Laboratory 7 (SPER)

Planificarea mișcării cu constrângeri de comunicație

29 martie 2022

Cuprins

compilat la: 29/03/2022, 01:15

Scopul lucrării

Problema comunicației într-un mediu cu obstacole este dificil de rezolvat. O abordare des utilizată este forțarea unei constrângeri de conectivitate în graful asociat (nodurile sunt agenții iar muchiile există dacă agenții sunt vizibili unul față de celălalt).

1 Noțiuni teoretice

Noțiunile de vizibilitate și de asigurare a comunicației într-un mediu cu obstacole sunt strâns legate și definesc probleme încă deschise. Sunt posibile mai multe variații (pe care nu le detaliem):

- graf de comunicație staționar / dinamic (al doilea caz apare în situațiile în care agenții/obstacolele sunt mobile);
- conectivitate robustă (în sensul de redundanță, un agent este conectat cu mai mulți agenți);
- conectivitate rezilientă (în sensul că există capacitatea în sistem de a recupera conectivitatea).

Există multiple modalități (sub-optimale, heuristice) pentru a garanta comunicația într-o echipă de agenți ce se deplasează într-un mediu cu obstacole:

- penalizarea distanței dintre 2 agenți dacă aceasta a crescut prea mult;
- menținerea unei linii de comunicație (LOS – line of sight).

Pentru metodele LOS, implementarea se face prin *programare cu variabile mixte* ceea ce echivalent cu a impune condiții de conectivitate asupra unui graf. **Ideea**:

- considerăm o echipă de agenți $\mathcal{V} = \{v_1, \dots v_N\}$;
- enumerăm o colecție de legături, variabilă în timp, între perechi de agenți $\epsilon(k) = \{(v_i, v_j) : \ v_i, v_j \in \mathcal{V}\}$;
- construim graful variabil în timp $\mathcal{G}(k) = (\mathcal{V}, \epsilon(k))$;
- echipa este conectată la pasul $k \Leftrightarrow$ graful $\mathcal{G}(k)$ este conectat.

Doi agenți, cu pozițiile p_i, p_j, vor fi conectați (se află în LOS) dpdv al unui obstacol $\mathcal{O}$ dacă relația

$$\{p_i(k) \cdot \alpha + p_j(k) \cdot (1 - \alpha), \ \forall \alpha \in [0, 1]\} \cap \mathcal{O} = \emptyset, \tag{1}$$

este respectată. Echivalent spus, dacă segmentul definit de pozițiile celor doi agenți intersectează obstacolul $\mathcal{O}$ nu există o linie de comunicație între cei doi. O astfel de relație este greu de integrat în formulările standard de planificare a mișcării (mai ales în cele care necesită rezolvarea unei probleme de optimizare la fiecare pas).

Soluția propusă este următoarea (într-o formă mai complexă, apare în [1]):

$$F_\ell^\top p_i(k) \leq \theta_\ell + M(1 - z_{i\ell}(k)), \forall i, \ell \tag{2a}$$
$$z_{i\ell}(k) \geq \bar{z}_{ij}(k) \geq z_{i\ell}(k) + z_{j\ell}(k) - 1, \forall i, j, \ell \tag{2b}$$
$$z_{j\ell}(k) \geq \bar{z}_{ij}(k) \geq z_{i\ell}(k) + z_{j\ell}(k) - 1, \forall i, j, \ell \tag{2c}$$
$$d_i(k) = \sum_{j \neq i} \bar{z}_{ij}(k), \forall i \tag{2d}$$
$$d_i(k) + d_j(k) \geq (N-1) - N\bar{z}_{ij}(k), \forall 1 \leq j < i \tag{2e}$$

Forțând doi agenți să stea în aceeași regiune convexă garantăm că există o linie de comunicație între ei. Mai precis:

- relația (2a) controlează prin intermediul variabilei binare $z_{i\ell}(k)$ dacă agentul i se află în interiorul regiunii $A_\ell = \{x : F_\ell^\top x \leq \theta_\ell\}$:

