represented as the *local average* algorithm by which the convergence value is decided solely based on the average of the initial values. This synchronization algorithm can be modeled into a generalized system with inputs and outputs, as depicted in Figure 7.

### 3. Applications of Synchronization

In this section, we introduce some major applications based on synchronization principles, focusing on communication and networking systems. The taxonomy of synchronization applications is provided in Figure 8, which is classified by the major applications used in communication and networking systems.

**3.1. Distributed Time Synchronization.** In an environment in which many nodes exist and the network dynamically changes according to the movements, entering, and exiting



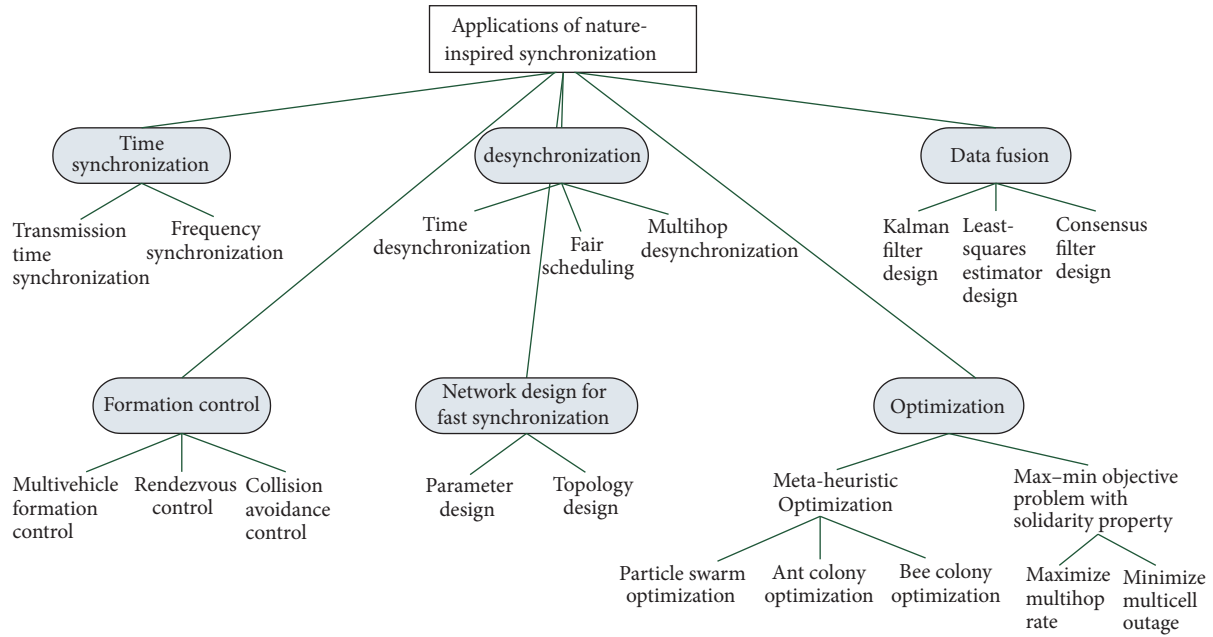


FIGURE 8: A taxonomy of applications of nature-inspired synchronization.

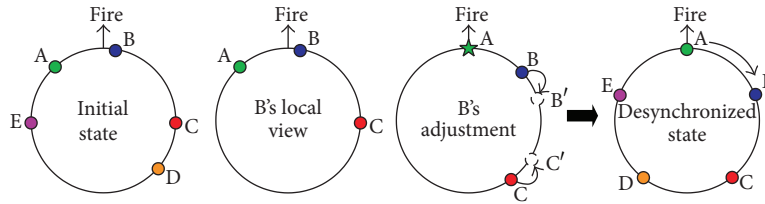


FIGURE 9: DESYNC algorithm.

of nodes—such as in mobile ad hoc, wireless mesh, and sensor networks—a nature-inspired distributed synchronization method is more suitable than the conventional centralized synchronization methods. Various studies on this premise have been conducted. A synchronization method for node transmission time was suggested by applying the synchronization principle of fireflies to a large-scale network [19, 20]. Therein, the Strogatz PCO model was applied to a large-scale sensor network. Synchronization performance was then analyzed according to the number of nodes and optimized according to the tradeoff of the energy efficiency and convergence rate [21]. A technique for synchronizing the timing of separate networks was suggested by using the positions and mobility of the nodes in mobile ad hoc networks [22]. A distributed frequency synchronization technique using a bioinspired algorithm was suggested to solve the problem of multiple frequency offsets in mesh networks wherein multiple nodes have different carrier frequencies [23]. Lastly, a synchronization method was suggested for realistic situations in which a time delay occurs across nodes and not all of them are connected [15, 24].

**3.2. Desynchronization.** Unlike synchronization, by which oscillators converge to the same time phase, desynchronization (DESYNC) separates the oscillators as far as possible

from each other. As a result, the phase differences of all the nodes become the same. In DESYNC, the synchronization target is not the phase value itself; instead, it is the phase difference between the nodes. A distributed DESYNC algorithm based on the PCO model was suggested for the desynchronizing of time event [25, 26]. As shown in Figure 9, each oscillator detects the firing of another oscillator immediately before and immediately after its own firing. It adjusts its own phase to the medium position of the firing time of its two neighbors. This process is repeated and DESYNC is completed.

