Coalition Coordination for Tightly Coupled Multirobot Tasks with Sensor Constraints

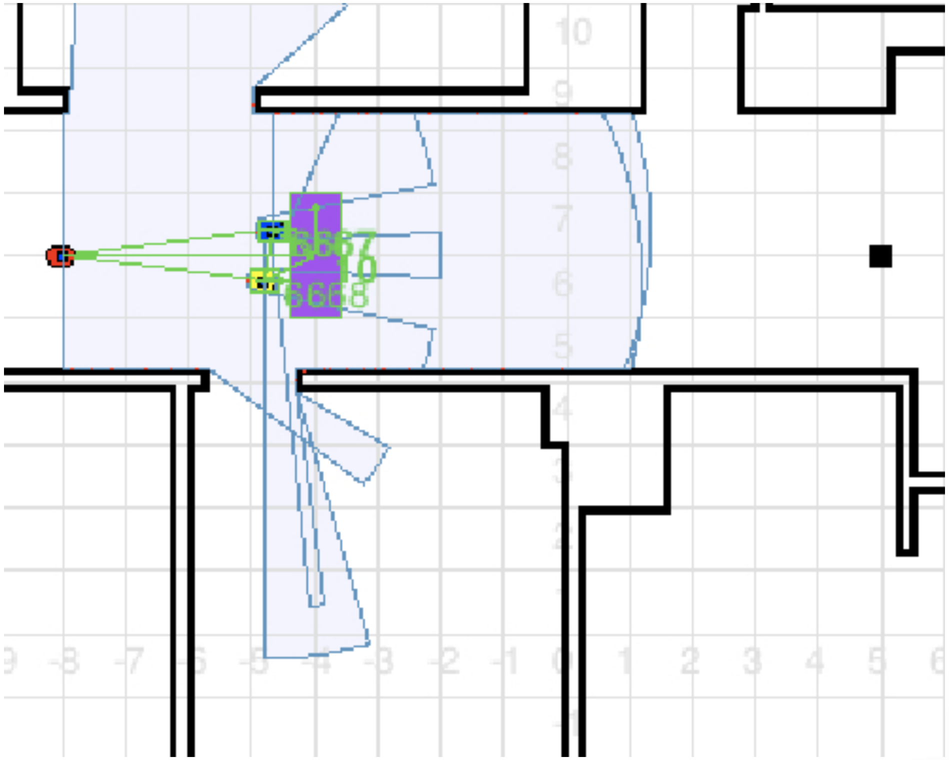
Yu Zhang, Lynne E. Parker and Subbarao Kambhampati

ASU Ira A. Fulton Schools of Engineering

THE UNIVERSITY of TENNESSEE KNOXVILLE
COLLEGE of ENGINEERING

Background

Why coalitions



Background

Before executing coalitions

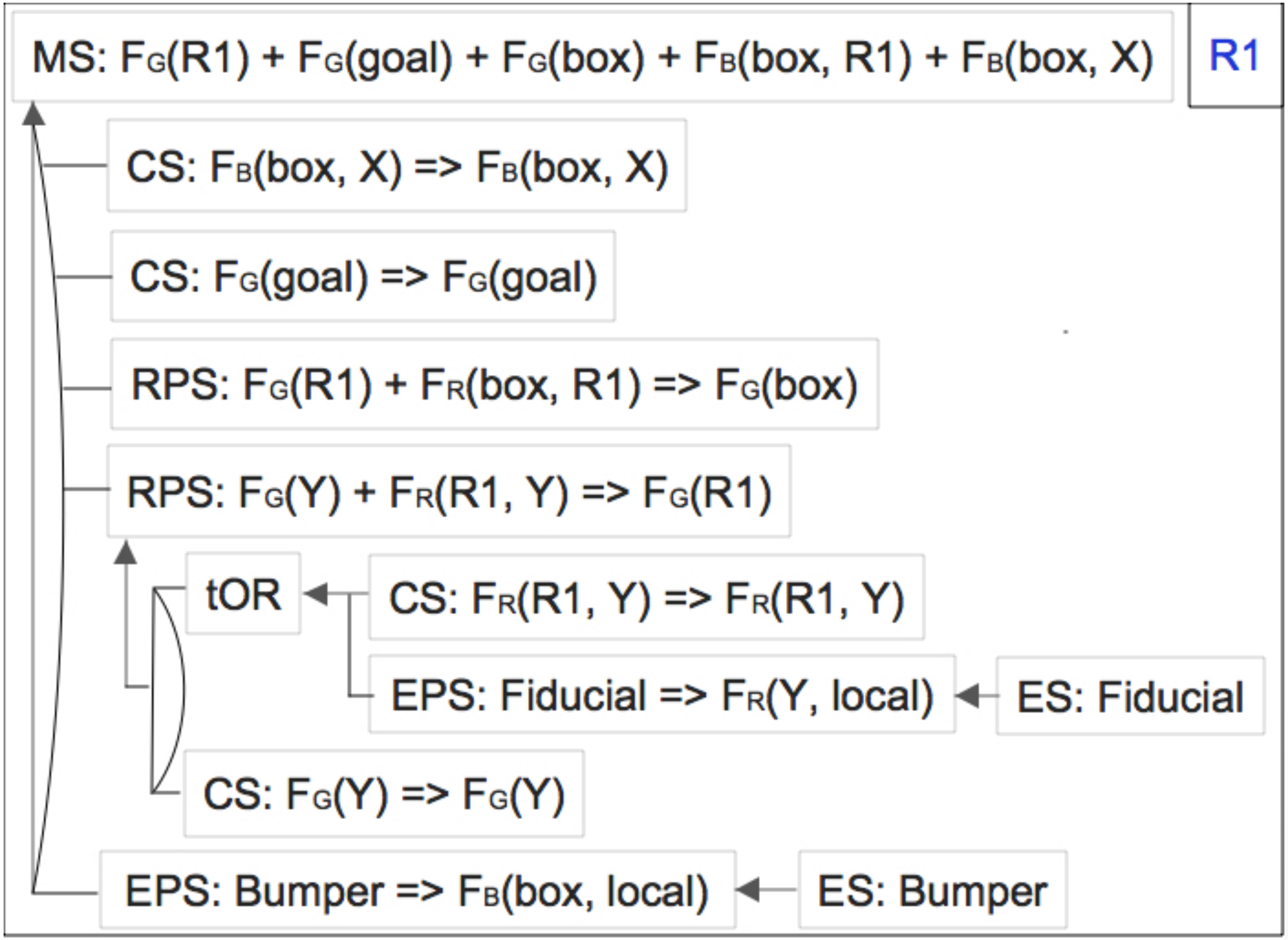
❖ Forming coalitions:

