Introduction to sampled-data methods in model predictive control

Mohamed Elobaid

April 22, 2022

What this talk is

- Basic, introductory and somewhat informal
- Discrete time representation of *systems under sampling*. Single and multi-rate sampling
- Sampling zero dynamics and relative degree
- Recalls on optimal control and MPC
- Cancellation of zero dynamics in MPC and instability
- Work-arounds using multi-rate sampling

What this talk is not

- Comprehensive and exhaustive
- Other methods of modeling systems under sampling (e.g. DDR, Hybrid representation, ...etc)
- 1^{st}, higher order and generalized holding schemes
- The mechanics of solving optimization problems (specially NLPs)

Prelude

Nonlinear control systems in continuous-time

$$\dot{x} = f(x) + g(x)u$$
$$y = h(x)$$

x, u, y are *states, controls, and measurements/outputs*. Some assumptions:

- **basic**: origin is an quilibrium, forward complete (or smoothness assumption)
- **technical**: geometry of the space, relative degree, Lipschitz constant, “hyperbolic” zero dynamics ...etc

Nonlinear control systems in continuous-time

$$\dot{x} = f(x) + g(x)u$$
$$y = h(x)$$

$x, \ u, \ y$ are *states, controls, and measurements/outputs*. Some assumptions:

- **basic**: origin is an quilibrium, forward complete (or smoothness assumption)
- **technical**: geometry of the space, relative degree, Lipschitz constant, "hyperbolic" zero dynamics ...etc

why always control affine?

Nonlinear control systems in continuous-time

Why not a general nonlinear dynamics;

$$\begin{aligned} \dot{x} &= f(x, u) \\ y &= h(x) \end{aligned}$$

Nonlinear control systems in continuous-time

Why not a general nonlinear dynamics;

$$\dot{x} = f(x, u)$$
$$y = h(x)$$

Can be put in control affine form via dynamics extension; i.e control through derivatives of u

$$u = \xi$$
$$\dot{\xi} = v$$

$$\underbrace{\begin{pmatrix} \dot{x} \\ \dot{\xi} \end{pmatrix}}_{\dot{\tilde{x}}} = \underbrace{\begin{pmatrix} f(x, \xi) \\ 0 \end{pmatrix}}_{\tilde{f}(\tilde{x})} + \underbrace{\begin{pmatrix} 0 \\ I \end{pmatrix} v}_{\tilde{g}(\tilde{x})\tilde{u}}$$
$$y = h(x)$$

Nonlinear control systems in continuous-time

Obviously systems in Euler-Lagrange form (e.g. robots) are control affine by definition;

$$M(q)\dot{\nu} + h(q, \nu) = S\tau + \sum_{k=1}^{m} J_k^\top F_k$$

$$x = \begin{pmatrix} q \\ \nu \end{pmatrix} \implies$$

$$\dot{x} = \underbrace{\begin{pmatrix} \nu \\ -M^{-1}(q)h(q, \nu) \end{pmatrix}}_{f(x)} + \underbrace{\begin{pmatrix} 0 \\ M^{-1}(q)\left[S\tau + \sum_{k=1}^{m} J_k^\top F_k\right] \end{pmatrix}}_{g(x)u}$$

Nonlinear control systems in continuous-time

Obviously systems in Euler-Lagrange form (e.g. robots) are control affine by definition;

$$M(q)\dot{\nu} + h(q,\nu) = S\tau + \sum_{k=1}^{m} J_k^\top F_k$$

$$x = \begin{pmatrix} q \\ \nu \end{pmatrix} \implies$$

$$\dot{x} = \underbrace{\begin{pmatrix} \nu \\ -M^{-1}(q)h(q,\nu) \end{pmatrix}}_{f(x)} + \underbrace{\begin{pmatrix} 0 \\ M^{-1}(q)\left[S\tau + \sum_{k=1}^{m} J_k^\top F_k\right] \end{pmatrix}}_{g(x)u + {\color{red}p(x)w}}$$

Relative degree in continuous-time

$$\dot{x} = f(x) + g(x)u$$
$$y = h(x)$$

Number of times one need to differentiate the output for the input to appear explicitly;

$$\dot{y} = \frac{\partial h}{\partial x} f(x) + \frac{\partial h}{\partial x} g(x)u := L_f h(x) + u \color{red}{L_g h(x)}$$

Relative degree in continuous-time

$$\dot{x} = f(x) + g(x)u$$
$$y = h(x)$$

Number of times one need to differentiate the output for the input to appear explicitly;

$$\dot{y} = \frac{\partial h}{\partial x} f(x) + \frac{\partial h}{\partial x} g(x)u := L_f h(x) + u L_g h(x)$$

$$\begin{aligned}
\ddot{y} &= \frac{\partial L_f h(x)}{\partial x} f(x) + \frac{\partial L_f h(x)}{\partial x} g(x)u + u \frac{\partial L_g h(x)}{\partial x} f(x) + u \frac{\partial u L_g h(x)}{\partial x} g(x)u \\
&= L_f^2 h(x) + u L_g L_f h(x) \\
&\vdots \\
y^{(r)} &= L_f^r h(x) + u L_g L_f^{r-1} h(x)
\end{aligned}$$

Relative degree in continuous-time

$$\begin{aligned}\dot{x} &= f(x) + g(x)u \\ y &= h(x)\end{aligned}$$

relative degree

Well defined relative degree r (at a point x_0) if and only if

- $L_g L_f^\ell h(x) = 0$ on a neighbourhood of x_0 for $\ell = 1, \ldots, r-2$
- $L_g L_f^{r-1} h(x_0) \neq 0$ at x_0.

Zero dynamics in continuous-time

Roughly: "dynamics characterizing the unobservable behavior of a system once **initial conditions** and **controls** are chosen in such a way as to constrain the output to be zero for all time"

Zero dynamics in continuous-time

Roughly: "dynamics characterizing the unobservable behavior of a system once **initial conditions** and **controls** are chosen in such a way as to constrain the output to be zero for all time"

$$
\begin{aligned}
\dot{x} &= f(x) + g(x)u \\
y &= h(x)
\end{aligned}
$$

when well defined r, change of coordinates $z = \phi_1(x)$ and $\eta = \phi_2(x)$ with

$$
\phi_1(x) = \begin{pmatrix} y \\ \dot{y} \\ \vdots \\ y^{(r-1)} \end{pmatrix}, \quad L_g\phi_2(x) = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}
$$

Zero dynamics in continuous-time

Roughly: "dynamics characterizing the unobservable behavior of a system once **initial conditions** and **controls** are chosen in such a way as to constrain the output to be zero for all time"

$$
\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
&\vdots \\
\dot{z}_{r-1} &= z_r \\
\dot{z}_r &= L_f^r h(\cdot) + u L_g L_f^{r-1} h(\cdot) \\
\dot{\eta} &= L_f \phi_2(x) + u L_g \phi_2(x) \\
y &= h(x)
\end{aligned}
$$

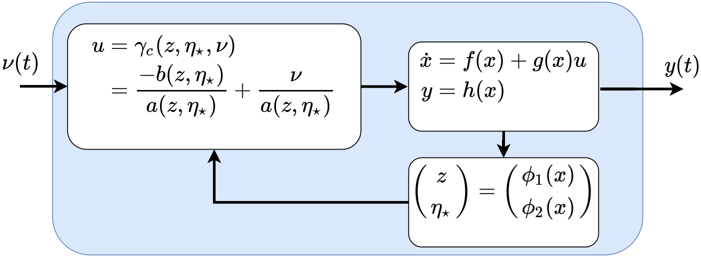
Zero dynamics in continuous-time

Roughly: "dynamics characterizing the unobservable behavior of a system once **initial conditions** and **controls** are chosen in such a way as to constrain the output to be zero for all time"

$$
\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
&\vdots \\
\dot{z}_{r-1} &= z_r \\
\dot{z}_r &= b(z,\eta) + a(z,\eta)u \\
\dot{\eta} &= q(z,\eta) \\
y &= z_1
\end{aligned}
$$

Zero dynamics in continuous-time

Roughly: "dynamics characterizing the unobservable behavior of a system once **initial conditions** and **controls** are chosen in such a way as to constrain the output to be zero for all time"

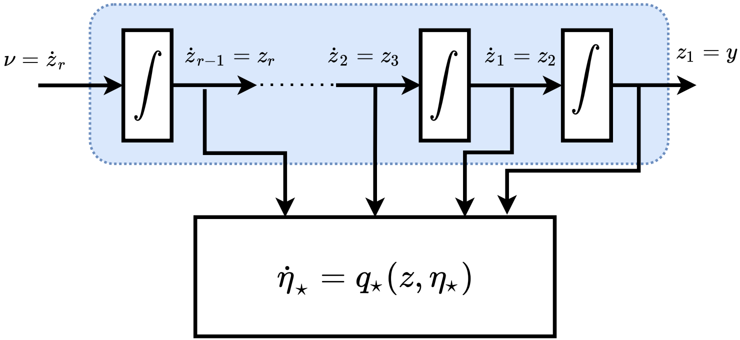


$$\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
&\vdots \\
\dot{z}_{r-1} &= z_r \\
\dot{z}_r &= \nu(t) \\
\dot{\eta}_\star &= q_\star(z, \eta_\star)
\end{aligned}$$

Zero dynamics in continuous-time

Roughly: "dynamics characterizing the unobservable behavior of a system once **initial conditions** and **controls** are chosen in such a way as to constrain the output to be zero for all time"

$$y(s) = \frac{1}{s^r}\nu(s)$$



Zero dynamics in continuous-time

Roughly: "dynamics characterizing the unobservable behavior of a system once **initial conditions** and **controls** are chosen in such a way as to constrain the output to be zero for all time"

The zero dynamics submanifold

$$\begin{aligned}\mathcal{Z}^\star &= \{(z,\ \eta_\star) : z = 0\} \\ &= \{x : h(x) = L_f h(x) = \ldots = L_f^{r-1} h(x) = 0\}\end{aligned}$$