A round-robin scheduling method that fairly allocates resources in a distributed manner without access collision was suggested by applying the DESYNC algorithm to time-division multiple access (TDMA) scheduling [25, 26]. DESYNC-based TDMA scheduling enables high throughput and low control overhead, even in a heavy load, because it does not induce signaling messages for scheduling. Furthermore, a distributed scheduling method was proposed by applying DESYNC to multihop networks. This approach flexibly controls the transmission intervals between nodes in a situation wherein the nodes randomly enter or exit [27]. Lastly, proportional fair scheduling was achieved by enabling each node to operate two oscillators through the expansion of the DESYNC algorithm [28].

**3.3. Distributed Fusion in Sensor Networks.** In a sensor network, it is necessary to reduce the amount of transmitted information through the fusion of sensed information in intermediate nodes, rather than to concentrate all information sensed from each node to a center and simultaneously compute it. In particular, as the number of nodes increases, the amount of sensed information to transmit rapidly increases. Thus, the fusion of sensed information is more necessary in various parts of the network. The fusion of sensed information is a process in which multiple sensor nodes cooperatively make a decision. This process is used to extract a useful piece of information by reaching a consensus among the nodes.

Various methods that compute the average in a distributed manner in sensor networks were suggested to realize the Kalman filter in this context [29, 30]. Additionally, the linear least-squares estimator (LSE) method was proposed to compute an average based on the consensus among network nodes [31]. Lastly, to more efficiently compute the average of sensed information, low and high pass consensus filters were suggested [32].

**3.4. Network Design for Fast Synchronization.** Nature-inspired synchronization algorithms require iterations and thus require time to obtain the desired convergence value. To overcome this constraint, the development of a network design that achieves fast synchronization is an active research topic. In [33], for example, a study on shortening the convergence time of a synchronization algorithm was presented. The convergence time was reduced by varying the weight, which is multiplied for each operation when the average is calculated.

In another study, the weight was fixed and the network topology was varied to stimulate faster synchronization [34, 35]. When a small-world network was created by rewiring random nodes in a given network topology, it was confirmed that the synchronization speed is dramatically faster. Figure 10 shows the difference in convergence times between a regular network, in which 100 nodes are wired within three hops, and a small-world network, wherein an additional connection is made between random 300 nodes on the regular network.

**3.5. Distributed Formation Control.** The multivehicle system is an important networking system for commercial and military applications. Controlling the number of vehicles to be automatically driven in a certain formation requires cooperation between nodes. It is thus necessary for the vehicles to synchronize to achieve their collective mission. It was proven that distributed multivehicle formation control is a matter of synchronization and a related theoretical framework was developed [36].

In addition, the matter of a space rendezvous is regarded as a synchronization problem concerning the position of various individuals that interact [37, 38]. In the rendezvous problem, the occurrence of convergence according to the network topology is important; however, it is irrelevant to the convergence value. Furthermore, a theoretical framework was developed for the design and analysis of a flocking algorithm that avoids collision with obstacles [39]. For flocking,

the relation with obstacles should be considered as well as the interaction between individuals. In this study, the synchronization algorithm is used to enable an individual to avoid colliding with obstacles and other individuals based only on the information of the surroundings, while synchronizing the velocities of all individuals.

**3.6. Solution for Optimization Problems.** Various metaheuristic optimization methods based on swarm intelligence were suggested, some examples of which include particle swarm optimization, ant colony optimization, and bee colony optimization [40]. In this type of optimization technique, numerous particles repeatedly explore a multidimensional solution space by exchanging information with each other to determine the most optimal value. This approach is similar to the synchronization phenomenon in the sense that various individuals interact in a distributed manner and converge toward a single optimal point. However, it does not align with the basic synchronization algorithm, which obtains the average of the local information of interest.

For a certain optimization problem, it was proven that the optimal solution is to synchronize specific variables [41–43]. When the value sought by each individual has a “solidarity” property (i.e., the increase/decrease of the value of an individual causes the decrease/increase of the values of the others) and the objective problem is to maximize the minimum of these values (i.e., the max–min objective problem), it was proven that the optimal solution is to make these values equal, that is, to synchronize them.

Based on the above proposition, [41] suggested a distributed transmit-power control algorithm that maximizes the end-to-end throughput in wireless multihop networks and [43] proposed a distributed transmit-power control algorithm that minimizes the outage probability in wireless cellular networks. The rate of each link in the wireless network has a solidarity property on account of mutual interference. Moreover, the end-to-end throughput in multihop networks is decided by the minimum value of the rates of links consisting of a multihop path. Thus, the maximization problem of multihop end-to-end throughput has the following max–min objective problem:

$$\max_{\mathbf{P}} R_{ee} = \max_{\mathbf{P}} \min \{R_{12}, R_{23}, \dots, R_{n(n+1)}\}, \quad (10)$$

where vector  $\mathbf{P}$  indicates the transmit power set of  $n$  transmitting nodes and  $R_{ij}$  denotes the rate of each link on the multihop path. Therefore, in this problem, the optimal solution is to make the transmission rate of each link equal. To this end, a synchronization algorithm of each link rate based on the Cucker–Smale flocking model [16] was proposed as

$$\begin{aligned} & R_{ij}(t+1) - R_{ij}(t) \\ &= \frac{1}{n_{\forall k,l=k+1}} \Psi(|x_k - x_l|) (R_{kl}(t) - R_{ij}(t)). \end{aligned} \quad (11)$$

To support the link rate determined by (11), each transmitting node performs the transmit power control in each time slot.

