Heartbeat Synchronization in Large Multi-Agent Systems Using One-Way Communication

Federico Bergenti
Università di Parma
Parma, Italy
federico.bergenti@unipr.it

Simone Dallospedale
Università di Ferrara
Ferrara, Italy
simone.dallospedale@unife.it

Stefania Monica
Università di Modena e Reggio Emilia
Reggio Emilia, Italy
stefania.monica@unimore.it

ABSTRACT

A novel decentralized protocol is introduced to achieve heartbeat synchronization in large multi-agent systems, where agents produce brief, periodic flashes that align nearly simultaneously. The protocol operates entirely through one-way messages without requiring agents to share a common frequency or phase. These messages do not require responses and never block agents, eliminating the risk of deadlocks. The protocol proceeds as agents repeatedly contact randomly chosen peers, with a random delay before each transmission. Starting from random initial states, agents update their states upon receiving messages and emit light while a prescribed condition is satisfied. The reported study establishes a closed-form expression for the resulting flash period, showing dependence only on a protocol parameter and on the average number of messages sent per unit time per agent. Experiments conducted using a custom-built simulator confirm that the derived expression predicts the observed flash period accurately and precisely. These experiments further show that the flashes remain sufficiently brief and stable to support effective heartbeat synchronization.

KEYWORDS

Heartbeat synchronization; Emergent behavior; Collective intelligence; Self-organization

ACM Reference Format:
Federico Bergenti, Simone Dallospedale, and Stefania Monica. 2026. Heartbeat Synchronization in Large Multi-Agent Systems Using One-Way Communication. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 9 pages. https://doi.org/10.65109/VFUP8012

1 INTRODUCTION AND MOTIVATION

The *heartbeat synchronization problem* [3] concerns coordinating a large *Multi-Agent System* (*MAS*) so that all agents periodically emit brief flashes of light in near synchrony. In such systems, the scale makes it impractical to designate a leader to regulate flashing. No individual agent governs the process; instead, each agent operates with only partial knowledge of the MAS. Synchronization is therefore expected to arise from decentralized interactions. This makes heartbeat synchronization a central challenge in large-scale distributed coordination, with relevant applications in distributed artificial intelligence [11, 25, 30] and distributed computing [1, 16, 26]. For example, relevant scenarios include cycle- or round-based distributed protocols, such as gossip protocols [9] for distributed averaging [7], where agents must collectively agree on the start of a new cycle. In protocols that involve periodic restarts, local views of cycle boundaries must remain aligned [15, 16]. The novel protocol presented in Section 3 is proposed to effectively solve the heartbeat synchronization problem in these situations.

The heartbeat synchronization problem does not impose strict constraints on the interval between flashes, provided it remains bounded and agents eventually produce near-simultaneous, repetitive flashes [3]. However, protocols that generate a predictable and sufficiently regular flash period are particularly desirable. Small variations in flash emission timing are acceptable as long as they remain within a prescribed fraction of the nominal flash period. For the target applications [3], deviations within 10% of the nominal flash period are generally tolerable. The proposed protocol has a closed-form expression of the flash period, as shown in Section 4, and it meets this requirement, as shown in Section 5.

Flash duration is another key property of the desired collective behavior. Effective heartbeat synchronization requires that each flash remains short relative to the nominal flash period. As with the flash period, protocols that ensure short and consistent flash durations are preferable, especially when analytical predictions can be made with precision and accuracy. Target applications [3] expect flash durations within 10% of the nominal flash period. The proposed protocol meets this requirement, as shown in Section 5.

The proposed protocol removes the common assumption that agents behave as coupled oscillators [3] that adjust their frequencies and phases to achieve synchronization. Instead, agents engage in pairwise interactions at random intervals, without explicitly aligning their timing. Only the average number of messages sent per unit time per agent is prescribed to obtain a nominal flash period.

In contrast to similar protocols [6], the proposed approach relies exclusively on one-way messages. An interaction occurs when an agent sends a message containing its current state to another agent and immediately updates its own state according to the protocol's rule presented in Section 3. The receiving agent also updates its state upon receipt of the message to complete the interaction. Both agents emit light whenever their states satisfy a prescribed condition. No reply is expected to significantly simplify the communication compared to reply-based protocols. The use of one-way messages provides three main advantages:

1. Fewer messages are exchanged, since replies are absent;
2. Agents can initiate new interactions without waiting for replies, thereby reducing idle periods; and
3. Deadlocks are avoided by design, as two agents cannot become mutually blocked while waiting for replies.

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/VFUP8012

Note that reply-based protocols require timeouts to resolve such deadlocks [21], discarding all interactions that are not completed within the allotted time. The use of one-way messages eliminates deadlocks by design, ensuring that all interactions contribute to the progress of synchronization. Finally, note that the adoption of one-way messages more accurately respects the common assumption in ordinary decentralized protocols, such as gossip protocols [9], that interactions are instantaneous.

Besides introducing the proposed protocol as a viable alternative to other protocols, the main contribution of this paper is an analytical characterization of the dynamics induced by the protocol. Section 4 develops analytical results showing that the nominal period of emergent flashes can be expressed in closed form. These results are grounded in a theoretical framework for studying the average behavior of the collective and emergent properties of large MAS [6, 7, 21, 22]. In particular, this framework allows showing that, on average, the proposed protocol constrains the agent states to evolve sinusoidally over time. These analytical results show that the near-synchronous flashes occur at a frequency determined solely by two factors: a protocol parameter and the average number of messages sent per unit time per agent. Thus, the flash period can be controlled by selecting appropriate values for these quantities.

The paper is organized as follows. Section 2 reviews related approaches and situates the proposed protocol within existing research on heartbeat synchronization. Section 3 introduces the proposed protocol and formalizes the notation. Section 4 presents the analytical results characterizing the emergent flash period. Section 5 describes experiments that confirm the theoretical predictions using a custom-built simulator. Section 6 concludes with a summary of findings and directions for future research.

2 RELATED WORK

A well-known approach to synchronization in large MAS is the Kuramoto model [17]. In its most common form, each agent is represented as an oscillator with a natural frequency. Synchronization arises by coupling these oscillators through their phases. The study of this model is typically carried out in the limit of infinitely many oscillators to prove that synchronization occurs for specific values of the coupling constants. However, this model does not specify the phase-coupling mechanism and, in particular, does not address communication via messages. Once message exchange is introduced as the basis for interaction, heartbeat synchronization becomes appropriate because the Kuramoto model no longer directly applies.

Decentralized solutions to the heartbeat synchronization problem rely on protocols in which agents interact iteratively to align their flashes, sometimes using the flashes themselves as signals. Among these, firefly-inspired protocols [3, 27] are especially well known. In certain firefly species, large swarms of males synchronize their flashes at dusk, creating the striking effect of collective unison. This phenomenon emerges despite each insect observing only its local neighborhood to adapt its flashing behavior.

Numerous mathematical models have been developed to explain synchronized firefly flashing [18] and design protocols for decentralized synchronization in engineering and computing contexts [3, 28]. Firefly protocols model agents as pulse-coupled oscillators [27], each with a frequency and a phase that are adjusted in response to flashes from neighboring agents. Synchronization emerges as agents gradually converge toward a shared frequency and phase. These protocols often share similarities with gossip-based mechanisms [9] in the way information propagates through the system to reach consensus [3, 14].