The zero dynamics

$$\begin{aligned}\dot{\eta}_\star &= q_\star(0, \eta_\star), \quad \eta_\star(t_0) = \eta_\star^\circ \\ \dot{x} &= \left[f(x) - g(x)\frac{L_f^r h(x(t))}{L_g L_f^{r-1} h(x(t))}\right]|_{\mathcal{Z}^\star}\end{aligned}$$

Zero dynamics in continuous-time

Roughly: "dynamics characterizing the unobservable behavior of a system once **initial conditions** and **controls** are chosen in such a way as to constrain the output to be zero for all time"

Note that

- the zero dynamics is of dimension $n - r$ (obv. same dimension as $\mathcal{Z}^\star$)
- in a linear setting $f(x) = Ax,\ g(x) = B,\ h(x) = Cx$, zero dynamics coincide with the zeros of the *transmission* zeros of the TF
- if the zero dynamics is stable, *minimum phase*, if not then *non-minimum phase*
- feedback laws requiring the output to follow closely the reference (tracking) typically make the zero dynamics unobservable. (zero dynamics cancellation)

Digression: zero dynamics cancellation

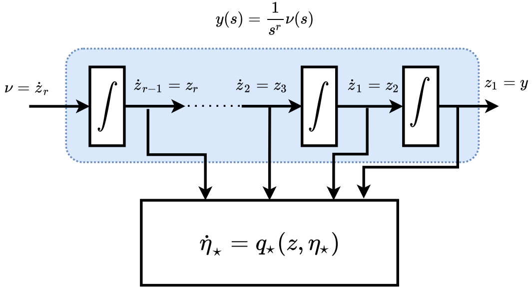
$$
\begin{aligned}
\dot{x} &= f(x) + g(x)u \\
y &= h(x)
\end{aligned}
$$

- assume well defined *ct relative degree* $r < n$
- given y_d, we want roughly $\frac{Y(s)}{Y_d(s)} = 1$, max unobservability/cancellation. In linear: full zero-pole cancellation

Digression: zero dynamics cancellation

$$\dot{x} = f(x) + g(x)u$$
$$y = h(x)$$

- assume well defined *ct relative degree* $r < n$
- given y_d, we want roughly $\frac{Y(s)}{Y_d(s)} = 1$, max unobservability/cancellation. In linear: full zero-pole cancellation



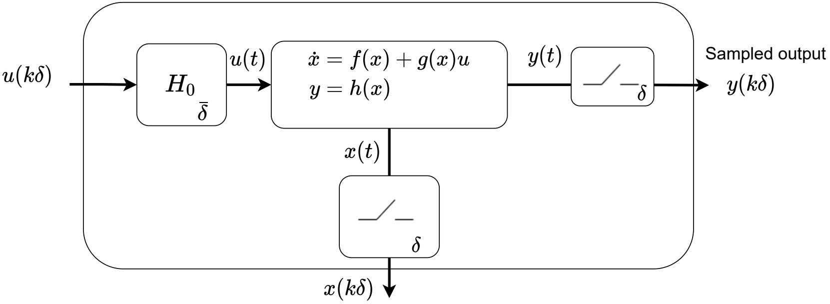
possible to design $\nu(s)$ s.t.

$$\frac{Y(s)}{Y_d(s)} = G_F(s) = \frac{1}{(1+\tau s)^r} \approx 1, \qquad \tau \approx 0$$

Recalls on sampling

Digital control and the sampled-data model

Sensors and actuators are mostly digital in nature;



$$x(k+1) = F^{\delta}(x(k), u(k))$$

Need to design the control taking this "sampled-data" nature in consideration.

Digital control and the sampled-data model

The dynamics at the sampling instants;

$$
\begin{aligned}
x(k+1) &= F^{\delta}(x(k), u(k)) \\
y(k) &= h(x(k))
\end{aligned}
$$

where

$$
F^{\delta}(x(k), u(k)) = x(k) + \int_{k\delta}^{(k+1)\delta} \Big(f(x(\tau)) + g(x(\tau))u(k) \Big) d\tau
$$

The integral is not always exactly computable in closed form.

Digital control and the sampled-data model

The dynamics at the sampling instants admits the Taylor series expansion;

$$
\begin{aligned}
F^{\delta}(x(k), u(k)) &= e^{\delta(\mathrm{L}_f + u(k)\mathrm{L}_g)} x(k) = x(k) + \sum_{i>0} \frac{\delta^i}{i!} (\mathrm{L}_f + u(k)\mathrm{L}_g)^i x(k) \\
&= x(k) + \delta\Big(f(x(k)) + g(x(k))u(k) \Big) \\
&+ \frac{\delta^2}{2!} (\mathrm{L}_f + u(k)\mathrm{L}_g)(f(x(k)) + g(x(k))u(k)) \\
&+ \frac{\delta^3}{3!} (\mathrm{L}_f + u(k)\mathrm{L}_g)(\mathrm{L}_f + u(k)\mathrm{L}_g)(f(x(k)) + g(x(k))u(k)) \\
&+ \dots
\end{aligned}
$$

The power series is not always finite.

Digital control and the sampled-data model

The dynamics at the sampling instants admits the Taylor series expansion;

$$
\begin{aligned}
F^\delta(x(k), u(k)) &= e^{\delta(\mathrm{L}_f + u(k)\mathrm{L}_g)} x(k) = x(k) + \sum_{i>0} \frac{\delta^i}{i!} (\mathrm{L}_f + u(k)\mathrm{L}_g)^i x(k) \\
&= x(k) + \delta\Big(f(x(k)) + g(x(k))u(k)\Big) \\
&+ \frac{\delta^2}{2!} (\mathrm{L}_f + u(k)\mathrm{L}_g)(f(x(k)) + g(x(k))u(k)) \\
&+ \frac{\delta^3}{3!} (\mathrm{L}_f + u(k)\mathrm{L}_g)(\mathrm{L}_f + u(k)\mathrm{L}_g)(f(x(k)) + g(x(k))u(k)) \\
&+ \dots
\end{aligned}
$$

Digital control and the sampled-data model

rewriting

$$F^{\delta}(x(k),u(k)) = x(k) + \sum_{i=1}^{p} \frac{\delta^i}{i!}(\mathrm{L}_f + u(k)\mathrm{L}_g)^i x(k) + O(\delta^{p+1}) = F^{\delta[p]}(x(k),u(k)) + O(\delta^{p+1})$$

Convergence of the series for some δ implies for any $p \geq 1$

$$\|F^{\delta}(x(k),u(k)) - F^{\delta[p]}(x(k),u(k))\| \leq O(\delta^{p+1})$$

The approximate single rate sampled-data model

Example

The differential drive

$$\begin{aligned}\dot{x} &= f(x) + g_1(x)u_1 + g_2(x)u_2 \\ &= \begin{pmatrix}\cos(x_3)\\ \sin(x_3)\\ 0\end{pmatrix} u_1 + \begin{pmatrix}0\\0\\1\end{pmatrix} u_2\end{aligned}$$

Example

Computing the approximate single rate sampled-data model at order $p = 2$

$$
\begin{aligned}
x(k+1) = x(k) &+ \delta\Big(g_1(x(k))u_1(k) + g_2(x(k))u_2(k)\Big) \\
&+ \frac{\delta^2}{2!}(u_1(k)\mathrm{L}_{g_1} + u_2(k)\mathrm{L}_{g_2})(g_1(x(k))u_1(k) + g_2(x(k))u_2(k)) + \ldots
\end{aligned}
$$

Example

Computing the approximate single rate sampled-data model at order $p = 2$

$$x(k+1) = x(k) + \delta\Big(g_1(x(k))u_1(k) + g_2(x(k))u_2(k)\Big)$$
$$+ \frac{\delta^2}{2!}(u_1(k)\mathrm{L}_{g_1} + u_2(k)\mathrm{L}_{g_2})(g_1(x(k))u_1(k) + g_2(x(k))u_2(k)) + \ldots$$

one obtains

$$u_1(k)L_{g_1}(g_1(x(k))u_1(k) + g_2(x(k))u_2(k)) = u_1(k)\frac{\partial}{\partial x}(g_1(x(k))u_1(k) + g_2(x(k))u_2(k))g_1(x)$$
$$= u_1(k)\begin{pmatrix} 0 & 0 & -u_1(k)\,\sin(x_3(k)) \\ 0 & 0 & u_1(k)\,\cos(x_3(k)) \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} \cos(x_3(k)) \\ \sin(x_3(k)) \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Example

Computing the approximate single rate sampled-data model at order $p = 2$

$$x(k+1) = x(k) + \delta\Big(g_1(x(k))u_1(k) + g_2(x(k))u_2(k)\Big) + \frac{\delta^2}{2!}\big(u_1(k)\mathrm{L}_{g_1} + u_2(k)\mathrm{L}_{g_2}\big)\big(g_1(x(k))u_1(k) + g_2(x(k))u_2(k)\big) + \ldots$$

$$u_2(k)L_{g_2}(g_1(x(k))u_1(k) + g_2(x(k))u_2(k)) = u_2(k)\frac{\partial}{\partial x}(g_1(x(k))u_1(k) + g_2(x(k))u_2(k))g_2(x)$$

$$= u_2(k)\begin{pmatrix} 0 & 0 & -u_1(k)\,\sin(x_3(k)) \\ 0 & 0 & u_1(k)\,\cos(x_3(k)) \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -u_1(k)u_2(k)\,\sin(x_3(k)) \\ u_1(k)u_2(k)\,\cos(x_3(k)) \\ 0 \end{pmatrix}$$

