Article

Evaluation of the Effects of Mobile Smart Object to Boost IoT Network Synchronization

Francesco Lamonaca * and Domenico Luca Carnì

Department of Informatics, Modelling, Electronics and Systems Science, University of Calabria,
87036 Arcavacata, Italy; dlcarni@dimes.unical.it

* Correspondence: f.lamonaca@dimes.unical.it

Abstract: This paper deals with the synchronization of Mobile Smart Objects (MSOs). Today, this scenario is becoming typical in Industrial IoT applications due to the plethora of MSOs available as robots, drones and wearables, equipped by sensors making them measurement instruments cooperating in distributed measurement systems. In this context, the synchronization accuracy is directly tied with the accuracy of the performed measurements. In hierarchical synchronization approaches, the presence of an MSO makes the network topology time varying, and this could prevent the synchronization of the whole network. Peer to peer approaches do not need node hierarchy to synchronize but could not converge to a common sense of time. To overcome these challenges, this paper proposes a consensus-based approach for which the convergence to a common sense of time is here demonstrated. The proposal deploys the MSO to bring the common sense of time from an SO to another, establishing new paths among SOs. The new paths are temporary and depend on the MSO's route. In the paper, the influence of the MSO's route on the synchronization accuracy σ and the time interval to synchronize all the SOs Δ_{TIS} is investigated, also. The mathematical proof, the simulations and the experimental tests confirm that the MSO can reduce both the values of σ and Δ_{TIS} , because the new connections introduced by the MSO can boost the exchange of information among SOs. Consequently, the criteria to a priori select the route ameliorating σ and Δ_{TIS} values are proposed.

Keywords: synchronization; consensus; mobile object; IoT network synchronization; IoT network topology

Citation: Lamonaca, F.; Carnì, D.L. Evaluation of the Effects of Mobile Smart Object to Boost IoT Network Synchronization. *Sensors* **2021**, *21*, 3957. <https://doi.org/10.3390/s21123957>

Academic Editor: Guillermo Villanueva

Received: 27 April 2021

Accepted: 5 June 2021

Published: 8 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Accurate time synchronization is a key enabler of real-time Industrial IoT applications [1–4], and numerous applications are available [1,5–10]. It permits the coordination in time between sensors and actuators equipping Smart Objects (SOs) [8,9,11] different SOs allowing the modelling of the system, and the design of the control systems [6,12–14]. Moreover, the collection of the measurements of a big number of SOs (Big Data) allows data retrieval [15–17] if their sampling time is accurately defined.

In recent years, low-power wireless SOs have gained popularity [18]. They interact to form large networks such as Machine to Machine (M2M) networks and Wearable Computing. In these contexts, the major challenge is to guarantee robust communication between elements while keeping the whole network energy efficient. The time synchronization permits them to coordinate communication slot for the SOs [4], implementing a time division multiple access communication protocols, i.e., the SO has to forward messages to its neighbors in only a short wakeup period.

The time synchronization of the measurements requires that all the network SOs share the common sense of time. The synchronization accuracy σ among SOs affects the efficacy of the data retrieval, while the time interval required to synchronize all the SOs (Δ_{TIS}), i.e., the time interval starting with the execution of the synchronization algorithm and ending

when the delay between any couple of SOs is lower than the admissible delay range, affects the “warm-up” time of the system. In the time interval Δ_{TIS} , no action should be performed, because the achieved data can refer to different time instants.

These days, time synchronization in IoT wireless network must take into account that several SOs can be Mobile SOs (MSOs). For example, moving robots are used in a warehouse to move goods [19,20] autonomous unmanned aerial vehicles (drones) [21,22] coordinate among themselves to achieve a common task, wearables are used to monitor health parameters [23–25], and so on.

The MSO makes the topology of the network varying in time. In this context, promising solutions are the consensus approaches since they are based on the local exchange of messages among SOs [26,27]. In [26], the MSO navigates toward each SO deployed on a straight-line route, bringing the sense of time of the previous visited SO. In [27], a bidimensional deployment of the SOs is considered, and the MSO moves on a straight-line route. The research proposed in [26,27] highlights that the MSO reduces Δ_{TIS} [28].

The novelties of this paper consist in: (i) proofing the convergence of the proposed synchronization method, (ii) considering that the MSO can move on a bidimensional route (that is a typical scenario), and (iii) detecting the update condition that MSO has to satisfy to promote the SOs’ synchronization, in the case it executes for the first time the synchronization procedure.

Moreover, in the paper, it is proved that the increased number of paths not only reduce Δ_{TIS} but could increase the synchronization accuracy σ . This is conceptually justified by the fact that, during the synchronization, the MSO visit a SO, and it is connected with it and all SOs in its Broadcast Range (BR). The MSO (i) takes part to the network synchronization and (ii) increases the number of connections among the set of SOs in its BR and, consequently, the number of paths among SOs. If the MSO makes higher the number of paths among SOs, faster is the propagation of the common sense of time: the influence of the sense of time of a SO on all the others occurs in a lower time interval. As a consequence, the effect of the connections introduced by the new time-varying paths on σ and Δ_{TIS} is analyzed. In particular, to link the MSO’s route to the improvement of the two synchronization parameters Δ_{TIS} and σ , two statistical indexes are defined: the average of the minimum number of connections in the path between any couple of SOs μ_{con} , and its variation σ_{con} versus the time. Based on such approach, the criteria to select more convenient MSO route are given.

