Modeling Self-Triggered Control Strategies Using the Hybrid System Framework

David Kooi

ECE 246 Hybrid Systems
University of California Santa Cruz

June 15, 2020
Motivation

Cyber-physical systems (CPSs) usually operate by taking sensor measurements, computing control, and applying control. In real systems, each of these steps takes time.

**Question**: How to design event-trigger-mechanisms (ETMs) for the sensor and control such that system requirements (stability, safety, temporal requirements) are satisfied?
Outline

1. Event-Triggered CPS Model
2. Specialization for Self-Triggered Strategies
3. Review and Comparison of Two Self-Triggered Strategies
Assumptions

1. The system is modeled as a differential equation
2. The control law $\kappa$ is static
3. Plant and controller output operate under zero-order hold
4. Plant and controller output are triggered by synchronous ETMs.
An Event Triggered CPS Model

- Developed from [Chai et al., 2017]

**Hybrid Model**

\[ \mathcal{H}_{et} := (F, G, C, D) \]

\[ z = (x_p, \hat{y}, \hat{u}, \chi)^\top \]

\[ \dot{z} = F(z) = \begin{bmatrix} F_p(x_p, \hat{u}) \\ 0 \\ 0 \\ F_{\chi}(z) \end{bmatrix} \]

\[ z^+ = G(z) = \begin{bmatrix} x_p \\ H_p(x_p) \\ \kappa(\hat{y}) \\ G_{\chi}(z) \end{bmatrix} \]

\[ C = \{ z \in Z : \gamma(\xi) = 0 \} \quad D = \{ z \in Z : \gamma(\xi) = 1 \} \]

(1)
An Event Triggered CPS Model

- Developed from [Chai et al., 2017]

Hybrid Model
\[ \mathcal{H}_{et} := (F, G, C, D) \]
\[ z = (x_p, \hat{y}, \hat{u}, \chi) \top \]

\[ \dot{z} = F(z) = \begin{bmatrix} F_p(x_p, \hat{u}) \\ 0 \\ 0 \\ F_\chi(z) \end{bmatrix} \]
\[ z^+ = G(z) = \begin{bmatrix} x_p \\ H_p(x_p) \\ \kappa(\hat{y}) \\ G_\chi(z) \end{bmatrix} \]
\[ C = \{ z \in Z : \gamma(\xi) = 0 \} \quad D = \{ z \in Z : \gamma(\xi) = 1 \} \quad (2) \]

- Updates occur through the synchronous ETM
- ETM is modeled by a function \( \gamma : \Xi \mapsto \{0, 1\} \)
- \( \Xi := \mathcal{Y} \times \mathcal{U} \times \mathcal{X} \) is the domain of the data available to the ETM
An Event Triggered CPS Model

- Developed from [Chai et al., 2017]

Hybrid Model

\[ \mathcal{H}_{et} := (F, G, C, D) \]

\[
\begin{align*}
\dot{z} &= F(z) = \\
&= \begin{bmatrix}
F_p(x_p, \hat{u}) \\
0 \\
0 \\
F_\chi(z)
\end{bmatrix} \\
\gamma(z) &= G(z) = \\
&= \begin{bmatrix}
x_p \\
H_p(x_p) \\
\kappa(\hat{y}) \\
G_\chi(z)
\end{bmatrix}
\end{align*}
\]

\[ C = \{ z \in \mathbb{Z} : \gamma(\xi) = 0 \} \quad D = \{ z \in \mathbb{Z} : \gamma(\xi) = 1 \} \quad (3) \]

- Specialization to a self-triggered ETM requires defining:

<table>
<thead>
<tr>
<th>Auxiliary State</th>
<th>Auxiliary Dynamics</th>
<th>Auxiliary Update</th>
<th>ETM Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \chi )</td>
<td>( F_\chi(\cdot) )</td>
<td>( G_\chi(\cdot) )</td>
<td>( \gamma(\cdot) )</td>
</tr>
</tbody>
</table>
Periodic and Self-Triggered Sampling

The Event-Triggered model define a sequence of times \( \{t_i\}_{i=0}^{\infty} \) when the system is updated.