The general framework of pulse-coupled oscillators [3] assumes that each agent maintains a phase and a period. The phase increases linearly in a sawtooth pattern until it reaches a maximum, at which point the agent emits a flash, transmits it to selected peers, and resets its phase. Initially, periods may differ among agents. Upon receiving a flash, an agent may adjust its phase and, in some cases, its period. Adjustments can involve advancing or delaying the phase depending on the agent's current state, potentially modifying the timing of subsequent flashes. Different protocol variants prescribe distinct adjustment rules and related details.

The simplest firefly protocol [3] requires that upon receiving a flash, an agent immediately sets its phase to the maximum, emits a flash, and resets, leading to pairwise synchronization between sender and receiver. This approach assumes a shared period that remains unaltered. An alternative protocol [18] adds a third state variable, which is defined by a nonlinear, concave-down function. This additional variable modulates the sensitivity of phase adjustments depending on the current phase: flashes received early in the cycle have little impact, whereas flashes received later exert stronger influence. Like the simplest firefly protocol, this model also presumes a common period across agents. A third protocol [10] relaxes this assumption by allowing agents to adapt their periods. Each agent is assigned a natural period that may vary across the MAS. The phase is used only to classify flashes as early or late: late flashes increase the period, delaying the next emission, while early flashes shorten it, anticipating the next emission.

The proposed protocol described in Section 3 departs from the pulse-coupled oscillator framework altogether. Agents do not adjust frequency or phase to achieve synchronization. Instead, each agent interacts with randomly selected peers at random intervals. The state changes triggered by these interactions determine when flashes occur. Light is emitted whenever an agent's state satisfies a predefined condition. Synchronization thus arises as an emergent property of the MAS as a whole. Agents operate independently, as only a tunable protocol parameter and the average number of messages sent per unit time per agent are prescribed to achieve a desired collective flash period, as shown in Section 4. The experiments described in Section 5 confirm that the flash period can be predicted with accuracy and precision using the closed-form expression presented in Section 4.

A distinctive feature of the proposed protocol is the use of one-way communication. When an agent transmits its current state to another, it immediately updates its own state according to the protocol update rule, while the receiving agent updates its state upon processing the message, also using the protocol update rule. Since replies are not required, communication overhead is reduced and the interaction model is simplified. This design choice yields several benefits. The main benefit is that the risk of deadlocks caused by mutual waiting for replies is completely eliminated. Protocols that rely on replies, such as the protocol proposed in [6], must incorporate timeout mechanisms to remedy potential deadlocks, often discarding incomplete interactions.

3 THE PROPOSED PROTOCOL

Consider a MAS $\mathcal{M}$ composed of $n \in \mathbb{N}_+$ agents under the following assumptions regarding agents and communication:

1. The number of agents does not change over time;
2. Each agent is permanently identified by a unique natural number between 1 and n;
3. Agents communicate by sending one-way messages whose time of flight is negligible; and
4. Every agent can send messages to any other agent.

Agent i, with $1 \leq i \leq n$, is the agent whose unique identifier is i. At time $t \in \mathbb{R}_{\geq 0}$, agent i has a state represented by a pair $(x_i(t), y_i(t)) \in \mathbb{R}^2$, where $x_i(t)$ is called *x state component* and $y_i(t)$ is called *y state component*. Furthermore, agent i has a third value, called *heartbeat value*, computed from its state as

$$z_i(t) = \frac{y_i(t)}{\sqrt{x_i^2(t) + y_i^2(t)}}. \tag{1}$$

Agent i continuously emits light when the condition

$$z_i(t) > 1 - \alpha, \tag{2}$$

holds, where $\alpha \in (0, 1)$ is a protocol parameter shared among all agents called *light-control parameter*. Normally, α is chosen very close to zero because the protocol is meant to make agents emit very short flashes. In the experiments discussed in Section 5, the value $\alpha = 10^{-2}$ is used.

The application of the proposed protocol starts at time $t = 0$ when each agent initializes its two state components with one positive random number, possibly different for each agent. Let $\overline{x}_0 \in \mathbb{R}_+$ be the mean of the x state components across the agents at $t = 0$. Let $\overline{y}_0$ be the mean of the y state components across the agents at $t = 0$. Note that $\overline{x}_0 = \overline{y}_0$ because each agent initializes its state using a single positive random number, which also implies that for all $1 \leq i \leq n$

$$z_i(0) = \overline{z}_0 = \frac{1}{\sqrt{2}}. \tag{3}$$

Let $\overline{x}(t)$ denote the mean of the x state components across the agents at time $t \in \mathbb{R}_{\geq 0}$, which implies $\overline{x}(0) = \overline{x}_0$. Similarly, let $\overline{y}(t)$ denote the mean of the y state components across the agents at time t, which implies $\overline{y}(0) = \overline{y}_0$. Analogously, let $\overline{z}(t)$ denote the mean of the heartbeat values across the agents at time t, which implies $\overline{z}(0) = \overline{z}_0$. Finally, let $\sigma_x^2(t)$, $\sigma_y^2(t)$, and $\sigma_z^2(t)$ denote the variances across the agents of the x state components, y state components, and heartbeat values at time t, which implies $\sigma_x^2(0) = \sigma_y^2(0)$ because of how agent states are initialized.

Given two arbitrary agents, the sender identified by $s \in \mathbb{N}_+$, with $1 \leq s \leq n$, and the receiver identified by $r \in \mathbb{N}_+$, with $1 \leq r \leq n$ and $r \neq s$, the protocol assumes that when the sender interacts with the receiver by sending a one-way message at time $t \in \mathbb{R}_{\geq 0}$, their respective states, $(x_s(t), y_s(t)) \in \mathbb{R}^2$ and $(x_r(t), y_r(t)) \in \mathbb{R}^2$, are updated using the *interaction rule*

$$\begin{cases} x_s'(t) = (1-\gamma)x_s(t) + \delta y_s(t) \\ y_s'(t) = (1-\gamma)y_s(t) - \delta x_s(t) \\ x_r'(t) = x_r(t) + \gamma x_s(t) + \delta y_r(t) \\ y_r'(t) = y_r(t) + \gamma y_s(t) - \delta x_r(t), \end{cases} \tag{4}$$

where $\gamma \in (0, 1)$ and $\delta \in (0, 1)$ are two protocol parameters shared among all agents. These two parameters can be chosen to modify the behaviors of the agents and of the MAS as a whole. As discussed in Section 4, only δ, often called *heartbeat parameter*, has a direct influence on the emergent flash period. The other parameter γ is normally chosen sufficiently small. Note that the updated states are also used to update the heartbeat values using Equation (1).