The paper is organized as follows: In Section 2, the hardware and software architectures of both SO and MSO and the synchronization procedure are abstracted. In Section 3, the effect of the MSO on the synchronization is analyzed. In Section 4, numerical tests are executed to evaluate the effect of the MSO route on Δ_{TIS} and σ together with μ_{con} and σ_{con} to detect the more convenient criteria for the MSO route selection. In Section 5, experimental results are presented to assess the theoretical and numerical results in a real-world scenario. The conclusions follow.

2. SO and MSO Architecture and Synchronization Procedure

Concerning the synchronization purpose, both the architectures of the SO and MSO can be related to the node architecture of [29] and can be modeled by three sections: clock, μ -processor, and radio. With respect to the SO, the MSO’s architecture also includes the hardware and software allowing the movement along the route.

The clock is typically composed of a nontunable crystal oscillator [30] driving a counter. The output of this counter is used as a hardware clock. For the i -th SO, SO# i , at time t , the hardware clock value τ_i is:

$$\tau_i(t) = \lfloor \alpha_i t \rfloor + \beta_i, \quad i = 1, \dots, Nc, \quad (1)$$

where $\lfloor \cdot \rfloor$ is the round toward negative infinity operator, Nc is the number of SOs in the IoT network, α_i is the frequency, and β_i the offset of the clock.

Since α_i and β_i vary according to the actual realization of the crystal oscillator and the starting condition of the SO#i, respectively, the μ -processor acts as clock-servo, evaluating the correction parameters for frequency $\hat{\alpha}_i(t_k)$ and offset $\hat{\delta}_i(t_k)$, respectively. The μ -processor computes the software time $\hat{\tau}_i(t)$ according to:

$$\hat{\tau}_i(t) = \hat{\alpha}_i(t)\tau_i(t) + \hat{\delta}_i(t), \quad (2)$$

The correction parameters $\hat{\alpha}_i(t)$, and $\hat{\delta}_i(t)$ are evaluated by each SO on the basis of synchronization messages (SynMsg) exchanged among the communicating SOs by the radio section.

The radio section is in charge of receive SynMsgs from communicating SOs and to send, in broadcast modality, the SynMsg each $t_k = kT_P, k \in N^+$, where T_P is the synchronization time period. The message SynMsg_{ji}, sent at t_k by SO#j and received by SO#i, includes: $\{id_j, \hat{\alpha}_j(t_k), \hat{\delta}_j(t_k), \tau_j(t_k)\}$, where id_j is the SO identifier that is unique for each SO.

At the reception of the SynMsg_{ji}, the radio section communicates this event to the μ -processor that starts the synchronization procedure of the block scheme of Figure 1.

Figure 1. Block scheme of the synchronization procedure executed by the i -th object (SO or MSO), when receiving a SynMsg_{ji} from the j -th object.

In Figure 1, τ_{ij} is the number of clock impulse of SO#i at the reception of SynMsg_{ij}. The variables τ_j^{old} and τ_{ij}^{old} refer to the previous reception of SynMsg_{ij}, ρ_v and ρ_o are design parameters (memory factors) that can be set in the range [0, 1], and t_k^+ indicates the upgraded value of the time dependent variables $\hat{\alpha}_i(t)$, and $\hat{\delta}_i(t)$, respectively [31]. In particular, if $\rho_v = 0$ and $\rho_o = 0$, the node will correct its $\hat{\alpha}_i(t)$ and $\hat{\delta}_i(t)$ on the basis of the time values of its neighbors only, and vice-versa: if $\rho_v = 0.9$ and $\rho_o = 0.9$, the upgrade of $\hat{\alpha}_i(t)$ and $\hat{\delta}_i(t)$ will mainly depend on its own value.

The MSO is used to bring the sense of the time from one SO to another. If it executes the synchronization procedure for the first time, the SOs ignore its message. This condition is checked by the $\hat{\alpha}_j(t)$ and $\hat{\delta}_j(t)$ values in the message received from the MSO and verified that they are equal to the initialization values.

In order to speed up the convergence of the MSO to the common sense of time of the network and to avoid that it could perturb the network synchronization, at the first synchronization time instant (t_k , with $k = 1$), the MSO takes the sense of time of the first received message. This condition is checked by the MSO by its own $\hat{\alpha}_i(t)$ and $\hat{\delta}_i(t)$ values and verified that they are equal to the initialization values. At the end of the first synchronization time instant (t_k^+ , with $k = 1$), MSO will have $\hat{\tau}_i(t)$ bounded by the maximum and minimum values of the $\hat{\tau}_j(t)$ of the SOs in its BR, because it: (i) is in the same neighborhood of the visited SO, (ii) receives the same synchronization messages, and (iii) executes the same upgrade of $\hat{\alpha}_i(t)$ and $\hat{\delta}_i(t)$. In the successive synchronization time instants (t_k , with $k > 1$), MSO will have $\hat{\tau}_i(t)$ bounded by the maximum and minimum values of the $\hat{\tau}_j(t)$ of the SOs.

The number of connections in the paths among SOs influences the time with whom the information of frequency and offset of each SO clock is shared among all SOs. In particular, the fully connected topology represents the best case, because each SO is directly connected to all others. The worst case is if a nonconnected SO is present. The isolated SO cannot be synchronized, because no information is shared with it.

Because the MSO moves, it increases the number of paths among SOs, creating new connections, but these connections vary in time according with the MSO route.

In the following Section 3, the conditions making the synchronization procedure success are investigated, and the effect of the MSO route on σ and Δ_{TIS} is analyzed.