- Previous approaches form coalitions based statically on robot capabilities
- E.g., [Vig and Adams 2006]

- Recent approaches form coalition dynamically based on information flow
- E.g., [Zhang and Parker 2013]

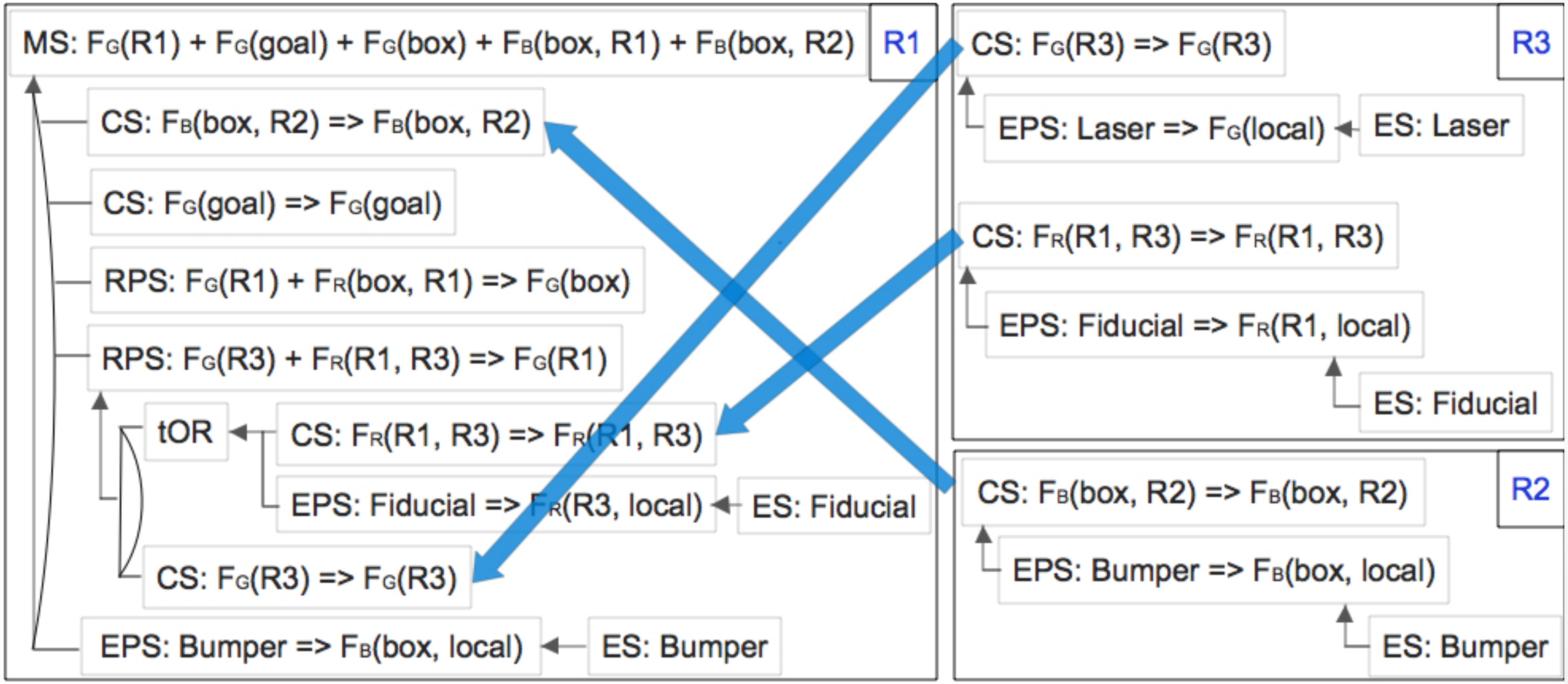
Background

Before executing coalitions



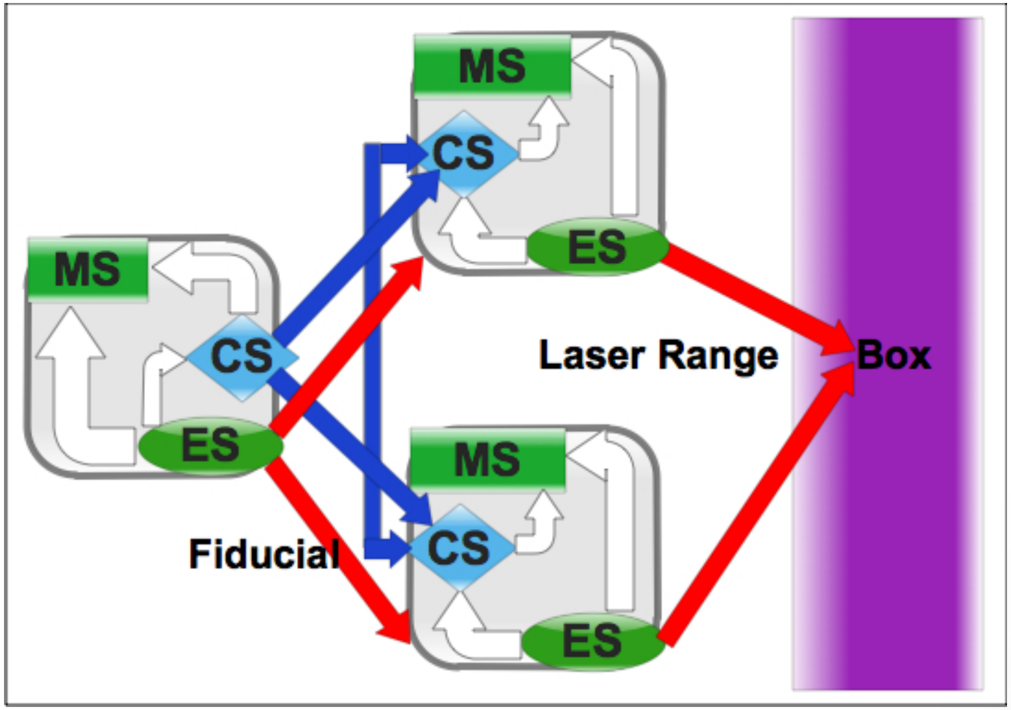
Background