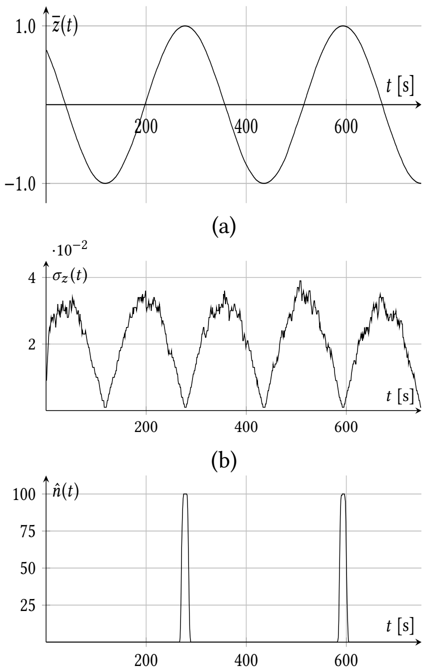
Controller 1 Pseudocode of the agent controller for the protocol

```
Require: $\alpha \in (0,1), \gamma \in (0,1), \delta \in (0,1), \hat{\nu} \in \mathbb{R}_+$
State: $x \in \mathbb{R}, y \in \mathbb{R}, z \in \mathbb{R}, \mathbf{X} \sim \mathcal{U}(0,1), \mathbf{D} \sim \mathcal{E}(\hat{\nu})$
procedure UPDATE($x', y'$)
    $x \leftarrow x'$; $y \leftarrow y'$; $z \leftarrow \frac{y}{\sqrt{x^2+y^2}}$        ▷ Update $x$, $y$, $z$
    if $z > 1 - \alpha$ then                                ▷ Is $z$ above threshold?
        LIGHT_ON                                            ▷ Turn or keep the light on
    else
        LIGHT_OFF                                           ▷ Turn or keep the light off
on CREATED
    $x_0 \leftarrow \mathbf{X}$                                         ▷ Set the initial state
    UPDATE($x_0, x_0$)                                      ▷ Update $x$, $y$, $z$
    SLEEP($\mathbf{D}$)                                           ▷ Sleep for a random interval
on RECEIVED ($x_s, y_s$)
    $x' \leftarrow x + \gamma x_s + \delta y$; $y' \leftarrow y + \gamma y_s - \delta x$
    UPDATE($x', y'$)                                        ▷ Update $x$, $y$, $z$
on AWAKENED
    $r \leftarrow$ GET_RANDOM_PEER                             ▷ Choose a random peer
    SEND($r$, ($x, y$))                                     ▷ Send a message to the peer
    $x' \leftarrow (1-\gamma)x + \delta y$; $y' \leftarrow (1-\gamma)y - \delta x$
    UPDATE($x', y'$)                                        ▷ Update $x$, $y$, $z$
    SLEEP($\mathbf{D}$)                                           ▷ Sleep for a random interval
```

The proposed protocol requires agents to engage cyclic behaviors. Each agent waits for a random interval and then interacts with another random agent in the MAS. Each agent can interact with any other agent in the MAS. The peer for an interaction is chosen by means of a uniform random number between 1 and n, which corresponds to the identifier of the chosen peer. Note that the interaction rule does not allow agents to interact with themselves.

The random interval that each agent waits before sending a message to another agent has an exponential distribution with rate $\hat{\nu} \in \mathbb{R}_+$. Therefore, each agent sends one-way messages to the other agents in the MAS at the instants of a Poisson point process with rate $\hat{\nu}$, which implies that each agent sends an average of $\hat{\nu}$ one-way messages per unit time. Note that $\hat{\nu}$, often called *sending rate*, is a protocol parameter and all agents share it. Also, note that $\hat{\nu}$ connects the occurrence of interactions with the passing of time, and therefore it is intimately related to the flash period.

Controller 1 shows the controller of the agents that perform the proposed protocol. The controller is configured using the protocol parameters α, γ, δ, and $\hat{\nu}$. The controller has a state composed of the x state component, the y state component and the heartbeat value. The controller uses the random variable $\mathbf{X}$ to initialize the agent state and the random variable $\mathbf{D}$ to sleep for random intervals. Finally, the controller uses GET_RANDOM_PEER to select peers uniformly at random for interactions.



(a)

(b)

(c)

Figure 1: For a simulation of $n = 100$ agents starting from random states in $(0, 1)$ and using the proposed protocol with $\alpha = 10^{-2}$, $\gamma = 10^{-1}$, $\delta = 10^{-2}$, and $\hat{\nu} = 1\,s^{-1}$, (a) shows the mean heartbeat value $\overline{z}(t)$, (b) shows the associated standard deviation of the heartbeat values $\sigma_z(t)$, and (c) shows the number of agents emitting light $\hat{n}(t)$.

Following the nomenclature in [7], the adopted interaction rule can be characterized as asymmetric because the sender does not rely on the receiver's state to perform its update. Consequently, no reply from the receiver is required and one-way messages are sufficient to enact the proposed protocol. Furthermore, note that the agents' states are updated using Equation (4) precisely at the time the sender sends its one-way message to the receiver because the messages are assumed to have a negligible time of flight.

Figure 1(a) shows the mean heartbeat value $\overline{z}(t)$ across all agents in a simulated MAS for the first $750\,s$ of the simulation. The simulation was performed using the simulator presented in Section 5. The simulated MAS is composed of $n = 100$ agents whose states are initialized with uniform random numbers in $(0, 1)$. The parameters used for this simulation are $\alpha = 10^{-2}$, $\gamma = 10^{-1}$, $\delta = 10^{-2}$, and the agents in the MAS send an average of one message per second because $\hat{\nu} = 1\,s^{-1}$ is used.

Figure 1(b) shows the standard deviation of the heartbeat values $\sigma_z(t)$ across the agents in this simulation. Note that $\sigma_z(t)$ has its peaks near the instants in which $\overline{z}(t)$ changes sign. These are the instants in which the heartbeat values are most widely spread across the agents. On the contrary, $\sigma_z(t)$ is close to zero near the instants in which $\overline{z}(t)$ is close to ±1. These are the instants in which all agents tend to share the same heartbeat value. In particular, when $\overline{z}(t)$ is close to one, the heartbeat values are all close to one. Figure 1(b) provides empirical support for the correctness of the proposed protocol. Each agent emits light when its heartbeat value is close to one, which tends to happen at unison when $\overline{z}(t)$ is close to one and $\sigma_z(t)$ is close to zero.

Figure 1(c) shows the number of agents emitting light $\hat{n}(t)$ during this simulation, with $1 \leq \hat{n}(t) \leq n$. Normally, $\hat{n}(t) = 0$ because for all $1 \leq i \leq n$, $z_i(t) \leq 1 - \alpha$, but $\hat{n}(t)$ quickly increases to n when $\overline{z}(t)$ approaches one and $\sigma_z(t)$ approaches zero. Then, $\hat{n}(t)$ quickly decreases to zero as $\overline{z}(t)$ decreases and $\sigma_z(t)$ increases.

Note that the peaks in Figure 1(c) can be used to measure the flash period. The experiments discussed in Section 5 use the peaks of $\hat{n}(t)$ to measure the flash periods in various situations to confirm that the flash period can be predicted precisely and accurately.

4 THE EMERGENT FLASH PERIOD

This section presents analytical results that characterize the flash period induced by the proposed protocol. The opening part of the section summarizes the fundamental elements of the theoretical framework employed to analyze the flash period. The subsequent part applies this framework to derive the analytical results, which include a closed-form expression of the flash period.

