Dealing with Time Asynchrony in CyMAC Design

In a practical sensor network, sender and receiver nodes are inevitably asynchronous. Typically, clocks of sensor nodes differ for two reasons: *clock skew* that is simply the initial difference between clocks, and *clock drift* that refers to different clocks counting time at slightly different rates, which results in varying clock skews over time. In general, clock asynchrony between sender and receiver nodes can be described with the following equation:

$$t_r = a \times t_s + b, \tag{1}$$

where t_s is the time instance at the sender, t_r is the corresponding time instance at the receiver, and a and b represent the clock drift and the clock skew, respectively. In this section, we analyze the effects of clock asynchrony on CyMAC performance, and discuss how we enhance CyMAC to deal with these issues.

In CyMAC design, for a relative delay bound of μ, let us set $I_{\text{allow}}(p_1)$ to

$$I_{\text{allow}}(p_1) = (1+\mu)\theta(p_1) - D(p_1). \tag{2}$$

1) $a < 1$: In this case, the sender clock counts time at a faster rate than the receiver clock, as shown in Fig. 1(a). After the sender delivers a packet p_1 successfully to the receiver, both sender and receiver know that $T_{\text{sent}}(p_1)$ on the sender clock corresponds to T_{LAST} on the receiver clock, and schedule the next rendezvous time to $I_{\text{allow}}(p_1)$ time later. Since the sender clock counts faster, when the sender wakes up at $T_{\text{SCHD}}(p_1)$ to listen for beacon from the receiver, the receiver won't wake up till $I_{\text{allow}}(p_1)(\frac{1}{a}-1)$ time later. As a result, an extra delay is introduced to the delivery of packet p_2:

$$D(p_2) = I_{\text{allow}}(p_1)\frac{1}{a} + D(p_1) - (T_{\text{arrv}}(p_2) - T_{\text{arrv}}(p_1)). \tag{3}$$

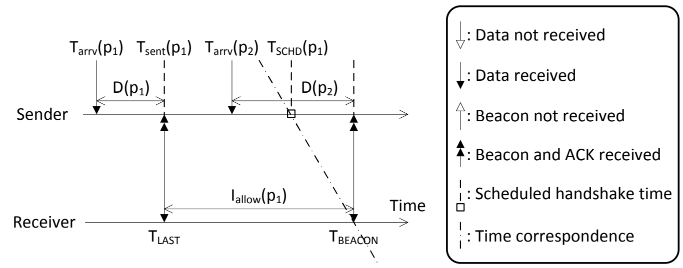
When the system stabilizes, $D(p_1) = D(p_2) \triangleq D$ and $T_{\text{arrv}}(p_2) - T_{\text{arrv}}(p_1) = \theta(p_1) \triangleq \theta$. Plugging in Eq. (2), we have

$$\begin{aligned} D &= ((1+\mu)\theta - D)\frac{1}{a} + D - \theta \\ \implies\quad D &= (\mu + 1 - a)\theta. \end{aligned} \tag{4}$$

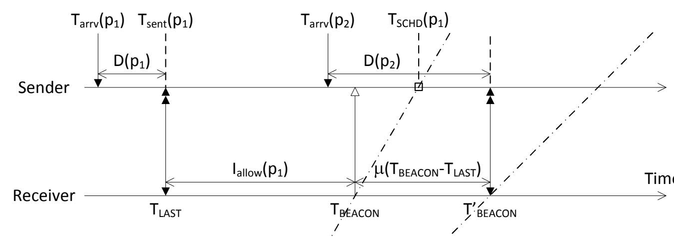
This means that an extra delay of $(1-a)\theta$ has been added to the packet delivery delay.

2) $a > 1$: In this case, the sender clock counts time at a slower rate than the receiver clock, as shown in Fig. 1(b). After the sender delivers a packet p_1 successfully to the receiver, both sender and receiver schedule the next rendezvous time to $I_{\text{allow}}(p_1)$ time later. Since the sender clock counts slower, when the sender wakes up at $T_{\text{SCHD}}(p_1)$ to listen for a beacon from the receiver, the receiver has already finished its beacon transmission. As a result, the sender has to remain awake to wait for the next beacon. We have:

$$D(p_2) = (1+\mu)I_{\text{allow}}(p_1)\frac{1}{a} + D(p_1) - (T_{\text{arrv}}(p_2) - T_{\text{arrv}}(p_1)). \tag{5}$$



(a) When the sender clock counts time faster than the receiver clock (i.e., $a < 1$).