Before executing coalitions



Background

Interactions within coalitions



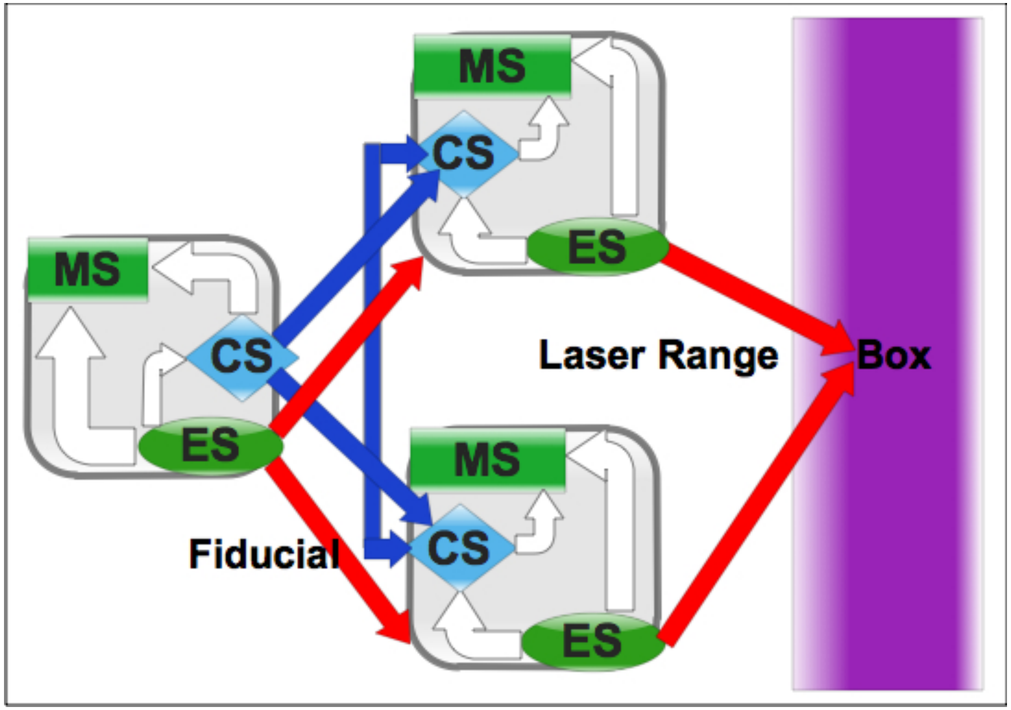
Approach

Coalition coordination

- ➢ Coalitions can overlap
- ➢ Sensor constraints can restrict the robot motion during execution

✧ Coalition coordination is required to reason about synergies between coalitions to enable multitasking