In both the maximization problem of multihop end-to-end throughput in [41] and the minimization problem of

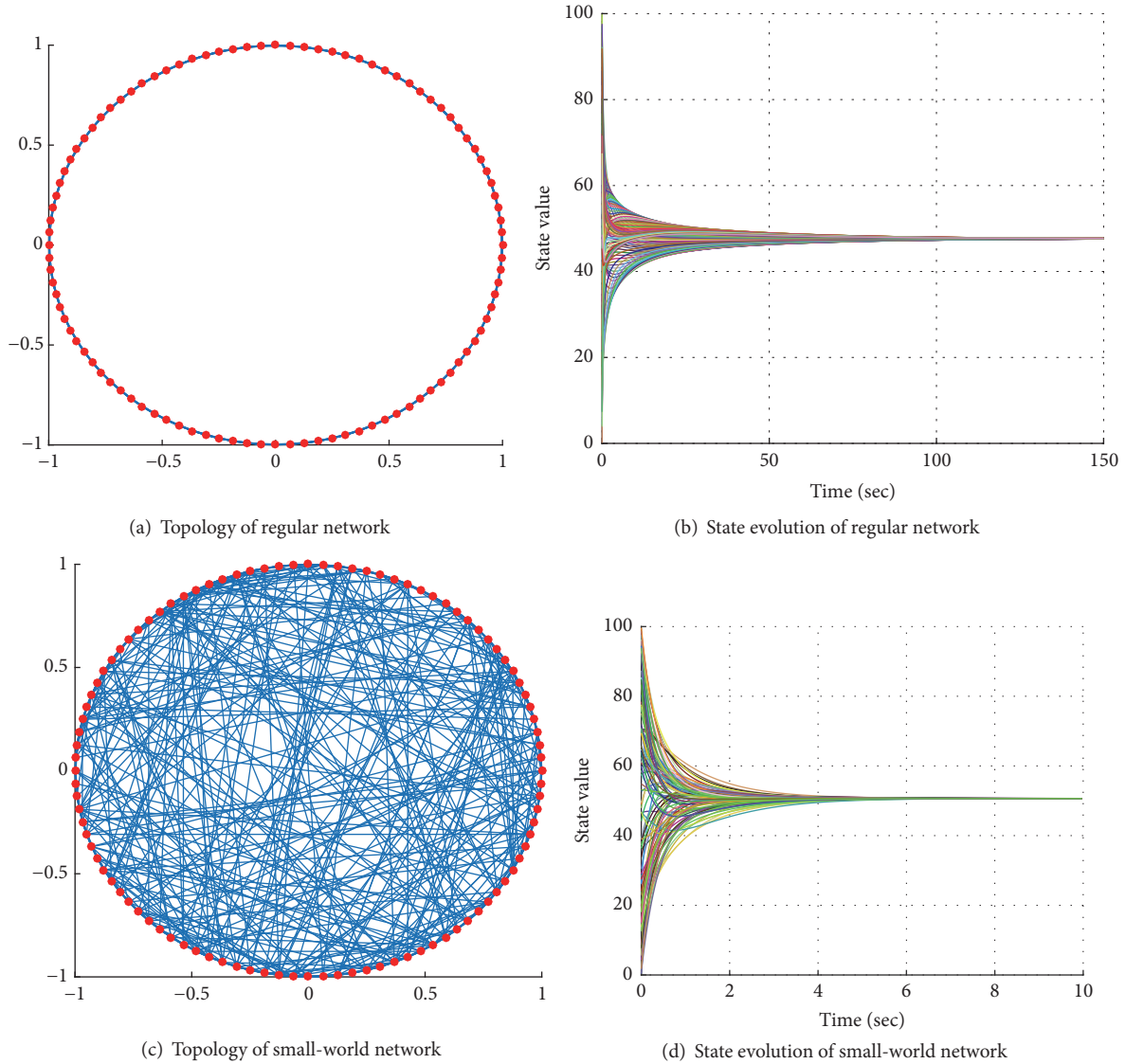


FIGURE 10: Comparison of convergence times between a regular network and a small-world network.

outage in [43], the synchronization phenomenon of each link rate and the transmission power variation in each node over time are depicted in Figure 11. The suggested algorithm basically follows the Cucker–Smale flocking model. Therefore, the convergence is guaranteed. The results also show that when the link transmission rates converge, the node with the best initial link rate uses the lowest transmit power, while the node with the worst initial link rate uses the maximum transmit power. Because the higher transmit power induces more interference, the optimal strategy is to enable the transmitting node with the best link to use the lowest transmit power, thereby causing the least interference and increasing the transmission rates of the other links.