3. Effect of MSO on the IoT Network Synchronization

3.1. Convergence Assessment

An effect that must be taken into account by introducing the MSO in an IoT wireless network synchronized by the consensus-based technique of Section 2 concerns the convergence assessment to a common sense of time. The MSO spreads its sense of time in the network and potentially can make the consensus process fail [26]. A formal proof that the synchronization procedure of the Section 2 makes the consensus process successful is to verify the convergence condition of the Theorem#1 of [26]. Theorem#1 assesses that the consensus is stable, i.e., the SOs converge to a common sense of time, if the variable estimated by MSO is always bounded by the maximum and minimum estimation of the corresponding variables of each SO. At the starting of the synchronization procedure, the SOs ignore the MSO's message, and the MSO takes the sense of time of the first received message. At the successive synchronization steps, the MSO and the SOs will execute the same synchronization procedure. Therefore, the MSO's $\hat{\tau}_i(t)$ is always bounded in the maximum and minimum values of the SOs, satisfying the Theorem#1.

3.2. Reduction of σ Value

Another effect, important for measurement purposes, is the reduction of the synchronization accuracy σ . This effect is not considered in [26] and in the theoretical approaches, since they assume that, after the convergence, the difference between the estimated time values of any couple of SOs is zero. This condition cannot be achieved in practical cases, due to the nonidealities of the SO's hardware components. In [31], the effect of the nonide-

alities is modeled by introducing the bounded time-varying disturbance vector $d(t)$, with maximum value $d_{\max}$, that depends on the hardware characteristics of the SO clocks as the clock noise, the quantization error introduced by the limited resolution of the clocks, and so on. The clock hardware characteristics determine the lower bound of the achievable synchronization accuracy. The SOs and the MSO update the hardware clock value each T seconds, according to (1) at $t_{s+1} = t_s + T$:

$$\tau_i(t_{s+1}) = \tau_i(t_s) + \bar{\alpha}T + d_i(t_s), \quad (3)$$

where s is positive integer, $\bar{\alpha}$ is the clock frequency nominal value, and $d_i(t_s)$ is the i -th element of $d(t)$ such that $\|d_i(t_s)\| \leq d_{\max}$. As a consequence, at $t_{s+m} = (s+m)T$, m positive integer (without executing the synchronization procedure) it is:

$$\tau_i(t_{s+m}) = \tau_i(t_s) + m\bar{\alpha}T + d_i(t_s), \quad (4)$$

where $\|d_i(t_s)\| < d_{\max}$.

At $t_s = t_k$ and $m = m_{\text{synch}} = T_P/T$, the synchronization time occurs. By considering, for sake of simplicity, the case where only offset synchronization is performed, it is assumed that $\hat{\alpha}_i(t_k^+) - \hat{\alpha}_i(t_k) = 0$. According to (2) and the upgrade rule of $\hat{\alpha}_i$, Figure 2, it is:

$$\hat{\tau}_i(t_k^+) = \tau_i(t_k) + \hat{\alpha}_i(t_k) + (1 - \rho_0)(\hat{\tau}_j(t_k) - \hat{\tau}_i(t_k)) = \hat{\tau}_i(t_k) + (1 - \rho_0)(\hat{\tau}_j(t_k) - \hat{\tau}_i(t_k)), \quad (5)$$

By considering a synchronous communication model among SOs and MSO, from (5), it is possible to evaluate the update of the software clock time of each SO and of the MSO. It is

$$\hat{\tau}(t_k^+) = (I + \varepsilon L)\hat{\tau}(t_k), \quad (6)$$

where $\hat{\tau}(t_k^+) = [\hat{\tau}_1(t_k^+), \dots, \hat{\tau}_{Nc+1}(t_k^+)]^T$, $\hat{\tau}(t_k) = [\hat{\tau}_1(t_k), \dots, \hat{\tau}_{Nc+1}(t_k)]^T$, I is identity matrix $(Nc+1) \times (Nc+1)$, L is the Laplacian matrix depending on the connections of the communication graph including the MSO, and $\varepsilon = 1 - \rho_0$. In Equation (6), $\hat{\tau}_i(t_k^+)$ is the sum of all the contributes, evaluated according to Equation (5), of the software clocks of the SOs and MSO in the BR.

From Equation (4), because no communication is performed between two consecutive synchronization time instants t_k and t_{k+1} , the software clock values are:

$$\hat{\tau}(t_{k+1}) = \hat{\tau}(t_k^+) + 1_{(Nc+1) \times 1} \bar{\alpha}T_P + d(t_k), \quad (7)$$

where $d(t_k) = [d_1(t_k), \dots, d_{(Nc+1)}(t_k)]^T$. It is bounded as $\|d(t_k)\|_\infty < d_{\max}$. By Equation (6), Equation (7) becomes:

$$\hat{\tau}(t_{k+1}) = (I + \varepsilon L)\hat{\tau}(t_k) + 1_{(Nc+1) \times 1} \bar{\alpha}T_P + d(t_k), \quad (8)$$

According to graph theory [32,33], if the network graph is connected, the matrix $I + \varepsilon L$ has only one eigenvalue with value equal to 1, $\lambda_1 = 1$, while all the others are strictly lower than one ($|\lambda|_i < 1, i = 2, \dots, (Nc+1)$). In addition, the eigenvector associated to λ_1 is the unit vector v_1 with dimension $Nc+1$.

To evaluate the influence of the MSO on the synchronization accuracy, it is defined the synchronization delay between SO# i and SO# j as $y_{ij}(k) = C_{ij}\tau(t_k)$, where $C_{ij} = (e_i - e_j)^T$ and e_i, e_j are the i -th and j -th vectors of the canonical basis. Since $C_{ij}v_1 = 0$, i.e., an object is with delay zero respect to itself, the v_1 does not contribute to the synchronization delay, and, consequently, on the synchronization accuracy. Moreover, the eigenvalues λ_1 has no influence on the synchronization delay. Since all the other eigenvalues are strictly lower than 1, the maximum value of the synchronization delay is bounded with respect to $d_{\max}$, as follows:

$$\|y_{ij}(k)\|_\infty < l_1 d_{\max} \quad \forall k, \quad (9)$$

where $l_1 = \sum_{r=1}^{N_c} \sum_{k=0}^{\infty} |C_{ij}(I + \varepsilon L)^k e_r|$ is the 1-norm of the impulsive response matrix between the disturbance vector and the output y_{ij} [33].

