# Poster Abstract: A switched control scheme to handle quantisation in the design of high-precision computing system components

Federico Terraneo, Alberto Leva, Maria Prandini
DEIB – Politecnico di Milano, Italy
{federico.terraneo, alberto.leva, maria.prandini}@polimi.it

Over the last years, several problems related to computing systems are being re-considered with a control-centric approach, leading to a system-theoretical component design and assessment, [2] and [4]. Notable examples are thread scheduling [3], memory management [7], and time synchronisation [1]. When addressed as control ones, many of said problems reveal a surprisingly similar structure, where the dynamics to be controlled is very simple, and the only source of uncertainty are exogenous disturbances caused by the external environment, other components of the computing system, or any combination thereof. Whenever the quantised nature of controls and measurements is negligible, standard control structures can be devised to compensate for disturbances. However, in high-precision applications, quantisation may become so relevant to cause undesired fluctuations in the controlled variables, and the same structures may not be adequate anymore. Starting from time synchronisation in Wireless Sensor Networks (WSN), a general technique is proposed to solve this issue.

For a long time, the problem of WSN synchronisation has been viewed essentially as the composition of two: *clock synchronisation*, which is the act of making the node clocks instantaneously agree with a reference one, and *skew compensation*, which means counteracting the node oscillators' discrepancies so as to maintain clock agreement in between the synchronisation actions. In the FLOPSYNC scheme [5], an entirely control-based approach to WSN synchronisation, there is no distinction between clock synchronisation and skew compensation. Rather, each node hosts a discrete-time Linear, Time-Invariant (LTI) controller that attempts to zero the synchronisation error, measured as the difference between the expected and the actual arrival times, counted in the node local clock, of special packets periodically flooded by the master node. FLOPSYNC does not require timestamp transmission which results in a very low traffic overhead. In turn, the precision achieved by FLOPSYNC has triggered another problem related to the input/output signals quantization.

Consider a WSN in which one node (hereinafter, the *master* node) sends out a special packet with a fixed period $T$, known to all the others (the *slave* nodes). The master holds the reference clock for the WSN, thus the period $T$ is assumed to be kept exactly. Synchronisation in FLOPSYNC is achieved with a decentralised control scheme, i.e., by controllers aboard each slave node, that communicate with the master only to periodically receive the synchronisation packets. Specifically, when the $(k+1)$-th packet is received, each slave takes a local clock timestamp, i.e., a measurement of the actual packet arrival time $t^a_{k+1}$. Also, the node can easily obtain an *estimated* arrival time $t^e_{k+1}$, by taking the previous one and adding $T$. The quantity $e_k := t^e_k - t^a_k$ – that, notice, only contains times counted in the slave local clock – measures thus the synchronisation error, and the system under control is ruled by

$$e_{k+1} = e_k + d_k, \tag{1}$$

where $d_k = -\int_{t^a_k}^{t^a_{k+1}} \frac{\delta_f(\tau)}{f_o} d\tau$, $f_o$ being the nominal frequency of the slave clock oscillator, and $\delta_f(t)$ its variation due to ageing, thermal stress, and short-term jitter.

Note that in (1) *all* causes of error lie together in the disturbance $d_k$. Differently from classical alternatives [6], in FLOPSYNC, the various physical sources of disturbances are counteracted by considering the different bands of their contributions: manufacturing tolerances result in a constant frequency error; ageing can be thought of as a constant disturbance contribution; short-term jitter acts at the time scale of electronic noise and provides the ultimate bound for the achievable synchronisation quality; and thermal stress acts on a time scale comparable to reasonable synchronisation periods. Since in a wide variety of operating conditions a WSN undergoes either abrupt but sporadic thermal stress episodes, or environmental variations that are slow compared to the thermal dynamics of typical nodes, control operation is optimized for a disturbance that is constant.

