DARC:
Dynamic Adaptation of Real-time Control Systems

Nils Vreman[1], Claudio Mandrioli[1]

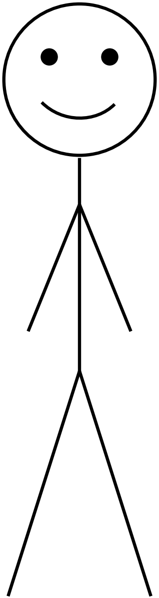
Control Systems Synthesis - Project

November 30, 2020

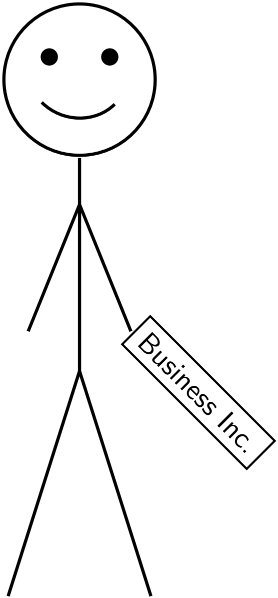
[1]{nils.vreman,claudio.mandrioli}@control.lth.se
Dept. of Automatic Control
Lund University

This story starts with...

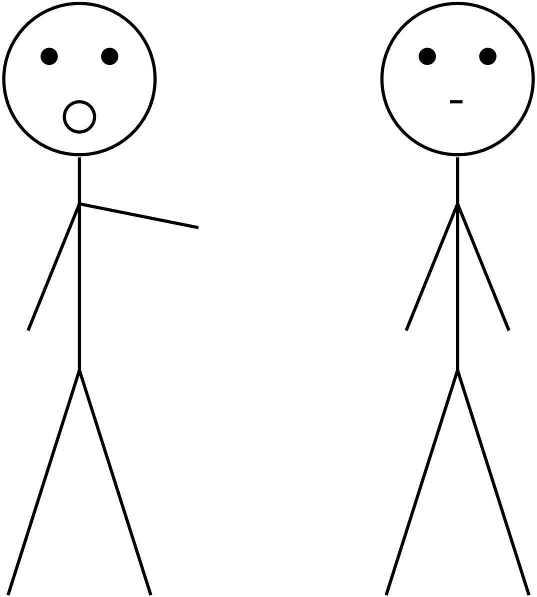
Fred...



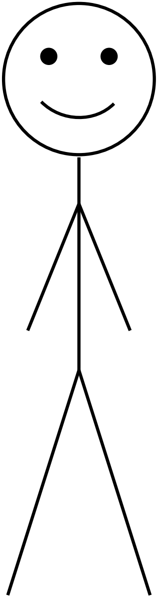
Fred works at Business Inc...

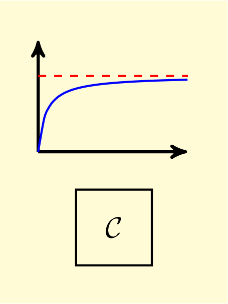


Fred's boss asks him to design a controller...

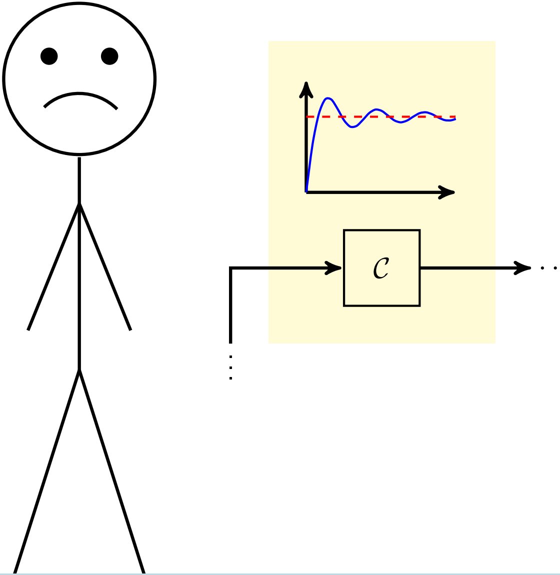


Fred designs an amazing controller...