Equation (9) shows that the maximum value of the synchronization delays, and consequently, the synchronization accuracy is linked in a proportional way with the parameter l_1 that is influenced by the second largest eigenvalue of the matrix, which depends upon the connectivity of the graph. The more the graph is connected, the lower is the l_1 value and, therefore, the influence of the disturbances on the synchronization accuracy. The MSO, by communicating with the SOs in its BR, creates new connections and, consequently, new path among SOs, increasing the connectivity of the graph.

Figure 2. Different routes for N = 25 lattice topology.

3.3. Reduction of Δ_{TIS} Value

Further effect of MSO concerns with the reduction of Δ_{TIS} . Indeed, the lower is $\|y_{ij}(k)\|_{\infty}$, the earlier the synchronization delay between any couple of SOs becomes lower of the admissible delay range. Consequently, more the graph is connected, the lowest is the Δ_{TIS} value. This result is also confirmed by Theorem#3 of [26], assessing that, if the MSO is used, the Δ_{TIS} is reduced with respect to the case of a static network, i.e., only SOs. This theorem is based on the comparison of the second eigenvalue of the matrix $I + \varepsilon L$ in the two-abovementioned cases. In particular, it is proved that, with the MSO, the second eigenvalue of the $I + \varepsilon L$ is higher with respect to the case the MSO is not used, then the second eigenvector is lower, i.e., the network converges faster.

Nevertheless, the quantitative analysis of the Δ_{TIS} reduction is still an open challenge [26].

4. Evaluation of the Effect of MSO on the IoT Network Synchronization

In order to evaluate the effect of the MSO on the synchronization, it is necessary to evaluate not only the increasing number of connections due to the newly introduced paths but also how such connections are spatially distributed. Indeed, in the extreme case that a SO is not linked to the network, if it is periodically visited by the MSO, as in [34], the

synchronization of the whole network is achieved. If the MSO visits only the connected SOs, it is still not possible to achieve the synchronization of all the SOs.

To evaluate the effect of the MSO route on the network connections, a lattice network topology is considered. In this topology, all the SOs (except the ones on the boundary) have the same number of connections. In this case, the MSO effect on the connections is evaluated by two statistical parameters: the mean and standard deviation of the minimum number of connections among any couple of nodes on the observing time window. In the following, node refers to both SOs and MSO. In particular, because the nodes exchange the SynMsg each T_p , the effect of the MSO on the connections among nodes is evaluated at each synchronization time t_k . By considering at t_k the network connection's adjacency matrix $A(t_k)$, with dimension $(Nc + 1) \times (Nc + 1)$, the average minimum number of connection among nodes μ_{con} , activated from the start of the synchronization procedure up to t_k , is:

$$\mu_{con} = \text{mean} \left(\text{DST} \left(\sum_{i=1}^k A(t_k) \right) \right), \quad (10)$$

The standard deviation is:

$$\sigma_{con}(t_k) = \text{st.dev.} \left(\text{DST} \left(\sum_{i=1}^k A(t_k) \right) \right), \quad (11)$$

The element $a_{ij}(t_k)$ of the matrix $A(t_k)$ is the number of connections between node#i and node#j at t_k as a consequence, the element in position i,j of the matrix $\sum_{i=1}^k A(t_k)$ is the number of connections between node#i and node#j available from the start of the synchronization procedure up to t_k .

The operator $\text{DST}()$ of Equations (10) and (11) receives as input the adjacency matrix and returns the matrix where the element dst_{ij} is the minimum distance between node#i and node#j. The operator $\text{mean}()$ of Equation (10) returns the mean value of all the element of the matrix. The operator $\text{st.dev.}()$ of Equation (11) returns the standard deviation of all the element of the matrix.

Because the synchronization parameters Δ_{TIS} and σ depend on (i) the network topology, (ii) the number of SOs, and (iii) the MSO's routes, numerical tests are designed to assess the theory in Section 3. In this way, the relationships among Δ_{TIS} , σ , $\mu_{con}(t_k)$, and $\sigma_{con}(t_k)$ [28] are highlighted. These relationships allow the establishment of the more convenient criteria in the design of the MSO route.

In the tests, the MSO visits the SOs in the predefined route. Since the MSO propagates the common sense of time at t_k , it estimates $\hat{\tau}_i(t_k^+)$ according to the synchronization parameters of the last visited SOs (see Section 2). It is assumed that, at t_k , the MSO (dashed node in Figure 2) is directly connected with the visited SO and the SOs are directly connected to the visited SO. Both MSO and SOs communicate in wireless broadcasting mode.

In order to easily understand the effect of the MSO on the SO connections, a square lattice topology with different values of Nc is considered.

The T_{clk} and $\hat{\tau}_i(t_0)$ values are selected, respectively, according to the hardware realization of the clock typically equipping commercial MSO and SO [30] and the time required to power-on all the SOs and the MSO. In particular, T_{clk} is selected equal to 1 ms with variation in the range $[0.999980, 1.000020]$ T_{clk} , $\hat{\tau}_i(t_0)$ in the range $[0, 30] \times 10^7 T_{clk}$. The observation time interval of each test is set equal to 10,000 s, and each node collects its own time interval each $T = 2 \sigma$ and sends a SynMsg each $T_p = 10$ s.