4.1 Essentials of the KTMAS Framework

When the number of agents in a MAS is large, analyzing the state dynamics of each individual agent becomes infeasible. For instance, imagine a large fleet of droids or autonomous taxis operating in urban environments. While monitoring each individual agent's position, velocity, and battery level may be prohibitively costly or even infeasible, the primary concern is that the collective system fulfills its overarching goals, such as executing a coordinated choreography in the case of droids, or ensuring uniform coverage of the urban area for taxis. In such cases, it is preferable to study the collective behavior of the MAS as a whole [13, 29], which involves examining the properties of the agent states that contribute to the collective and emergent characteristics of the MAS.

Under the mild assumption that the relevant features of the agent states can be represented as real numbers, the collective and potentially emergent properties of the MAS can be investigated using statistical methods. These methods focus primarily on aggregate quantities, which are typically insufficient to describe individual agent states in detail but are adequate for characterizing the overall collective dynamics of the MAS as a whole.

The *Kinetic Theory of Multi-Agent Systems* (*KTMAS*) [22] has been developed to provide a general framework for analyzing the average collective dynamics of large MAS in which the agents employ decentralized protocols. As its name suggests, this theory is a specific instantiation of mathematical kinetic theories [4, 5, 23]. These theories are normally limited to physical systems, but they can be generally employed to study the collective properties of interacting elements under the assumption that the collective characteristics dynamically emerge from interactions among elements and from environmental influences [2, 12, 19, 20, 24].

Mathematical kinetic theories share a common general framework [4, 5, 23] for studying the collective dynamics of interacting elements. The KTMAS framework [22] instantiates the general framework of mathematical kinetic theories to study the average behavior of large MAS in which agents influence each other's states through direct messaging [8] and decentralized protocols. The notation and terminology used in Section 3 to describe the proposed protocol are derived from the KTMAS framework to facilitate its application to the current study.

The KTMAS framework considers MAS composed of a static and large number of agents $n \in \mathbb{N}_+$, each uniquely identified by an integer between 1 and n. Each agent's state is represented by a real vector $\mathbf{q} \in Q$, with $Q \subseteq \mathbb{R}^m$ for some $m \in \mathbb{N}_+$. Q is assumed to be an open set containing all admissible agent states.

Agents interact autonomously and every agent can communicate with any other agent. Interactions are modeled as message exchanges between two agents. Accordingly, an interaction involves agent s, with $1 \leq s \leq n$, sending a message to agent r, with $1 \leq r \leq n$. Messages are allowed to call for replies, which happens when agent r sends a reply message to agent s. Time is modeled as a nonnegative real variable and interactions are supposed to occur at the instants of a Poisson point process with rate $\nu \in \mathbb{R}_+$, often called *interaction rate*. Interactions are assumed to be instantaneous and mutually independent, meaning that an agent cannot experience multiple interactions concurrently [22].

The current KTMAS framework does not incorporate environmental forces; thus, agent states evolve solely due to interactions. These interactions are characterized by interaction rules that map preinteraction states to postinteraction states. Interaction rules are specific to the MAS under study and remain unspecified within the general framework until a concrete model is built. The only constraint imposed by the KTMAS framework on these rules is that they must be total and locally invertible over Q.

Hereafter, only the case $m = 2$ is discussed because the proposed protocol assumes that agents have states in $Q = \mathbb{R}^2$. Following the framework of mathematical kinetic theories, $f : Q \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ is a function, called *density function*, such that $f(x, y, t)\,\mathrm{d}x\mathrm{d}y$ is the number of agents whose states are at time $t \in \mathbb{R}_{\geq 0}$ in the open box $(x, x + \mathrm{d}x) \times (y, y + \mathrm{d}y)$. Note that f is assumed to be sufficiently regular to support the development of the two master equations that characterize the KTMAS framework [22].

The density function enables the expression of the total number of agents in the MAS via the integral

$$n = \int_Q f(x, y, t)\,\mathrm{d}x\mathrm{d}y, \tag{5}$$

where the time dependence of n is omitted, reflecting the assumption that the total number of agents remains constant. Similarly, the mean of the x state components can be expressed as

$$\overline{x}(t) = \frac{1}{n}\int_Q x\, f(x, y, t)\,\mathrm{d}x\mathrm{d}y. \tag{6}$$

Finally, the mean of the y state components can be expressed as

$$\overline{y}(t) = \frac{1}{n}\int_Q y\, f(x, y, t)\,\mathrm{d}x\mathrm{d}y. \tag{7}$$

The introduction of the density function can be used to derive the two master equations that form the core of the KTMAS framework, which are normally called *(strong form of the) Boltzmann equation* [22] and *weak form of the Boltzmann equation* [22]. The former provides a fine-grained characterization of the dynamics of the agent states, which is normally too fine-grained to be feasible for complex MAS. Therefore, this equation is not further detailed here. The latter is relevant here because it is used to prove the analytical results discussed in Section 4. In particular, the weak form of the Boltzmann equation is used to derive a closed-form expression of the flash period induced by the proposed protocol.

For any sufficiently regular *test function* $\phi : Q \to \mathbb{R}$, the weak form of the Boltzmann equation is

$$\int_Q \frac{\partial f}{\partial t}(\mathbf{q}, t)\phi(\mathbf{q})\,\mathrm{d}^2\mathbf{q} = \frac{\beta}{2}\int_{Q^2} \Delta(\mathbf{q}_s, \mathbf{q}_r) f(\mathbf{q}_s, t) f(\mathbf{q}_r, t)\,\mathrm{d}^2\mathbf{q}_r \mathrm{d}^2\mathbf{q}_s, \tag{8}$$

where

$$\Delta(\mathbf{q}_s, \mathbf{q}_r) = \phi(\mathbf{q}'_r) + \phi(\mathbf{q}'_s) - \phi(\mathbf{q}_r) - \phi(\mathbf{q}_s) \tag{9}$$

and $\beta \in \mathbb{R}_+$ is defined for large MAS as

$$\beta = \frac{2}{n^2}\nu. \tag{10}$$

Note that Equation (8) is relevant for the analytical study of the proposed heartbeat protocol because if $\phi(x, y) = x$ is chosen, the left-hand side of Equation (8) reduces to

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_Q x f(x, y, t)\,\mathrm{d}x\mathrm{d}y = n\frac{\mathrm{d}\overline{x}}{\mathrm{d}t}(t), \tag{11}$$

which turns Equation (8) into a condition on the time derivative of the mean of the x state components. Similarly, if $\phi(x, y) = y$ is chosen, the left-hand side of Equation (8) reduces to

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_Q y f(x, y, t)\,\mathrm{d}x\mathrm{d}y = n\frac{\mathrm{d}\overline{y}}{\mathrm{d}t}(t), \tag{12}$$

which turns Equation (8) into a condition on the time derivative of the mean of the y state components.

4.2 Analytical Results

Consider the MAS $\mathcal{M}$ described in Section 3. The dynamics of $\mathcal{M}$ can be studied using the KTMAS framework if the interaction rule that characterizes the protocol is total and locally invertible over Q^2, where $Q = \mathbb{R}^2$ is the set of admissible agent states. The following lemma states this property.