Example

Computing the approximate single rate sampled-data model at order $p = 2$

$$x(k+1) = x(k) + \delta\Big(g_1(x(k))u_1(k) + g_2(x(k))u_2(k)\Big) + \frac{\delta^2}{2!}(u_1(k)\mathrm{L}_{g_1} + u_2(k)\mathrm{L}_{g_2})(g_1(x(k))u_1(k) + g_2(x(k))u_2(k)) + \ldots$$

$$x(k+1) = x(k) + \delta \begin{pmatrix} u_1(k)\cos(x_3(k)) \\ u_1(k)\sin(x_3(k)) \\ u_2(k) \end{pmatrix} + \frac{\delta^2}{2!} \begin{pmatrix} -u_1(k)u_2(k)\ \sin(x_3(k)) \\ u_1(k)u_2(k)\ \cos(x_3(k)) \\ 0 \end{pmatrix} + \ldots$$

Example

Comparisons for different values of p

Example

Fortunately, the sampled-data equivalent model can be exactly computed in this case;

$$
\begin{aligned}
x_1(k+1) &= x_1(k) + \int_{k\delta}^{(k+1)\delta} \cos(x_3(\tau))u_1(k)d\tau \\
&= x_1(k) + \frac{u_1(k)}{u_2(k)} \int_{x_3(k)}^{x_3(k+1)} \cos(s)ds = x_1(k) + \frac{u_1(k)}{u_2(k)} \sin(x_3(k+1)) - \sin(x_3(k))
\end{aligned}
$$

Example

$$
\begin{aligned}
x_1(k+1) &= x_1(k) + \int_{k\delta}^{(k+1)\delta} \cos(x_3(\tau))u_1(k)d\tau \\
&= x_1(k) + \frac{u_1(k)}{u_2(k)} \int_{x_3(k)}^{x_3(k+1)} \cos(s)ds = x_1(k) + \frac{u_1(k)}{u_2(k)} \sin(x_3(k+1)) - \sin(x_3(k)) \\
x_2(k+1) &= x_2(k) + \int_{k\delta}^{(k+1)\delta} \sin(x_3(\tau))u_1(k)d\tau \\
&= x_2(k) + \frac{u_1(k)}{u_2(k)} \int_{x_3(k)}^{x_3(k+1)} \sin(s)ds = x_2(k) + \frac{u_1(k)}{u_2(k)} \cos(x_3(k)) - \cos(x_3(k+1)) \\
x_3(k+1) &= x_3(k) + \int_{k\delta}^{(k+1)\delta} u_2(k)d\tau = x_3(k) + \delta u_2(k)
\end{aligned}
$$

Example

Comparisons

Relative degree in discrete-time

For a discrete-time system

$$x(k+1) = F(x(k), u(k))$$
$$y(k) = h(x(k))$$

and denote $F_0^j(x) = \underbrace{F_0(\cdot) \circ F_0(\cdot) \circ \ldots \circ F_0(x)}_{j-times}$ the composition along time with $F_0(x) = F(x, 0)$. The **discrete relative degree** r_d is the integer satisfying

$$\frac{\partial h \circ F_0^\ell \circ F(x,u)}{\partial u} = 0, \quad \ell = 0 \ldots r_d - 2$$
$$\frac{\partial h \circ F_0^\ell \circ F(x,u)}{\partial u} \neq 0, \quad \ell = r_d - 1.$$

$$f : X \to Y$$
$$g : Y \to Z$$
$$g \circ f : X \to Z$$

r_d is the number of time steps for the control to influence the output.

Relative degree under sampling

Applying the definition to the sampled-data equivalent model

$$
\begin{aligned}
x(k+1) &= F^{\delta}(x(k), u(k)) \\
y(k) &= h(x(k))
\end{aligned}
$$

one has, the first delay step

$$
\begin{aligned}
\frac{\partial h \circ F^{\delta}(x,u)}{\partial u} &= \frac{\partial}{\partial u} h \circ \Big(x(k) + \sum_{i>0} \frac{\delta^i}{i!} (\mathrm{L}_f + u(k)\mathrm{L}_g)^i x(k) \Big) \\
&= \frac{\delta^r}{r!} L_g L_f^{r-1} h(x)\big|_{x_0} + O(\delta^{r+1}) \neq 0
\end{aligned}
$$

The discrete relative degree falls to $r_d = 1$ under sampling for all continuous-time systems with well defined $r \geq 1$.

Zero dynamics under sampling

$$\dot{x} = f(x) + g(x)u$$
$$y = h(x)$$

- zero dynamics sub-manifold $n - r$

$$\mathcal{Z}^\star = \{x : h(x) = \ldots = L_f^{r-1} h(x) = 0\}$$

- zero dynamics

$$\dot{x} = \big[f(x) - g(x)\frac{L_f^r h(x(t))}{L_g L_f^{r-1} h(x(t))}\big]|_{\mathcal{Z}^\star}$$

Zero dynamics under sampling

$$\dot{x} = f(x) + g(x)u$$
$$y = h(x)$$

- zero dynamics sub-manifold $n - r$

$$\mathcal{Z}^\star = \{x : h(x) = \ldots = L_f^{r-1} h(x) = 0\}$$

- zero dynamics

$$\dot{x} = \left[f(x) - g(x)\frac{L_f^r h(x(t))}{L_g L_f^{r-1} h(x(t))}\right]\Big|_{\mathcal{Z}^\star}$$

$$x(k+1) = F^\delta(x(k), u(k))$$
$$y(k) = h(x(k))$$

- zero dynamics sub-manifold $n - 1$

$$\mathcal{Z}_{sd}^\star = \{x : h(x(k)) = 0\}$$

- sampling induces additional zero dynamics of dimension $r - 1$

Zero dynamics under sampling

Zero-dynamics (and zeros) under sampling; modified (unstable zero-dynamics may appear[2]). Possible to characterize precisely for LTI SISO systems, no as straighforward for LTI MIMO systems or nonlinear systems.

$$
\begin{aligned}
\dot{x} &= Ax + Bu \\
x(k+1) &= x(k) + \delta(Ax(k) + Bu(k)) + \frac{\delta^2}{2!}A(Ax(k) + Bu(k)) + \dots \\
&= \left[I_n + \delta A + \frac{\delta^2}{2}A^2 + \dots\right]x(k) + \left[\delta B + \frac{\delta^2}{2!}AB + \dots\right]u(k) \\
&:= e^{\delta A}x(k) + \int_0^\delta e^{\tau A}Bd\tau \; u(k) \\
&\equiv e^{\delta A}x(k) + A^{-1}\left[e^{\delta A} - I_n\right]B \; u(k)
\end{aligned}
$$

Zero dynamics under sampling

Zero-dynamics (and zeros) under sampling; modified (unstable zero-dynamics may appear[2]). Possible to characterize precisely for LTI SISO systems, no as straighforward for LTI MIMO systems or nonlinear systems.

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$
$$C = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}$$

Zero dynamics under sampling

Zero-dynamics (and zeros) under sampling; modified (unstable zero-dynamics may appear[2]). Possible to characterize precisely for LTI SISO systems, no as straighforward for LTI MIMO systems or nonlinear systems.

Example triple integrator

$$A_d = \begin{pmatrix} 1 & \delta & \frac{\delta^2}{2!} \\ 0 & 1 & \delta \\ 0 & 0 & 1 \end{pmatrix}, \quad B_d = \begin{pmatrix} \frac{\delta^3}{3!} \\ \frac{\delta^2}{2!} \\ \delta \end{pmatrix}$$

$$C_d = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}$$

$$z_d^\star = \det \begin{pmatrix} z-1 & -\delta & -\frac{\delta^2}{2} & -\frac{\delta^3}{6} \\ 0 & z-1 & -\delta & -\frac{\delta^2}{2} \\ 0 & 0 & z-1 & -\delta \\ 1 & 0 & 0 & 0 \end{pmatrix} = \text{root}\left(\frac{\delta^3}{6}(z^2+4z+1)\right)$$

Zero dynamics under sampling

Zero-dynamics (and zeros) under sampling; modified (unstable zero-dynamics may appear[2]). Possible to characterize precisely for LTI SISO systems, no as straighforward for LTI MIMO systems or nonlinear systems.

Equivalently from the pulse transfer function

$$G_d(z) = \frac{\delta^3(z^2 + 4z + 1)}{3!(z-1)^3}$$

Two new **sampling zeros** $z_d^\star = \{-\sqrt{3} - 2,\ \sqrt{3} - 2\}$. One outside unit disk!

Zero dynamics under sampling

Sampling zeros

As $\delta \to 0$, the $r-1$ sampling zeros are the roots to the Euler-Frobenius polynomials;

$$E_r(z) = b_1^r z^{r-1} + b_2^r z^{r-2} + \ldots + b_r^r$$

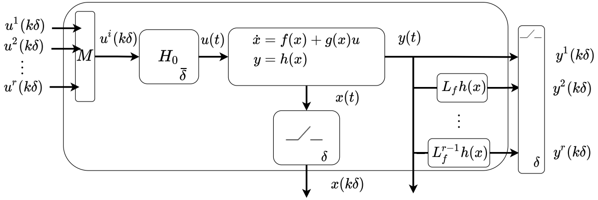
$$b_k^j = \sum_{\ell=1}^{k} (-1)^{k-\ell} \ell^j \binom{j+1}{k-\ell}$$

Sampling zero dynamics

The sampling process induces a further (possibly) unstable <u>sampling zero dynamics</u> of dimension $r-1$ whenever $r \geq 1$

How to preserve the minimum phase property under sampling ?

