Cooperative Motion and Task Planning Under Temporal Tasks

KTH Electrical Engineering

Meng Guo

Automatic Control Lab, EES
Royal Institute of Technology, KTH, Sweden

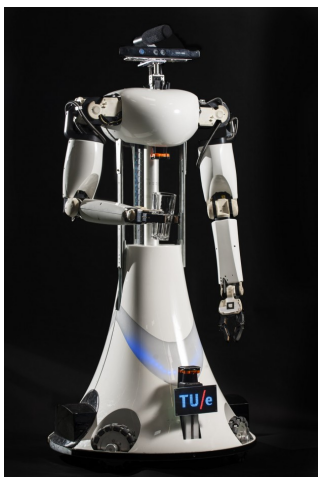
Licentiate Seminar

KTH Electrical Engineering

Motivation

Industrial and domestic robots[1] boosted by

- Development of digital processing units
  - more powerful, in speed and capacity
  - more affordable



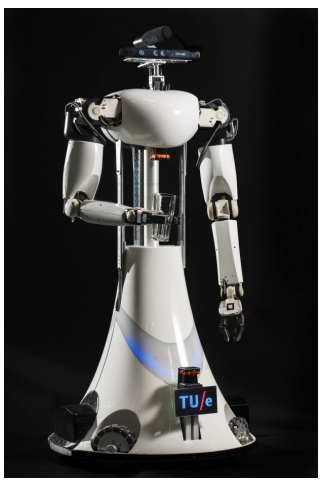
[1] Gostai, Roomba, Romo, Amigo service robots

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Motivation

Industrial and domestic robots[1] boosted by

- Development of digital processing units
  - more powerful, in speed and capacity
  - more affordable
- Wireless communication technology
  - potentially connects the robots
  - integrates with other "smart" devices

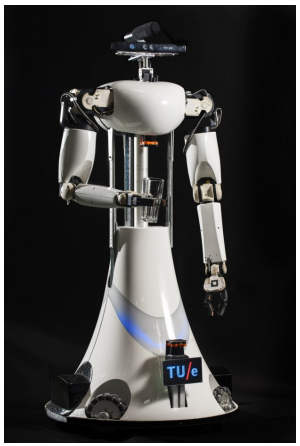


[1] Gostai, Roomba, Romo, Amigo service robots

KTH Electrical Engineering

Motivation

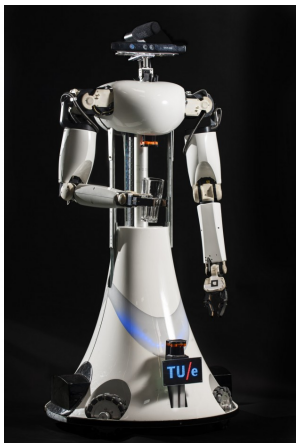
- Imagine ... several care robots in your house



KTH Electrical Engineering

Motivation

- Imagine ... several care robots in your house

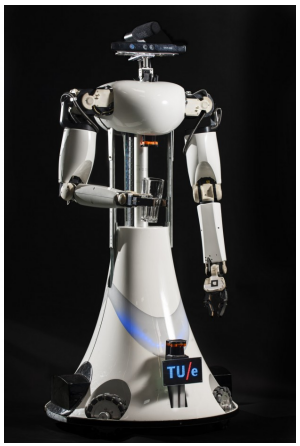


- "Amigo, go to the kitchen, find an apple and bring it to me"

KTH Electrical Engineering

Motivation

- Imagine ... several care robots in your house

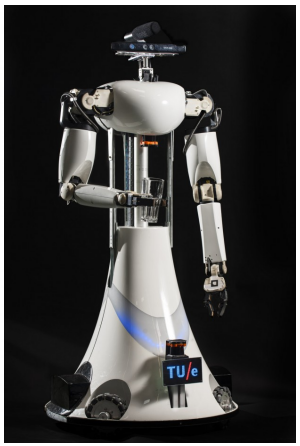


- "Amigo, go to the kitchen, find an apple and bring it to me"
- "Gostai, keep an eye on the kids in the living room and bedroom"

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Motivation

- Imagine ... several care robots in your house



- "Amigo, go to the kitchen, find an apple and bring it to me"
- "Gostai, keep an eye on the kids in the living room and bedroom"
- "Amigo, put all the toys in the basket"
- ...