Approach

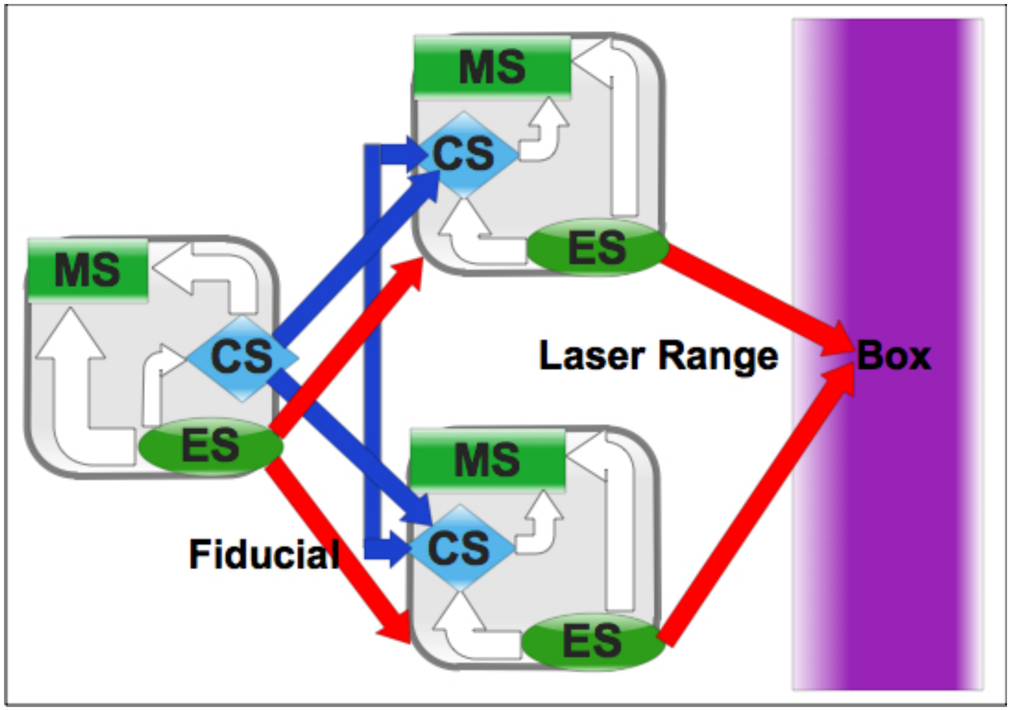


Sensor constraints restrict coalition execution

Definition V.2 (Information Instance). *An information instance, or $F(\mathcal{E})$ (F for short), is an information type with the associated referent set $\mathcal{E}$ ($\mathcal{E}$ is an ordered set). The number of referents or $|\mathcal{E}|$ is determined by the information type.* □

Approach

Coalition coordination



Sensor constraints restrict coalition execution

$$e \rightarrow_{F(\mathcal{E})} \mathcal{E}$$

Definition V.3 (Information Configuration). *The information configuration of an information instance F, denoted by $cf(F)$, represents the set of semantic configurations of the associated referents that are included in the specification of F.* □

Approach

✧ How to maintain a sensor constraint:

$$e \rightarrow_{F(\mathcal{E})} \mathcal{E}$$

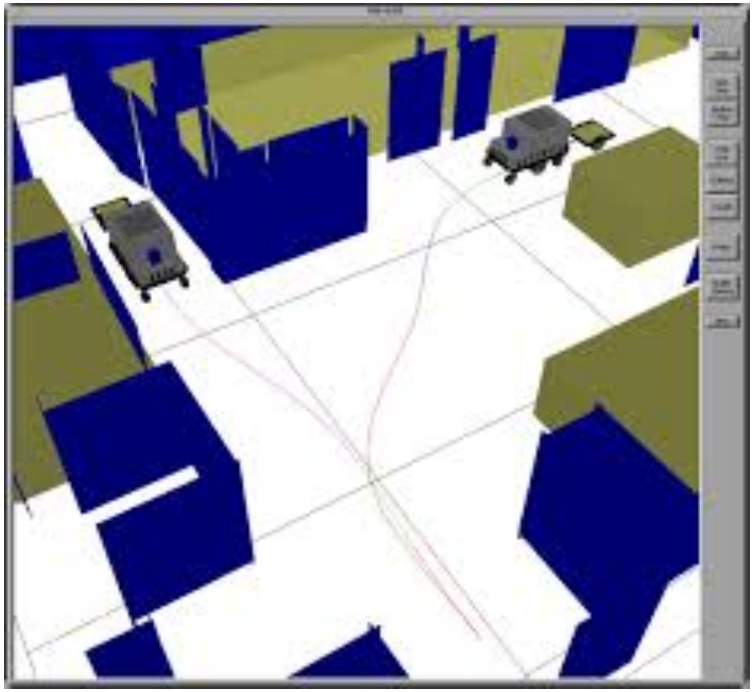
- Find a set of information configurations for individual robots $\{cf(F(e))\}_{e \in \mathcal{E}}$ such that $cf(\{F(e)\}_{e \in \mathcal{E}}) \supseteq cf(F(\mathcal{E}))$

- All except one of these robots can execute a specific MS, which can update v(F(e)) (i.e., the value of F(e))

Definition VI.1 (MS Type)**.** *Given the input IIS s, the type of MS, denoted by $\mathcal{T}(MS(s))$, is an information instance F, such that $v(F)$ is updated by the MS when activated. (If an MS can influence the configuration of more than one information instance, it can be considered as multiple MSs.)* □

Approach

Domain knowledge



- ✧ To identify synergies

Approach

Domain knowledge

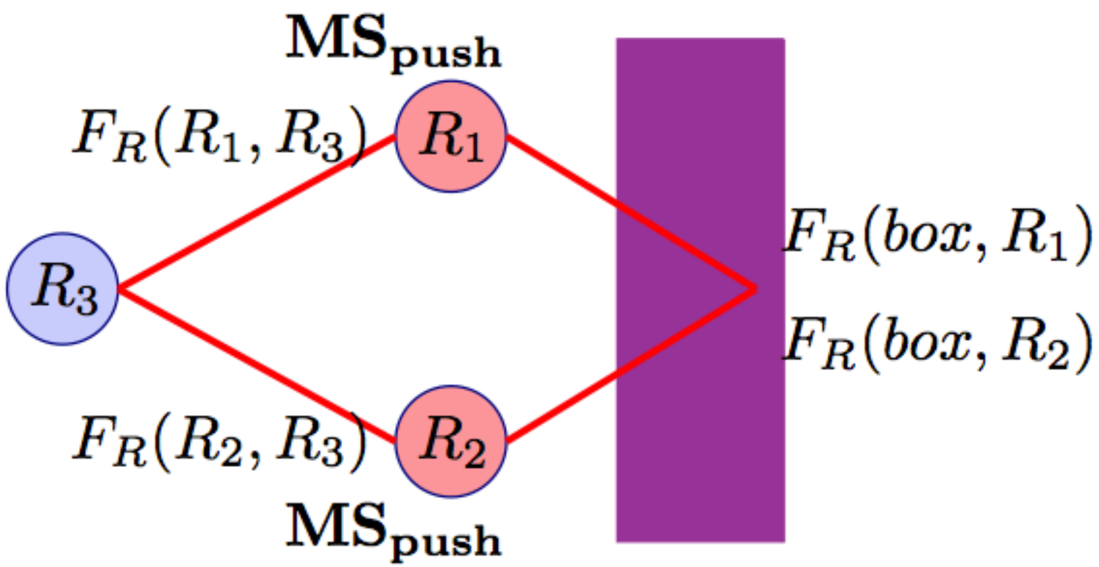
- Domain compatibility,
  E.g., DOM(m(box), MS_{push}({F_G(local), ...})) = F_R(box, local)