To illustrate the difference between periodic and self-triggered sampling consider the following

- **Periodic ETMs:** \( t_{i+1} - t_i = T_s \)
- **Self-Triggered ETMs:** \( t_{i+1} - t_i = \Gamma(\xi), \) For \( \xi \in \Xi, \Gamma : \Xi \rightarrow \mathbb{R}_{\geq 0}. \) (Recall that \( \Xi := \mathcal{Y} \times \mathcal{U} \times X. \))
Self-Triggered Model

\[ \mathcal{H}_{st} := (F, G, C, D) \]
\[ z = (x_p, \hat{y}, \hat{u}, \chi)^\top \]

\[ \dot{z} = F(z) = \begin{bmatrix} \dot{x}_p(x_p, \hat{u}) \\ 0 \\ 0 \\ F_\chi(z) \end{bmatrix} \quad z^+ = G(z) = \begin{bmatrix} x_p \\ H_p(x_p) \\ \kappa(\hat{y}) \\ G_\chi(z) \end{bmatrix} \]

\[ C = \{ z \in \mathbb{Z} : \gamma(\xi) = 0 \} \quad D = \{ z \in \mathbb{Z} : \gamma(\xi) = 1 \} \] (4)

- Define:
  - **Auxiliary State**: \( \chi := (\tau, T_s^*, \chi_3, \ldots, \chi_{n\chi})^\top \)
  - **Auxiliary Dynamics**: \( F_\chi(\cdot) := (1, 0, \dot{\chi}_3, \ldots, \dot{\chi}_n)^\top \)
  - **Auxiliary Update**: \( G_\chi(\cdot) := (0, \Gamma(\xi), \chi_3^+, \ldots, \chi_n^+)^\top \)
  - **ETM Function**: \( \gamma(\xi) := \begin{cases} 0 & \text{if } \tau < T_s^* \\ 1 & \text{if } \tau \geq T_s^* \end{cases} \)
Assumptions

1. There exists a feedback law $\kappa : \mathcal{X} \mapsto \mathcal{U}$ such that the origin is asymptotically stable.

2. There is full state feedback. That is, at sample point $t_i$, the sampled state is $\hat{y} = H_p(x_p) = x_p := \hat{x}$

Self-Triggered Sampler

$$\Gamma_1(\hat{y}) = \frac{1}{2L} \ln(1 + 2\delta/\|f(\hat{y}, \kappa(\hat{y}))\|)$$

(5)

where $\delta > 0$, and $L = L_{f,u}L_{\kappa,x}$ is the product of the Lipschitz constants $L_{f,u}$ (Lipschitz constant of $f$ with respect to $u$) and $L_{\kappa,x}$ (Lipschitz constant of $\kappa$ with respect to $x$). The triggering condition assumes that

$$\|f(\hat{y}, \kappa(\hat{y}))\| > m$$

for some $m > 0$.

$\implies$ Globally ultimately uniformly bounded (GUUB). I.e, For $a$ arbitrarily large, $\|x_o\| < a$,

$$\|x_p\| \leq b > 0 \forall t \geq T$$
Assumptions

1. $F_p$ represents linear dynamics.

2. There exists a feedback law $\kappa : X \mapsto U$ such that ideal system $F_p$ is exponentially asymptotically stable (E.A.S) to the origin.

3. There is full state feedback. That is, at sample point $t_i$, the sampled state is $\hat{y} = H_p(x_p) = x_p := \hat{x}$
Since the linear system is E.A.S there exists a Lyapunov function $V(x_p)$ such that the following hold.

1. $\alpha_1 \|x_p\|^2 \leq V(x) \leq \alpha_2 \|x_p\|^2$,
2. $\dot{V}(x_p) \leq -\alpha_3 \|x_p\|^2$,
3. $\|\frac{\partial V}{\partial x_p}(x_p)\| \leq \alpha_4 \|x_p\|$.

Consequently, the ideal system solutions starting at $x_o$ evolve as

$$V(x_p) \leq V(x_o)e^{\lambda t} \quad \forall t \in \text{dom } x_p, \quad \forall x_o \in \mathcal{X},$$

where $\lambda \geq \alpha_3/(2\alpha_2)$. 

Self Triggered Strategies [Heemels et al., 2012]
Self Triggered Strategies [Heemels et al., 2012]

Strategy is to enforce

\[ h_c(\hat{x}, t) = V(x(t)) - V(\hat{x})e^{-\lambda} \leq 0 \quad \forall t \in [0, T_s]. \] (6)

- If this is enforced the system remains (E.A.S) to the origin.
- The entire duration between samples must be exponentially decreasing as compared with the previous state \( \hat{x} \).

Note: \( T_s \) is not known before hand.

\[ \Rightarrow \] Predict over a finite horizon \([\tau_{min}, \tau_{max}]\) to estimate the sampling period \( T_s \) such that (6) holds. Design parameters: \( \tau_{min}, \tau_{max} \).