Lemma 1. *The interaction rule used by the agents in $\mathcal{M}$ is total and locally invertible over $\mathbb{R}^4$.*

Proof. The agents in $\mathcal{M}$ interact according to Equation (4), which depends on the two protocol parameters $\gamma \in (0, 1)$ and $\delta \in (0, 1)$. If the interaction rule is expressed as a transformation $(x'_s, y'_s, x'_r, y'_r) = \mathbf{t}(x_s, y_s, x_r, y_r)$, it is possible to observe that this transformation is linear and, hence, total over $\mathbb{R}^4$. The Jacobian matrix of this transformation is

$$\partial\mathbf{t}(x_s, y_s, x_r, y_r) = \begin{pmatrix} 1-\gamma & \delta & 0 & 0 \\ -\delta & 1-\gamma & 0 & 0 \\ \gamma & 0 & 1 & \delta \\ 0 & \gamma & -\delta & 1 \end{pmatrix}. \tag{13}$$

The determinant of this Jacobian matrix is

$$\det \partial\mathbf{t}(x_s, y_s, x_r, y_r) = (1 + \delta^2)\left[(1 - \gamma)^2 + \delta^2\right], \tag{14}$$

and therefore it is always nonzero for $\gamma \in (0, 1)$ and $\delta \in (0, 1)$, which implies that the interaction rule is locally invertible. □

This lemma ensures that the proposed protocol can be studied using the KTMAS framework. In particular, the following proposition applies the KTMAS framework to find a closed-form expression of the flash period induced by the protocol.

Proposition 1. *On average, the mean of the x state components of the agents in $\mathcal{M}$ evolves for $t \in \mathbb{R}_{\geq 0}$ as*

$$\overline{x}(t) = \overline{x}_0 \cos(\beta\delta n t) + \overline{y}_0 \sin(\beta\delta n t), \tag{15}$$

and the mean of the associated y state components evolves as

$$\overline{y}(t) = \overline{y}_0 \cos(\beta\delta n t) - \overline{x}_0 \sin(\beta\delta n t). \tag{16}$$

Proof. The evolution of $\overline{x}(t)$ can be studied by considering the weak form of the Boltzmann equation with the test function $\phi(x, y) = x$. Substituting this test function into Equation (8) and using Equation (6), the left-hand side of Equation (8) becomes

$$n \frac{\mathrm{d}}{\mathrm{d}t} \overline{x}(t) \tag{17}$$

and the associated right-hand side becomes

$$\frac{\beta}{2} \int_{\mathbb{R}^2} (x_s' + x_r' - x_s - x_r) f(x_s, y_s, t) f(x_r, y_r, t)\, \mathrm{d}x_s \mathrm{d}y_s \mathrm{d}x_r \mathrm{d}y_r, \tag{18}$$

where x_r' and x_s' denote the components of the postinteraction states in Equation (4). Basic algebraic manipulations using the interaction rule result in

$$x_s' + x_r' - x_s - x_r = \delta(y_s + y_r), \tag{19}$$

and therefore Equation (8) for $\phi(x, y) = x$ becomes

$$\frac{\mathrm{d}}{\mathrm{d}t} \overline{x}(t) = \frac{\beta\delta}{2n} \int_{\mathbb{R}^2} (y_s + y_r) f(x_s, y_s, t) f(x_r, y_r, t)\, \mathrm{d}x_s \mathrm{d}y_s \mathrm{d}x_r \mathrm{d}y_r.$$

Using Equation (5), which states a relationship between the density function and n, together with Equation (7), which expresses a relationship between the density function and $\overline{y}(t)$, the integral on the right-hand side of the previous equation becomes

$$\int_{\mathbb{R}^2} (y_s + y_r) f(x_s, y_s, t) f(x_r, y_r, t)\, \mathrm{d}x_s \mathrm{d}y_s \mathrm{d}x_r \mathrm{d}y_r = 2n^2 \overline{y}(t).$$

Therefore, Equation (8) for $\phi(x, y) = x$ reduces to the first-order homogeneous linear differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t} \overline{x}(t) = \beta\delta n\, \overline{y}(t). \tag{20}$$

Analogously, the evolution of $\overline{y}(t)$ is derived by choosing the test function $\phi(x, y) = y$ in the weak form of the Boltzmann equation. In this case, the left-hand side of Equation (8) becomes

$$n \frac{\mathrm{d}}{\mathrm{d}t} \overline{y}(t) \tag{21}$$

and the corresponding right-hand side becomes

$$\frac{\beta}{2} \int_{\mathbb{R}^2} (y_s' + y_r' - y_s - y_r) f(x_s, y_s, t) f(x_r, y_r, t)\, \mathrm{d}x_s \mathrm{d}y_s \mathrm{d}x_r \mathrm{d}y_r, \tag{22}$$

where, according to the adopted interaction rule,

$$y_s' + y_r' - y_s - y_r = -\delta(x_s + x_r). \tag{23}$$

Therefore, Equation (8) for $\phi(x, y) = y$ becomes

$$\frac{\mathrm{d}}{\mathrm{d}t} \overline{y}(t) = -\frac{\beta\delta}{2n} \int_{\mathbb{R}^2} (x_s + x_r) f(x_s, y_s, t) f(x_r, y_r, t)\, \mathrm{d}x_s \mathrm{d}y_s \mathrm{d}x_r \mathrm{d}y_r.$$

Using Equation (5) and Equation (6), the previous equation simplifies to the first-order homogeneous linear differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t} \overline{y}(t) = -\beta\delta n\, \overline{x}(t). \tag{24}$$

In summary, $\overline{x}(t)$ and $\overline{y}(t)$ are governed by the homogeneous system of linear differential equations with constant coefficients

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}t} \overline{x}(t) = \beta\delta n\, \overline{y}(t) \\ \dfrac{\mathrm{d}}{\mathrm{d}t} \overline{y}(t) = -\beta\delta n\, \overline{x}(t) \end{cases} \tag{25}$$

with the initial conditions $\overline{x}(0) = \overline{x}_0$ and $\overline{y}(0) = \overline{y}_0$. This system can be easily solved to obtain

$$\begin{cases} \overline{x}(t) = \overline{x}_0 \cos(\beta\delta n t) + \overline{y}_0 \sin(\beta\delta n t) \\ \overline{y}(t) = \overline{y}_0 \cos(\beta\delta n t) - \overline{x}_0 \sin(\beta\delta n t) \end{cases} \tag{26}$$

for $t \in \mathbb{R}_{\geq 0}$, which proves the proposition. □

Note that the proposed protocol assumes $\overline{x}_0 = \overline{y}_0$, and therefore the previous proposition leads to the following corollary when

$$\omega = \beta\delta n. \tag{27}$$

Corollary 1. *On average, the mean of the x state components of the agents in $\mathcal{M}$ evolves for $t \in \mathbb{R}_{\geq 0}$ as*

$$\overline{x}(t) = \overline{x}_0 \sqrt{2} \sin\left(\omega t + \frac{\pi}{4}\right) \tag{28}$$

and the mean of the associated y state components evolves as

$$\overline{y}(t) = -\overline{x}_0 \sqrt{2} \sin\left(\omega t - \frac{\pi}{4}\right). \tag{29}$$

Proof. The corollary readily follows from Proposition 1 using $\overline{x}_0 = \overline{y}_0$ and $\omega = \beta\delta n$. □

The previous corollary shows that the mean of the x state components and the mean of the y state components have oscillatory behavior with period

$$T = \frac{2\pi}{\omega} = \frac{2\pi}{\beta\delta n}. \tag{30}$$

This is the period shown in Figure 1(a) for a sample simulation. A comparison with Figure 1(c) suggests that this period equals the associated flash period. Therefore, Figure 1(a) and Figure 1(c) suggest that Equation (30) can be used to readily express the nominal period of the flashes induced by the proposed protocol. The experiments discussed in Section 5 confirm that Equation (30) accurately predicts the measured flash period.