To counteract $d_k$, it suffices that the slave can alter the expected time for the next packet by a quantity $u_k$, which is the control signal. Hence, (1) becomes $e_{k+1} = e_k + u_k + d_k$ or, in the $\mathscr{Z}$ transform domain, $E(z) = P(z)\big(U(z) + D(z)\big)$ where $P(z) = \frac{1}{z-1}$ is the process transfer function. In the FLOPSYNC scheme, $P(z)$ is controlled by a PI written in the form $R(z) = \frac{1-\alpha z}{z-1}$ so that in closed-loop $\frac{E(z)}{D(z)} = \frac{z-1}{z(z+\alpha-2)}$, and the system is asymptotically stable if $1 < \alpha < 3$. Thus, the PI controller guarantees that the error converges to zero in the presence of a constant disturbance, i.e., the slave clock keeps the master pace. If absolute time is needed, then, a single timestamp transmission at boot time is needed to eliminate the initial offset of the slave clock with respect to the master.

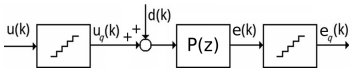


Figure 1: Model of the controlled system.

When the relevant quantisations are introduced, the system un-


*ICCPS* '15 Seattle, USA

der control in the FLOPSYNC scheme takes the form in Figure 1, where the "$q$" subscript denotes quantised variables. We assume without loss of generality that the quantisation step is unitary. In general, in presence of a constant disturbance $d_k = \overline{d}$, the closed-loop system composed of $P(z)$, $R(z)$, and the two quantisers enters a condition characterised by small error fluctuations around zero, not necessarily a limit cycle. In order to force the quantised error $e_q(k)$ to lie in the range $[0,1]$, or $[-1,0]$, we introduce a conditional quantisation of the state of the PI integrator, leading to the switching scheme in Figure 2. The efficacy of this extension can be proven through a two-step procedure.

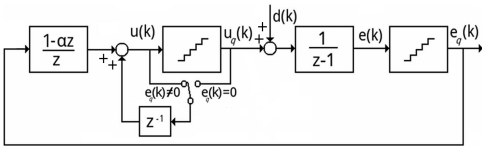


Figure 2: Control scheme incorporating a switching mechanism.

**Step 1 – Invariance analysis:** The system in Figure 2 admits an invariant set characterized by a quantised error $e_{q,k}$ taking values either in the set $\{0,1\}$ or $\{-1,0\}$, depending on the constant value $\bar{d}$ for the disturbance.

The system can be described as a hybrid automaton with three modes. If $e_{q,k} \neq 0$, the system is ruled by

$$\begin{aligned} e_k &= e_{k-1} + \rho(u_{k-1}) + d_{k-1} \\ u_k &= u_{k-1} + \rho(e_{k-1}) - \alpha\rho\big(e_{k-1} + \rho(u_{k-1}) + d_{k-1}\big) \end{aligned} \tag{2}$$

where $\rho(\cdot)$ denotes the *round* operator. If $e_{q,k} = 0$, then,

$$e_k = e_{k-1} + \rho(u_{k-1}) + d_{k-1}; \quad u_k = \rho(u_{k-1}) + \rho(e_{k-1}) \tag{3}$$

which simplifies to the linear dynamics

$$e_k = e_{k-1} + u_{k-1} + d_{k-1}; \quad u_k = u_{k-1} \tag{4}$$

when $e_{q,k}$ remains zero for two or more steps.

THEOREM 1. *Let $\overline{u} = -\rho(\overline{d})$, $\Delta_d = \overline{d} + \overline{u}$. Observe that, by the way it is defined, $-0.5 \leq \Delta_d \leq 0.5$. Also, suppose that $1 < \alpha < 1.5$. If at a certain step $k$*

$$-0.5 < e(k) < 0.5 \text{ (thus } e_q(k) = 0) \text{ and } u(k) = \overline{u}, \tag{5}$$

*then, for all subsequent steps: i) $(e_q, u_q) \in \{(0,\overline{u}), (1,\overline{u}-1)\}$, if $0 < \Delta_d \leq 0.5$; ii) $(e_q, u_q) \in \{(0,\overline{u}), (-1,\overline{u}+1)\}$, if $-0.5 \leq \Delta_d < 0$; iii) $(e_q, u_q) \in \{(0,\overline{u})\}$, if $\Delta_d = 0$.*

The proof is omitted due to space limitations.

**Step 2 – Reachability analysis:** The system in Figure 2 eventually reaches set $\mathscr{S} = \{(u,e) : u = \overline{u}, -0.5 < e < 0.5\}$ in Theorem 1 from any initial state in a bounded region of the $(u,e)$ plane.