The above reasoning is supported by the notion that the strongest bird positions itself at the head of the flock, where the air resistance is the greatest, in order to support the weaker birds that follow when migrating as a flock. The velocity vectors of birds have a solidarity property owing

to the air resistance. Additionally, they have a max–min objective because their flying speed is limited by the speed of the slowest bird. Accordingly, we can understand that the synchronization of velocity in the flights of flocking birds is the optimal solution for achieving their objective [42, 43].

**3.7. Comparisons between Nature-Inspired and Conventional Synchronizations.** In this section, the nature-inspired and conventional approaches are compared in terms of time synchronization. The characteristics, advantages, and disadvantages of nature-inspired time synchronization methods are evaluated compared with the conventional centralized and distributed time synchronization methods.

Centralized time synchronization methods use global time orientations, such as global positioning system (GPS) time [44] and network time protocol (NTP) [45]. All nodes listen for the reference time provided from the central clock and adjust their own time in order to synchronize. This



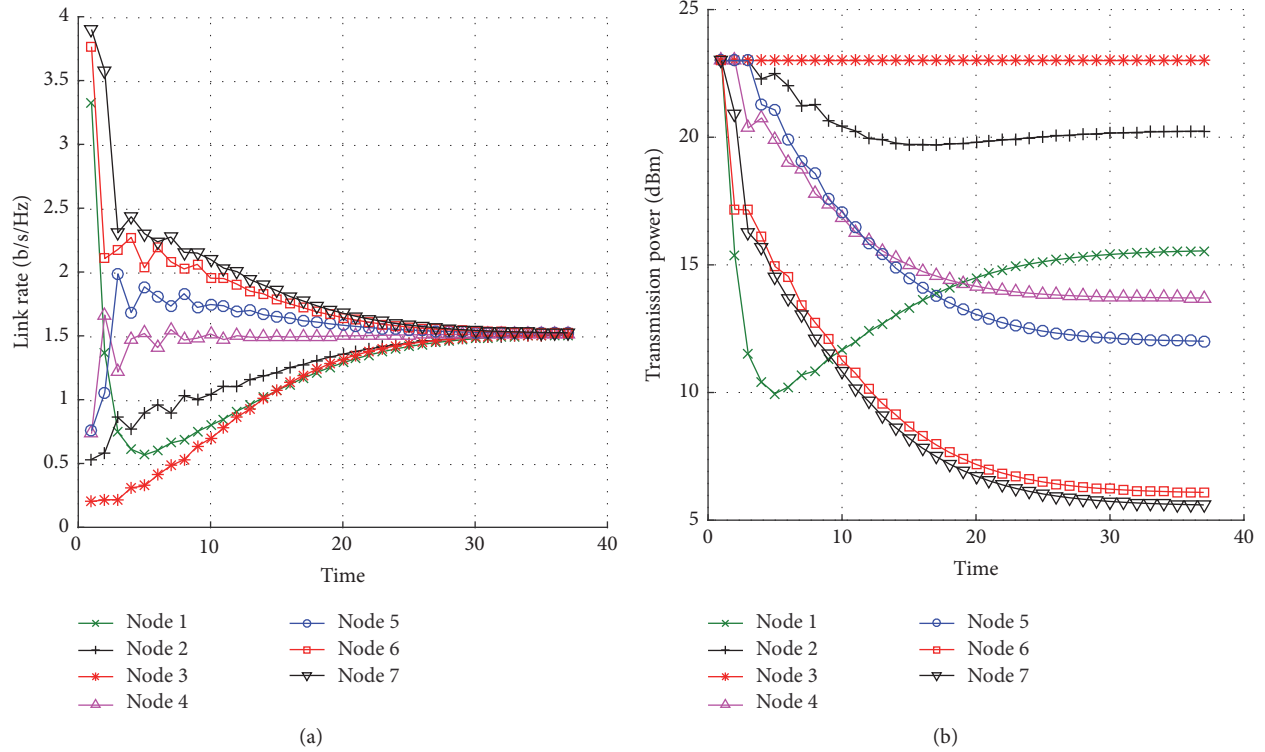


FIGURE 11: (a) Link rate and (b) transmission power over time for the maximization problem of end-to-end throughput in wireless multihop networks.

process enables accurate synchronization with high reliability; however, it involves a high cost in terms of hardware implementation and energy consumption. Moreover, the synchronization performance is related to the respective distance to the central clock, which limits the network scalability.

The conventional reference broadcast synchronization (RBS) method [46] is a typical distributed time synchronization method that is not nature-inspired. Without using a central global clock, RBS engenders mutual synchronization among the network nodes. Thus, precise synchronization is possible without a delay from the central clock. However, the time information of each node should be shared with all the other nodes so that an immense amount of data can be exchanged between the nodes. As the number of nodes increases, multihop RBS is required and the data exchange overhead likewise increases, resulting in the decrease of performance.

Unlike these methods, nature-inspired time synchronization methods [19–24] can achieve synchronization in a distributed and efficient manner regardless of the number of nodes in a large-scale network. Table 1 summarizes the characteristics, advantages, and disadvantages of the conventional synchronization methods and those of the nature-inspired synchronization methods.