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Challenges

Challenges for system engineers (us)

- Provide the non-expert end-users

- Design algorithms for robots that

Challenges

Challenges for system engineers (us)

- Provide the non-expert end-users
  - a formal but flexible way to specify daily tasks
  - task execution status as feedback
- Design algorithms for robots that

KTH Electrical Engineering

Challenges

Challenges for system engineers (us)

- Provide the non-expert end-users
  - a formal but flexible way to specify daily tasks
  - task execution status as feedback
- Design algorithms for robots that
  - synthesize and execute a plan to satisfy the task
  - without or with minimal human intervention

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Challenges

Challenges for system engineers (us)

- Provide the non-expert end-users
  - a formal but flexible way to specify daily tasks
  - task execution status as feedback
- Design algorithms for robots that
  - synthesize and execute a plan to satisfy the task
  - without or with minimal human intervention
  - accommodate changes in the workspace
  - initiate communication with other devices
  - handle collaborative tasks

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

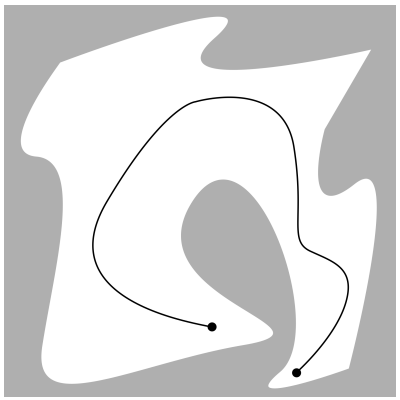
Background

- Motion and task planning

[2] S. M. LaValle. *Planning algorithms*, 2006.
[3] M. Ghallab et al., *Automated planning: theory & practice*, 2004.
[4] C. Baier, J.-P Katoen. *Principles of model checking*, 2008.

Background

- Motion and task planning
  - motion plan of dynamic systems[2]

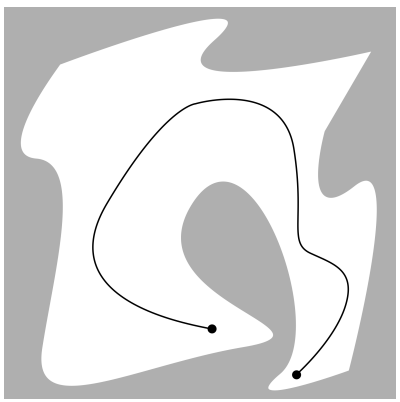


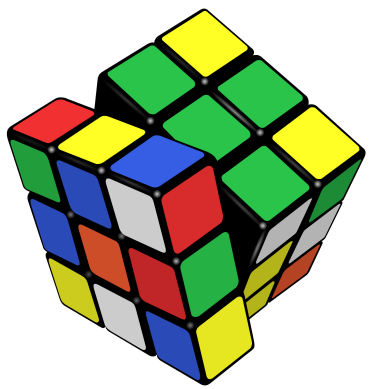
[2] S. M. LaValle. *Planning algorithms*, 2006.
[3] M. Ghallab et al., *Automated planning: theory & practice*, 2004.
[4] C. Baier, J.-P Katoen. *Principles of model checking*, 2008.