Multi-rate sampling

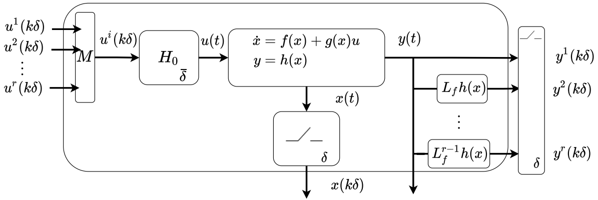


$$\begin{aligned} x(k+1) &= F_r^{\delta}(x(k), u^1(k), \dots, u^r(k)) \\ &= F^{\bar{\delta}}(\cdot, u^r(k)) \circ \cdots \circ F^{\bar{\delta}}(x(k), u^1(k)) \\ Y(k) &= H(x(k)) \end{aligned}$$

Integrating the continuous-time dynamics under the piecewise constant multi-rate controls $\underline{u}(k) = \begin{pmatrix} u^1(k) & u^2(k) & \dots & u^r(k) \end{pmatrix}^\top$, for which the following exapnsion holds;

$$\begin{aligned} F_r^{\delta}(x(k), \underline{u}(k)) &= e^{\bar{\delta}(\mathrm{L}_f + u^1(k)\mathrm{L}_g)} \dots e^{\bar{\delta}(\mathrm{L}_f + u^r(k)\mathrm{L}_g)} x\big|_{x(k)} \\ &= F^{\bar{\delta}}(\cdot, u^r(k)) \circ \cdots \circ F^{\bar{\delta}}(x(k), u^1(k)) \end{aligned}$$

Multi-rate sampling



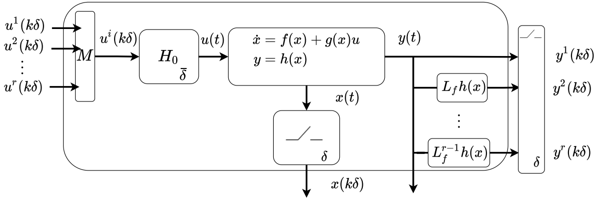
$$
\begin{aligned}
x(k+1) &= F_r^{\delta}(x(k), u^1(k), \dots, u^r(k)) \\
&= F^{\bar\delta}(\cdot, u^r(k)) \circ \cdots \circ F^{\bar\delta}(x(k), u^1(k)) \\
Y(k) &= H(x(k))
\end{aligned}
$$

multiple variations of the input in one sampling interval; extra degrees of freedom.

If in addition; augmented output vector

$$
H(x) = \begin{pmatrix} h(x) & L_f h(x) & \dots & L_f^{r-1} h(x) \end{pmatrix}^\top
$$

Multi-rate sampling



$$
\begin{aligned}
x(k+1) &= F_r^{\delta}(x(k), u^1(k), \dots, u^r(k)) \\
&= F^{\bar\delta}(\cdot, u^r(k)) \circ \cdots \circ F^{\bar\delta}(x(k), u^1(k)) \\
Y(k) &= H(x(k))
\end{aligned}
$$

Statement

a MIMO system with well defined discrete vector relative degree $r_d = (1\ 1\ \dots\ 1)$. More importantly, same as in ct; $\mathcal{Z}^\star_{sd} = \{x \in \mathbb{R}^n : H(x) = 0\} \equiv \mathcal{Z}^\star$

Example, linear systems

$$\Sigma_c : \begin{cases} \dot{x} &= Ax + Bu \\ y &= Cx \end{cases} \xrightarrow{\text{Single rate}} \Sigma_d : \begin{cases} x(k+1) &= A_d x(k) + B_d u(k) \\ y(k) &= C_d x(k) \end{cases}$$

A multi-rate of order r, being a composition of single rate models at sub-intervals of length $\bar{\delta} = \frac{\delta}{r}$;

$$\begin{aligned} x(k+1) &= (A_{\bar{d}})^r x(k) + (A_d)^{r-1} B_{\bar{d}} u^1(k) + \ \dots \ + A_{\bar{d}} B_{\bar{d}} u^{r-1}(k) + \ B_d u^r(k) \\ &= A_m x(k) + B_m \underline{u}(k) \\ y(k) &= \begin{pmatrix} C_{\bar{d}} x(k) \\ \vdots \\ C_{\bar{d}} (A_{\bar{d}})^{r-1} x(k) \end{pmatrix} \\ &= C_m x(k) \end{aligned}$$

$$\begin{aligned} A_d &= e^{\delta A}, \quad B_d = \int_0^{\delta} e^{\tau A} B d\tau \\ A_{\bar{d}} &= e^{\bar{\delta} A}, \quad B_{\bar{d}} = \int_0^{\bar{\delta}} e^{\tau A} B d\tau \end{aligned}$$

Example, linear systems

For the triple integrator, the single-rate model gave us;

$$A_d = \begin{pmatrix} 1 & \delta & \frac{\delta^2}{2!} \\ 0 & 1 & \delta \\ 0 & 0 & 1 \end{pmatrix}, \quad B_d = \begin{pmatrix} \frac{\delta^3}{3!} \\ \frac{\delta^2}{2!} \\ \delta \end{pmatrix}$$

$$C_d = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}$$

$$z_d^\star = \det \begin{pmatrix} z-1 & -\delta & -\frac{\delta^2}{2} & -\frac{\delta^3}{6} \\ 0 & z-1 & -\delta & -\frac{\delta^2}{2} \\ 0 & 0 & z-1 & -\delta \\ 1 & 0 & 0 & 0 \end{pmatrix} = \text{root}\left(\frac{\delta^3}{6}(z^2+4z+1)\right) = \{{\color{red}-\sqrt{3}-2},\ \sqrt{3}-2\}$$

Example, linear systems

The continuous-time relative degree $r = 3$, apply a multi-rate of order r

$$x(k+1) = A_m x(k) + B_m \underline{u}(k)$$

$$A_{m=3} = \begin{pmatrix} 1 & 3\bar{\delta} & \frac{9\bar{\delta}^2}{2!} \\ 0 & 1 & 3\bar{\delta} \\ 0 & 0 & 1 \end{pmatrix}, \quad B_{m=3} = \begin{pmatrix} \frac{19\bar{\delta}^3}{6} & \frac{7\bar{\delta}^3}{6} & \frac{\bar{\delta}^3}{6} \\ \frac{5\bar{\delta}^2}{2} & \frac{3\bar{\delta}^2}{2} & \frac{\bar{\delta}^2}{2} \\ \bar{\delta} & \bar{\delta} & \bar{\delta} \end{pmatrix}$$

For which, taking as output an extended vector $y = C_{m=3} x(k)$ we can verify that;

$$z_d^\star = \emptyset$$

We recovered the fact that the original system has no zeros !. In fact, if the original system had some zeros, we will recover precisely those!

Example, linear systems

The choice of the multi-rate order is crucial; applying a multi-rate of order two the same triple integrator

$$A_{m=1} = \begin{pmatrix} 1 & 2\bar{\delta} & \frac{4\bar{\delta}^2}{2!} \\ 0 & 1 & 2\bar{\delta} \\ 0 & 0 & 1 \end{pmatrix}, \quad B_{m=3} = \begin{pmatrix} \frac{7\bar{\delta}^3}{6} & \frac{\bar{\delta}^3}{6} \\ \frac{3\bar{\delta}^2}{2} & \frac{\bar{\delta}^2}{2} \\ \bar{\delta} & \bar{\delta} \end{pmatrix}$$

$$z_d^\star = \text{det} \begin{pmatrix} z-1 & -2\,\bar{\delta} & -2\,\bar{\delta}^2 & -\frac{7\,\bar{\delta}^3}{6} & -\frac{\bar{\delta}^3}{6} \\ 0 & z-1 & -2\,\bar{\delta} & -\frac{3\,\bar{\delta}^2}{2} & -\frac{\bar{\delta}^2}{2} \\ 0 & 0 & z-1 & -\bar{\delta} & -\bar{\delta} \\ 1 & 0 & 0 & 0 & 0 \\ 1 & \bar{\delta} & \frac{\bar{\delta}^2}{2} & 0 & 0 \end{pmatrix} = \text{root}\,\frac{\bar{\delta}^6(5z+1)}{6} = \{-\frac{1}{5}\}$$

The differential drive revisited

Back to **the exact SD model** of our differential drive;

$$x(k+1) = F^{\delta}(x(k), u_1(k), u_2(k))$$

$$F^{\delta}(x(k), u_1(k), u_2(k)) = \begin{pmatrix} x_1(k) + \frac{u_1(k)}{u_2(k)}\big[\sin(x_3(k) + \delta u_2(k)) - \sin(x_3(k))\big] \\ x_2(k) + \frac{u_1(k)}{u_2(k)}\big[\cos(x_3(k)) - \cos(x_3(k) + \delta u_2(k))\big] \\ x_3(k) + \delta u_2(k) \end{pmatrix}$$

The differential drive revisited

Applying a multi-rate of order 2 on u_2, a multi-rate equivalent model can be computed as;

$$\begin{aligned} x(k+1) = F_2^{\delta}(x(k), u_1(k), u_2(k)) &= F^{\delta}(x(k+\frac{1}{2}), u_1(k), \underline{u_2(k)}) \\ &= F^{\bar{\delta}}(\cdot, u_1(k), u_2^2(k)) \circ F^{\bar{\delta}}(x(k), u_1(k), u_2^1(k)) \end{aligned}$$

We already have;

The differential drive revisited

Applying a multi-rate of order 2 on u_2, a multi-rate equivalent model can be computed as;

$$
\begin{aligned}
x(k+1) = F_2^{\delta}(x(k), u_1(k), u_2(k)) &= F^{\delta}(x(k+\frac{1}{2}), u_1(k), \underline{u_2(k)}) \\
&= F^{\bar{\delta}}(\cdot, u_1(k), u_2^2(k)) \circ F^{\bar{\delta}}(x(k), u_1(k), u_2^1(k))
\end{aligned}
$$

We already have for $\bar{\delta} = \frac{\delta}{2}$;

$$
F^{\delta}(x(k), u_1(k), u_2^1(k)) = \begin{pmatrix} x_1(k) + \frac{u_1(k)}{u_2^1(k)}\left[\sin(x_3(k) + \bar{\delta} u_2^1(k)) - \sin(x_3(k))\right] \\ x_2(k) + \frac{u_1(k)}{u_2^1(k)}\left[\cos(x_3(k)) - \cos(x_3(k) + \bar{\delta} u_2^1(k))\right] \\ x_3(k) + \bar{\delta} u_2^1(k) \end{pmatrix}
$$

