

Multirobotické systémy

Miroslav Kulich

Intelligent and Mobile Robotics Group
Gerstner Laboratory for Intelligent Decision Making and Control
Czech Technical University in Prague

27/02/2014

Multirobotické systémy

Systémy kooperujících robotů

- Znalost o existenci ostatních robotů
- Plnění společných úkolů na základě stavu, akcí a možností ostatních robotů
- Silně kooperující - komunikace, synchronizace
- Slabě kooperující - období, kdy roboty operují nezávisle

Robotická hejna

- Velké množství typicky homogenních robotů
- Lokální řízení
- Malá/žádná explicitní komunikace

Je N robotů lepších než jeden?

Pozitiva

- Výkonnější, rychlejší
- Lze řešit složitější úlohy
- Menší náchylnost k chybám
- Distribuované snímání
- Akce mohou být prováděny na vzdálených místech současně
- Vytvořit tým jednoduchých robotů může být levnější než jeden složitý

Zápory

- Roboty se mohou navzájem rušit a plést
- Cena komunikace
- Komplexita/údržba
- Cena
- Nejistota znalosti o záměrech ostatních robotů

Studované problémy multirobotických týmů

- **Explorace a pokrývání:** sbírání náhodně rozmístěných objektů
- **Utváření formací a shlukování:** tým udržuje zadaný geometrický tvar během svého pohybu
- **Kooperativní manipulace/tlačení krabice:** tým kolektivně manipuluje s objekty
- **Sledování více objektů:** tým se pohybuje tak, aby udržel objekty ve svém zorném poli
- **Řízení dopravy/multirobotické plánování cesty:** koordinace akcí ve společném prostoru
- **Multi-robotický fotbal:** hra/soutěž, kde jsou řešeny klíčové aspekty kooperace

Architektury multirobotických systémů

Centralizovaná

- Koordinace týmu z jednoho centra
- Zranitelnost centrálního prvku
- Centrální prvek musí být informován o aktuálním stavu (nutnost komunikace)
- Vhodné, pokud lze centrální prvek vhodně umístit

Hierarchická

- Inspirováno uspořádáním vojenských jednotek
- Špatné zotavení při chybě/výpadku robotu, který je v hierarchii vysoko

Architektury multirobotických systémů

Decentralizovaná

- Robot typicky rozhoduje pouze na základě lokální znalosti
- Nikdo nezodpovídá za řízení jiného robotu
- Cíle musí být zahrnuty v lokálním řízení každého robotu
- Nejrozšířenější architektura

Hybridní

- Kombinace lokálního řízení s řízením vyšší úrovně
- Robustnost
- Možnost ovlivňovat chování celého týmu globálně
- Velmi rozšířená

Komunikace

- **Cíl:** umožnit robotům vyměňovat informace o vnitřním stavu robotů a o prostředí s minimálními nároky na přenosové médium
- **Implicitní komunikace** - komunikace prostřednictvím prostředí
 - Sledování změn prostředí, které jsou důsledkem činnosti jiných robotů
 - Sledování přímo akcí jiných robotů
- **Explicitní komunikace** - přenos informace přímo mezi roboty
 - Zahrnuje: občasné požadavky, informace o vnitřním stavu, senzorická měření
 - Nutno specifikovat: CO, KDY, JAK, S KÝM komunikovat
 - Komunikační médium: dosah, šířka pásma, spolehlivost
- **Příklady:**
 - Feromony u mravenců
 - „Pózování“ zvířat během páření, před bojem
 - Psaní, kreslení

Je komunikace potřeba?

- **Dobré si uvědomit:**
 - Komunikace není zadarmo a může být nespolehlivá
 - V nepřátelském prostředí může být rušená
- **Role komunikace:**
 - Synchronizace akcí - koordinace v uspořádání akcí
 - Výměna informací - sdílení informace získané z různých zdrojů a perspektiv
 - Dohadování - kdo co bude dělat
- **Studie ukazují:**
 - Explicitní komunikace zlepšuje výkonnost týmu, pokud je malá implicitní komunikace
 - Explicitní komunikace není třeba, pokud je k dispozici implicitní komunikace
 - Komplexní komunikační strategie přinášejí malý benefit oproti základní komunikaci

Komunikační dosah

Kategorie

- Žádná komunikace
- Omezený dosah (lokální komunikace)
- Neomezený dosah

Vlastnosti

- Všeobecná představa: větší komunikační dosah je lepší.
- Platí pro malé týmy, ale pro velké to již pravda být nemusí:
 - Efektivita přenosu informace je malá, pokud je jedno komunikační medium použito mnoha roboty (rádio).
 - Pokud zajišťuje komunikaci centrální prvek, stává se úzkým hrdlem a zvyšuje se chybovost přenosu.
 - Ve většině případů stačí robotu k naplánování další akce pouze lokální komunikace.