#### 4. Research Challenges

In this section, we examine further research challenges that should be considered in utilizing natural synchronization phenomena for the future communication and networking

systems. The considered research challenges are classified as in Figure 12.

**4.1. Challenges for Convergence Performance.** The synchronization algorithm conducts iterations in a dynamic environment. Thus, the performance related to convergence is very important. According to the number of nodes, initial value, network topology, and coupling strength, we must theoretically analyze the existence, value, and rate of convergence, or we must empirically recognize the tendency of convergence. To increase the convergence rate, we must design a network topology that can stimulate rapid synchronization based on the results of many experiments in a variety of network environments. These include random, small-world, and large-scale networks, as well as locally connected regular networks.

According to studies conducted thus far, a network wherein all nodes are fully connected is known to guarantee the convergence regardless of the number of nodes and the initial value [9]. Therefore, in order to guarantee the convergence, it is necessary to arbitrarily make the network fully connected. Alternatively, a hierarchical method can be considered in which a network is clustered as a fully connected network at a local level. Then, synchronization is engaged at the upper level with the locally converged information.

The convergence rate has a tradeoff relationship with the accuracy of the convergence value. If the convergence value is set at a discrete level instead of a real value from a practical perspective, the convergence rate can be controlled according to the number of levels. For example, the convergence value in the synchronization of transmission rate is mapped to

TABLE 1: Comparison of conventional and nature-inspired synchronization methods.

Item	Conventional synchronization method	Nature-inspired synchronization method
Reliability	High reliability and accuracy; short convergence time	Guaranteed reliability; accurate synchronization in large-scale networks
Scalability	Limited by the numbers of nodes and hops	Scalability maintained regardless of the number of nodes and changes
Robustness	Vulnerable to network topology changes, such as cluster head failures	Robust to poor communication environments and network topology changes
Cost	Very high; message exchange of upper layers is required	Low amount of data exchange; can be simply operated through pulse exchange at the physical layer without upper layer protocols
Complexity	Very high hardware (HW)/software (SW) complexity	Low HW/SW complexity (not necessary to save time information of other nodes in the memory)
Influential factors for performance	Synchronization accuracy depends on the distance to root nodes	Affected by node density, coupling strength, delay, noise, path loss, and modulation method
Advantage	High reliability; effective with an appropriate number of nodes; not dependent on the global clock in the case of RBS	High scalability and robustness; low cost and complexity; effective with a large number of nodes
Disadvantage	Poor performance of outer nodes in the topology in cases of centralized synchronization; a large amount of exchanged data is required in distributed synchronization	Synchronization is impossible when the transmission and reception delays take a long time; synchronization speed decreases with weak coupling strength and a small number of nodes

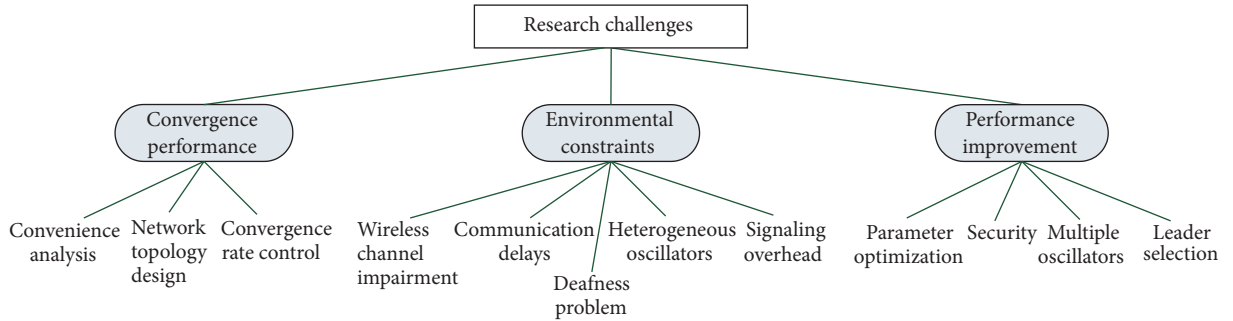


FIGURE 12: Classification of research challenges.

the modulation and coding scheme (MCS) level in reality. Thus, the convergence rate can be controlled according to the number of MCS levels. On the contrary, the number of MCS levels can be determined according to the required convergence rate.

**4.2. Challenges of Environmental Constraints.** Path loss, noise, delay, and packet loss occurring in real environments influence synchronization. The weakening of coupling strength due to path loss and packet loss reduces the convergence rate, while delay and noise are major factors that interfere with synchronization [10]. When applying a synchronization algorithm to a real environment, it is therefore necessary to design the algorithm with consideration of all these interfering factors.

A receiving node may be unable to immediately respond on account of a communication delay. Consequently, the synchronization may not be completed. Specifically, the communication delay can be divided into a transmission delay,

decoding delay, and propagation delay. It is necessary to redesign a synchronization algorithm considering the statistical ranges of these delays. Moreover, a wireless communication node cannot receive data while simultaneously transmitting. Thus, it cannot recognize the firing of other nodes while it is firing. Therefore, the firing period should be set to cross one another to avoid such “deafness” problems in wireless communication environments.