The differential drive revisited

Applying a multi-rate of order 2 on u_2, a multi-rate equivalent model can be computed as;

$$\begin{aligned} x(k+1) &= F_2^\delta(x(k), u_1(k), u_2(k)) = F^\delta(x(k+\frac{1}{2}), u_1(k), \underline{u_2(k)}) \\ &= F^\delta(\cdot, u_1(k), u_2^2(k)) \circ F^\delta(x(k), u_1(k), u_2^1(k)) \end{aligned}$$

and for the first part;

$$F^\delta(\cdot, u_1(k), u_2^2(k)) = \begin{pmatrix} x_1(k+\frac{1}{2}) + \frac{u_1(k)}{u_2^2(k)}\left[\sin(x_3(k) + \bar{\delta}u_2^1(k) + \bar{\delta}u_2^2(k) - \sin(x_3(k) + \bar{\delta}u_2^1(k))\right] \\ x_2(k+\frac{1}{2}) + \frac{u_1(k)}{u_2^2(k)}\left[\cos(x_3(k) + \bar{\delta}u_2^1(k)) - \cos(x_3(k) + \bar{\delta}u_2^1(k) + \bar{\delta}u_2^2(k)\right] \\ x_3(k+\frac{1}{2}) + \bar{\delta}u_2^2(k) \end{pmatrix}$$

The differential drive revisited

Applying a multi-rate of order 2 on u_2, a multi-rate equivalent model can be computed as;

$$
\begin{aligned}
x(k+1) &= F^{\delta}(\cdot, u_1(k), u_2^2(k)) \circ F^{\delta}(x(k), u_1(k), u_2^1(k)) \\
&= x(k) + \begin{pmatrix} \frac{u_1(k)}{u_2^1(k)}\left[\sin(x_3(k) + \bar{\delta}u_2^1(k)) - \sin(x_3(k))\right] \\ \frac{u_1(k)}{u_2^1(k)}\left[\cos(x_3(k)) - \cos(x_3(k) + \bar{\delta}u_2^1(k))\right] \\ \bar{\delta}u_2^1(k) \end{pmatrix} \\
&+ \begin{pmatrix} \frac{u_1(k)}{u_2^2(k)}\left[\sin(x_3(k) + \bar{\delta}u_2^1(k) + \bar{\delta}u_2^2(k) - \sin(x_3(k) + \bar{\delta}u_2^1(k))\right] \\ \frac{u_1(k)}{u_2^2(k)}\left[\cos(x_3(k) + \bar{\delta}u_2^1(k)) - \cos(x_3(k) + \bar{\delta}u_2^1(k) + \bar{\delta}u_2^2(k)\right] \\ \bar{\delta}u_2^2(k) \end{pmatrix}
\end{aligned}
$$

The differential drive revisited

Matlab break

Digression on exact compuatbility

Even when the continuous-time system may not admit a closed form finitely computable exact sampled-data model, a transformed (through coordinates change and/or feedback) version may admit a closed form SD finitely computable equivalent model.

Digression on exact compuatbility

$$\dot{x} = \begin{pmatrix} u_1 \cos(x_3) \\ u_1 \sin(x_3) \\ u_2 \end{pmatrix} \xrightarrow{\text{scaling}} \dot{x} = \begin{pmatrix} u_1 \\ u_1 \tan(x_3) \\ u_2 \end{pmatrix}$$

Then applying coordinates

$$z = \phi(x) = \begin{pmatrix} x_1 \\ \tan(x_3) \\ x_2 \end{pmatrix} \implies \dot{z} = \begin{pmatrix} u_1 \\ \frac{u_2}{\cos^2(x_3)} \\ z_2 u_1 \end{pmatrix}$$

And feedback transformation

$$v = \gamma(x, u) = \begin{pmatrix} u_1 \\ \cos^2(x_3) u_2 \end{pmatrix} \implies \dot{z} = \begin{pmatrix} v_1 \\ v_2 \\ z_2 v_1 \end{pmatrix}$$

Digression on exact compuatbility

Starting from this *chained form*, applying the single rate SD equivalent model definition;

$$
\begin{aligned}
F^{\delta}(z(k), v(k)) &= z(k) + \sum_{i>0} \frac{\delta^i}{i!} (v_1(k)\mathrm{L}_{g_1} + v_2(k)\mathrm{L}_{g_2})^i z(k) \\
&= z(k) + \delta\Big(g_1(z(k))v_1(k) + g_2(z(k))v_2(k)\Big) \\
&+ \frac{\delta^2}{2!}(\mathrm{L}_f + u(k)\mathrm{L}_g)(f(x(k)) + g(x(k))u(k)) \\
&+ \cancel{\frac{\delta^3}{3!}(\mathrm{L}_f + u(k)\mathrm{L}_g)(\mathrm{L}_f + u(k)\mathrm{L}_g)(f(x(k)) + g(x(k))u(k))} \\
&+ \cancel{\dots}
\end{aligned}
$$

Digression on exact compuatbility

Thus getting

$$z(k) = z(k) + \delta \begin{pmatrix} v_1(k) \\ v_2(k) \\ v_1(k)\, z_2(k) \end{pmatrix} + \frac{\delta^2}{2!} \begin{pmatrix} 0 \\ 0 \\ v_1(k)\, v_2(k) \end{pmatrix}$$

Obv. the multi-rate sampled-data model will be finitely and exactly compuatble as well!

What does this mean? possible to design the control based on the exact finite simple SD model under assumed transformations

MPC and multi-rate sampling

MPC and cheap MPC

Model predictive control: solves a receding horizon optimal control problem. The go-to tool, nowadays, for constrained control problems;

Started in the industry before academia caught up (MAC, DMC, GPC);

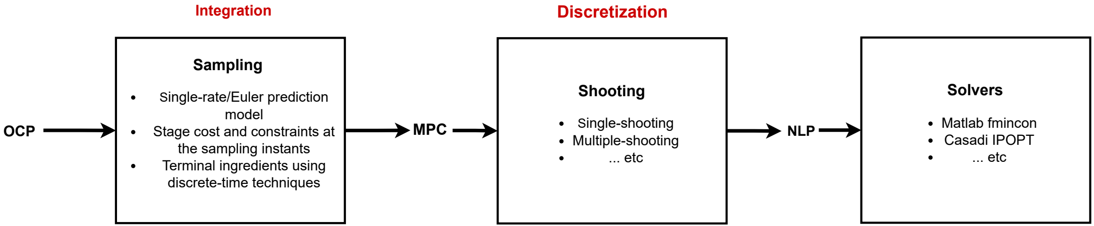
- J. Richalet et. al, "*Model Predictive Heuristic Control: Application to Industrial Processes*". Automatica, pp. 413–428, 1978
- C. Cutler and B. Ramaker, "*Dynamic Matrix Control - A Computer Control Algorithm*". Automatic Control Conference, 1980
- D.W. Clarke et. al, "*Generalized Predictive Control - Part I*". Automatica, pp. 149–160, 1987

Stability guarantees by modifying the optimization problem and horizons;

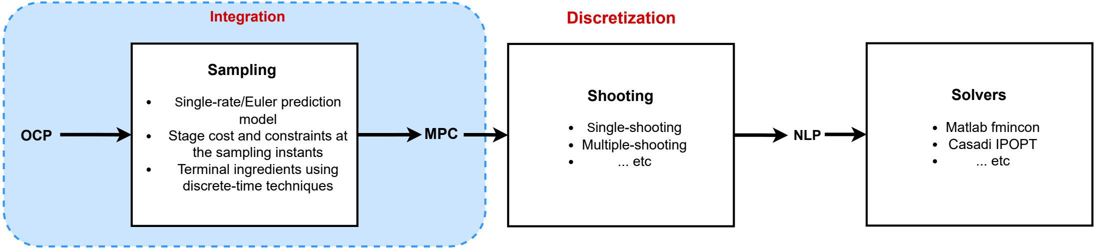
- E. Camacho and C. Bordons, "*Model Predictive control*". Springer, 2007
- L. Grüne and J. Pannek, "*Nonlinear model predictive control*", Springer, 2017
- F. Borelli, A. Bemporad and M. Morari, "*Predictive Control for Linear and Hybrid Systems*", Springer, 2017

MPC and cheap MPC

Model predictive control: solves a receding horizon optimal control problem. The go-to tool, nowadays, for constrained control problems;



MPC and cheap MPC



MPC and cheap MPC

ct ocp

$$
\begin{aligned}
V^\star = \min & V_{t_f}(x(t_f)) + \\
& \int_{t_0}^{t_f} \ell(y(t), y_d(t), u(t)) \\
st.\ \dot{x} = f(x(t)) + g(x(t))u(t), & \quad y(t) = h(x(t)) \\
& x(t) \in \mathcal{X}, t \in [t_0, t_f], \\
& u(t) \in \mathcal{U}, t \in [t_0, t_f]
\end{aligned}
$$