Komunikační topologie

Kategorie

- Broadcast - co pošle jeden, slyší všichni
- Adresovaná (peer-to-peer)
- Strom (supervize)
- Graf - obecnější případ, robustnější než strom

Vlastnosti

- Strom a adresovaná topologie jsou náchylné na chybu jednotlivých robotů.
- Adresování vede k diferenciaci (a tedy k horší zaměnitelnosti) robotů \rightsquigarrow dynamická změna role (aukce).
- Množina robotů, která může spolu komunikovat je dána:
 - komunikační topologií,
 - komunikační dosahem a
 - rozmístěním robotů v prostředí.

Komunikace - šířka pásma

Kategorie

- Vysoká (neomezená) - předpoklad v teoretických úvahách
- Jako pohyb - cena komunikace a pohybu je srovnatelná (včelí tanečky)
- Nízká - cena za komunikaci je vysoká (velmi nezávislé roboty)
- Žádná

Vlastnosti

- V některých případech může robot buď komunikovat, nebo plnit úlohu
- Nízká šířka pásma je akceptovatelná, pokud cílem použití více robotů je spíše redundance než efektivita.

Složení týmu

Identické

- Stejný hardware i software
- Roboty mohou mít různé role (v závislosti na prostředí a náhodě)
- Roboty mohou mít unikátní identifikaci (přidělenou deterministickým procesem)

Homogenní

- Stejný hardware, jiný software

Heterogenní

- Fyzicky různé roboty

Alokace úloh

- **Alokace úloh** je problém určení, který robot bude provádět kterou úlohu
- **Mějme** n robotů $\{r_1, r_2, \dots, r_n\}$
 m úloh $\{t_1, t_2, \dots, t_m\}$
- **Cíl:** nalézt mapování úloh k robotům tak, že každá úloha je provedena nejlepším možným způsobem
- Většina konkrétních specifikací problému je NP-úplná
- **Taxonomie:** (Gerkey)
 - Úloha: single-robot (SR) \times multi-robot (MR)
 - Roboty: single-task (ST) \times multi-task (MT)
 - Přiřazení: okamžité (IA) \times v čase (TA)
- Kombinace předchozích kritérií do jednotného popisu:
 - SR-ST-TA: single-robot, single-task, přiřazení v čase
 - MR-ST-IA: multi-robot, single-task, okamžité přiřazení

ST-SR-IA

- Instance Optimal Assignment Problem (OAP)
- Nejjednodušší problém, studován v operačním výzkumu (Gale 1960)
- Formulace: Mějme m pracovníků, každý schopný vykonávat jednu úlohu a n úloh s prioritou w_j vyžadujících jednoho pracovníka. Pro každého pracovníka je definován užitek (utility) U_{ij} , s kterým vykoná danou úlohu. Cílem je každému pracovníkovi přiřadit jednu úlohu tak aby se maximalizoval celkový užitek a zohlednila se prioritita úloh:

$$U = \sum_{i=1}^m \sum_{j=1}^n \alpha_{ij} U_{ij} w_j \quad \text{tak, aby}$$

$$\sum_{i=1}^m \alpha_{ij} = 1, 1 \leq j \leq n \quad \text{a} \quad \sum_{j=1}^n \alpha_{ij} = 1, 1 \leq i \leq m$$

ST-SR-IA - přístupy k řešení

- Lineární programování - maďarská metoda, Kuhn(1955), složitost $O(mn^2)$
- Řešení duální úlohy, Gale (1960):

$$P = \sum_{i=1}^m u_i + \sum_{j=1}^n v_j, \text{ kde}$$

$$u_i + v_j \geq U_{ij}, \forall i, j$$

- Aukce: makléř nabízí úlohy, roboty vybírají
 $t_i = \arg \max_j \{h_{ij} - p_j\}$
- Iterované přiřazování (BLE - Broadcast of Local Eligibility):
 1. Pokud existuje robot bez úlohy, najdi dvojici robot-úloha (i, j) s nejvyšším užitekem. V opačném případě skonči.
 2. Přiřaď úlohu j robotu i a oba vyřaď z dalšího uvažování.
 3. Jdi na krok 1.

ST-SR-IA - přístupy k řešení

- On-line přiřazení (MURDOCH) - úlohy jsou předkládány postupně:
 1. Pokud se objeví nová úloha, přiřad' ji k robotu tak, aby užitek byl co největší.
- Bez znalosti o úlohách, které budou prezentovány, je MURDOCH nejlepší možný.