- Disjoint configuration
  E.g., orthogonal motion

- Propagated Configuration Constraint
  E.g., cf(F_R(box, R1)) + cf(F_R(box, R2)) -> cf(F_R(R1, R2))

Approach

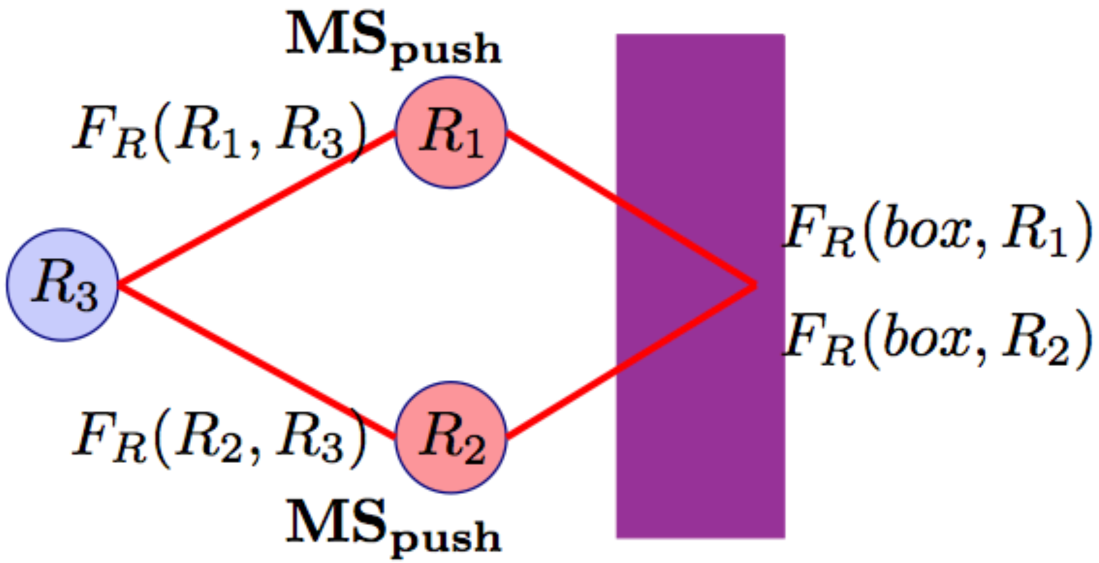
Enable multitasking



Definition VI.2 (Constraint Graph). *Constraint graph is an undirected graph (V, E), in which each node $v \in V$ represents an entity. Two nodes are connected if both are present in the same sensor constraint, and the edge is labeled by the information instance associated with the constraint.* □

Approach

Enable multitasking

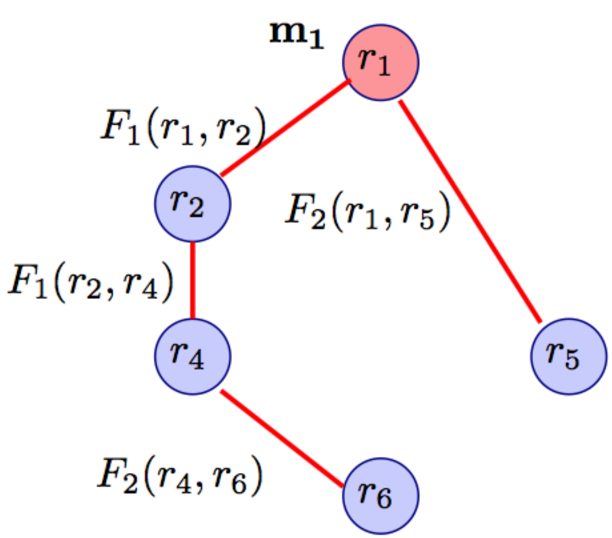


- Identify constraints for individual tasks
- Overlay the constraint graphs
- Search graph for compatible solutions to enable multitasking

