PULSE COUPLED OSCILLATORS' PRIMITIVE FOR LOW COMPLEXITY SCHEDULING

Y.-W. Peter Hong¹, Anna Scaglione², and Roberto Pagliari³

¹ Institute of Communications Engineering, National Tsing Hua University, Taiwan
² Department of Electrical and Computer Engineering, UC Davis, USA
³ Department of Electrical and Computer Engineering, Cornell University, USA

ABSTRACT

Pulse coupled oscillators (PCOs) are pulsing devices that pulse individually in a periodic manner but alter their pulsing patterns in response to the pulsing of other nodes. A network of PCOs can produce a number of different dynamics from their pulsing activities, among which the synchrony of pulsing is perhaps the most well known. In this paper, we study the primitive that falls into the class of "desynchronization". Specifically, we propose a simple pulsecoupling mechanism that allows each node in the network to converge to a desynchronized state where the nodes will pulse periodically with a constant spacing among each others firing times. We discuss the convergence of the PCO mechanism and propose to apply this primitive to resolve contention in the reservation phase of a reservation-based MAC protocol.

Index Terms— pulse-coupled oscillators, medium access control, synchronization, scheduling.

1. INTRODUCTION

Pulse coupled oscillators (PCOs) are pulsing elements that pulse individually in a periodic manner and alter their pulsing patterns in response to the pulsing heard from other elements (hence the adjective "coupled"). The pulse emission is often referred as a *firing* event. In particular, the recipient of a pulse (or coupling) moves earlier or later its own firing by altering a local state that regulates the pulse emission of the node. It is well known that networks of PCOs can produce a variety of pulsing patterns, among which the synchrony of pulsing is the most well-known [1]. These patterns are often used to model the dynamics of spiking neurons in the brain [2].

The use of the PCO model as a powerful primitive for wireless networks' synchronization has been proposed by the authors in [3]. In this paper, we utilize the PCO network's ability to generate a variety of pulsing patterns to solve one of the most difficult tasks in wireless networks – multiplexing of communication resources. We ask: "*Can the PCO dynamic be changed so as to have a fixed point where the*

nodes fire the desired multiplexing schedule?" Solutions to this question were first investigated in [4], where the authors found that it was possible to enforce in a decentralized fashion a fair Time Division Multiple Access (TDMA) schedule which would assign to each node 1/n-th of the frame duration, where n denotes the number of nodes in a fully connected network. Specifically, pulse-coupling mechanisms were designed such that the nodes will converge to a desynchronized state where they pulse periodically with a constant spacing of 1/n among each others' firings. The authors in [4] proposed two PCO protocols to achieve the fair TDMA schedule: 1) *Desync*, where the nodes' state-update depends only on the neighboring firing events; and 2) *Inverse-MS*, which is essentially the same PCO technique proposed for synchrony in [3], except with a negative coupling.

Here we propose an alternative method to the simplest of the two strategies, the *Inverse-MS* which, as discussed in [4], can only achieve *weak convergence*. Our first contribution in this paper is to incorporate in the dynamics of each node a parameter which can be adapted to capture the number of nodes in the network. This method allows for a stable convergence (i.e. *strong convergence*) to the desired schedule when the parameter matches or is greater than the number of active nodes. Secondly, we describe how this pulsing mechanism can be used to perform contention in the reservation phase of a reservation-based MAC protocol.

2. A PCO PRIMITIVE FOR DESYNCRHONIZATION

Consider a network of n sensors that are fully-connected via direct transmission links. When in isolation, each sensor pulses periodically with period T, which we refer to as the *firing cycle*. However, in an interconnected network, the pulsing serves as the basis of interaction among sensors and, with properly designed local coupling rules, allows the network to attain desirable emergent properties.

We assume that, similar to CSMA, the sensors are able to sense the transmission of their peers and react to the event by adjusting their local pulsing times. Unlike CSMA, however, the procedure has memory which is stored into the state vari-

Emails: ywhong@ee.nthu.edu.tw, ascaglione@ucdavis.edu, and rp294@cornell.edu. This research was supported in part by the National Science Council, Taiwan, under grant NSC-95-2221-E-007-043- MY3, NSC-96-2628-E-007-012-MY2, and NSC-97-2221-E-007-037.