ST-SR-TA

- V systému je více úloh než robotů a chceme je vyřešit všechny.
- Problém (rozvrhování) je NP-těžký \Rightarrow aproximační algoritmus:
 1. Optimálně řeš počáteční $m \times n$ OAP problém
 2. Použij hladový algoritmus k postupnému přiřazování úloh robotům tak, jak roboty dokončují předchozí úlohy.
- Čím více úloh je přiřazeno v prvním kroku, tím lepší je celkový výsledek.
- Jiný přístup Iterativní alokace úloh: price-based market Dias, Stentz (2001), aplikace pro exploraci Zlot et.al (2002)
- Bez znalosti o konkrétním způsobu „obchodování“ nelze o chování algoritmu nic říct.

ST-MR-IA

- Více robotů pro jednu úlohu \Rightarrow *kombinovaná* užitečnost skupiny robotů.
- Multi-agentní přístupy - formování koalic.
- Teorie množin - Set Partitioning Problem (SPP): Mějme konečnou množinu E , množinu akceptovatelných podmnožin F a funkci užitečnosti $u : F \rightarrow \mathbb{R}_+$, najdi podmnožinu F s maximálním užitekem, která je rozdělením E .
- NP-těžké, ale existují kvalitní heuristiky (problémy řádu stovek úloh lze řešit za jednotky sekund).
- Je nutno vyčíslit velké množství různých „koalic“ \Rightarrow je vhodné ořezávat (vzdálenost robotů).

ST-MR-TA

- Formování koalic + rozvrhování
- NP-těžký
- Příklad: doručování balíků různých velikostí z jednoho skladu do různých cílů.
- Řešení: ST-MR-IA + hladový algoritmus pro zbytek.
- Kvalita výsledku je nepredikovatelná.
- Jiný přístup: Několik lídrů, které si formují koalice a pro tyto koalice se pak vytváří rozvrh.

MT-SR-IA a MT-SR-TA

- Tyto problémy nemají v robotice moc velký význam.
- Lze řešit převodem na úlohy ST-MR-IA a ST-MR-TA (záměnou robotů a úloh).

MT-MR-IA

- Příklad (hlídání budovy): každý robot hlídá určitou část budovy, ale je schopen detekovat pouze některé události (oheň, otevřené dveře, pohybující se osoba).
- Set Covering Problem z teorie množin: Mějme konečnou množinu E , F množinu podmnožin E (ne nutně všech) a funkci užitečnosti $u : F \rightarrow \mathbb{R}_+$, najdi podmnožinu F s maximálním užitekem, která je pokrytím E .
- U pokrytí (na rozdíl od rozdělení) nemusí být množiny disjunktní.
- Byly vyvinuty heuristiky, které fungují dobře pro omezené množiny F (množství možných koalic je o hodně menší než množství všech koalic).
- MT-MR-TA - nikdo neřeší.

Koordinace více robotů

- Předpokládáme p robotů, každý s n stupni volnosti.
- Plánovací metody jsou pro mobilní roboty i manipulátory stejné.

Centralizované plánování

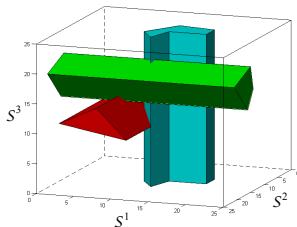
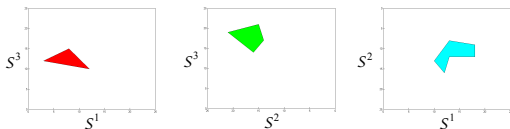
- Více robotů uvažovat jako jeden robot s více DOF.
- Plánování probíhá v kompozitním C-space.
- Algoritmus je úplný, tj. vždy najde řešení, pokud existuje
- Složitost $\approx e^{pn}$.
- Nelze aplikovat tam, kde není centrální znalost o všech robotech.

Oddělené (decoupled) plánování

- Plánování pro každý robot zvlášť.
- Korekce jednotlivých plánů (koordinace) později.
- Složitost $\approx pe^n$.
- Není úplný, není optimální.

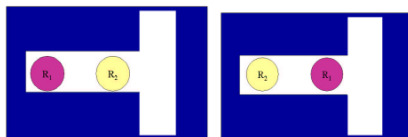
Centralizované plánování

- Plánování bezkolizní cesty τ v $C_1 \times C_2 \times \dots \times C_p$.
- Překážky v kompozitním konfiguračním prostoru jsou všechny konfigurace, kde robot koliduje s překážkou nebo s jiným robotem.
- Projekce τ do C_i je cesta i -tého robotu.



