Laborator 2 (SPER)

Harta de viteză într-un mediu cu obstacole

21 februarie 2022

Cuprins

compilat la: 21/02/2022, 18:50

Scopul lucrării

Considerând un mediu cu obstacole și constrângeri de accelerație pentru un sistem dinamic liniar generăm harta sa de viteză ("velocity map") folosind mulțimi.

1 Noțiuni teoretice

Un sistem dinamic liniar este dat în forma de *reprezentare pe stare* de relațiile

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{1a}$$
$$y(t) = Cx(t) + Du(t). \tag{1b}$$

Vectorii $x \in \mathbb{R}^n, u \in \mathbb{R}^m, y \in \mathbb{R}^p$ denotă starea, intrarea (comanda) și respectiv ieșirea sistemului. Matricele A, B, C sunt alese corespunzător.

Un exemplu des utilizat în robotică este cel al "dublului integrator" unde starea este compusă din poziție și viteză, intrarea este accelerația iar ieșirea este poziția:

$$\dot{x}(t) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} u(t), \tag{2a}$$
$$y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} x(t). \tag{2b}$$

Discretizăm acest sistem folosind metoda Euler implicită ($\dot{x}(t) \approx (x_d[k+1] - x_d[k])/T$) unde $x_d[k] := x(kT)$ iar T este pasul de eșantionare:

$$x_d[k+1] = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x_d[k] + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ T & 0 \\ 0 & T \end{bmatrix} u_d[k], \tag{3a}$$
$$y_d[k] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} x_d[k]. \tag{3b}$$

Lucrăm sub ipoteza că intrarea rămâne constantă de-a lungul unui pas de eșantionare: $u(t) = u_d[k], \forall t \in [KT, (K+1)T)$.

Folosind notația compactă din (1) observăm că evoluția stării (traiectoria sistemului) este dată de:

$$x_d[k+1] = Ax_d[k] + Bu_d[k] = A\left(x_d[k-1] + Bu_d[k-1]\right) + Bu_d[k] = \dots \tag{4a}$$
$$= A^{k+1}x_d[0] + \sum_{i=0}^{k} A^i B u_d[k-i]. \tag{4b}$$

Cu alte cuvinte, traiectoria unui sistem depinde de starea sa inițială $x_d[0]$ (a cărei influență se atenuează dacă sistemul e stabil, $A^{k+1}x_d[0] \xrightarrow{k\to\infty} 0$) și de secvența de comenzi $u_d[0], \dots u_d[k]$ aplicată acestuia.

Orice mulțime convexă poate fi reprezentată ca un poliedru. Acesta are două formulări echivalente:

half-space regiunea din spațiu ce verifică o listă finită de inegalități[1]:

$$P = \{x \in \mathbb{R}^d : \; F_i x \leq \theta_i, \forall i = 1 : N_h\}; \tag{5}$$

vertex regiunea din spațiu definită ca suma convexă a unei liste finite de vârfuri[2]:

$$P = \{x \in \mathbb{R}^d : \; x = \sum_{j=1}^{N_v} \alpha_j v_j, \sum_{i=j}^{N_v} \alpha_j = 1, \alpha_j \geq 0, \forall j = 1 : N_v\}. \tag{6}$$

Pentru două mulțimi $P, Q \in \mathbb{R}^d$ definim operațiile:

sumă Minkowski generalizarea noțiunii de sumă pentru mulțimi:

$$P \oplus Q = \{p + q : \; \forall p \in P, \forall q \in Q\}; \tag{7}$$

diferență Pontryagin generalizarea noțiunii de diferență pentru mulțimi:

$$P \ominus Q = \{p \in P : \; p + q \in P, \forall q \in Q\}; \tag{8}$$

distanță Hausdorff generalizarea noțiunii de distanță pentru mulțimi:

$$d_H(P, Q) = \min \left\{ \max_{p \in P} \min_{q \in Q} d(p, q), \max_{q \in Q} \min_{p \in P} d(p, q) \right\}. \tag{9}$$

Considerând dinamica (1) și constrângeri pe comandă ($u \in U$) putem asocia unui obiect P noțiunea de *mulțime accesibilă ("forward reachable set")*:

$$\Omega_0 = P, \tag{10a}$$

$$\Omega_{k+1} = A\Omega_k \oplus BU. \tag{10b}$$

Mulțimea Ω_{k+1} mărginește valorile posibile ale stării după $k + 1$ iterații atunci când starea inițială pornește din P.