able of the PCO clock. This is done as follows. Each sensor, say sensor *i*, has a PCO clock corresponding to the variable $\phi_i(t)$, which represents the position of the PCO clock in [0, 1) at the absolute time *t*. We refer to these as the nodes' state variables $\phi_i(t)$, i = 1, ..., n, where $0 \le \phi_i(t) < 1$. In the absence of coupling, the state variable of each node, say node s_i , evolves as

$$\phi_i(t) = (\Phi_i + t/T) \mod 1.$$

where $\Phi_i \in (0, 1)$ is an initial random value. That is, a node with phase $\phi_i(t)$ indicates that it is scheduled to fire after time $T - \phi_i(t)T$. When the phase reaches 1, the node fires and resets its phase to 0. The firing will cause an instantaneous coupling to all other nodes in the network. Let t_p be the *p*-th firing event time. In describing the protocol we assume that the pulse is sensed perfectly and instantaneously by all nodes. However, in practice, channel noise, finite pulse durations, and propagation delays may affect detection of the pulse or the correct coupling time, which will be described briefly in Section 2.1. Without loss of generality let us assume that the initial phases Φ_i are in descending order, so that node 1 is the first firing and the rest of the nodes follows in increasing order of the indexes.

Suppose that a firing occurs at time t_p and that $\phi_i(t_p)$ and $\phi_i(t_p^+)$ are the phases of a receiving node i at time t_p before and after a coupling is imposed. Every time a node fires, node i will update its phase as

$$\phi_i(t_p^+) = \begin{cases} f^{-1} \left(f \left(\phi_i(t_p) \right) + \varepsilon \right), & 0 \le \phi_i(t_p) < \frac{1}{n_0}; \\ \phi_i(t_p), & \text{else.} \end{cases}$$
(1)

or

$$\phi_i(t_p^+) = \begin{cases} 1 - f^{-1}(f(1 - \phi_i(t_p))) + \varepsilon), & 1 - \frac{1}{n_0} < \phi_i(t_p) \le 1 \\ \phi_i(t_p), & \text{else.} \end{cases}$$
(2)

where $n_0 \geq 2$ is a parameter that controls the spacing between nodes and f(x) is a function that is convex and monotonically grows from 0 to ∞ in $[0, 1/n_0]$ (see Fig.1). $\varepsilon \in$ $(0, \infty)$ is the coupling strength. Specifically, the speed of convergence increases with ε .

The update rule in (1) is such that, when node *i* hears a pulse at time t_p that is less than T/n_0 (i.e. $1/n_0$ of a firing cycle) away from its last firing, node *i* will increase its state variable by an amount ε so that its firing in the next cycle is pushed earlier and further away from the firing of the current node. On other hand, in (2), when node *i* hears a pulse at time t_p that is less than T/n_0 away from its scheduled firing, it will push its firing to a later instant in time. However, if a node is expected to fire beyond T/n_0 of the current firing, then it will not adjust its phase since it is safely beyond the minimum required spacing (assuming that it is T/n_0). By employing the update rule of (1) or (2) (or both), the spacing between firings will be pushed apart, but by no more than T/n_0 .

We can distinguish between the states of *strict* and *weak* desynchronization as follows:



Fig. 1. An example of the PCO dynamics and update.

Definition 1: The network is *strictly desynchronized* after *t* if consecutive firing events are always separated by a constant interval and the firing time of each node within each firing cycle remains the same.

Definition 2: It is *weakly desynchronized* after t if consecutive firing events are always separated by a constant interval but the firing time within each firing cycle may be shifting.

In the proposed protocol, three cases may occur.

<u>Case I</u>: If the number of nodes $n > n_0$, the network will be *weakly synchronized*. In this case, even though the separation of consecutive firings remains constant, the firing time does not remain fixed in each firing cycle.

<u>Case II</u>: If $n = n_0$, there is only one fixed point for the protocol, which corresponds to have all nodes separated by T/nwith a stable firing pattern.

<u>Case III</u>: If $n < n_0$, then there is an entire interval of possible fixed points, all leaving a temporal gap between two nodes which is at least $\geq T/n_0$.

The most interesting aspect of these cases is the fact that the dynamics tend to force the nodes to be separated by T/n_0 , but if $n > n_0$ that is physically impossible. When n and n_0 are very different the nodes keep moving their firing time without rest, even though they eventually will tend to move in a pattern that makes the separation among pulses constant (a fixed point). The closer is n_0 to n, the greater are the odds of being strictly synchronized (see Section 4). We would like to remark that, based on the number of pulses heard, the correct number of nodes within the network can be correctly adjusted, and updated in view of departures and new nodes joining the system.