Oddělené plánování - druhá fáze

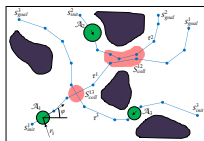
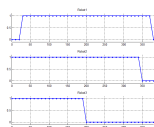
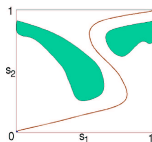
- Ladění rychlostí.
 - Každý robot se pohybuje po trajektorii generované v prvním kroku.
 - V rámci této trajektorie může robot, zastavit, couvat, nebo měnit rychlost.
 - Koordinace párů (pairwise planning)
 - Globální koordinace
- Prioritní plánování



zdroj: BP V. Krtičky

Oddělené plánování - Koordinace párů

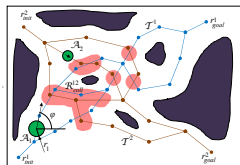
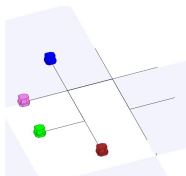
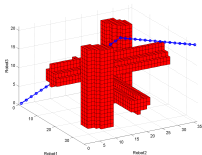
- Cesty τ_1 a τ_2 prvních dvou robotů jsou zkoordinovány v 2-dimenzionálním koordinačním prostoru.
- Konfigurace robotu je jedno-dimenzionální (pozice na trajektorii v čase).
- Plánování cesty mezi $(0, 0)$ a $(1, 1)$.
- Výsledkem je bezkolizní koordinovaná cesta $\tau_{1,2}$.
- V dalším kroku se provádí koordinace cesty $\tau_{1,2}$ a τ_3 .
- V procesu se pokračuje, dokud se nezíská $\tau_{1,2,\dots,p}$



zdroj: BP V. Krtičky

Oddělené plánování - Globální koordinace

- Vytvoří se p -dimenzionální koordinační prostor.
- Plánování cesty mezi $(0, 0, 0, \dots)$ a $(1, 1, 1, \dots)$.
- Zjednodušení: diskretní koordinační prostor, diskretní konfigurační trajektorie.
- Lze rozšířit pro případ, kde se roboty pohybují po grafu.



zdroj: BP V. Krtičky

Koordinace pohybu

- **Vzhledem k ostatním robotům:** formace, vločkování (flocking), shlukování, rozptýlení
- **Vzhledem k prostředí:** prohledávání, hledání, spásání (foraging), pokrývání
- **Vzhledem k externím faktorům:** pronásledování, sledování cíle
- **Vzhledem k ostatním robotům a prostředí:** patrolování
- **Vzhledem k ostatním robotům, externím faktorům a prostředí:** robotický fotbal

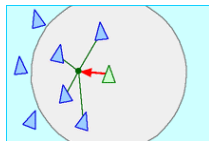
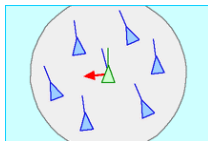
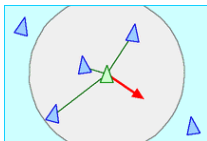
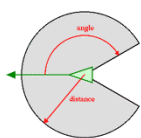


zdroj: <http://en.wikipedia.org>

Boids (Craig Reynolds)

Základní typy chování (steering behaviours) na základě lokální informace:

- **Separace** - řízení za účelem vyhnutí se shlukům
- **Zarovnání** - nastavení orientace dle lokálního průměru
- **Soudržnost** - řízení do lokálního průměru



zdroj: <http://www.red3d.com/cwr/boids>

The Nerd Herd

- Princip: základní typy chování jako základní bloky pro stavbu složitějších strategií
- Základní chování:
 - Avoidance
 - Save-wandering
 - Following
 - Aggregation
 - Dispersion
 - Homing
- Složitější chování:
 - Flocking
 - Foraging

Algoritmus Safe-Wandering

Avoid-Kin

- Whenever an agent is within d_{avoid}
 - If the nearest agent is on the left
 - Turn right
 - Otherwise, turn left

Avoid-Everything-Else

- Whenever an obstacle is within d_{avoid}
 - If obstacle is on right only, turn left
 - If obstacle is on left only, turn right
 - After 3 consecutive identical turns, backup and turn
 - If an obstacle is on both sides, stop and wait.
 - If an obstacle persists on both sides, turn randomly and back up

Move-Around

- Otherwise move forward by $d_{forward}$, turn randomly

Algoritmus Following

Follow

- Whenever an agent is within d_{follow}
 - If an agent is on the right only, turn right
 - If an agent is on the left only, turn left

Algoritmus Dispersion

Dispersion

- Whenever one or more agents are within $d_{disperse}$
 - Move away from $centroid_{disperse}$

Algoritmus Aggregation

Aggregation

- Whenever nearest agent is outside $d_{aggregate}$
 - Turn toward the local $centroid_{aggregate}$, go.
- Otherwise, stop.

Algoritmus Homing

Home

- Whenever at home
 - Stop
- Otherwise, turn toward home, go.