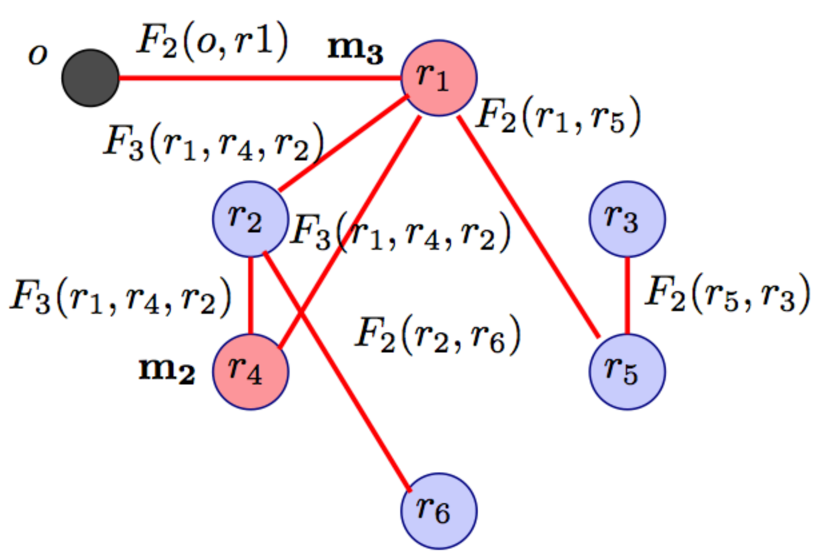
Approach

Enable multitasking

✧ More complex scenarios



1.	$\mathcal{T}(m_2\{F_4(local)\}) = F_4(local)$
2.	$\mathcal{T}(m_3\{F_2(o, local)\}) = F_4(local)$
3.	$DOM(m(o), m_3(\{F_2(o, local)\})) = F_2(o, local)$
4.	$cf(F_4) \cap cf(F_3) \equiv \emptyset$



Approach

Enable multitasking

Algorithm 1 Algorithm for Coalition Coordination

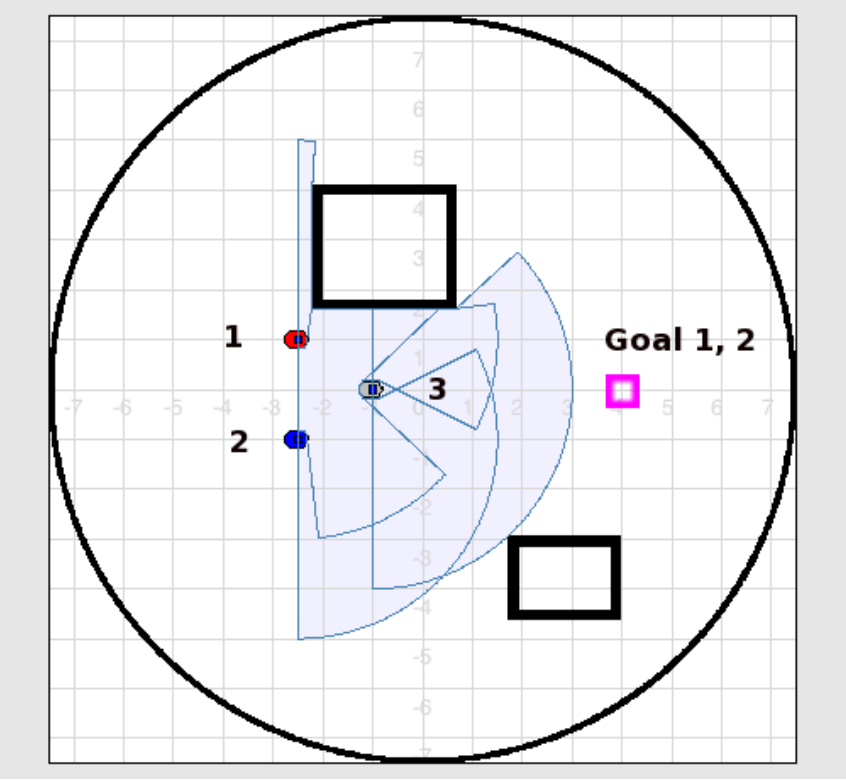
INPUT: a constraint graph $G = (V, E)$; domain compatibilities D; disjoint configurations C.
Create a queue Q and push in all nodes (i.e., robots).
Create a set s for currently satisfied constraints.
Create a set c for checked edges.
Add domain compatibilities to s.
$node$ = POP(Q); Coordinate($node, G, C, D, Q, s, c$).

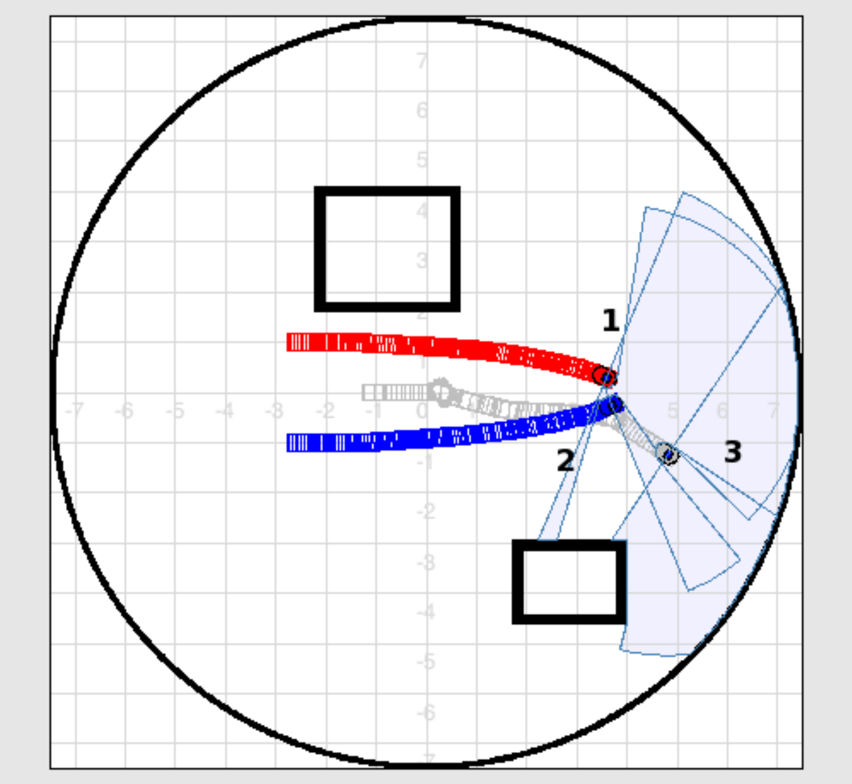
PROCEDURE: Coordinate(v, G, C, D, Q, s, c):
while $(U = \{u : (u, v)_F \in E \ \& \ (u, v)_F \notin c\}) \equiv \emptyset$ **do**
  If $Q \equiv \emptyset$, return solution; else $c = c \cup v$, $v =$ POP(Q).
end while
Add $(u, v)_F \in U$ to c.
if c_1: $cf(F) \not\subseteq cf(s)$ **then**
  for all $x \in \{u, v\}$ **do**
    for all constraint MS m for F on x **do**
      if c_2: $cf(\mathcal{T}(x)) \cap cf(\mathcal{T}(m)) \equiv \emptyset$ or $cf(\mathcal{T}(x)) \equiv \emptyset$ **then**
        $G_x^m =$ Duplicate(G); assign m to x in G_x^m.
        $s_x^m =$ Duplicate(s); add F to s_x^m if the constraint is satisfied.
        Add domain compatibilities to s_x^m.
        $r_x^m =$ Coordinate($v, G_x^m, C, D, Q, s_x^m, c$).
      else
        $r_x^m =$ no solution.
      end if
    end for
  end for
  return $\{r_x^m\}_{m,x}$.
end if
return Coordinate(v, G, C, D, Q, s, c).