An example of f in the PCO update rules (1) and (2) is

$$f(\phi) = -\ln\left(c(1/n_0 - \phi)\right), c > 0. \tag{3}$$

In particular, if we consider only the coupling for $\phi_i(t_p) \in (1 - \frac{1}{n_0}, 1]$ and by setting $n_0 = 1$, c = 1 and $\epsilon = -\ln(1 - \alpha) > 0$, we have equivalently the dynamics of the *Inverse-MS* [4]. Hence, this shows that the PCO desynchronization primitive described above subsumes the *Inverse-MS* protocol. Note that precisely because $n_0 = 1 < n$, *Inverse MS* can only weakly synchronize, but strict synchronization is possible with a simple change in dynamics.

An alternative model with concave dynamics and positive coupling, as shown in Figure 1, is given by:

$$f(\phi) = \tan(0.5\pi n_0 \phi).$$
 (4)

We will refer to the models (3) and (4), inspired by the FireFly synchronization mechanism, to as DEFLYLOG and DEFLY-TAN, respectively.

2.1. Practical Issues

In the proposed strategy, the interaction among nodes is achieved through the emission and reception of short pulses. Perfect coupling is attained when the pulses have zero pulse duration, i.e. delta functions, or when the channel is noiseless. In practice, the actual coupling can be carried out by having each node transmit a pulse p(t) with root-mean square (rms) bandwidth β , which has good time localization properties.

Noise causes false alarms and missed detections as well as time of arrival estimation errors. The false detection of a pulse may cause a node to wrongly shift to a new phase according to the dynamics of the function f. The effect of this error is determined by the convexity of the function f and the greater the convexity the greater the error. Missed detection simply delay the convergence. In general, given that false alarms are error sources, a detector based on Newman Pearson (fixed false alarm) criterion seems the most appropriate. Concurrently, the node can estimate the arrival time. Assuming a deterministic frequency flat channel with gain $|h_{ij}|$ and perfect detection, the Cramer-Rao lower bound (CRLB) of time estimation is $\frac{N_0}{2|h_{ij}|^2 E_p \beta}$ [6], where E_p is the energy of the pulse. We can see that the estimation error reduces as the energy or the bandwidth of the pulse increases. This poses a tradeoff between the bandwidth of the system (or the length of the reservation period) and the accuracy of the schedule. Propagation delays also affect the convergence. However, if T >> p(t) this effect can be neglegted.

3. A PCO-BASED MAC MECHANISM

For PCO to be functional, some amount of idle time has to be left between the pulses that are fired. The model clearly relies on having signals concentrated in time relative to the period. If these signals were data packets, the model essentially stipulates the absence of collisions from the start and spacing these non colliding packets equally in time has no real purpose. On the other hand, waiting for the PCO to converge, as proposed by [4] is in many ways problematic, because it does not acknowledge the need to adapt to a changing number of nodes trying to access the medium. Hence, the question that we briefly address in this section is how to negotiate the transmission of data for a prolonged period.

To utilize the PCO primitive for multiple access, a reasonable option is to dedicate to it a fraction of time at the beginning of each frame (i.e. a reservation phase) while leaving the rest of the frame (i.e. the transmission phase) for data transmission, according to a schedule proportional to what is agreed upon in the reservation phase through the spacing of



Fig. 2. DEFLYTAN: (Bottom) Phase difference for $n_0 = 7$ and $\varepsilon = 5$ (black), $\varepsilon = .5$ (blue), $\varepsilon = .05$ (dotted black). (Top) Firing times of the nodes within each firing cycle.

the firing events. There is a vast literature that discusses reservation based MAC strategies (see e.g. [5]). Typically, these protocols use a CSMA access policy on the reservation phase of the frame transmitting mini-packets. Then, the nodes transmit proportionally longer packets containing their data in the dedicated part of the frame. The PCO primitive can be used in a similar fashion, as a reservation mechanism. The main advantages of PCO are architectural, because PCO requires an extremely elementary logic and circuitry to work.

Normally in reservation-based MAC protocols a node that needs to transmit and that is successful in reserving its slot transmits only once. When using the PCO mechanism, this approach does not make sense. In fact, because PCO allows to reach a fair schedule among the nodes after a few iterations, the nodes cannot stick with the schedule that they obtain at their first attempt; they should, instead, divide their payload in several sub-packets, each containing a fraction of the data that is proportional to the time between their firing and the next firing in the reservation phase. If the same group of sources will persist sufficiently, their data will be served equally. Additions of new nodes and departures will be gracefully accomodated. Unfortunately, the greater the fragmentation the greater is the overhead. The study of this trade-off, however, goes beyond the scope of this paper.

4. NUMERICAL RESULTS

In this section, we study the performance of the proposed PCO-based primitives under two different PCO dynamics, namely, DEFLYLOG and DEFLYTAN described in (3) and (4), respectively. In the simulations, we consider only the update rule in (1) since the other rules perform similarly.