To evaluate the effect of the MSO, different MSO's routes are considered. Among all these routes, in the following, particular interest is devoted to two different sets.

One set that includes straight-line routes, the main diagonal (Route#1–3 of Figure 2), is selected, because it splits symmetrically the network. The selection of the second diagonal or the central row or central column would not give further information, since they are

a rotation of the main diagonal respect to the center of the matrix. Therefore, the results referred to these routes are not reported. Another set includes the routes selected with the following two criteria (Route#4–6 of Figure 2): (i) peripheral nodes of the network are visited, and (ii) the number of the node connections at each t_k is maximized.

Random topologies of the network are not considered in this research, because they make difficult to understand the relationship between the MSO's route and Δ_{TIS} , since Δ_{TIS} depends not only on the number of the visited nodes but also on their position in the network and the order with which MSO visits the nodes.

Let $N_{i,j}$, $i, j = 1, \dots, n$, the node of the square lattice topology, where $n = \sqrt{Nc}$. Imposing that at time t_k , MSO is on $N_{i,j}$, the following routes are selected:

- **Route#1:**

$$\begin{cases} i = j = k \bmod n & k \bmod n \neq 0 \\ i = j = n & k \bmod n = 0 \end{cases}, k \in N^+$$
- **Route#2:**

$$\begin{cases} i = j = V[p] \\ V = [1; n; 2; n-1; \dots; \lfloor \frac{n}{2} \rfloor; \lceil \frac{n}{2} \rceil + 1; \lceil \frac{n}{2} \rceil; \lceil \frac{n}{2} \rceil + 1; \lfloor \frac{n}{2} \rfloor; \dots; n-1; 2; n] \end{cases}, p = k \bmod \text{length}(V), k \in N^+$$
- **Route#3:**

$$\begin{cases} i = j = V[p] \\ V = [2; n-1; \dots; \lfloor \frac{n}{2} \rfloor - 2; \lfloor \frac{n}{2} \rfloor + 2; \lfloor \frac{n}{2} \rfloor; \lfloor \frac{n}{2} \rfloor + 2; \lfloor \frac{n}{2} \rfloor - 2; \dots; n-1] \\ V = [2; n-1; \dots; \lfloor \frac{n}{2} \rfloor; \lceil \frac{n}{2} \rceil; \lceil \frac{n}{2} \rceil; \dots; n-1] \end{cases}, p = k \bmod \text{length}(V), \begin{matrix} \lfloor \frac{n}{2} \rfloor \bmod(2) = 0 \\ \lceil \frac{n}{2} \rceil \bmod(2) \neq 0 \end{matrix}, k \in N^+$$
- **Route#4:**

$$\begin{cases} i = 2, & j = 2 & k \bmod(4) = 1 \\ i = 2, & j = n-1 & k \bmod(4) = 2 \\ i = n-1, & j = n-1 & k \bmod(4) = 3 \\ i = n-1, & j = 2 & k \bmod(4) = 0 \end{cases}, k \in N^+$$
- **Route#5:**

$$\begin{cases} i = 2, & j = 2 & k \bmod(8) = 1 \\ i = 2, & j = \frac{n}{2} & k \bmod(8) = 2 \\ i = 2, & j = n-1 & k \bmod(8) = 3 \\ i = \frac{n}{2}, & j = n-1 & k \bmod(8) = 4 \\ i = n-1, & j = n-1 & k \bmod(8) = 5 \\ i = n-1, & j = \frac{n}{2} & k \bmod(8) = 6 \\ i = n-1, & j = 2 & k \bmod(8) = 7 \\ i = \frac{n}{2}, & j = 2 & k \bmod(8) = 0 \end{cases}, k \in N^+$$
- **Route#6:**

$$\begin{cases} i = 2, & j = \frac{n}{2} & k \bmod(4) = 1 \\ i = \frac{n}{2}, & j = n-1 & k \bmod(4) = 2 \\ i = n-1, & j = \frac{n}{2} & k \bmod(4) = 3 \\ i = \frac{n}{2}, & j = n-1 & k \bmod(4) = 0 \end{cases}, k \in N^+$$

Figure 2 shows the routes in the case $Nc = 25$ and the SOs visited by MSO. The dashed arches are the temporary connections created by the MSO in the route.

To evaluate Δ_{TIS} , the admissible delay range equal to $100 T_{clk}$, is imposed; σ is evaluated on the last 1000 s of the observation time interval. This time interval is selected on the basis of preliminary tests to guarantee that σ is evaluated when the common sense of time among the SOs is achieved.

Figure 3 shows the percentage improvement of Δ_{TIS} and σ and percentage variation of μ_{con} and σ_{con} , with respect to the case of static topology.

Figure 3. Percentage improvement of Δ_{TIS} and σ and percentage variation of $\mu_{con}\%$ and $\sigma_{con}\%$, with respect to the case of static topology.

The numerical results show that, in all cases, the MSO increases the number of connections among SOs. Lower $\sigma_{con}\%$ corresponds with higher improvement in σ . Concerning Δ_{TIS} , according to the theory, the MSO improves its values, but the quantitative analysis is still an open challenge, and it will be studied in future research.

Figure 4 shows the trend of the delay between node#25 taken as a reference and all the others in the case Figure 4a static topology and Figure 4b MSO moving according to route#4.

Figure 4. Trend of the delay among nodes in the case $N_c = 49$ and (a) static topology, and (b) MSO is moving according to route#4.

By analyzing the numerical data, Δ_{TIS} and σ are reduced of the 35% and 36%, respectively.