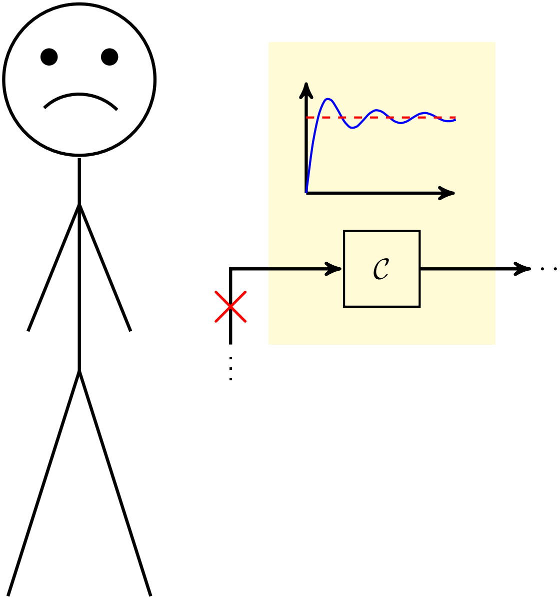




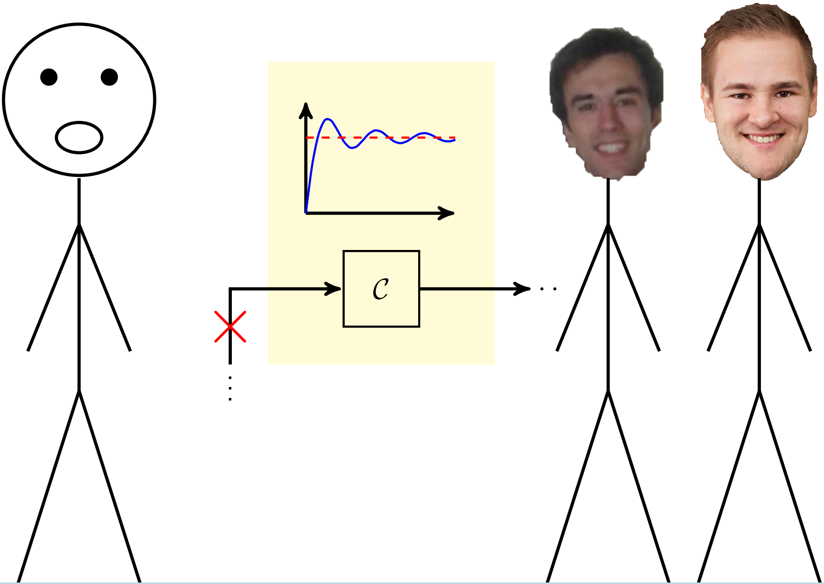
Fred's controller performs worse when connecting it to to the real process...



Fred's problems are due computation and connection issues...



Fred asks his colleagues if they can help...



Control system setup

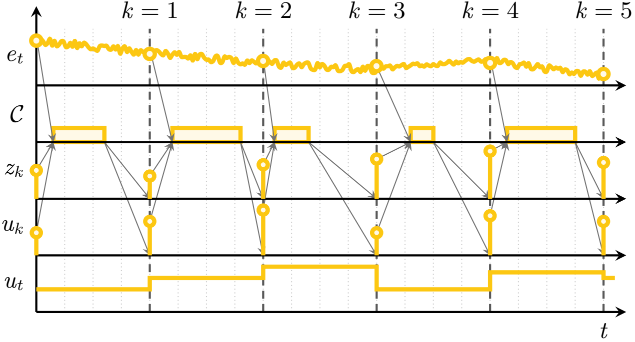
$$\mathcal{P}: \begin{cases} x_{k+1} &= A_p x_k + B_p u_k + w_k \\ y_k &= C_p x_k + D_p u_k + n_k \end{cases}$$

$$\mathcal{C}: \begin{cases} z_{k+1} &= A_c z_k + B_c y_k \\ u_{k+1} &= C_c z_{k+1} + D_c y_k \end{cases}$$

