

Teória automatického riadenia III.

Cvičenie X, Parametrické programovanie, Explicitné MPC

G. Takács, G. Batista

Ústav automatizácie, merania a aplikovanej informatiky
Strojnícka fakulta, Slovenská technická univerzita

Na dnešnom cvičení

- Zostavenie MPC regulátora pomocou rozhrania LTI system
- Transformácia implicitného MPC problému do explicitného
- Porovnanie výkonu

Z prednášky

Riadiaci zákon MPC sa dá riešiť explicitne ako súbor po častiach afinných (lineárnych) riadiacich zákonov. Dá sa teda povedať že optimálny vstup systému je funkcia stavu systému $\mathbf{u} = \mathbf{u}(x)$ v príslušnom regióne \mathcal{R}_n

$$\mathbf{u}^* = \mathbf{f}(x) = \begin{cases} \mathbf{f}_1 x + \mathbf{g}_1 & \text{ak } x \in \mathcal{R}_1 \\ \mathbf{f}_2 x + \mathbf{g}_2 & \text{ak } x \in \mathcal{R}_2 \\ \vdots & \\ \mathbf{f}_{n_r} x + \mathbf{g}_{n_r} & \text{ak } x \in \mathcal{R}_{n_r} \end{cases}$$

Pre MATLAB sa dá použiť MPT toolbox (MPT3). Je dostupný na <http://people.ee.ethz.ch/~mpt/3/>

Inštalácia:

- skopírovať súbor "install_MPT3.m" zložky C:\MATLAB
- spustiť ho v MATLABe

Ako nato

Vytvoríte model lineárneho dynamického systému pomocou príkazu **"LTISystem"**

```
model = LTISystem('A', A, 'B', B, 'C', C, 'D', D);
```

Potom sa vytvorí MPC regulátor

```
ctrl = MPCController(model, np);
```

Zadajú sa váhové matice systému

```
ctrl.model.x.penalty = QuadFunction(Q);  
ctrl.model.u.penalty = QuadFunction(R);
```

Ohraničenia kvadratického problému sa zadajú príkazmi:

```
ctrl.model.u.min = Umix;  
ctrl.model.u.max = Umax;  
ctrl.model.x.min = Xmix;  
ctrl.model.x.max = Xmax;
```

Ako nato

Takto vytvorený MPC regulátor sa premení do explicitnej formy pomocou

```
expctrl = ctrl.toExplicit();
```

Keď máme vytvorený model, vypočítame akčný zásah na základe stavu systému:

```
U(i) = ctrl.evaluate(X(:,i));
```

a pre explicitné riešenie:

```
U(i) = expctrl.evaluate(X(:,i));
```

Ďalej je možné graficky znázorniť

- stavové partície:

```
expctrl.partition.plot();
```

- PWA riadiaci zákon

```
expctrl.feedback.fplot();
```

- PWQ účelvu funkciu

```
expctrl.cost.fplot();
```


Zadanie

- načítajte váš identifikovaný systém, a vytvorte MPC regulátor podľa doterajšieho postupu
- vytvorte implicitný a explicitný regulátor pomocou "**LTISystem**" a nalaďte ich tak aby podávali podobný výkon ako váš MPC regulátor
- urobte simuláciu regulačného pochodu pre všetky tri prípady, zmerajte čas výpočtu celého cyklu pomocou príkazu "**tic - toc**"
- graficky porovnajte priebehy všetkých troch simulácií
- graficky znázorníte stavové partície, PWA riadiaci zákon a PWQ účelovú funkciu explicitného regulátora
- otvorte šampanské a oslávte dokončenie posledného zadania TARIII