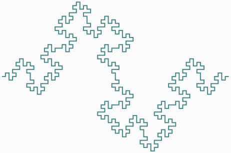
CoCoMA – Mini-projet

N. Maudet

2020–2021



CoCoMA — Master ANDROIDE

Au choix: Sujet #1

On considère (pour la visualisation, et pour donner un sens concret au coût de visite des différents sites) un monde de cases, avec:

- un ensemble de robots
- un ensemble de tâches
- les robots disposent d'une visibilité identique d
- les cases à l'intersection entre i et j constituent la zone de négociation $Z(i, j)$
- l'ensemble constitue un graphe non-orienté que l'on suppose connexe
- on va se limiter à une topologie simple: 3 agents, pas de site à l'intersection de plus de 2 agents. (Donc ligne de 3 agents).
- Optionnel: plus de 3 agents (pose pas mal de problèmes...)

Au choix: Sujet #1

On suppose que chaque agent ne peut trouver un accord qu'avec un seul autre agent (cf. Network Exchange Theory, cours 2).

Si un agent i ne trouve pas d'accord avec j il devra ramasser tous les objets de la zone $Z(i, j)$.

L'objectif est d'implémenter le processus de négociation en supposant:

- que chaque négociation est menée selon un Monotonic Concession Protocol (et n'est pas interrompue) bilatéral (pas plus de deux agents à la fois dans la négociation)
- mais en supposant comme dans Network Exchange Theory que les opportunités de recours constituent le point de conflit

		r_1			o_2				
		o_1		o_3					
					r_2			o_6	
					o_4				
							o_5	r_3	

perception distance $d = 4$ (le robot perçoit 4 cases autour)

$Z(r_1, r_2) = \{o_1, o_2, o_3\}$, $Z(r_2, r_3) = \{o_4, o_5, o_6\}$

Pas besoin de revenir à la position initiale.
Coût d'un ensemble de sites = coût de la tournée la plus courte.
Utilités des agents: $u_i(tour) = 10 - cost(tour)$
Négo entre r_1 et r_2 (arbitrairement, autre ordre possible)
Par ex. pour la tournée entière sur o_1, o_2, o_3, on a $u_1(o_1, o_2, o_3) = 10 - 7 = 3$, et $u_2(o, o_2, o_3) = 10 - 8 = 2$. Donc conflict point entre r_1 et r_2: $o_\downarrow : (3, 2)$
(negociation par MCP...)
Agreement: $o* = \langle(o_2), ((o_1, o_3)\rangle$, $u_1(o*) = 10 - 3 = 7$, $u_2(o*) = 10 - 4 = 6$.
Note: Nash prod: $(u_1(o*) - u_1(o_\downarrow) * (u_2(o*) - u_2(o_\downarrow)) = (7 - 3) * (6 - 2) = 16$
Négo entre r_2 et r_3: cette fois r_2 annonce une utilité en cas de conflit de 6 (résultat négo précédente), donc conflict pt entre r_2, r_3 pour o_4, o_5, o_6: $(6, 2)$;
(negociation par MCP...)
Agreement: $o* = \langle(o_4), ((o_5, o_6)\rangle$, $u_2(o*) = 10 - 1 = 9$, $u_3(o*) = 10 - 4 = 6$.
Note: Nash prod: $(u_2(o*) - u_2(o_\downarrow) * (u_3(o*) - u_3(o_\downarrow)) = (9 - 6) * (6 - 2) = 12$
Balanced outcome. End.

Au choix: Sujet #2

Reproduire (partiellement) l'article "Autonomous Vehicle-Target Assignment: A Game-Theoretical Formulation"

Il conviendra d'implémenter, la coordination avec négociation:

- par dynamique de meilleure réponse
- par fictitious play
- par regret matching
- par spatial adaptive play (version de base)

Voir: `http://www2.hawaii.edu/~gurdal/JDSMC07.pdf`

Rendu

Dans tous les cas le rendu consistera en un notebook:

- présentant le problème et les approches tudiées
- permettant de tester simplement des instances (dans le notebook)
- dont la vocation est illustrative: on ne demande pas d'analyse expérimentale sur un nombre significatif d'instances