For sake of completeness, a random movement of the MSO is considered, and Figure 5 shows the trend of the delay among nodes in the case $N_c = 49$ and (a) static topology, and (b) MSO is randomly moving. In particular, the position of the MSO is established, time by time, by extracting an integer number in the range 1–49. As shown in Figure 5, as foreseen from theory, this random path gives better results with respect the other paths. In

particular, Δ_{TIS} is equal to 1400 s, and σ is equal to 10 T_{clk} . Such result is justified by the fact that the probability that the MSO links far nodes is increased.

Figure 5. Trend of the delay among nodes in the case $N_c = 49$ and (a) static topology, and (b) MSO is randomly moving.

5. Experimental Tests and Discussion

Experimental tests were carried out to evaluate the effect of MSO in a real scenario.

The experimental testbed consisted of 23 Crossbow TelosB nodes, 30 xm1000 nodes and a PC. In particular, 21 TelosB nodes (indexed from 1 to 21) and all the xm1000 nodes were used to implement the synchronization algorithm. The TelosB node (indexed as 101) was plugged into the PC and used as ZigBee/USB protocol converter. The MSO is implemented in a xm1000 node indexed as 50. The TelosB node (indexed as 102) was used to trigger the 50 nodes to send the service messages containing their identification number id_i , the software clock frequency $\hat{\alpha}_i(t)$, the software clock offset $\hat{\delta}_i(t)$, and the hardware clock τ_i value to the ZigBee/USB protocol converter (and, in turn, to the PC) at sampling period $T = 2$ s. The data collected by the PC was used to evaluate the synchronization accuracy in terms of discrepancy among the software clocks $\hat{\tau}_i(t)$ for all $i \in [1, 50]$. Figure 6 shows the actual realization of the testbed.

In the TelosB [32] and xm1000 [35], the microcontroller is a TIMSP430 featuring a Digitally Controlled Oscillator (DCO) running at 8 MHz with a clock period $T_{DCO} = 0.125 \mu\text{s}$ and 10 kbyte of RAM.

Furthermore, it also features an External Crystal Oscillator (ECO) running at 32,768 Hz. Regarding the radio equipment, the nodes feature the IEEE 802.15.4/ZigBee-compliant module equipped with a RF transceiver operating within the range 2.4000–2.4835 GHz, compatible with ISM band, which allows a data rate of 250 kbps. An important feature of the radio chip CC2420 is the MAC-layer timestamp capability. This allows each node to read the local clock at the beginning of the transmission or reception of the start frame delimiter (SFD) of a message. This mechanism strongly reduces the random delays introduced by the transmission and the readings of the synchronization messages. Indeed, this mechanism was used in [36] to support their major assumption, i.e., communication delays are negligible with respect to T_{clk} . The adoption of the precision tag “TMilli” giving a millisecond resolution of T_{clk} was not cosmetic, as there are several hardware platforms, such as the Memsic’s IRIS mote [37], for which this represents the only option available. Thus, this reflects a realistic operating condition in several application contexts.

Figure 6. Experimental testbed.

TelosB and xm1000 motes have been programmed by TinyOS, an open-source operating system specifically designed for WSN [38].

In order to allow SOs and MSO to periodically send their clock time values to Node#101 and then to the PC, all them must be in the Node#101 BR [28,31,36]. To this aim, the nodes were deployed in a region of approximately one square meter. The high spatial density of nodes was chosen to evaluate the effectiveness of the proposed algorithm in a saturated spectrum network. For each node i -th, the neighborhood was predefined in order to achieve a lattice topology.

As a consequence, the lattice topology and the movement of the MSO are realized by defining in each SOs and in the MSO a time variant Boolean vector $ND_i(t_k)$ of the neighbors with $i = 1 \dots (Nc + 1)$. If $ND_i(t_k)[j] = 1$, the j -th node, at time t_k , is in the neighborhood of the i -th node. The MSO routes are implemented by changing the update rules of $ND_i(t_k)$.

The following parameters were used: $\rho_v = 0.9$ and $\rho_o = 0.9$. These are memory factors [36], and then their values influence the speed with which the whole network converges to a common sense of time. In order to compare numerical and experimental tests, ρ_v and ρ_o are the same for all the tests.

Figure 7 shows the trend of the delay between SO#25 as reference and all other nodes of the network in the case (a) static topology, and (b) MSO moves according to the route#4.

By comparing Figure 7 with Figure 4, the same trend of the delay among SOs can be noted. The reduction of the delay among nodes shows the robustness of the consensus to the packet loss, as proved in [28,36].

By analyzing the experimental data, Δ_{TIS} and σ are reduced of the 28% and 38%, respectively.

Figure 7. Experimental results of the delay among nodes in the case $N_c = 49$ and, (a) static topology, and (b) MSO moves according to the route#4.

6. Conclusions

To overcome the convergence challenge to a common sense of time in the case of a Mobile Smart Object (MSO), in the paper, a novel synchronization algorithm based on consensus is proposed. This challenge is introduced by the fact that an MSO makes the network topology time varying, provoking the continuous evaluation of the logical tree to spread the sense of time. The consensus does not need logical tree, and, in this paper, the proof of the convergence was given for the proposed algorithm. Theoretical results, here presented, have highlighted that the MSO boosts the time synchronization of the network. This result is logically justified by the fact that the MSO brings the common sense of time among SOs by increasing their connections according to its path. The effects of the MSO's route on the statistical parameters describing the number of connections during the time and then on the values of σ and the Δ_{TIS} is also investigated both by numerical and experimental tests that give compatible results and confirm the theoretical findings. Consequently, the criteria to select the route reducing σ and the Δ_{TIS} are given. Ongoing activities are concerned with the investigation of the effect of the distance among nodes, so as of the speed of the MSO in reaching SOs, on the ameliorant of the σ and the Δ_{TIS} values. Moreover, the case of multiple MSOs cooperating in bring the common sense of the time to the SOs will be also taken into account.