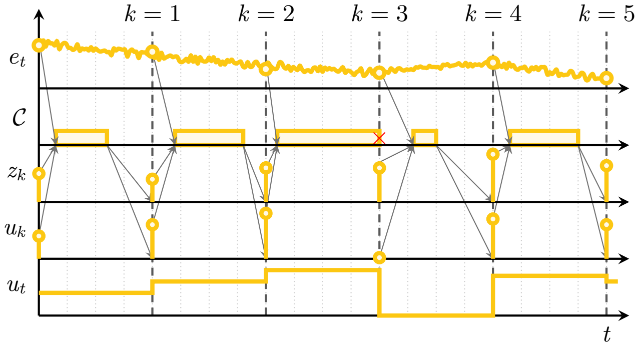
$$\tilde{x}_k = (x_k,\ z_k,\ u_k)$$

Note the *one-step delay* controller.

Control Design vs. Implementation

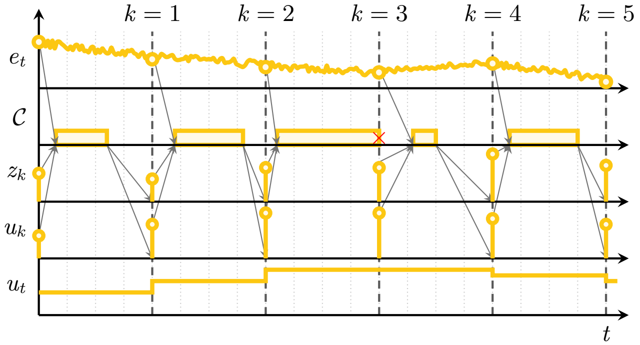


Faults - Zeroing



- $z_3 = z_2$,
- $u_3 = 0$,
- The measurement at $k = 2$ gets thrown away.

Faults - Hold



- $z_3 = z_2$,
- $u_3 = u_2$,
- The measurement at $k = 2$ gets thrown away.

Control adaptation

$$\mathcal{C}\left(q\right): \begin{cases} z_{k+1} & = A_c\left(q\right) z_k + B_c\left(q\right) y_k \\ u_{k+1} & = C_c\left(q\right) z_{k+1} + D_c\left(q\right) y_k \end{cases}$$

Where q is the number of control task invocations missed in a row.

Idea: Express $\left(A_c\left(q\right), B_c\left(q\right), C_c\left(q\right), D_c\left(q\right)\right)$ in terms of (A_c, B_c, C_c, D_c).

Control adaptation - Advantages

$$\mathcal{C}(q): \begin{cases} z_{k+1} &= A_c(q)\, z_k + B_c(q)\, y_k \\ u_{k+1} &= C_c(q)\, z_{k+1} + D_c(q)\, y_k \end{cases}$$

Advantages of the control adaptation include (not limited to):

- No plant knowledge necessary,
- Valid for *all* linear controllers,
- Modular $(A_c(q), B_c(q), C_c(q), D_c(q))$,
- Intuitive (never underestimate the power of simplicity),
- Minimal overhead (counting deadline misses),
- No performance loss in the nominal case.

Control adaptation - Disadvantages

$$\mathcal{C}\left(q\right): \begin{cases} z_{k+1} & = A_c\left(q\right)z_k + B_c\left(q\right)y_k \\ u_{k+1} & = C_c\left(q\right)z_{k+1} + D_c\left(q\right)y_k \end{cases}$$

Disadvantages of the control adaptation include (not limited to):

- No stability guarantees (a posteriori analysis),
- Could increase performance using knowledge about process (at the cost of the complexity involved in changing control law),
- Inherits problems with the nominal controller.