Noțiunea duală este cea de *mulțime accesibilă inversă ("backwards reachable set")*:

$$\Omega_0^- = P, \tag{11a}$$

$$\Omega_{k+1}^- = A^{-1}\Omega_k^- \oplus \{-A^{-1}BU\}. \tag{11b}$$

Mulțimea Ω_{k+1}^- mărginește valorile posibile din care, după $k+1$ iterații, traiectoriile ajung în P (se aplică aceeași relație ca în (10) dar pentru dinamica inversată în timp $x_d[k] = A^{-1}x_d[k+1] - A^{-1}Bu_d[k]$).

[1] Atenție, rezultatul poate să fie mulțimea vidă!

[2] Presupunem că obiectul ce îl modelăm este mărginit (nu este un con).

2 Calculul unei hărți de viteze admisibile

Considerăm un mediu cu obstacole poliedrale date în reprezentare "sumă convexă"[3] în subspațiul "poziție". Plotăm obstacolele folosind librăria MPT3:

```
plot(P,'Color','b','Alpha',.4)
```

Considerând constrângeri pe comandă ($U = \{|u| \leq \bar{u}\}$) devine evident că viteza maximă (în sensul că există o secvență de comenzi ce împiedică contactul) pe care putem să o avem în vecinătatea unui obstacol depinde de distanța până la acesta.

Considerăm sistemul dinamic de la (2) și, observând că $A^\ell = \begin{bmatrix} I & \ell T \cdot I \\ 0 & I \end{bmatrix}$, obținem traiectoria (4) descompusă în poziție (p) și viteză (v):

$$p_{k+1} = p_0 + (k+1)Tv_0 + T\sum_{\ell=0}^{k} \ell \cdot u_{k-\ell}, \tag{12a}$$

$$v_{k+1} = v_0 + T\sum_{\ell=0}^{k} u_{k-\ell}. \tag{12b}$$

Dacă dorim ca la pasul $k+1$ să fim în repaus, avem că $v_{k+1} = 0$ ceea ce, extrăgând v_0 din (12b) și introducând în (12a), conduce la următoarea formă a lui p_{k+1}:

$$p_{k+1} = p_0 - T\sum_{\ell=0}^{k} \left[\ell - (k+1)T\right] u_{k-\ell}. \tag{13}$$

Putem prin urmare să construim un poliedru în variabila $\begin{bmatrix} p_0^\top & u_0^\top & \dots & u_k^\top \end{bmatrix}^\top$ pornind de la incluziunile $p_{k+1} \in P$ și $u_{k-\ell} \in U, \forall \ell = 0 : k$:

```
Polyhedron([P(i).A*[eye(2) -T*kron([(0:r)-(r+1)*T],eye(2))];...
           [zeros(2*(r+1),2) eye(2*(r+1))];...
           [zeros(2*(r+1),2) -eye(2*(r+1))]],...
           [P(i).b; ones(2*(r+1),1)*ubar; ones(2*(r+1),1)*ubar]);
```

Combinațiile de poziție inițială și secvență de comenzi ce sunt în interiorul acestei regiuni se interpretează astfel: pornind din p_0 există o secvență de comenzi $u_0, \dots, u_k$ astfel încât, în interiorul obstacolului viteza finală să fie nulă (echivalent spus, putem evita coliziunea).

Pentru ilustrații, ne interesează atât regiunile din care garantăm că în '$k+1$' pași ne oprim cât și vitezele admisibile asociate. Ambele regiuni se obțin folosind metoda **projection** a clasei Polyhedron, întâi pentru poziție:

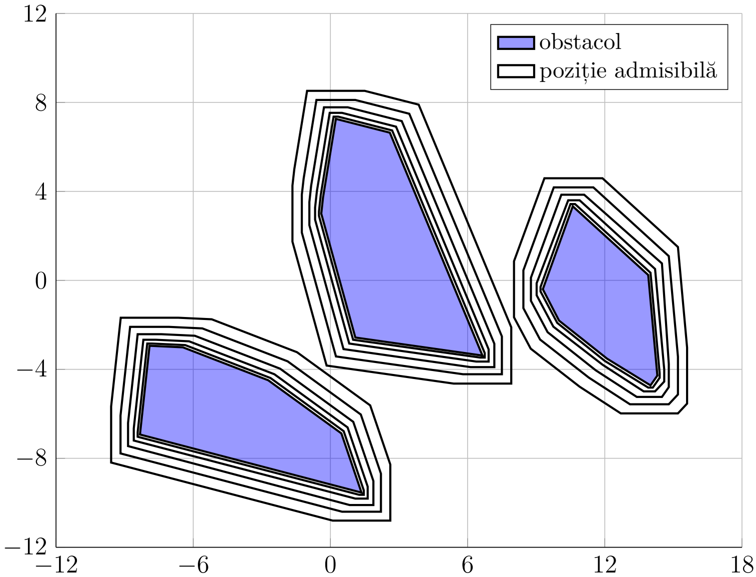
```
plot(VM{i}(r).projection([1:2]),'Wire',1,'WireColor','r','Alpha',.4)
```