Author Contributions: Conceptualization, F.L.; methodology, F.L.; software, F.L. and D.L.C.; validation, F.L. and D.L.C.; formal analysis, F.L. and D.L.C.; investigation, F.L. and D.L.C.; resources, F.L.; data curation, F.L. and D.L.C.; writing—original draft preparation, F.L. and D.L.C.; writing—review and editing, F.L. and D.L.C.; visualization, F.L. and D.L.C.; project administration, F.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ferrari, P.; Flammini, A.; Sisinni, E.; Rinaldi, S.; Brandao, D.; Rocha, M.S. Delay Estimation of Industrial IoT Applications Based on Messaging Protocols. *IEEE Trans. Instrum. Meas.* **2018**, *67*, 2188–2199. [\[CrossRef\]](#)
2. Lu, X.; Liu, J.; Zhao, H. Collaborative target tracking of IoT heterogeneous nodes. *Measurement* **2019**, *147*, 106872. [\[CrossRef\]](#)
3. Geller, B. Advanced synchronization techniques for the Internet of Things. In Proceedings of the 2016 International Symposium on Signal, Image, Video and Communications (ISIVC), Tunis, Tunisia, 21–23 November 2016; pp. 180–184. [\[CrossRef\]](#)

4. Yadav, P.; McCann, J.A.; Pereira, T. Self-Synchronization in Duty-Cycled Internet of Things (IoT) Applications. *IEEE Internet Things J.* **2017**, *4*, 2058–2069. [\[CrossRef\]](#)
5. Pegoraro, P.A.; Meloni, A.; Atzori, L.; Castello, P.; Sulis, S. PMU-Based Distribution System State Estimation with Adaptive Accuracy Exploiting Local Decision Metrics and IoT Paradigm. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 704–714. [\[CrossRef\]](#)
6. Li, Y.; Xu, Y.; Liu, Z.; Hou, H.; Zheng, Y.; Xin, Y.; Zhao, Y.; Cui, L. Robust detection for network intrusion of industrial IoT based on multi-CNN fusion. *Measurement* **2020**, *154*, 107450. [\[CrossRef\]](#)
7. Hu, Q.; Ohata, E.F.; Silva, F.H.; Ramalho, G.L.B.; Han, T.; Filho, P.P.R. A new online approach for classification of pumps vibration patterns based on intelligent IoT system. *Measurement* **2020**, *151*, 107138. [\[CrossRef\]](#)
8. Mo, L.; Cao, X.; Song, Y.; Kritikakou, A. Distributed Node Coordination for Real-Time Energy-Constrained Control in Wire-less Sensor and Actuator Networks. *IEEE Internet Things J.* **2018**, *5*, 4151–4163. [\[CrossRef\]](#)
9. Sangeetha, A.L.; Bharathi, N.; Ganesh, A.B.; Radhakrishnan, T.K. Particle swarm optimization tuned cascade control system in an Internet of Things (IoT) environment. *Measurement* **2018**, *117*, 89–90. [\[CrossRef\]](#)
10. Pawar, P.; TarunKumar, M. An IoT based Intelligent Smart Energy Management System with accurate forecasting and load strategy for renewable generation. *Measurement* **2020**, *152*, 107187. [\[CrossRef\]](#)
11. Rizzi, M.; Ferrari, P.; Flammini, A.; Sisinni, E. Evaluation of the IoT LoRaWAN Solution for Distributed Measurement Applications. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 3340–3349. [\[CrossRef\]](#)
12. Akbar, A.; Kousiouris, G.; Pervaiz, H.; Sancho, J.; Ta-Shma, P.; Carrez, F.; Moessner, K. Real-Time Probabilistic Data Fusion for Large-Scale IoT Applications. *IEEE Access* **2018**, *6*, 10015–10027. [\[CrossRef\]](#)
13. Nesa, N.; Banerjee, I. IoT-Based Sensor Data Fusion for Occupancy Sensing Using Dempster–Shafer Evidence Theory for Smart Buildings. *IEEE Internet Things J.* **2017**, *4*, 1563–1570. [\[CrossRef\]](#)
14. Mois, A.G.; Sanislav, T.; Folea, S.C. A Cyber-Physical System for Environmental Monitoring. *IEEE Trans. Instrum. Meas.* **2016**, *65*, 1463–1471. [\[CrossRef\]](#)
15. Lee, H.; Ke, K. Monitoring of Large-Area IoT Sensors Using a LoRa Wireless Mesh Network System: Design and Evaluation. *IEEE Trans. Instrum. Meas.* **2018**, *67*, 2177–2187. [\[CrossRef\]](#)
16. Celesti, A.; Lay-Ekuakille, A.; Wan, J.; Fazio, M.; Celesti, F.; Romano, A.; Bramanti, P.; Villari, M. Information management in IoT cloud-based tele-rehabilitation as a service for smart cities: Comparison of NoSQL approaches. *Measurement* **2020**, *151*, 107218. [\[CrossRef\]](#)
17. Jabbar, S.; Malik, K.R.; Ahmad, M.; Aldabbas, O.; Asif, M.; Khalid, S.; Han, K.; Ahmed, S.H.; Aldabbas, O.; Han, K.J.; et al. A Methodology of Real-Time Data Fusion for Localized Big Data Analytics. *IEEE Access* **2018**, *6*, 24510–24520. [\[CrossRef\]](#)
18. Mois, G.; Folea, S.; Sanislav, T. Analysis of Three IoT-Based Wireless Sensors for Environmental Monitoring. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 2056–2064. [\[CrossRef\]](#)