Controller dynamics

Problem: The controller's states diverge when execution fails.

Solution: Adapt the control state dynamics to the number of execution fails.

Controller dynamics - Example

$$\mathcal{C}: \begin{cases} z_{k+1} &= A_c z_k + B_c y_k \\ u_{k+1} &= C_c z_{k+1} + D_c y_k \end{cases}$$

Independent of the number of faults in a row,

$$C_c(q) = C_c, \quad D_c(q) = D_c$$

are chosen.

We adapt the state dynamics as

$$A_c(q) = A_c^{q+1}, \quad B_c(q) = \left(\sum_{i=0}^{q} A_c^i \right) B_c.$$

Note that this is *one* example of how to adapt the controller.
More possibilities are under investigation.

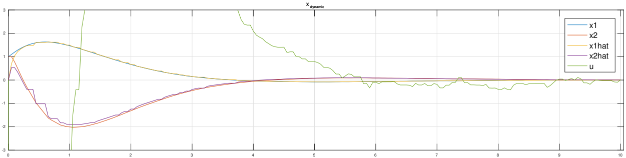
Results - Setup

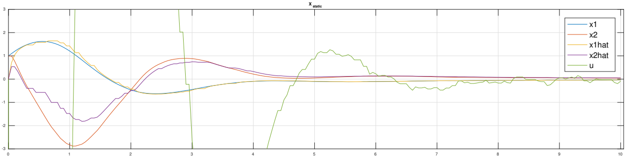
$$\mathcal{P}: \begin{cases} x_{k+1} &= \begin{pmatrix} 1.046 & 0.051 \\ 0 & 0.995 \end{pmatrix} x_k + \begin{pmatrix} 0.000127 \\ 0.004988 \end{pmatrix} u_k + w_k \\ y_k &= \begin{pmatrix} -1 & 0 \end{pmatrix} x_k + 0 u_k + n_k \end{cases}$$

$$\mathcal{C}: \begin{cases} z_{k+1} &= \begin{pmatrix} 0.3397 & 0.0475 \\ -0.7283 & 0.8549 \end{pmatrix} z_k + \begin{pmatrix} -0.7014 \\ -0.5352 \end{pmatrix} y_k \\ u_{k+1} &= \begin{pmatrix} -38.72 & -28.1 \end{pmatrix} z_{k+1} + 0 y_k \end{cases}$$

- Unstable process (fast unstable pole)
- LQG controller
- $w_k = [2 \times 1] \in \mathcal{N}\left(0, 10^{-3/2}\right)$
- $n_k = [1 \times 1] \in \mathcal{N}\left(0, 10^{-3/2}\right)$

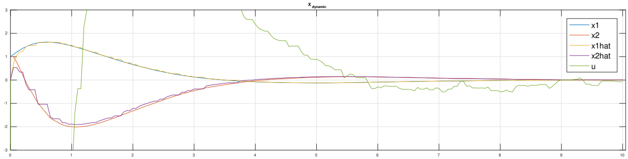
Results

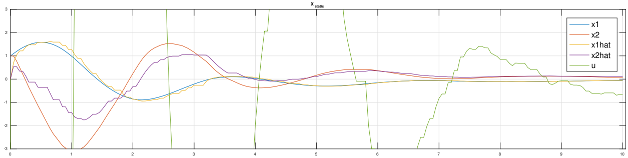




$$p = 0.25, \quad q_{max} = 3$$

Results





$$p = 0.35, \quad q_{max} = 3$$

Conclusion

- We have developed a novel, simplistic adaptation methodology used to tackle execution faults in real-time control systems.
- Complete without system or controller knowledge.
- Stability analysis can be performed a posteriori using *switching stability analysis under weakly-hard constraints*[1].
- More research will be dedicated to finding the situations in which this approach is feasible.

[1] Paper in submission (ask Nils for more information.)