MPC and cheap MPC

ct ocp

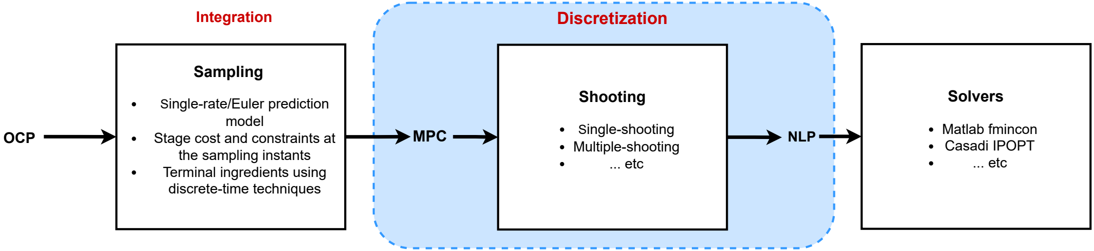
$$
\begin{aligned}
V^\star = \min & V_{t_f}(x(t_f)) + \\
& \int_{t_0}^{t_f} \ell(y(t), y_d(t), u(t)) \\
st. \;\; & \dot{x} = f(x(t)) + g(x(t))u(t), \;\; y(t) = h(x(t)) \\
& x(t) \in \mathcal{X}, t \in [t_0, t_f], \\
& u(t) \in \mathcal{U}, t \in [t_0, t_f]
\end{aligned}
$$

MPC

$$
\begin{aligned}
V^\star = \min & V_{n_p}(x(k+n_p)) + \\
& \sum_{i=1}^{n_p-1} \ell(y(k+i), y_d(k+i), u(k+i-1)) \\
st. \;\; & x(k+1) = F^\delta(x(k), u(k)), \;\; y(k) = h(x(k)) \\
& x(k+i) \in \mathcal{X}, i = 1 \ldots, n_p - 1 \\
& u(k+j) \in \mathcal{U}, \;\; j = 0, \ldots, n_c - 1 \\
& u(k+j) = u_{term}, \;\; j = n_c, \ldots, n_p - 1 \\
& x(k+n_p) \in \mathcal{X}_{n_p}
\end{aligned}
$$

- terminal ingredients: e.g. designed using stabilizing LQR ingredients
- bounds on minimum length of n_p and imposing $n_c << n_p$

MPC and cheap MPC



MPC and cheap MPC

Example: single shooting

MPC

$$
\begin{aligned}
V^\star = \min & V_{n_p}(x(k+n_p)) + \\
& \sum_{i=1}^{n_p-1} \ell(y(k+i), y_d(k+i), u(k+i-1)) \\
st.\ & x(k+1) = F^\delta(x(k), u(k)), \quad y(k) = h(x(k)) \\
& x(k+i) \in \mathcal{X}, i = 1 \ldots, n_p - 1 \\
& u(k+j) \in \mathcal{U}, \quad j = 0, \ldots, n_c - 1 \\
& u(k+j) = u_{term}, \quad j = n_c, \ldots, n_p - 1 \\
& x(k+n_p) \in \mathcal{X}_{n_p}
\end{aligned}
$$

MPC and cheap MPC

Example: single shooting

MPC

$$
\begin{aligned}
V^\star = \min & V_{n_p}(x(k+n_p)) + \\
& \sum_{i=1}^{n_p-1} \ell(y(k+i), y_d(k+i), u(k+i-1)) \\
st.\ x(k+1) = & F^\delta(x(k), u(k)), \quad y(k) = h(x(k)) \\
x(k+i) \in & \mathcal{X}, i = 1 \ldots, n_p - 1 \\
u(k+j) \in & \mathcal{U}, \quad j = 0, \ldots, n_c - 1 \\
u(k+j) = & u_{term}, \quad j = n_c, \ldots, n_p - 1 \\
x(k+n_p) \in & \mathcal{X}_{n_p}
\end{aligned}
$$

NLP

$$
\begin{aligned}
V^\star = & \min \phi^{ss}(\mathbf{w}, x_0) \\
st.\ & h_{\mathsf{ineq}}(\mathbf{w}, x_0) \leq 0 \\
& h_{\mathsf{eq}}(\mathbf{w}, x_0) = 0
\end{aligned}
$$

MPC and cheap MPC

Example: single shooting

MPC

$$V^\star = \min V_{n_p}(x(k+n_p)) + \sum_{i=1}^{n_p-1} \ell(y(k+i), y_d(k+i), u(k+i-1))$$

$$st.\ x(k+1) = F^\delta(x(k), u(k)),\quad y(k) = h(x(k))$$

$$x(k+i) \in \mathcal{X}, i = 1\ldots, n_p - 1$$

$$u(k+j) \in \mathcal{U},\quad j = 0, \ldots, n_c - 1$$

$$u(k+j) = u_{term},\quad j = n_c, \ldots, n_p - 1$$

$$x(k+n_p) \in \mathcal{X}_{n_p}$$

NLP

$$V^\star = \min \phi^{ss}(\mathbf{w}, x_0)$$

$$st.\ h_{\mathsf{ineq}}(\mathbf{w}, x_0) \leq 0$$

$$h_{\mathsf{eq}}(\mathbf{w}, x_0) = 0$$

$$\begin{aligned}
w &= [u(k),\ u(k+1), \ldots, u(k+N_p-1)] \\
F_0 &= F^\delta(x_0, w_1) \\
F_k &= F^\delta(F_{k-1}, w) = F^\delta(\cdot, w) \circ \cdots \circ F_0 \\
\ell(k+i) &= \ell(y(k+i), y_d(k+i), w) \\
&= \ell(h \circ F_{k+i}, y_d(k+i), w) \\
&\vdots
\end{aligned}$$

Example

Linear: revisiting the triple integrator, our MPC problem;

$$V = \text{min}_u \sum_{i=1}^{n_p} \left(\|y(k+i) - y_d(k+i])\|_Q + \|u(k+i-1])\|_R \right)$$

$$\text{s.t } x(k+i) = A_d x(k+i-1) + B_d u(k+i-1)$$
$$y(k+i) = C_d x(k+i)$$

where

$$A_d = \begin{pmatrix} 1 & \delta & \frac{\delta^2}{2} \\ 0 & 1 & \delta \\ 0 & 0 & 1 \end{pmatrix}, \quad B_d = \begin{pmatrix} \frac{\delta^3}{3!} \\ \frac{\delta^2}{2!} \\ \delta \end{pmatrix}, \quad C_d = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}$$

Example

Single shooting, propagating the cost using the dynamics from initial state;

$$\begin{pmatrix} y(k+1) \\ y(k+2) \\ \vdots \\ y(k+n_p) \end{pmatrix} = \begin{pmatrix} C_d A_d \\ C_d A_d^2 \\ \vdots \\ C_d A_d^{n_p} \end{pmatrix} x_0 + \begin{pmatrix} C_d B_d & 0 & \dots & 0 \\ C_d A_d B_d & C_d B_d & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C_d A_d^{n_p-1} B_d & C_d A_d^{n_p-2} B_d & \dots & C_d A_d^{n_p-n_c} B_d \end{pmatrix} w$$

$$y_e(k) = A_e x_0 + B_e w$$

Example

we get the optimization problem (in this case QP, not NLP)

$$
\begin{aligned}
V(w, x_0) &= \mathsf{min}_w \ (A_e x_0 + B_e w - \underline{y}_d)^\top Q_e (A_e x_0 + B_e w - \underline{y}_d) + w^\top R_e w \\
&= \mathsf{min}_w \ \frac{1}{2} w^\top 2(R_e + B_e^\top Q_e B_e) w + (A_e x_0 - \underline{y}_d)^\top 2(Q_e B_e) w \\
&+ \frac{1}{2}(A_e x_0 - \underline{y}_d)^\top Q_e (A_e x_0 - \underline{y}_d) \\
&= \mathsf{min}_w \ \frac{1}{2} w^\top H w + (A_e x_0 - \underline{y}_d)^\top F w + (A_e x_0 - \underline{y}_d)^\top Q_e (A_e x_0 - \underline{y}_d)
\end{aligned}
$$

MPC and cheap MPC

Back to our MPC problem:

$$
\begin{aligned}
V^{\star} = \min & V_{n_p}(x(k+n_p)) + \\
& \sum_{i=1}^{n_p-1} \ell(y(k+i), y_d(k+i), u(k+i-1)) \\
st.\ & x(k+1) = F^{\delta}(x(k), u(k)), \quad y(k) = h(x(k)) \\
& x(k+i) \in \mathcal{X}, i = 1 \ldots, n_p - 1 \\
& u(k+j) \in \mathcal{U}, \ \ j = 0, \ldots, n_c - 1 \\
& u(k+j) = u_{term}, \ \ j = n_c, \ldots, n_p - 1 \\
& x(k+n_p) \in \mathcal{X}_{n_p}
\end{aligned}
$$

MPC and cheap MPC

cheap MPC: quadratic cost with $\epsilon \approx 0$

$$\ell(y(k+i), y_d(k+i), u(k+i-1)) = \|y(k+i) - y_d(k+i)\|_Q^2 + \epsilon \|u(k+i-1)\|_R^2$$

unconstrained: both "path" and terminal constraints are not considered

MPC and cheap MPC

cheap MPC: quadratic cost with $\epsilon \approx 0$

$$\ell(y(k+i), y_d(k+i), u(k+i-1)) = \|y(k+i) - y_d(k+i)\|_Q^2 + \epsilon\|u(k+i-1)\|_R^2$$

unconstrained: both "path" and terminal constraints are not considered

Statement

The feedback solving the cheap optimal control problem achieves *zero ideal performance* if and only if the system is minimum phase

If not minimum phase: at best no perfect tracking/regulation and at worst instability.