To date, research has been conducted only on homogeneous oscillators that follow the same rule. However, in reality, heterogeneous oscillators may exist that adhere to different synchronization rules in the same situation on account of the network heterogeneity. Therefore, research should be conducted on synchronization algorithms when the heterogeneous oscillators are intended to cooperatively achieve a certain collective purpose.

For the real operation of the synchronization algorithms described in Section 2, periodic data exchange among neighboring nodes is required. Consequently, signaling overhead

occurs. Considering this kind of overhead cost, the point at which it is more effective to apply nature-inspired synchronization methods over conventional synchronization methods should be determined. Depending on the situation, a hybrid synchronization algorithm can be considered that adaptively applies both types of methods.

**4.3. Challenges to Performance Improvement.** Depending on network topology changes in the wireless environment, synchronization performance can be improved by optimizing the operation parameters of the synchronization algorithm, such as the coupling strength, weight, and step size. The mapping relations between operation parameters and environment variables should be configured through various experiments.

In terms of network security, a method is required to enhance the tolerance of systems against malicious nodes that intentionally interfere with the synchronization. For example, a synchronization algorithm can be designed to ignore certain information with the maximum deviation from the average of the information received from neighboring nodes. A synchronization algorithm that considers unnecessary information should be devised and it should be verified whether it is possible to achieve a desired convergence performance.

When there are two or more oscillators in a single node, a method that can utilize multiple oscillators should be considered. In [28], proportional fair scheduling was conducted in a distributed manner using two oscillators. Therefore, complex synchronization phenomena with several oscillators need to be studied for developing a new application that utilizes them.

It is possible to converge the values of all nodes into a desired value by selecting a leader and setting only its value. For example, in aircraft engineering, by controlling the single leader in flock-like flight formation, this method can enable all the rest of the aircrafts to automatically follow that specific aircraft. In addition, the transmission rate of all nodes can be equalized by controlling only the rate of the leader. However, this kind of approach that controls the convergence value only through a leader may be unable to guarantee the convergence depending on the environment. Thus, a theoretical study to address that issue should be conducted.

## 5. Discussion and Prospect

The reason why we focus on synchronization phenomena can be explained as follows. First of all, the natural environment in which synchronization occurs is similar to the environments in which many engineering problems occur. Synchronization occurs among a number of individuals. It also occurs in a completely distributed system in which each individual decides its own action with consideration of only the surrounding environment or situation. In addition, individuals perfectly synchronize in dynamic environments where obstacles or adversaries may appear, and nodes enter or exit. These kinds of environments in which synchronization occurs inspire many engineering problems in large-scale networking, distributed, and dynamic environments.

Secondly, the purpose of synchronization in nature can be understood as the same as engineering objectives. Insects,

birds, fish, and other animals that form groups are known to synchronize for migration, mating, protection from predators, and efficient searching for food [3]. These creatures exhibit solidarity and thus engage in synchronization for collective goals. These goals are typically for survival purposes. Such animal populations use synchronization as an optimal strategy for surviving while evolving through time. Likewise, a number of nodes in systems, such as sensor networks, multivehicle systems, and multihop network, must achieve a collective objective with solidarity. Their objectives are to rapidly obtain useful information, quickly move together over long distances, and maximize the end-to-end throughput, respectively. These goals are not for maximizing the performance of individual nodes; rather, they serve to maximize the performance of the entire system. Therefore, the purposes of engineering systems correspond to those demonstrated by collectives of creatures. It is thus reasonable to employ synchronization as the optimal solution to problems that exist in engineering systems.

Lastly, synchronization in nature is more efficient than any centralized or distributed system available. For example, thousands of fireflies engender the synchronization phenomenon in the fastest and most efficient way with limited resources. They use a very simple method and achieve synchronization within a short period of time. Therefore, mimicking these natural approaches can enable engineering systems to achieve performances at a similar level of low complexity and cost in situations with limited resources.

The point at which nature-inspired technologies will become more effective than existing approaches will be when the number of nodes comprising a system increases beyond a certain level. Owing to the advancement of communication network technologies, the number of nodes connected to networks is drastically increasing each year. The increase of nodes accelerates the shortage of resources and incurs the high cost of centralized management. Conventional centralized technologies developed in consideration of a limited number of nodes can neither operate with a large number of nodes nor guarantee the required performance. Thus, naturally, a shift to distributed systems is required. However, distributed system technologies developed under the centralized paradigm still bear the limitation concerning the increase in the number of nodes. Furthermore, they are vulnerable to environmental changes due to the incompleteness of the distribution [46]. Therefore, a paradigm shift to new distributed systems is necessary and it is imperative to focus on the distributed algorithms of collective organisms in nature as an effective methodology to be applied.