iar apoi, combinând și cu relația din (12b), pentru viteză:

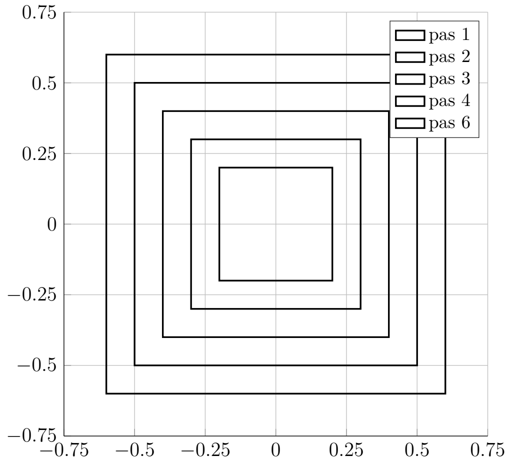
[3] Codul complet se regăsește în fișierul atașat laboratorului, în continuare ilustrăm fragmente relevante.

```
plot(-T*repmat(eye(2),1,r+1)*VM{i}(r).projection(3:(2+(r+1)*2)),'Alpha'
    ,.4)
```

Pentru claritate ilustrăm și figurile obținute în urma rulării codului anterior.



(a) poziții admisibile



(b) viteze admisibile

Figura 1: Hartă de viteze admisibile pentru un mediu cu obstacole.

3 Exerciții propuse

Exercițiul 1. Discretizarea unui sistem implică întotdeauna o aproximare. Prin metode de integrare numerică este posibil să ilustrăm traiectoria "adevărată" a sistemului (tot o aproximare dar mai apropiată de realitate).

Se poate folosi **ode45** pentru a integra numeric un semnal de forma $\dot{y} = F(t, y)$ pornind de la o stare inițială y_0 pe un orizont de timp de lungime T:

```
[t,y]=ode45(F,[0 T],y0)
```

În general, un sistem dinamic este definit de relația $\dot{x} = f(x, u)$. Presupunem (rezonabil) că $u(t) = u_k$ de-a lungul intervalului de timp $[KT, (k+1)T)$. Atunci putem rescrie:

$$\begin{cases} \dot{x}(t) & = f(x, u), \\ \dot{u}(t) & = 0, \end{cases}$$

și alegând $y_0 = \begin{bmatrix} x_k^\top & u_k^\top \end{bmatrix}^\top$, $y(t) = \begin{bmatrix} x(t)^\top & u(t)^\top \end{bmatrix}^\top$, construind $F(\cdot)$ pe baza ecuației de mai sus, apelăm **ode45** pentru a integra numeric și obține evoluția sistemului $x_k \to x_{k+1} = x((k+1)T)$.

Exercițiul 2. Metodele de discretizare propun întotdeauna un compromis între complexitatea reprezentării și precizia aproximării. Pe lângă metode uzuale pentru Euler explicit/implicit sau Tustin, există și clasa metodelor Runge-Kutta.

Implementați următoarea aproximare RK4 pentru a discretiza sistemul dublu integrator folosind următoarele relații:

$$\begin{aligned} x_{k+1} &= x_k + \frac{T}{6}\left(w_1 + 2w_2 + 2w_3 + w_4\right) \\ w_1 &= f(x_k, u(kT)), \\ w_2 &= f\left(x_k + \frac{T}{2}w_1, u(kT + T/2)\right), \\ w_3 &= f\left(x_k + \frac{T}{2}w_2, u(kT + T/2)\right), \\ w_4 &= f\left(x_k + \frac{T}{2}w_3, u((k+1)T)\right). \end{aligned}$$

Exercițiul 3. Pentru a calcul harta de viteze admisibile am exploatat structura particulară a sistemului. Pentru cazul general, vor trebui folosite construcțiile de mulțime direct/invers admisibilă. Care este procedura?