Note that Equation (30) can be further simplified to reveal an interesting property of the flash period induced by the proposed protocol. First, Equation (10) is used to remove β from Equation (30). Then, the relationship between the sending rate $\hat{\nu}$ used in the proposed protocol and the interaction rate ν used in the KTMAS framework is considered. Notably, the classical superposition theorem for Poisson point processes yields

$$\nu = n\hat{\nu}, \tag{31}$$

which can be used to further simplify T and obtain

$$T = \frac{\pi}{\delta\hat{\nu}}. \tag{32}$$

5 EXPERIMENTAL EVALUATION

The experiments documented in this section were conducted to validate relevant characteristics of the flash period implied by the proposed protocol. In particular, the flash period depends on δ and $\hat{v}$ according to Equation (32) and it does not depend on n. Additionally, the experiments were designed to confirm that the protocol produces sufficiently brief flashes.

Overall, the experiments confirm that the proposed protocol induces flashes with sufficiently stable periods and sufficiently stable and brief durations to effectively solve the heartbeat synchronization problem in target applications [3].

5.1 The Simulator

A custom-built simulator was designed and implemented to evaluate the characteristics of the proposed protocol. The simulator was implemented in Python (version 3.12.4) without using additional libraries or frameworks. The simulator is accompanied by an interactive web application to help users visually inspect the protocol before using the simulator. These tools are available from https://www.github.com/bergenti/heartbeat.

The developed simulator is a discrete event simulator that depends on the following parameters to simulate a MAS:

1. The number of agents $n \in \mathbb{N}_+$ in the simulated MAS;
2. The value of $\hat{v} \in \mathbb{R}_+$ used to delay the sending of messages;
3. The value of $\alpha \in (0, 1)$ used for the protocol;
4. The value of $\delta \in (0, 1)$ used for the protocol;
5. The value of $\gamma \in (0, 1)$ used for the protocol;
6. The number of flash periods $p \in \mathbb{N}_+$ to be simulated; and
7. The values of κ_0 and κ_1 used to measure flash periods and flash durations (see below).

The results of a simulation are the following four values used to evaluate the characteristics of flash periods and flash durations:

1. The mean flash period $\overline{T}$ and its standard deviation σ_T over the requested p periods; and
2. The mean flash duration $\overline{\tau}$ and its standard deviation σ_τ over the requested p flashes.

Note that exactly p flash durations are measured because one flash duration is measured for each measured period.

A simulation starts by initializing the agent states using a uniform random variable in $(0, 1)$. Then, the delay that each agent waits before sending messages is initialized using an exponential random variable with rate $\hat{v}$. After the initialization, a sender agent is chosen among the agents with minimum delay for the next interaction. Then, a receiver agent is chosen uniformly at random and the interaction is performed using Equation (4). Before advancing to another interaction, the delay of the sender agent is reinitialized, the delays of the other agents are reduced, and the simulated time is advanced. The simulator iteratively performs interactions until exactly p flash periods and flash durations are measured.

The simulator continuously monitors two conditions on the number of agents currently emitting light, denoted by $\hat{n}$, with $0 \leq \hat{n} \leq n$, to measure flash periods and flash durations. The first condition is called *trough condition* and uses the parameter $\kappa_0 \in \left(0, \frac{1}{2}\right)$ to identify when flashes start:

$$\hat{n} > \kappa_0\, n. \tag{33}$$

The second condition is called *peak condition* and uses the parameter $\kappa_1 \in \left(0, \frac{1}{2}\right)$ to identify when flashes reach their peaks:

$$\hat{n} > (1 - \kappa_1)\, n. \tag{34}$$

The measurement of flash periods works as follows. The simulator initially monitors the peak condition. If this condition starts holding at time $\tilde{t}_1$, then the simulator continues its work but stops monitoring the peak condition and starts monitoring the trough condition, which surely holds. If the trough condition no longer holds at time $\tilde{t}_2$, then the simulator continues its work but stops monitoring the trough condition and restarts monitoring the peak condition, which surely does not hold. If the peak condition starts holding at time $\tilde{t}_3$, then the current period is measured as $\tilde{t}_3 - \tilde{t}_1$. The measurement of the subsequent period immediately starts because the event that concludes a period coincides with the event that initiates the subsequent period.

The simulator measures flash durations concurrently with the measurement of flash periods. The simulator initially monitors the trough condition. If this condition starts holding at time $\hat{t}_1$, then the simulator continues its work but stops monitoring the trough condition and starts monitoring the peak condition, which surely does not hold. If the peak condition starts holding at time $\hat{t}_2$, the simulator continues its work but stops monitoring the peak condition and restarts monitoring the trough condition, which surely holds. If the trough condition stops holding at time $\hat{t}_3$, then the current flash duration is measured as $\hat{t}_3 - \hat{t}_1$.

In the simulations discussed in the rest of this section, the values $\kappa_0 = 10^{-2}$ and $\kappa_1 = 10^{-2}$ are used. These values imply that the simulator identifies a flash period when $\hat{n}$ exceeds $\frac{99}{100}n$, then $\hat{n}$ drops to $\frac{1}{100}n$ or less before exceeding $\frac{99}{100}n$ again. Essentially, the simulator identifies consecutive peaks to measure flash periods. Concurrently, the simulator identifies a flash when $\hat{n}$ exceeds $\frac{1}{100}n$, then $\hat{n}$ exceeds $\frac{99}{100}n$ before dropping to $\frac{1}{100}n$ or less. Essentially, the simulator identifies peaks to measure flash durations.

5.2 Experimental Results

The simulator described previously was used to perform all the simulations discussed in the rest of this section. These simulations were organized into a set of configurations, each of which fixes the parameters of a simulation. For each configuration, a corresponding MAS was simulated fixing $\kappa_0 = \kappa_1 = 10^{-2}$ and $p = 100$. Each simulated MAS is conventionally denoted by $\mathcal{M}_{n,\alpha,\gamma,\delta,\hat{v}}$.

Table 1 shows the results of the simulations of the thirty MAS $\mathcal{M}_{n,10^{-2},10^{-1},\delta,1}$ obtained using $\delta \in \{i \cdot 10^{-3} \mid 1 \leq i \leq 10\}$ and $n \in \{10^2, 5 \cdot 10^2, 10^3\}$. The table reports the considered δ, the nominal flash period $T \in \mathbb{R}_+$ obtained using Equation (32), and the relevant measured values for each considered n. In particular, the table reports for each considered n: the measured average period $\overline{T}$ relative to the nominal flash period (in percentage)

$$\tilde{T} = \frac{\overline{T}}{T}, \tag{35}$$

the associated standard deviation σ_T relative to the nominal flash period (in percentage)

$$\tilde{\sigma}_T = \frac{\sigma_T}{T}, \tag{36}$$

Table 1: The results of the simulations of the thirty MAS $\mathcal{M}_{n,10^{-2},10^{-1},\delta,1}$ for $\delta \in \{i \cdot 10^{-3} \mid 1 \leq i \leq 10\}$, $n \in \{10^2, 5 \cdot 10^2, 10^3\}$, $p = 100$, and $\kappa_0 = \kappa_1 = 10^{-2}$ in terms of the measured average period relative to the nominal flash period $\tilde{T}$ (in percentage), its associated standard deviation $\tilde{\sigma}_T$ (in percentage), the measured average flash duration relative to the nominal flash period $\tilde{\tau}$ (in percentage), and its associated standard deviation $\tilde{\sigma}_\tau$ (in percentage).