## 6. Conclusion

In this article, we identified the principles of various natural synchronization phenomena and studied how they can be applied in practical communication and networking systems. In terms of time synchronization methods in networks, the strengths and weaknesses of conventional approaches and nature-inspired approaches were compared. Further research challenges that are necessary for realizing nature-inspired synchronization technologies were presented. The

rapid increase of the number of networking nodes will accelerate the occurrence of complex and distributed network environments, which are getting similar to the environment of nature. Therefore, it would be a great help for us to mimic and apply the fundamental principles of biocommunities in which each entity has shown collective behaviors to sustain their lives in the midst of complex and chaotic environments. In the future, various studies considering practical issues are required to effectively apply these principles to mobile communication and networking systems.

## Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2016R1C1B1016261), by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2015R1D1A1A01060207), and by the Human Resources Development of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korea government Ministry of Trade, Industry and Energy (no. 20154030200860).

## References

- [1] A. Pikovsky, M. Rosenblum, and J. Kurths, *Synchronization: A Universal Concept in Nonlinear Sciences*, Cambridge University Press, 2001.
- [2] H.-H. Choi and J.-R. Lee, "Communication and networking technologies based on bio-inspired algorithms," *KICS Information & Communications Magazine*, vol. 29, no. 4, pp. 62–71, 2012 (Korean).
- [3] S. H. Strogatz, *Sync: How Order Emerges from Chaos in the Universe, Nature, and Daily Life*, Hyperion, New York, NY, USA, 2004.
- [4] F. Dressler and O. B. Akan, "A survey on bio-inspired networking," *Computer Networks*, vol. 54, no. 6, pp. 881–900, 2010.
- [5] Z. Zhang, K. Long, J. Wang, and F. Dressler, "On swarm intelligence inspired self-organized networking: its bionic mechanisms, designing principles and optimization approaches," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 513–537, 2014.
- [6] C. Zheng and D. C. Sicker, "A survey on biologically inspired algorithms for computer networking," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1160–1191, 2013.
- [7] C. W. Reynolds, "Flocks, herds, and schools: a distributed behavioral model," *ACM Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987.
- [8] C. S. Peskin, "Mathematical aspects of heart physiology," Tech. Rep., Courant Institute of Mathematical Sciences, New York University, 1975.
- [9] R. E. Mirollo and S. H. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM Journal on Applied Mathematics*, vol. 50, no. 6, pp. 1645–1662, 1990.
- [10] U. Ernst, K. Pawelzik, and T. Geisel, "Synchronization induced by temporal delays in pulse-coupled oscillators," *Physical Review Letters*, vol. 74, no. 9, pp. 1570–1573, 1995.
- [11] Y. Kuramoto and H. Araki, Eds., *Lecture Notes in Physics, International Symposium on Mathematical Problems in Theoretical Physics*, Springer, New York, NY, USA, 1975.
- [12] S. H. Strogatz, "From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators," *Physica D. Nonlinear Phenomena*, vol. 143, no. 1-4, pp. 1–20, 2000.
- [13] J. A. Acebrón, L. L. Bonilla, C. J. Perez-Vicente, F. Ritort, and R. Spigler, "The Kuramoto model: a simple paradigm for synchronization phenomena," *Reviews of Modern Physics*, vol. 77, no. 1, pp. 137–185, 2005.
- [14] R. Sepulchre, D. Paley, and N. Leonard, "Collective motion and oscillator synchronization," in *Proceedings of the Block Island Workshop Cooperative Control*, Block Island, RI, USA, June 2003.
- [15] A. Papachristodoulou and A. Jadbabaie, "Synchronization in oscillator networks: switching topologies and non-homogeneous delays," in *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference (CDC-ECC '05)*, pp. 5692–5697, December 2005.
- [16] F. Cucker and S. Smale, "Emergent behavior in flocks," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 852–862, 2007.
- [17] S.-Y. Ha and J.-G. Liu, "A simple proof of the Cucker-Smale flocking dynamics and mean-field limit," *Communications in Mathematical Sciences*, vol. 7, no. 2, pp. 297–325, 2009.
- [18] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
- [19] A. Tyrrell, G. Auer, and C. Bettstetter, "Fireflies as role models for synchronization in ad hoc networks," in *Proceedings of the 1st Bio-Inspired Models of Network, Information and Computing Systems*, pp. 1–7, Madonna di Campiglio, Italy, December 2006.
- [20] A. Tyrrell and G. Auer, "Imposing a reference timing onto firefly synchronization in wireless networks," in *Proceedings of the IEEE 65th Vehicular Technology Conference (VTC '07)-Spring*, pp. 222–226, April 2007.
- [21] Y.-W. Hong and A. Scaglione, "A scalable synchronization protocol for large scale sensor networks and its applications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 5, pp. 1085–1099, 2005.
- [22] G. A. Puerta, E. A. Aguirre, and M. A. Alzate, "Effect of topology and mobility in bio-inspired synchronization of mobile ad hoc networks," in *Proceedings of the IEEE Latin-American Conference on Communications (LATINCOM '10)*, pp. 1–6, September 2010.
- [23] H. Yoo, M. Lee, and Y. Cho, "A distributed frequency synchronization technique for OFDMA-based mesh networks using bio-inspired algorithm," *The Journal of Korea Information and Communications Society*, vol. 37B, no. 11, pp. 1022–1032, 2012.
- [24] G. Werner-Allen, G. Tewari, A. Patel, M. Welsh, and R. Nagpal, "Firefly-inspired sensor network synchronicity with realistic radio effects," in *Proceedings of the 3rd ACM International Conference on Embedded Networked Sensor Systems (SenSys '05)*, pp. 142–153, ACM, San Diego, Calif, USA, November 2005.
- [25] J. Degeys, I. Rose, A. Patel, and R. Nagpal, "DESYNC: self-organizing desynchronization and TDMA on wireless sensor networks," in *Proceedings of the 6th International Symposium on*