MPC and cheap MPC

Matlab break

Cheap unconstrained MPC implies cancellation

Example: triple integrator continued $(r = 3)$ $\begin{cases} G(s) & = \frac{1}{s^3} \\ G_d(z) & = \frac{\delta(z+0.26)(z+3.73)}{z^3-3z^2+3z-1} \end{cases}$

$$V(w, x_0) = \mathsf{min}_w \ \frac{1}{2} w^\top H w + (A_e x_0 - \underline{y}_d)^\top F w + (A_e x_0 - \underline{y}_d)^\top Q_e (A_e x_0 - \underline{y}_d)$$

$$\frac{\partial V(w, x_0)}{\partial w} = Hw + F(A_e x_0 - \underline{y}_d) = 0$$

$$w^\star = -H^\# F(A_e x_0 - \underline{y}_d)$$

Cheap unconstrained MPC implies cancellation

Example: triple integrator continued $(r = 3)$ $\begin{cases} G(s) & = \frac{1}{s^3} \\ G_d(z) & = \frac{\delta(z+0.26)(z+3.73)}{z^3-3z^2+3z-1} \end{cases}$

$$
\begin{aligned}
V(w, x_0) &= \min_w \frac{1}{2} w^\top H w + (A_e x_0 - \underline{y}_d)^\top F w + (A_e x_0 - \underline{y}_d)^\top Q_e (A_e x_0 - \underline{y}_d) \\
\frac{\partial V(w, x_0)}{\partial w} &= Hw + F(A_e x_0 - \underline{y}_d) = 0 \\
w^\star &= -H^\# F(A_e x_0 - \underline{y}_d)
\end{aligned}
$$

receding horizon implementation

$$
u^\star(k) = \begin{pmatrix} 1 & 0 & \dots & 0 \end{pmatrix} (B_e^\top Q_e B_e + R_e)^{-1} B_e^\top Q_e (A_e x_0 - \underline{y}_d)
$$

Cheap unconstrained MPC implies cancellation

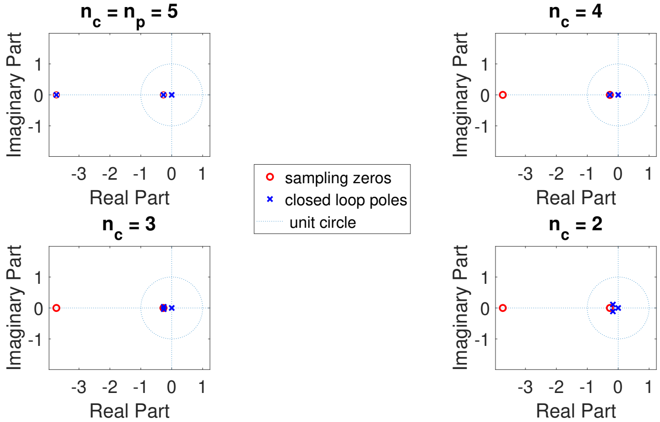
Example: triple integrator continued $(r = 3)$ $\begin{cases} G(s) & = \frac{1}{s^3} \\ G_d(z) & = \frac{\delta(z+0.26)(z+3.73)}{z^3-3z^2+3z-1} \end{cases}$

Matlab break

Cheap unconstrained MPC implies cancellation

Example: triple integrator continued $(r=3)$ $\begin{cases} G(s) &= \frac{1}{s^3} \\ G_d(z) &= \frac{\delta(z+0.26)(z+3.73)}{z^3-3z^2+3z-1} \end{cases}$

fixing $R=0$, $Q_e = \mathsf{I}$, $n_p=5$, and checking for different n_c;



Cheap unconstrained MPC implies cancellation

- From the example, when $n_p = n_c$ the MPC performs cancellation
- Since $G_d(z)$ by construction is nonminimum phase due to sampling zeros $\rightarrow$ possibly unstable closed loop
- This is true from the general explicit solution of linear unconstrained MPC, and is very intuitive to see for tracking nonlinear MPC
- This issue of internal stability was not emphasized
- Understanding of this pathology intuitively leads to the consideration of sampled-data multi-rate techniques

Idea: multi-rate sampling in cheap MPC

- **cheap unconstrained tracking** MPC cancels the prediction model zero dynamics;

  prediction model is typically the sampled-data model (non-minimum phase)

  multi-rate preserves the zero dynamics sub-manifold and zero dynamics stability properties

- idea: use multi-rate sampling to design the control (prediction model) and recover the stability properties and the zero-dynamics submanifold of the underlying continuous-time process

- simpler MPC problem. no need for terminal ingredients

Solution: multi-rate for prediction

problem

Find a bounded digital feedback ensuring that at the sampling instants that:
$y(k) = y_d(k)$, $k \geq k^\star$ with $y_r(k) = y_r(k\delta)$ by minimizing:

$$V^\star = \min \sum_{i=1}^{n_p} \left(\|y(k+i) - y_d(k+i)\|_Q^2 + \epsilon \|u(k+i-1)\|_R^2 \right)$$

with $Q, R > 0$ being appropriate penalizing weights and $\epsilon \in \mathbb{R}$ small.

Solution: multi-rate for prediction

modified/equivalent problem

$$V^\star = \min \sum_{i=1}^{n_p} \left(\|Y(k+i) - Y_d(k+i)\|_Q^2 + \epsilon \|\underline{u}(k+i-1)\|_R^2 \right) + \cancel{V_{n_p}(\cdot)}$$

$$st. \quad x(k+1) = F_r^\delta(x(k), u^1(k), \ldots, u^r(k)), \quad \cancel{x(k+n_p) \in \mathcal{X}_{n_p}}$$

Solution: multi-rate for prediction

modified/equivalent problem

$$V^\star = \min \sum_{i=1}^{n_p} \left(\|Y(k+i) - Y_d(k+i)\|_Q^2 + \epsilon \|\underline{u}(k+i-1)\|_R^2 \right) + \cancel{V_{n_p}(\cdot)}$$

$$st. \quad x(k+1) = F_r^\delta(x(k), u^1(k), \ldots, u^r(k)), \quad \cancel{x(k+n_p) \in \mathcal{X}_{n_p}}$$

solution overview:

- replace prediction model with multi-rate model: more optimization variables
- extend ref and output vectors with their higher $r-1$ derivatives
- solve in the extended optimization variables vector (the multi-rate controls)

Solution: multi-rate for prediction

Statement

Given a ct nonlinear input affine SISO system possessing relative degree $r \leq n$. There exists $\delta^\star > 0$ such that for all $\delta \in [0, \delta^\star[$ the modified nmpc problem is solvable with internal stability for all $n_p = n_c \geq 1$ and ϵ small enough. The feedback is *unique* and defined implicitly as a formal series in powers of δ solution to;

$$K(x, u_e)Q_e(B_e(x) + R_e)u_e = B_e(.)(Y_{d_e} - A_eY_e - \Theta(x, u_e))$$

for $Y_{d_e},\ Y_e,\ u_e$ predictions of the augmented reference, output and multi-rate feedback respectively. Moreover, $K_e(x, u_e),\ B_e(x),\ A_e,\ Q_e,\ R_e$ are matrices and matrix-valued functions depending on the dynamics, penalizing weights and horizons respectively[1]

[1] M. Elobaid, M. Mattioni, S. Monaco and D. Normand-Cyrot, "*On unconstrained MPC through multirate sampling*", IFAC PapersOnline, pp 388-393, 2019

Application example

Example: the unicycle revisited

$$\begin{aligned}\dot{x} &= v\cos\theta\\ \dot{y} &= v\sin\theta\\ \dot{\theta} &= \omega\end{aligned}$$

can be put in chained form[1]

$$\begin{aligned}\dot{z}_1 &= \nu_1\\ \dot{z}_2 &= \nu_2\\ \dot{z}_3 &= z_2\nu_1\end{aligned}$$

Application example

Example: the unicycle revisited

multi-rate of order 2 on ν_2;

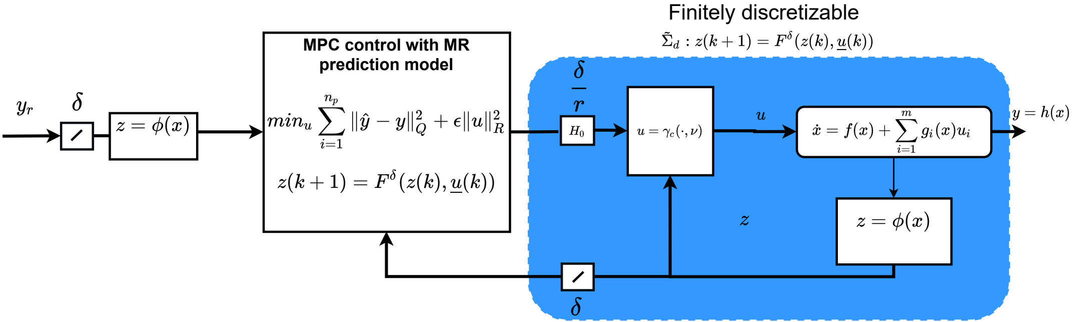
$$z_1(k+1) = z_1(k) + \delta\nu_1(k)$$
$$\underline{z}(k+1) = A^2(\bar{\delta}, \nu_1(K))\underline{z}(k) + R(\bar{\delta}, \nu_1(k))\underline{\nu_2}(k)$$

with $\underline{z} = (z_2 \quad z_3)^\top$ and

$$A^2(\bar{\delta}, \nu_1(k)) = \begin{pmatrix} 1 & 0 \\ 2\bar{\delta}\nu_1(k) & 1 \end{pmatrix}$$
$$R(\bar{\delta}, \nu_1) = \begin{pmatrix} \bar{\delta} & \bar{\delta} \\ \frac{3\bar{\delta}^2}{2}\nu_1(k) & \frac{\bar{\delta}^2}{2}\nu_1(k) \end{pmatrix}$$

Application example

Example: the unicycle revisited First test: with preliminary coordinates change and feedback



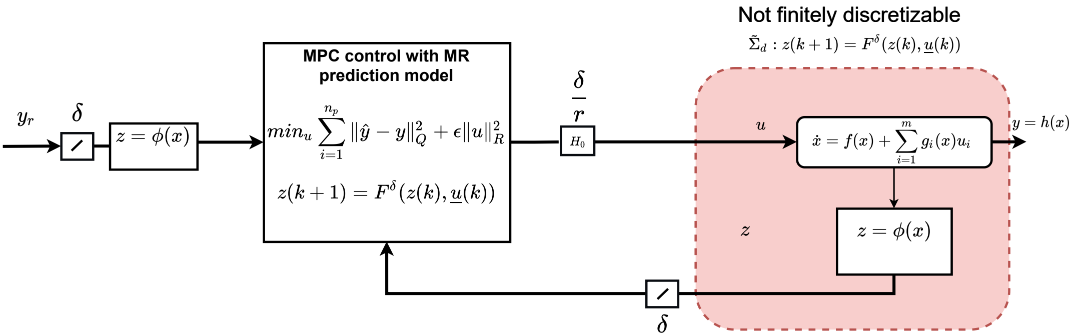
Application example

Matlab/Simulink break

Application example

Example: the unicycle revisited

Second test: remove the preliminary feedback



Application example

Matlab/Simulink break

Drawbacks and can we do better?