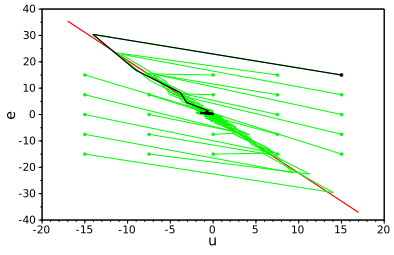


Figure 3: Phase-plane trajectories with quantisations.

To verify that this reachability condition is satisfied starting from a sufficiently large set of initial conditions, we exploit the simplicity of the control system, and study its behaviour through a phase-plane approach. Figure 3 shows the evolution of the state $(u,e)$ starting from a certain rectangular region, when $\overline{d} = 0.4$. In this case, $\mathscr{S} = \{(0,e) : -0.5 < e < 0.5\}$ and we verified that the reachability condition is always satisfied for a very dense grid of initial conditions. Note that all trajectories approach the straight line drawn in red in one step, and then evolve towards $\mathscr{S}$ keeping close to that same line. This resembles the behaviour that can be observed when quantisation is removed and the system becomes linear and ruled by

$$\begin{bmatrix} e(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1-\alpha & 1-\alpha \end{bmatrix} \begin{bmatrix} e(k-1) \\ u(k-1) \end{bmatrix} + \begin{bmatrix} 1 \\ -\alpha \end{bmatrix} \overline{d}. \tag{6}$$

Indeed, if we multiply by $\begin{bmatrix} 1-\alpha & -1 \end{bmatrix}$ both sides of (6), we get the equation $(1-\alpha)e(k) - u(k) = \overline{d}$ of the red line in Figure 3. This entails that in one step the linear system state ends up exactly on that line and then evolves along it towards the equilibrium $(0, -\overline{d})$.

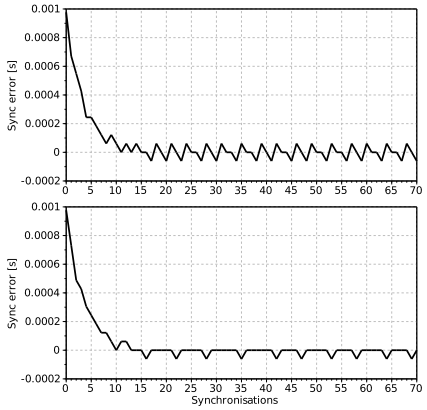


Figure 4: Synchronisation error with bare FLOPSYNC (top) and with FLOPSYNC endowed with the proposed extension (bottom).

Experiments were run on a WSN. Figure 4 shows that once zero is approached, the difference between bare and extended FLOPSYNC comes to evidence. From the 15th period onward, the extended scheme (bottom plot) makes the error oscillate taking two values: one is zero, while the negative one is about $61\mu$s. This corresponds to the resolution of the used clock, that in the previous treatise was coded as unitary for simplicity. The same is not true (see the top plot) for the bare FLOPSYNC scheme, where the fluctuations of the error around zero are not limited to two values.

## References


[1] J. Chen, Q. Yu, Y. Zhang, H. Chen, and Y. Sun. Feedback-based clock synchronization in wireless sensor networks: A control theoretic approach. *IEEE Transactions on Vehicular Technology*, 59(6), 2010.

[2] J. Hellerstein, Y. Diao, S. Parekh, and D. Tilbury. *Feedback Control of Computing Systems*. John Wiley & Sons, Hoboken, NJ, USA, 2004.

[3] A. Leva and M. Maggio. Feedback process scheduling with simple discrete-time control structures. *IEEE Proceedings – Control Theory and Applications*, 4(11):2331–2342, 2010.

[4] A. Leva, M. Maggio, A. Papadopoulos, and F. Terraneo. *Control-based Operating System Design*. IET, London, UK, 2013.

[5] A. Leva and F. Terraneo. Low power synchronisation in wireless sensor networks via simple feedback controllers: the FLOPSYNC scheme. In *Proc. 2013 American Control Conference*, pages 5017–5022, Washington, DC, USA, 2013.

[6] I. Rhee, J. Lee, J. Kim, E. Serpedin, and Y. Wu. Clock synchronization in wireless sensor networks: an overview. *Sensors*, 2009.

[7] F. Terraneo and A. Leva. Feedback-based memory management with active swap-in. In *Proc. ECC 2013*, pages 620–625, July 2013.