- Information Processing in Sensor Networks (IPSN '07)*, pp. 11–20, April 2007.
- [26] A. Patel, J. Degesys, and R. Nagpal, “Desynchronization: the theory of self-organizing algorithms for round-robin scheduling,” in *Proceedings of the 1st International Conference on Self-Adaptive and Self-Organizing Systems (SASO '07)*, pp. 87–96, Boston, Mass, USA, July 2007.
  - [27] J. Degesys and R. Nagpal, “Towards desynchronization of multi-hop topologies,” in *Proceedings of the 2nd IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO '08)*, pp. 129–138, Venice, Italy, October 2008.
  - [28] R. Pagliari, Y.-W. P. Hong, and A. Scaglione, “Bio-inspired algorithms for decentralized round-robin and proportional fair scheduling,” *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 4, pp. 564–575, 2010.
  - [29] R. Olfati-Saber and J. S. Shamma, “Consensus filters for sensor networks and distributed sensor fusion,” in *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference (CDC-ECC '05)*, pp. 6698–6703, December 2005.
  - [30] D. P. Spanos, R. Olfati-Saber, and R. M. Murray, “Approximate distributed kalman filtering in sensor networks with quantifiable performance,” in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 133–139, Los Angeles, Calif, USA, April 2005.
  - [31] L. Xiao, S. Boyd, and S. Lall, “A Scheme for robust distributed sensor fusion based on average consensus,” in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 63–70, April 2005.
  - [32] D. Spanos, R. Olfati-Saber, and R. M. Murray, “Dynamic consensus on mobile networks,” in *Proceedings of the 16th International Federation of Automatic Control World Congress (IFAC '05)*, Prague, Czech Republic, 2005.
  - [33] L. Xiao and S. Boyd, “Fast linear iterations for distributed averaging,” *Systems and Control Letters*, vol. 53, no. 1, pp. 65–78, 2004.
  - [34] D. J. Watts and S. H. Strogatz, “Collective dynamics of ‘small-world’ networks,” *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.
  - [35] R. Olfati-Saber, “Ultrafast consensus in small-world networks,” in *Proceedings of the American Control Conference (ACC '05)*, pp. 2371–2378, June 2005.
  - [36] J. A. Fax and R. M. Murray, “Information flow and cooperative control of vehicle formations,” *Institute of Electrical and Electronics Engineers. Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004.
  - [37] J. Lin, A. S. Morse, and B. D. O. Anderson, “The multi-agent rendezvous problem,” in *Proceedings of the 42nd IEEE Conference on Decision and Control*, pp. 1508–1513, Maui, Hawaii, USA, December 2003.
  - [38] J. Cortes, S. Martinez, and F. Bullo, “Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions,” *Institute of Electrical and Electronics Engineers. Transactions on Automatic Control*, vol. 51, no. 8, pp. 1289–1298, 2006.
  - [39] R. Olfati-Saber, “Flocking for multi-agent dynamic systems: algorithms and theory,” *Institute of Electrical and Electronics Engineers. Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.
  - [40] X.-S. Yang, Z. Cui, R. Xiao, A. H. Gandomi, and M. Karamanoglu, *Swarm Intelligence and Bio-Inspired Computation, Theory and Applications*, Elsevier, 1st edition, 2013.
  - [41] H.-H. Choi and J.-R. Lee, “Distributed transmit power control for maximizing end-to-end throughput in wireless multi-hop networks,” *Wireless Personal Communications*, vol. 74, no. 3, pp. 1033–1044, 2014.
  - [42] H.-H. Choi and J.-R. Lee, “A bio-inspired transmit power control algorithm for linear multi-hop wireless networks,” in *Proceedings of the IARIA International Conference on Networks (ICN '14)*, February 2014.
  - [43] H. Choi and J. Lee, “A biologically-inspired power control algorithm for energy-efficient cellular networks,” *Energies*, vol. 9, no. 3, article no. 161, 2016.
  - [44] J. Mannermaa, K. Kalliomaki, T. Mansten, and S. Turunen, “Timing performance of various GPS receivers,” in *Proceedings of the IEEE International Frequency Control Symposium, 1999. Proceedings of the 1999 Joint Meeting of the European*, pp. 287–290, April 1999.
  - [45] D. L. Mills, “Internet time synchronization: the network time protocol,” *IEEE Transactions on Communications*, vol. 39, no. 10, pp. 1482–1493, 1991.
  - [46] J. Elson, L. Girod, and D. Estrin, “Fine-grained network time synchronization using reference broadcasts,” *SIGOPS Operating Systems Review*, vol. 36, pp. 147–163, 2002.