(b) When the sender clock counts time slower than the receiver clock (i.e., $a > 1$).

Fig. 1. Effects of time asynchrony on CyMAC performance.

When the system stabilizes, $D(p_1) = D(p_2) \triangleq D$ and $T_{\text{arrv}}(p_2) - T_{\text{arrv}}(p_1) = \theta(p_1) \triangleq \theta$. Plugging in Eq. (2), we have

$$\begin{aligned} D &= (1+\mu)((1+\mu)\theta - D)\frac{1}{a} + D - \theta \\ \implies\quad D &= \left(\mu + 1 - \frac{a}{1+\mu}\right)\theta. \end{aligned} \tag{6}$$

This means that an extra delay of $(1-\frac{a}{1+\mu})\theta$ has been added to the packet delivery delay.

To ameliorate the effects of time asynchrony, we have employed the following schemes in CyMAC:

- To guarantee a relative delay bound of μ, CyMAC does it more conservatively by replacing μ with $\mu^* = \mu - |1-\hat{a}|$ as the target delay bound in sensor nodes' operations, where $\hat{a}$ is the upper limit of clock drift between sensor nodes. When $|1-\hat{a}| < \mu$, CyMAC works fine. However, if $\mu \leqslant |1-\hat{a}|$, CyMAC won't be able to provide the desired delay bound. Fortunately, this situation rarely occurs in practice as it makes little sense to ask a sensor network to provide a delay bound that is even tighter than the degree of clock asynchrony between sensor nodes.
- In CyMAC, the sender wakes up a bit earlier prior to the scheduled listen time to wait for beacons. Specifically, if the time between the previous listen time and the next listen time is ψ seconds, the sender will wake up at $\left(\frac{\mu\psi}{2+2\mu}\right)$ prior to the next listen time.

With these two enhancements, time asynchrony can be dealt with effectively and the original relative delay bound of μ can be satisfied. The proof is as follows.

Proof: As $\mu^* = \mu - |1 - \hat{a}|$, we have $\mu^* = \mu - 1 + a$ when $a < 1$, and $\mu^* = \mu + 1 - a$ when $a > 1$.

- Case I: $a < 1$. By simply replacing μ with μ^* in Eq. (4), we have $D = \mu\theta$, meaning that the desired delay bound is achieved. This indicates that when the sender clock counts time faster than the receiver clock (as shown in Fig. 1(a)), using a conservative μ would guarantee the target delay bound.
- Case II: $a \geqslant \frac{2(1+\mu)}{2+\mu}$. Since $\mu^* = \mu + 1 - a < \mu$, we have $a \geqslant \frac{2(1+\mu)}{2+\mu} > \frac{2(1+\mu^*)}{2+\mu^*}$. Then, by replacing μ with μ^* in Eq. (6), we have

$$\begin{aligned} D &= \left(\mu^* + 1 - \frac{a}{1+\mu^*}\right)\theta \\ &= \left(\mu + 2 - a - \frac{a}{1+\mu^*}\right)\theta \\ &< \left(\mu + 2 - \frac{2(1+\mu^*)}{2+\mu^*} - \frac{\frac{2(1+\mu^*)}{2+\mu^*}}{1+\mu^*}\right)\theta \\ &= \left(\mu + 2 - \frac{2(1+\mu^*)}{2+\mu^*} - \frac{2}{2+\mu^*}\right)\theta \\ &= \mu\theta. \end{aligned} \tag{7}$$

  Thus the desired delay bound is achieved.
- Case III: $1 < a < \frac{2(1+\mu)}{2+\mu}$. Since the sender clock counts time slower than the receiver clock, the scheduled listen time for the sender is at $\left(1 - \frac{1}{a}\right)\psi$ after the beacon arrival time, where

$$\left(1 - \frac{1}{a}\right)\psi < \left(1 - \frac{2+\mu}{2+2\mu}\right)\psi = \frac{\mu\psi}{2+2\mu}. \tag{8}$$

  Therefore, if the sender wakes up at $\left(\frac{\mu\psi}{2+2\mu}\right)$ prior to the scheduled listen time, we can guarantee that the sender receives the first beacon frame from the receiver. As a result, the packet delay must be smaller than the originally planned delay bound.

Till now, we have proved that, with the proposed two enhancements, CyMAC can deal with time asynchrony effectively to guarantee the original relative delay bound of μ. ■