		$n = 100$				$n = 500$				$n = 1{,}000$			
δ	T [s]	$\tilde{T}$ [%]	$\tilde{\sigma}_T$ [%]	$\tilde{\tau}$ [%]	$\tilde{\sigma}_\tau$ [%]	$\tilde{T}$ [%]	$\tilde{\sigma}_T$ [%]	$\tilde{\tau}$ [%]	$\tilde{\sigma}_\tau$ [%]	$\tilde{T}$ [%]	$\tilde{\sigma}_T$ [%]	$\tilde{\tau}$ [%]	$\tilde{\sigma}_\tau$ [%]
0.001	3141.593	100.036	0.197	4.737	0.051	99.995	0.083	4.754	0.022	99.997	0.049	4.754	0.014
0.002	1570.796	100.042	0.283	4.959	0.065	100.002	0.126	5.003	0.037	99.990	0.078	5.014	0.031
0.003	1047.198	100.082	0.374	5.195	0.102	100.013	0.140	5.250	0.055	99.986	0.103	5.262	0.038
0.004	785.398	100.064	0.409	5.431	0.137	100.003	0.175	5.499	0.069	99.993	0.121	5.516	0.050
0.005	628.319	100.066	0.437	5.675	0.165	99.992	0.208	5.750	0.084	99.996	0.131	5.763	0.060
0.006	523.599	100.084	0.487	5.891	0.156	99.991	0.236	5.998	0.092	100.000	0.152	6.022	0.064
0.007	448.799	100.094	0.545	6.141	0.252	99.994	0.264	6.242	0.118	99.990	0.177	6.264	0.070
0.008	392.699	100.086	0.636	6.343	0.251	99.993	0.272	6.472	0.122	99.990	0.193	6.511	0.083
0.009	349.066	100.099	0.634	6.638	0.256	99.998	0.320	6.741	0.146	99.988	0.191	6.772	0.098
0.010	314.159	100.088	0.669	6.836	0.251	99.998	0.330	6.995	0.155	99.982	0.231	7.027	0.117

the measured average flash duration τ relative to the nominal flash period (in percentage)

$$\tilde{\tau} = \frac{\tau}{T}, \tag{37}$$

and its associated standard deviation σ_τ relative to the nominal flash period (in percentage)

$$\tilde{\sigma}_\tau = \frac{\sigma_\tau}{T}. \tag{38}$$

Table 1 shows that the measured flash period accurately and precisely approximates the nominal flash period. The table shows that the flash duration is much less than 10% of the nominal flash period.

Figure 2 shows the results of the simulations for the fifty MAS $\mathcal{M}_{100,10^{-2},10^{-1},\delta,\hat{\nu}}$ obtained using $\delta \in \{i \cdot 10^{-3} \mid 1 \leq i \leq 10\}$ and $\hat{\nu} \in \{1\,s^{-1}, \sqrt{10}\,s^{-1}, 10\,s^{-1}, 10\sqrt{10}\,s^{-1}, 100\,s^{-1}\}$. The marks are the measured flash periods and the lines are the plots of T from Equation (32). The figure confirms that the flash period can be predicted with high precision and accuracy.

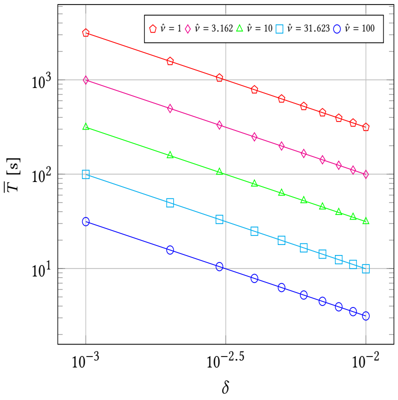


Figure 2: The lines are the plots of T from Equation (32) obtained using $\hat{\nu} \in \{1\,s^{-1}, \sqrt{10}\,s^{-1}, 10\,s^{-1}, 10\sqrt{10}\,s^{-1}, 100\,s^{-1}\}$ and $\delta \in \{i \cdot 10^{-3} \mid 1 \leq i \leq 10\}$, while the marks are the average flash periods obtained by simulating the MAS $\mathcal{M}_{100,10^{-2},10^{-1},\delta,\hat{\nu}}$ for the same values of δ and $\hat{\nu}$ with $p = 100$ and $\kappa_0 = \kappa_1 = 10^{-2}$.

6 CONCLUSION

The paper has introduced and examined a novel decentralized protocol for heartbeat synchronization in MAS. The defining characteristic of this protocol lies in its reliance on one-way communication to eliminate the need for agents to align frequencies and phases. By dispensing with blocking operations and message replies, the protocol inherently avoids the risk of deadlocks and maintains uninterrupted agent activity.

The analytical contribution of the paper is the derivation of a closed-form expression for the flash period induced by the protocol. This result establishes a clear and concise relationship between the emergent collective behavior induced by the protocol and two key system-level quantities: a tunable protocol parameter and the average number of messages sent per unit time per agent. This result not only provides predictive power but also allows practitioners to design systems with tunable synchronization properties while maintaining confidence in their long-term stability.

Experimental validation through simulations has demonstrated that the theoretical expression aligns with observed outcomes with high precision and accuracy. The empirical findings confirm that the flashes generated under the protocol remain sufficiently brief and stable, thereby fulfilling the practical requirements of heartbeat synchronization. The combination of rigorous analysis and empirical validation strongly supports the conclusion that the protocol achieves both robustness and scalability.

Future extensions of the protocol include relaxing the assumption of identical interaction rates. Allowing agents to operate with individual interaction rates would further reduce restrictions on their cyclic behavior, but would also require corresponding adaptations of the KTMAS framework. Preliminary considerations suggest that such an extension is feasible.

Another direction for further investigation concerns the impact of failures and heterogeneity on synchronization. Deviations from the prescribed protocol or variations in parameter values across agents inevitably alter collective behavior and influence both the flash period and flash duration. An open question concerns the robustness of the protocol not only against communication failures and delays but also against malevolent and erratic agents.