$$p_i(k) \in \mathcal{A}_\ell \quad \Leftrightarrow \quad z_{i\ell}(k) = 1;$$

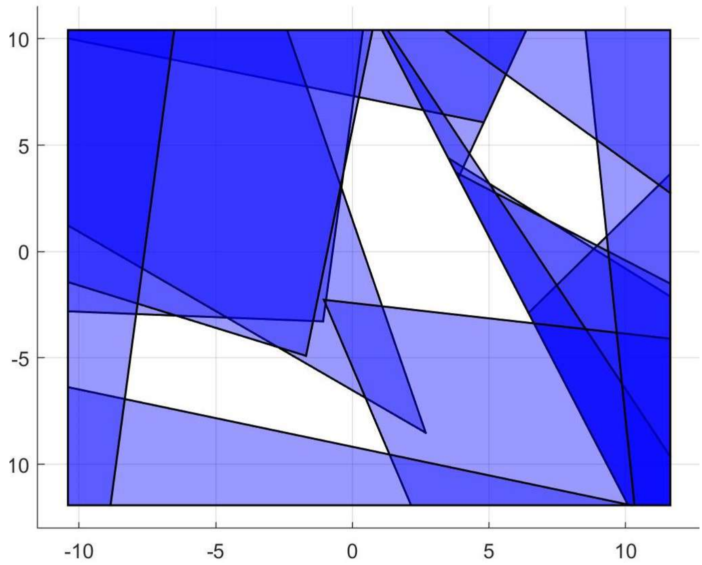
- relațiile (2b)–(2c) leagă variabilele binare $z_{i\ell}, z_{j\ell}$ de variabila binară $\bar{z}_{ij}$ ce arată că 2 agenți sunt sau nu adiacenți (se află în aceeași regiune); echivalent, ecuațiile se pot înlocui cu $\bar{z}_{ij} = (z_{i\ell} == z_{j\ell} == 1)$;
- relația (2d) stochează în variabila $d_i(k)$ gradul corespunzător agentului i (câte muchii din graf se termină/încep în nodul i);
- relația (2e) impune o condiție pe gradele fiecărei perechi de noduri distincte din graf, în așa fel încât să fie garantată conectivitatea echipei ("1-connectivity").

  Mai precis: dacă suma gradelor pentru oricare două noduri ce nu sunt adiacente este $\geq N - 1$, atunci graful este conectat.

O astfel de abordare este sub-optimală deoarece se fac presupuneri restrictive: se pre-calculează regiunile $\mathcal{A}_\ell$ ce descriu spațiul fezabil iar condiția de conectivitate în graf este una suficientă (nu necesară și suficientă). Totuși, mecanismul din (2) oferă o modalitate de a garanta conectivitatea unei echipe de agenți. Probleme vor apărea la dimensiuni mari ale echipei datorită conservatismului condiției de conectivitate și datorită numărului ridicat de variabile binare.

2 Implementare a unui algoritm de conectivitate

Considerăm o echipă de agenți și o colecție de obstacole, ilustrate în figură.



În plus față de (2), considerăm și un cost ce implică pozițiile agenților astfel încât problema de optimizare să fie corect definită:

$$\min_{d,p,z} \sum_{i=0}^{N} \|p_i\|^2 \tag{3a}$$

$$\text{s.t. } F_\ell^\top p_i \leq \theta_\ell + M(1 - z_{i\ell}) \tag{3b}$$

$$z_{i\ell} \geq \bar{z}_{ij} \geq z_{i\ell} + z_{j\ell} - 1, \forall i, j, \ell \tag{3c}$$

$$z_{j\ell} \geq \bar{z}_{ij} \geq z_{i\ell} + z_{j\ell} - 1, \forall i, j, \ell \tag{3d}$$

$$d_i = \sum_{j \neq i} \bar{z}_{ij}, \forall i \tag{3e}$$

$$d_i + d_j \geq (N - 1) - N\bar{z}_{ij}, \forall 1 \leq j < i. \tag{3f}$$

Pentru simplitate am considerat problema ca fiind una statică (nu ne mai interesează pasul curent k și nici dinamica internă a agenților).

```
for i in range(N):
    for l in range(M):
        solver.subject_to(mat['F'][0][l] @ P[:,i]<=mat['theta'][0][l]+
     Mbig*Z[i,l]*np.ones((len(mat['F'][0][l]),1)))
    solver.subject_to(np.ones((1,M)) * Z[i,:]>=1)

# conditions (3c) and (3d)
for i in range(N):
    for j in range(N):
        for l in range(M):
```

```
            solver.subject_to(Zbar[i,j]>=Z[i,l]+Z[j,l]-1)
            solver.subject_to(Z[i,l]>=Zbar[i,j])
            solver.subject_to(Z[j,l]>=Zbar[i,j])

# condition (3e)
for i in range(N):
    tmp=0
    for j in range(N):
        if j is not i:
            tmp=tmp+Z[i,j]
        solver.subject_to(D[i]==tmp)

# condition (3f)
for i in range(N):
    for j in range(i-1):
        solver.subject_to(D[i]+D[j]>=(N-1)-N*Zbar[i,j])

# cost (3a)
for i in range(N):
    objective = objective + \
                casadi.transpose(P[:, i]) @ P[:, i]
```

3 Exerciții propuse

Exercițiul 1. Codul asociat acestui laborator implementează doar problema statică unde nu presupunem o dinamică internă pentru aagenți. Implementați o problemă de tip MPC (cu orizont de predicție dat) pentru dinamică de tip "dublu-integrator".

Exercițiul 2. Modificați codul (inclusiv varianta obținută în exercițiul anterior) pentru cazul mașinii Dubins a cărei dinamică este dată de ecuațiile:

$$\begin{cases} \dot{x} &= u_V \cos\phi, \\ \dot{y} &= u_V \sin\phi, \\ \dot{\phi} &= \frac{u_V}{L}\tan u_\phi. \end{cases}$$

Bibliografie

[1] Rubens JM Afonso, Marcos ROA Maximo **and** Roberto KH Galvão. "Task allocation and trajectory planning for multiple agents in the presence of obstacle and connectivity constraints with mixed-integer linear programming". **in** *International Journal of Robust and Nonlinear Control*: 30.14 (2020), **pages** 5464–5491.