✧ Implemented distributedly as DisCSP

Results

Scenario 1

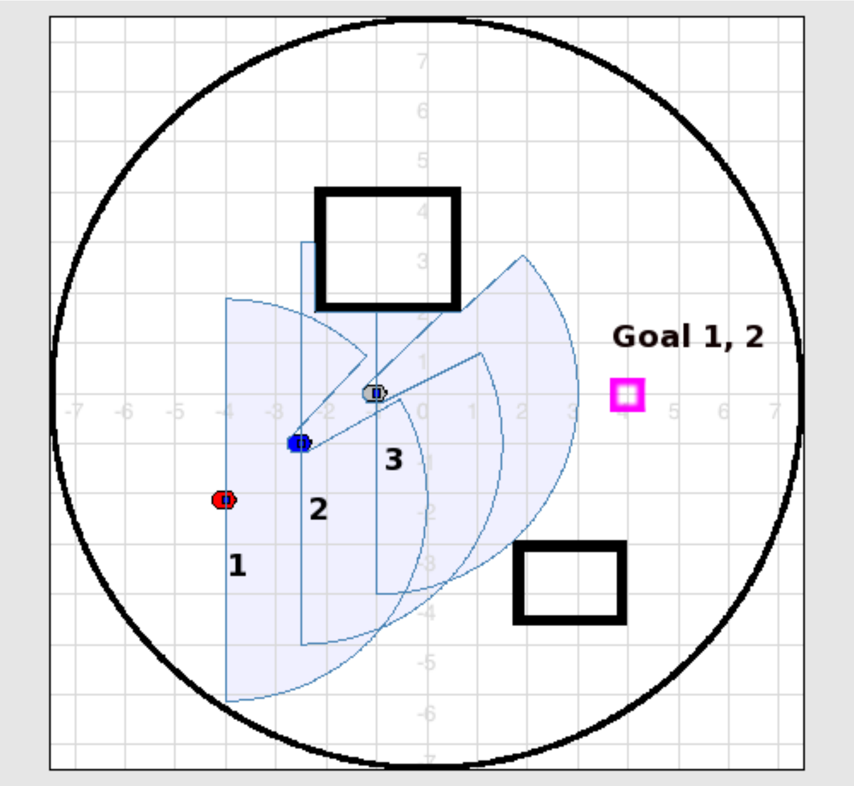


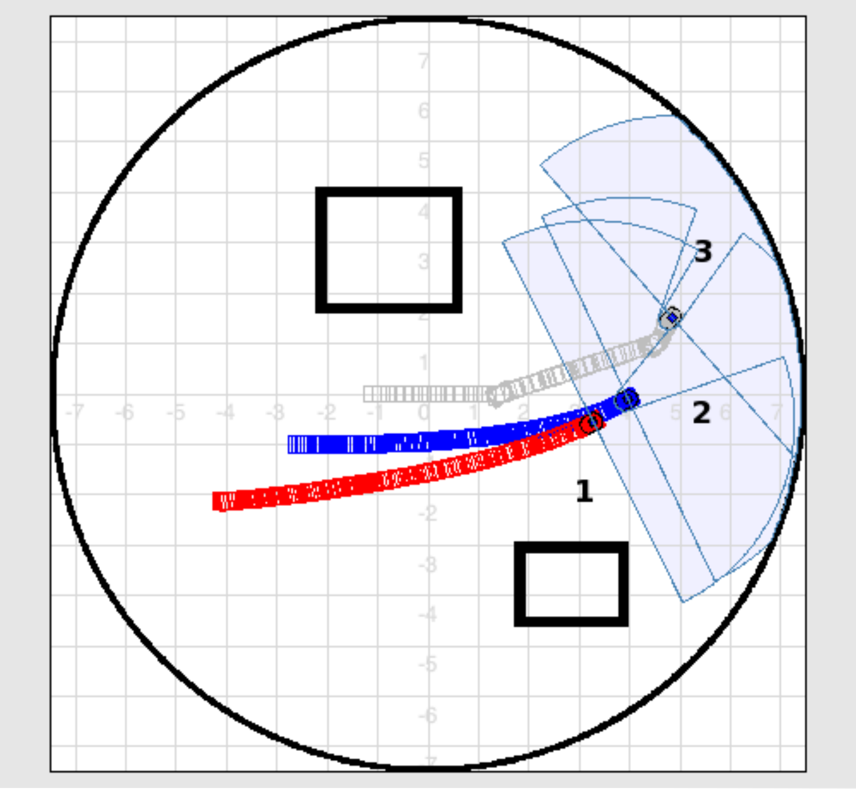


✧ Synergy is utilized to enable multitasking

Results

Scenario 2

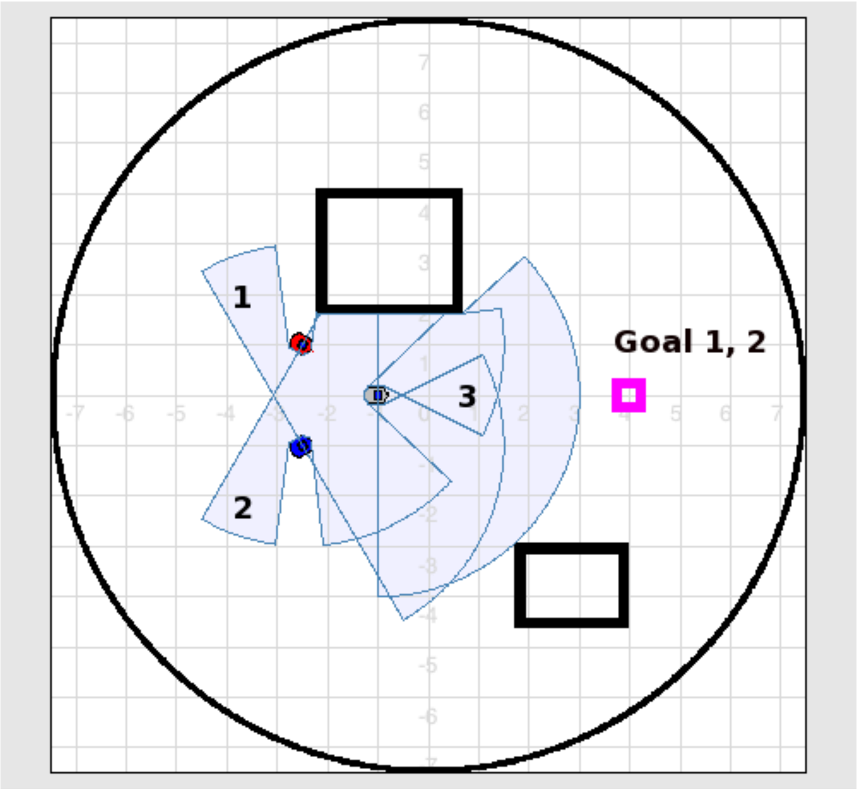


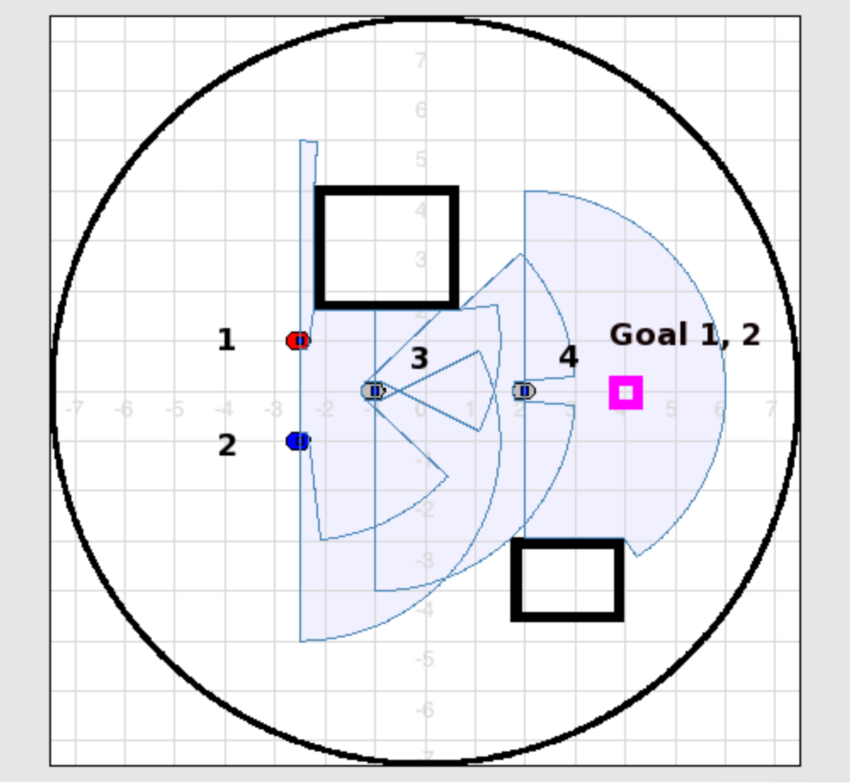


✧ Enable multitasking that is flexible to different configurations

Results

Scenario 3 & 4





- ✧ Identify invalid coalition coordination scenarios

Conclusions

✧ Multitasking in heterogeneous robots:

- Enable coalition execution even when resources are limited
- Automatically identify opportunities for synergies to conserve resources
- Improve task performance

Conclusions

- Proposes a coalition coordination strategy based on sensor constraint for coalition execution in tightly coupled multirobot tasks
- Introduces constraint graph and presents algorithms to search coordination solutions
- Represents the first approach to enabling multi-tasking robots in multirobot tasks (i.e., overlapping coalitions) for improved task performance and resource conservation.