REFERENCES

- [1] Zeyad Ghaleb Al-Mekhlafi, Zurina Mohd Hanapi, and Ahmed Mohammed Shamsan Saleh. 2019. Firefly-inspired time synchronization mechanism for self-organizing energy-efficient wireless sensor networks: A Survey. *IEEE Access* 7 (2019), 115229–115248.
- [2] Giacomo Albi, Elisa Calzola, and Giacomo Dimarco. 2024. A data-driven kinetic model for opinion dynamics with social network contacts. *European Journal of Applied Mathematics* 36, 2 (2024), 1–27.
- [3] Ozalp Babaoglu, Toni Binci, Mark Jelasity, and Alberto Montresor. 2007. Firefly-inspired heartbeat synchronization in overlay networks. In *Proceedings of the 1st International Conference on Self-Adaptive and Self-Organizing Systems (SASO 2007)*. IEEE, New York, 77–86.
- [4] Nicola Bellomo, Diletta Burini, Giovanni Dosi, Livio Gibelli, Damian Knopoff, Nisrine Outada, Pietro Terna, and Maria Enrica Virgillito. 2021. What is life? A perspective of the mathematical kinetic theory of active particles. *Mathematical Models and Methods in Applied Sciences* 31, 9 (2021), 1821–1866.
- [5] Abdelghani Bellouquid and Marcello Delitala. 2006. *Mathematical Modeling of Complex Biological Systems*. Springer, Berlin.
- [6] Federico Bergenti and Stefania Monica. 2025. Experimental evaluation of a decentralized protocol for emergent heartbeat synchronization. In *DAI '25: Proceedings of the 2025 7th International Conference on Distributed Artificial Intelligence*. ACM, New York, 19–27.
- [7] Federico Bergenti, Stefania Monica, Gloria Tamboroni, and Franco Zambonelli. 2025. A study of the dynamics of the average in a residual gossip protocol. In *Proceedings of the 21st European Conference on Multi-Agent Systems (EUMAS 2024) (Lecture Notes in Computer Science, Vol. 15685)*. Springer, Berlin, 255–273.
- [8] Federico Bergenti and Alessandro Ricci. 2002. Three approaches to the coordination of multiagent systems. In *Proceedings of the 2002 ACM Symposium on Applied Computing (SAC 2002)*. ACM, New York, 367–372.
- [9] Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah. 2006. Randomized gossip algorithms. *IEEE Transactions on Information Theory* 52, 6 (2006), 2508–2530.
- [10] Bard Ermentrout. 1991. An adaptive model for synchrony in the firefly pteroptyx malaccae. *Journal of Mathematical Biology* 29, 6 (1991), 571–585.
- [11] Alessandro F. Garcia, Carlos José Pereira de Lucena, Franco Zambonelli, Andrea Omicini, and Jaelson Castro (Eds.). 2002. *Software Engineering for Large-Scale Multi-Agent Systems, Research Issues and Practical Applications*. Lecture Notes in Computer Science, Vol. 2603. Springer, Berlin.
- [12] Marco Günther, Axel Klar, Thorsten Materne, and Raimund Wegener. 2002. An explicitly solvable kinetic model for vehicular traffic and associated macroscopic equations. *Mathematical and Computer Modelling* 35, 5–6 (2002), 591–606.
- [13] Jane Hillston, Jeremy Pitt, Martin Wirsing, and Franco Zambonelli. 2014. Collective adaptive systems: Qualitative and quantitative modelling and analysis. *Dagstuhl Reports* 4, 12 (2014), 68–113.
- [14] Konrad Iwanicki, Maarten van Steen, and Spyros Voulgaris. 2006. Gossip-based clock synchronization for large decentralized systems. In *Self-Managed Networks, Systems and Services (Lecture Notes in Computer Science, Vol. 3996)*, Alexander Keller and Jean-Philippe Martin-Flatin (Eds.). Springer, Berlin, 28–42.
- [15] Márk Jelasity and Ozalp Babaoglu. 2006. T-Man: Gossip-based overlay topology management. In *Engineering Self-Organising Systems: Third International Workshop (ESOA 2005) (Lecture Notes in Computer Science, Vol. 3910)*, Sven A. Brueckner, Giovanna Di Marzo Serugendo, David Hales, and Franco Zambonelli (Eds.). Springer, Berlin, 1–15.
- [16] Márk Jelasity, Alberto Montresor, and Ozalp Babaoglu. 2005. Gossip-based aggregation in large dynamic networks. *ACM Transactions on Computer Systems* 23, 3 (2005), 219–252.
- [17] Yoshiki Kuramoto. 1984. *Chemical Oscillations, Waves, and Turbulence*. Springer, Berlin.
- [18] Renato E. Mirollo and Steven H. Strogatz. 1990. Synchronization of pulse-coupled biological oscillators. *SIAM Journal on Applied Mathematics* 50, 6 (1990), 1645–1662.
- [19] Stefania Monica and Federico Bergenti. 2017. An analytic study of opinion dynamics in multi-agent systems. *Computers & Mathematics with Applications* 73, 10 (2017), 2272–2284.
- [20] Stefania Monica and Federico Bergenti. 2017. Opinion dynamics in multi-agent systems: Selected analytic models and verifying simulations. *Computational & Mathematical Organization Theory* 23, 3 (2017), 423–450.
- [21] Stefania Monica, Federico Bergenti, and Franco Zambonelli. 2022. Toward a kinetic framework to model the collective dynamics of multi-agent systems. In *Proceedings of the 11th International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (ISoLA 2022) (Lecture Notes in Computer Science, Vol. 13703)*. Springer, Berlin, 165–180.
- [22] Stefania Monica, Federico Bergenti, and Franco Zambonelli. 2023. A kinetic approach to investigate the collective dynamics of multi-agent systems. *International Journal on Software Tools for Technology Transfer* 25, 5–6 (2023), 693–705.
- [23] L. Pareschi and G. Toscani. 2013. *Interacting Multiagent Systems: Kinetic Equations and Montecarlo Methods*. Oxford University Press, Oxford.
- [24] Lorenzo Pareschi and Giuseppe Toscani. 2014. Wealth distribution and collective knowledge: A Boltzmann approach. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 372 (2014), 20130396.
- [25] Fernando Perez-Diaz, Ruediger Zillmer, and Roderich Groß. 2015. Firefly-inspired synchronization in swarms of mobile agents. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015)*. IFAAMAS, Richland, 279–286.
- [26] Gonzalo Marcelo Ramírez-Ávila, Juergen Kurths, and Jean-Louis Deneubourg. 2018. Fireflies: A paradigm in synchronization. In *Chaotic, Fractional, and Complex Dynamics: New Insights and Perspectives (Understanding Complex Systems)*. Springer, Berlin, 35–64.
- [27] Moisés Santillán. 2025. Pulse-coupled oscillator synchronization: Bridging theory and experiments with electronic firefly networks. *Chaos, Solitons & Fractals* 196 (2025), 116377.
- [28] Geoffrey Werner-Allen, Geetika Tewari, Ankit Patel, Matt Welsh, and Radhika Nagpal. 2005. Firefly-inspired sensor network synchronicity with realistic radio effects. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems (SenSys 2005)*. ACM, New York, 142–153.
- [29] D. H. Wolpert and K. Tumer. 2002. Collective intelligence, data routing and Braess' paradox. *Journal of Artificial Intelligence Research* 16 (2002), 359–387.
- [30] Yang Xu, Elizabeth Liao, Paul Scerri, Bin Yu, Mike Lewis, and Katia Sycara. 2006. Towards flexible coordination of large scale multi-agent teams. In *Coordination of Large-Scale Multiagent Systems*, Paul Scerri, Régis Vincent, and Roger Mailler (Eds.). Springer, Berlin, 287–309.