- \([\tau_{min}, \tau_{max}]\) is discretized into \( N_{max} \) pieces
- \( N_{max} = \frac{\tau_{max}}{\Delta} \) and \( \Delta \) is the discretization step.

The self-triggered sampler is then given by:

\[ \Gamma_2(\hat{x}) := \max\{\tau_{min}, n(\hat{x})\Delta\} \] (7)

\[ n(\hat{x}) := \arg\max\{h_c(\hat{x}, n\Delta) : n \in [0, N_{max}]\} \] (8)
Double Integrator Model

\[ \dot{x}_p = F_p(x_p, u) = (x_2, u)^\top \]

- Full state feedback \( \hat{y} = H_p(x_p) := \hat{x} \)
- Stabilized with \( u = \kappa(x_p) = -Kx \).
Double Integrator Model

\[ \dot{x}_p = F_p(x_p, u) = (x_2, u)^\top \]

- Full state feedback \( \hat{y} = H_p(x_p) := \hat{x} \)
- Stabilized with \( u = \kappa(x_p) = -Kx \).

[Tiberi and Johansson, 2013]

\[ \Gamma_1(\hat{y}) = \frac{1}{2L} \ln(1 + 2\delta/\|f(\hat{y}, \kappa(\hat{y}))\|), \]

where \( L = L_{f,u}L_{\kappa,x} \), and \( \delta > 0, L_{f,u} = 1 \) and \( L_{\kappa,x} = \max k_1, k_2 \). Further, the norm of the dynamics is \( \|f(\hat{y}, \kappa(\hat{y}))\| = \sqrt{x_2^2 + u^2} \).
Comparison of Methods

[Tiberi and Johansson, 2013]

Average Sample Time: 1.29s
Average Sample Time \( t > 15 \): 1.64s

Table 1: Simulation Parameters

| \( k_1 \) | 1 |
| \( k_2 \) | 1 |
| \( L_{f,u} \) | 1 |
| \( L_{\kappa,x} \) | 1 |
| \( \delta \) | 0.5 |
| \( x(0) \) | \((1, 0)\)\(\top\)|
| \( \min ||f(\hat{y}, \kappa(\hat{y}))|| \) | 0.1 |

Figure 1: State Evolution, Control, and \( V(x) \)
Comparison of Methods

[Heemels et al., 2012]

Recall the sampling method of

$$\Gamma_2(\hat{x}) := \max\{\tau_{min}, n(\hat{x})\Delta\}$$

$$n(\hat{x}) := \arg \max\{h_c(\hat{x}, n\Delta) : n \in [0, N_{max}]\}$$

with

$$h_c(\hat{x}, n\Delta) = V(x(n\Delta)) - V(\hat{x})e^{-\lambda} \leq 0,$$

where $x(n\Delta)$ is a prediction of the state evolution. In the case for linear systems an explicit solution is given by

$$x(t) = e^{At}x(0) - \int_0^t e^{-At}BK\hat{x}d\tau$$

$$= e^{At}x(0) - A^{-1}(e^{-At} - I)BK\hat{x}$$

if $A$ is invertable. In the simulation an approximation of $A^{-1}$. 
Comparison of Methods

[Heemels et al., 2012]

Average Sample Time: 1.15s
Average Sample Time($t > 15$): 1.01s

- Stable system has Lyapunov function $V(x) = x^\top P x$
- $\lambda_{\text{max}}$ is largest eigenvalue of $P$

Table 2: Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k_1$</td>
<td>1</td>
</tr>
<tr>
<td>$k_2$</td>
<td>1</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>$1/(2\lambda_{\text{max}})$</td>
</tr>
<tr>
<td>$\Delta$</td>
<td>0.1</td>
</tr>
<tr>
<td>$\tau_{\text{max}}$</td>
<td>4</td>
</tr>
<tr>
<td>$\tau_{\text{min}}$</td>
<td>0.1</td>
</tr>
<tr>
<td>$x(0)$</td>
<td>$(1,0)^\top$</td>
</tr>
</tbody>
</table>

Figure 2: State Evolution, Control, and $V(x)$
Conclusion

- Hybrid system framework can model general event-triggered CPSs.
- Can be adapted to self-triggered strategies by adding a timer and predictive function $\Gamma$.
- [Tiberi and Johansson, 2013]: Larger sample periods at the cost of boundedness,
- [Heemels et al., 2012]: Smaller sample periods, but keeps asymptotic stability.