19. Wan, J.; Tang, S.; Hua, Q.; Li, D.; Liu, C.; Lloret, J. Context-Aware Cloud Robotics for Material Handling in Cognitive Industrial Internet of Things. *IEEE Internet Things J.* **2017**, *5*, 2272–2281. [\[CrossRef\]](#)
20. Younan, M.; Housseina, E.H.; Elhoseny, M.; Ali, A.A. Challenges and recommended technologies for the industrial internet of things: A comprehensive review. *Measurement* **2020**, *151*, 107–198. [\[CrossRef\]](#)
21. Yanmaz, E.; Quaritsch, M.; Yahyanejad, S.; Rinner, B.; Hellwagner, H.; Bettstetter, C. Communication and coordination for drone networks, Ad Hoc Networks, Lecture Notes of the Institute for Computer Sciences. *Soc. Inform. Telecommun. Eng.* **2016**, *184*, 79–91.
22. Carvajal-Ramírez, F.; Navarro-Ortega, A.D.; Agüera-Vega, F.; Martínez-Carricondo, P.; Mancini, F. Virtual reconstruction of damaged archaeological sites based on Unmanned Aerial Vehicle Photogrammetry and 3D modelling. Study case of a southeastern Iberia production area in the Bronze Age. *Measurement* **2019**, *136*, 225–236. [\[CrossRef\]](#)
23. Yang, P.; Stankevicius, D.; Marozas, V. Lifelogging Data Validation Model for Internet of Things Enabled Personalized Healthcare. *IEEE Trans. Syst. Man Cybern. Syst.* **2018**, *48*, 50–64. [\[CrossRef\]](#)
24. Vellappally, S.; Al Kheraif, A.A.; Anil, S.; Wahba, A.A. IoT medical tooth mounted sensor for monitoring teeth and food level using bacterial optimization along with adaptive deep learning neural network. *Measurement* **2019**, *135*, 672–677. [\[CrossRef\]](#)
25. Fouad, H.; Hassanein, A.S.; Soliman, A.M.; Al-Feel, H. Analyzing patient health information based on IoT sensor with AI for improving patient assistance in the future direction. *Measurement* **2020**, *159*, 107757. [\[CrossRef\]](#)
26. Duan, X.; He, J.; Cheng, P.; Chen, J. Exploiting a Mobile Node for Fast Discrete Time Average Consensus. *IEEE Trans. Control Syst. Technol.* **2016**, *24*, 1993–2001. [\[CrossRef\]](#)
27. Lamonaca, F.; Carni, D.L.; Grimaldi, D.; Sciammarella, P. Mobile object to speed up the synchronization of IoT network. In Proceedings of the IEEE International Conference on Measurements and Networking, Naples, Italy, 27–29 September 2017; pp. 1–6.
28. Garone, E.; Gasparri, A.; Lamonaca, F. Clock synchronization protocol for wireless sensor networks with bounded communication delays. *Automatica* **2015**, *59*, 60–72. [\[CrossRef\]](#)
29. Lamonaca, F.; Carni, D.L.; Riccio, M.; Grimaldi, D.; Andria, G. Preserving Synchronization Accuracy from the Plug-in of Non-Synchronized Nodes in a Wireless Sensor Network. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 1058–1066. [\[CrossRef\]](#)
30. Telosb Mote Platform, Crossbow. Available online: https://www.willow.co.uk/TelosB_Datasheet.pdf (accessed on 6 June 2021).
31. Lamonaca, F.; Gasparri, A.; Garone, E.; Grimaldi, D. Clock Synchronization in Wireless Sensor Network with Selective Convergence Rate for Event Driven Measurement Applications. *IEEE Trans. Instrum. Meas.* **2014**, *63*, 2279–2287. [\[CrossRef\]](#)

32. Diestel, R. *Graph Theory*, 5th ed.; Heidelberg Graduate Texts in Mathematics; Springer: Berlin/Heidelberg, Germany, 2017; Volume 173.
33. Dahleh, M.; Dahleh, A.; Verghese, G.C. *Chapter 15—External Input Output Stability (Lectur Notes MIT Course)*; MIT Press: Cambridge, MA, USA, 1999.
34. Lamonaca, F.; Grimaldi, D.; Morello, R.; Nastro, A. Sub- μ s Synchronization Accuracy in Distributed Measurement System by PDA and PC triggers realignment. In Proceedings of the IEEE International Instrumentation and Measurement Technology Conference, Minneapolis, MN, USA, 6–9 May 2013; pp. 801–806.
35. Advanticsys-Model AS-XM1000-Mote Module. Available online: <https://www.environmental-expert.com/products/advanticsys-model-as-xm1000-mote-module-262761> (accessed on 6 June 2021).
36. Schenato, L.; Fiorentin, F. Average timesynch: A consensus-based protocol for clock synchronization in wireless sensor networks. *Automatica* **2011**, *47*, 1878–1886. [[CrossRef](#)]
37. Memsic, IRIS—Wireless Measurement System Datasheet. Available online: http://www.nr2.ufpr.br/~adc/documentos/iris_datasheet.pdf (accessed on 6 June 2021).
38. Kim, D.; Nam, H.; Kim, D. Adaptive Code Dissemination Based on Link Quality in Wireless Sensor Networks. *IEEE Internet Things J.* **2017**, *4*, 685–695. [[CrossRef](#)]