Drawbacks

- asynchronous Sample and hold devices.
- faster actuators are required (expensive and more opti. variables).
- upgrading existing mpc control loops is not straightforward.

multi-rate was originally introduced as a reference planner [1], and mpc greatly affected by quality of reference:

use multi-rate to provide apriori admissible references for the mpc block

[1] S. Monaco and D. Normand-Cyrot, "*An introduction to motion planning under multirate digital control*", IEEE CDC 1992

Better solution: multi-rate for reference planning

original problem

Find a bounded digital feedback ensuring that at the sampling instants that:
$y(k) = y_d(k)$, $k \geq k^\star$ with $y_r(k) = y_r(k\delta)$ by minimizing:

$$V^\star = \min \sum_{i=1}^{n_p} \left(\|y(k+i) - y_d(k+i)\|_Q^2 + \epsilon \|u(k+i-1)\|_R^2 \right)$$

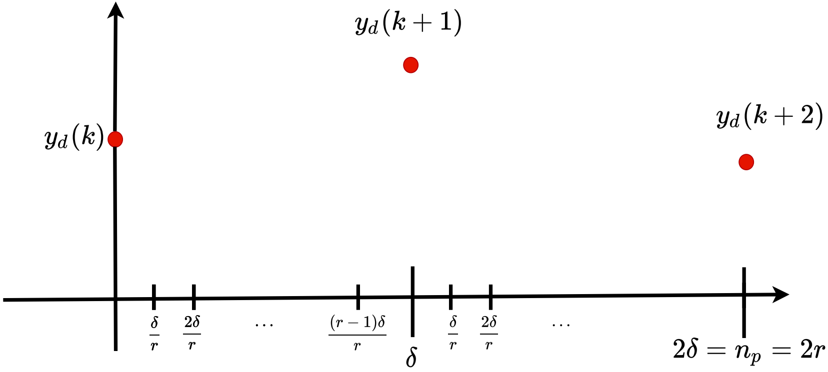
with $Q, R > 0$ being appropriate penalizing weights and $\epsilon \in \mathbb{R}$ small.

recall, at the limit of cheap control, optimal feedback solves

$$H(F_r^\delta(x(k), u^1(k), \ldots, u^r(k)) = \underline{y}_d(k+1)$$

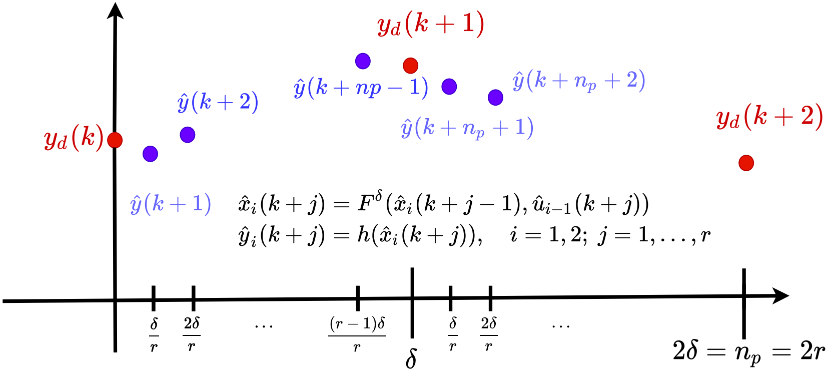
use this over two big intervals

Better solution: multi-rate for reference planning



Better solution: multi-rate for reference planning

Apriori admissible: for which exists a multirate control achieving objectives. Coincide with optimal solution at the cheap unconstrained limit!



Better solution: multi-rate for reference planning

Statement

Given a continuous time nonlinear input affine SISO system, and let $v(t)$ be a reference signal to be tracked at $t = k\delta$ for $k \to \infty$. Denote by $\{\nu_k = \nu(k\delta), k \geq 0\}$ the sequence of samples of the reference that is assumed to be multi-rate admissible for some multirate order r. Then, the unconstrained mpc problem

$$V^\star = \min \sum_{i=1}^{n_p} \left(\|y_d(k+i) - \hat{y}(k+i)\|_Q^2 + \epsilon \|u(k+i-1)\|_R^2 \right)$$

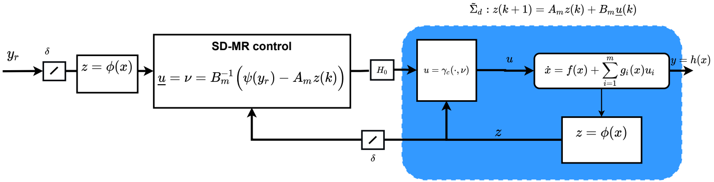
$$st. \quad x(k+1) = F^\delta(x(k), u(k))$$

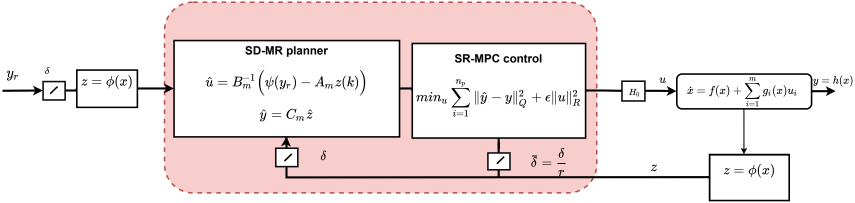
admits a solution which is bounded for $n_p \geq n_c \geq r$ and ϵ small enough [1]

[1] M. Elobaid, M. Mattioni, S. Monaco and D. Normand-Cyrot, "*Sampled-data tracking under model predictive control and multi-rate planning*", IFAC PapersOnline pp 3620-3625, 2020

Better solution: multi-rate for reference planning

Possible to use simplified models for the planner, e.g.

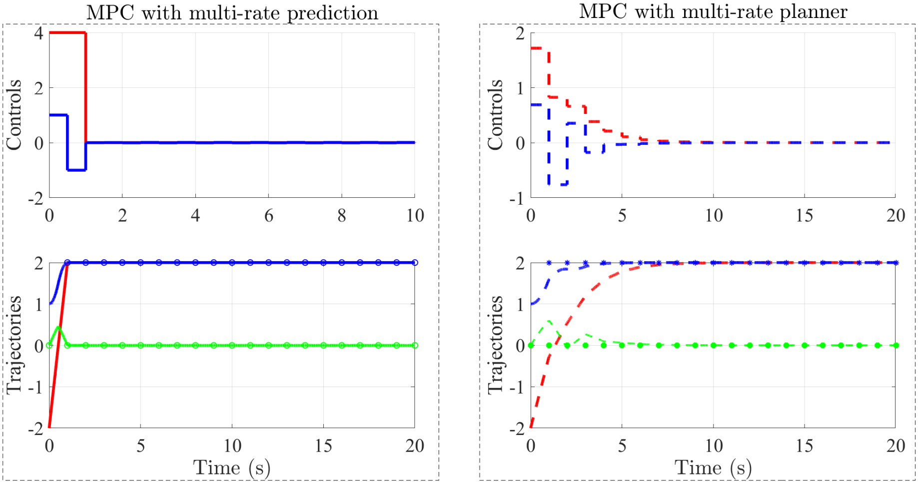




The unicycle revisited

Matlab/Simulink break

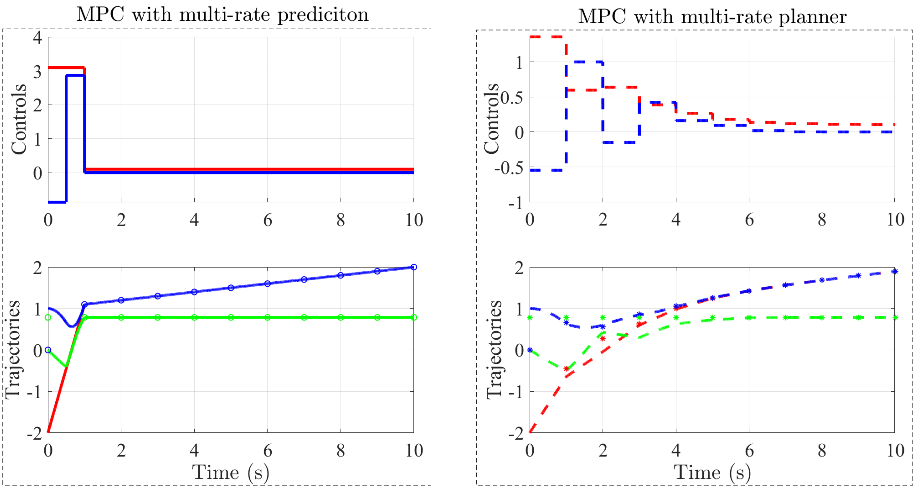
The unicycle revisited



The unicycle revisited

Matlab/Simulink break

The unicycle revisited



summary

- recalls on discrete-time representations of systems under sampling.
- we noticed that these models, used for control design, lose the original system's relative degree and its zero dynamics stability properties.
- non-minimum phase systems are harder to design tracking controllers for. Stability issues may arise.
- MPC, in the limiting case, can lead to zero dynamics cancellation resulting in poor tracking/instability
- multi-rate can be utilized to alleviate this issue either as a planner or as a control design model

Thanks