Case 1 : $n_0 < n$ **:**

When $n_0 < n$, the nodes in the network will only be able to reach *weak synchronization* as mentioned in Section 2. To see this, we show in Fig. 2 the evolution of the difference in firing times between each node and node 1 for DEFLYTAN



Fig. 3. Accuracy of desired schedule with respect to ε , with n = 10 and different values of n_0 , after 50 firing cycles.

scheme. That is, with $t_{pi}^{(m)}$ being the *m*-th firing time of node i, we plot $\Delta_i[m] = t_{pi}^{(m)} - t_{p1}^{(m)}$ with respect to the rounds of firing *m*. Here, we set $n_0 = 7$ and n = 10. We can see that, in fact, the spacing between neighbors' firings converge to a constant value. However, the convergence state and the speed of convergence may depend on the coupling strength ε . Moreover, even though the spacing is fixed, it is not guaranteed that the exact firing time of a node within each firing cycle (i.e. within each length-*T* interval) may remain fixed. To confirm this, we reported, in the upper right corner of the same figure, the absolute firing time of each node within each length-*T* interval, for $\varepsilon = 0.5$. We can observe that the exact firing time in each firing cycle does not remain fixed, even though the phase difference between the nodes tends to be constant. The DEFLYLOG scheme behaves similarly and thus the plot is omitted due to space limitations.

In order to achieve a fair and constant scheduling in each frame, the firing of nodes should converge to a state where the spacing is equal to T/n for any neighboring firings. When this is achieved, the absolute firing time in each cycle will remain fixed, i.e. strong convergence is reached. To observe the system's ability to achieve this state, we consider the MSE metric $MSE[m] = E \left[\sum_{i=1}^{n} (T/n - \Delta_i[m])^2 \right]$. In Fig. 3, we plot the MSE as a function of the coupling strength ε for both DEFLYTAN and DEFLYLOG with $n_0 = 3, 5, 7, 9$ (< n = 10), after 50 firing cycles. We can see that an optimal value of ε exists but varies for different values of n_0 . Moreover, the error tends to be smaller as n_0 becomes closer to n.

We would like to remark that the Inverse MS scheme proposed in [4] corresponds to DeFlyLOG with $n_0 = 1$. In this case, only weak convergence can be attained and the convergence depends greatly on the coupling strength ε .

Case 2: $n_0 = n$ **:**

For $n_0 = n = 10$, we shown in Fig. 4 (top) the evolution of the phase differences and the firing times of the nodes for DEFLYLOG, with $\varepsilon = 5$. We see that, as time evolves, the exact firing time of a node within each firing cycle (i.e. within



Fig. 4. DEFLYLOG for n = 10. Top: phase differences (left) and firing times (right) with $n_0 = 10$. Bottom: phase differences (left) and firing times (right) with $n_0 = 20$.

each length-T interval) remains fixed after a sufficient number of iterations, and the nodes achieve the desired T/n spacing. **Case 3:** $n_0 > n$:

In Fig. 4 (bottom), we consider the case where $n_0 = 20$ while n = 10 for DEFLYLOG. Again, the network converges to a state where a spacing of at least T/n_0 is guaranteed for all nodes, and the firing times of the nodes within each firing cycle tend to a constant value. We would like to remark that, in our experiments, we also observe that the convergence occurs faster for the same coupling strengths when n_0 is larger since the nodes get pushed more easily to the boundary of T/n_0 , but this is omitted due to space limitations. As shown in Fig. 4, strong convergence is achieved for $n_0 \ge n$.

5. REFERENCES

- R. Mirollo and S. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM J. Appl. Math.*, vol. 50, no. 6, pp. 1645–1662, 1990.
- [2] F. C. Hoppensteadt and E. M. Izhikevich, "Weakly connected neural networks," *Springer*, 1997.
- [3] Y. Hong and A. Scaglione, "A scalable synchronization protocol for large scale sensor networks and its applications," *JSAC*, vol. 23, no. 5, pp. 1085–1099, 2005.
- [4] A. Patel, J. Degesys, and R. Nagpal, "Desynchronization: The theory of self-organizing algorithms for round-robin scheduling," in *IEEE Conference on Self-Adaptive and Self-Organizing Systems (SASO)*, 2007.
- [5] S. Kumar, V. S. Raghavan, and J. Deng, "Medium access control protocols for ad hoc wireless networks: A survey," *Elsevier*, 2004.
- [6] V. Poor, "An introduction to signal detection and estimation," Springer, 1994.