KTH Electrical Engineering

Background

- Motion and task planning
  - motion plan of dynamic systems[2]
  - task plan for discrete-event systems[3]



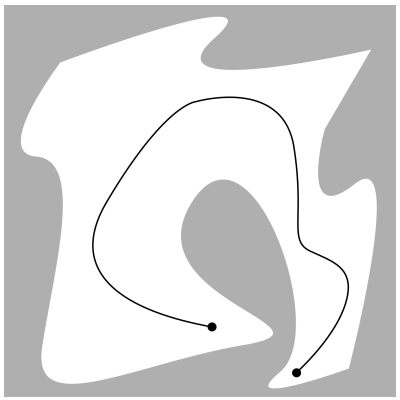


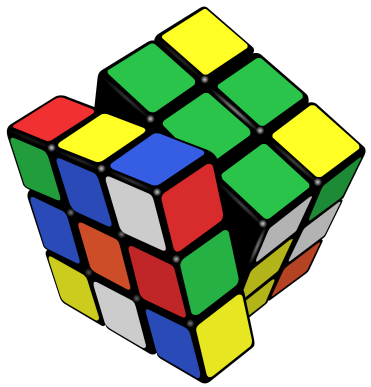
[2] S. M. LaValle. *Planning algorithms*, 2006.
[3] M. Ghallab et al., *Automated planning: theory & practice*, 2004.
[4] C. Baier, J.-P Katoen. *Principles of model checking*, 2008.

KTH Electrical Engineering

Background

- Motion and task planning
  - motion plan of dynamic systems[2]
  - task plan for discrete-event systems[3]





- Model checking
  - for verification[4]
  - for plan synthesis

[2] S. M. LaValle. *Planning algorithms*, 2006.
[3] M. Ghallab et al., *Automated planning: theory & practice*, 2004.
[4] C. Baier, J.-P Katoen. *Principles of model checking*, 2008.

KTH Electrical Engineering

Multi-agent System

- Multi-agent system to solve a global task
  - decompose into sub-tasks
  - top-down, tightly-coupled
- Multi-agent system with local tasks
  - favour individual interests
  - bottom-up, loosely-coupled

KTH Electrical Engineering

Main Contributions

- Reconfiguration for single- and multi-agent systems
  - M. Guo, K. H. Johansson and D. V. Dimarogonas. Motion and Action Planning under LTL Specification using Navigation Functions and Action Description Language. *IROS*, Japan, 2013.
  - M. Guo and D. V. Dimarogonas. Reconfiguration in Motion Planning of Single- and Multi-agent Systems under Infeasible Local LTL Specifications. *CDC*, Italy, 2013.
- Real-time adaptation for single- and multi-agent systems
  - M. Guo, K. H. Johansson and D. V. Dimarogonas. Revising Motion Planning under Linear Temporal Logic Specifications in Partially Known Workspaces. *ICRA*, Germany, 2013.
  - M. Guo and D. V. Dimarogonas. Distributed Plan Reconfiguration via Knowledge Transfer in Multi-agent Systems under Local LTL Specifications. *ICRA*, Hongkong, 2014.
  - M. Guo and D. V. Dimarogonas. Multi-agent Plan Reconfiguration under Local LTL Specifications. *IJRR*. Submitted.

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

KTH Electrical Engineering

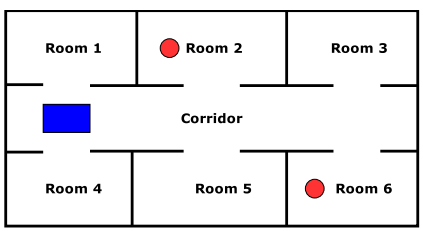
Motion Abstraction

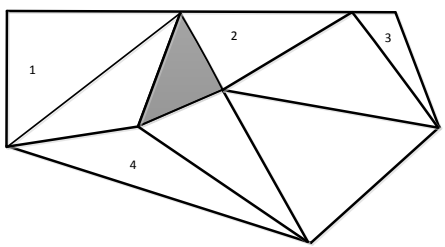
- Abstracted as the weighted finite transition system (FTS)

$$\mathcal{T}_c = (\Pi, \longrightarrow_c, \Pi_0, AP, L_c, W_c)$$

where the finite regions $\Pi = \{\pi_1, \pi_2, \cdots, \pi_W\}$

- Examples:





KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Motion Abstraction

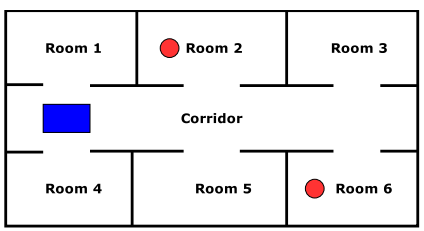
- Abstracted as the weighted finite transition system (FTS)

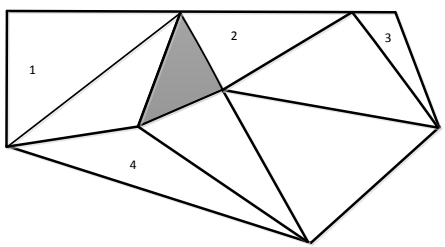
$$\mathcal{T}_c = (\Pi, \longrightarrow_c, \Pi_0, AP, L_c, W_c)$$

where the finite regions $\Pi = \{\pi_1, \pi_2, \cdots, \pi_W\}$

  - properties of interest $AP = AP_r \cap AP_p$
  - Π_0 initial states
  - $\longrightarrow_c \subseteq \Pi \times \Pi$, control-driven transition
  - $L_c : \Pi \to 2^{AP}$, labelling function
  - $W_c : \Pi \times \Pi \to \mathbb{R}^+$, weight function

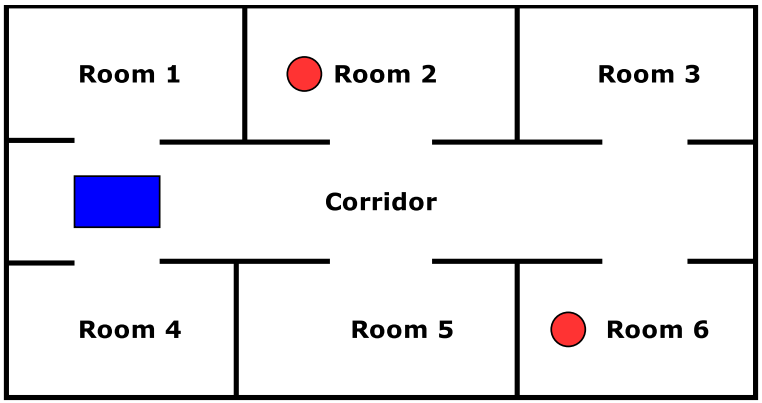
- Examples:

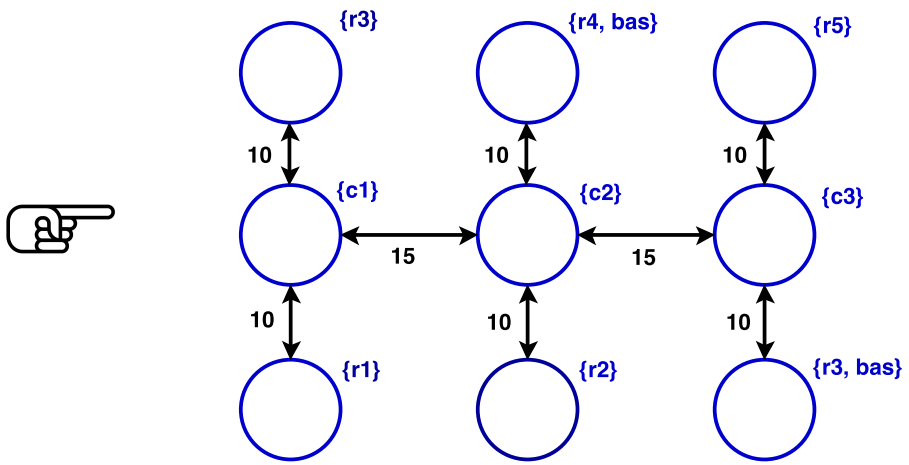




KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Abstraction Outcome





KTH Electrical Engineering

Temporal Task Specification

How to formally specify the task?

- Language: Linear-time Temporal Logic (LTL) formula
- Syntax:

$$\varphi ::= \texttt{True} \mid a \mid \varphi_1 \vee \varphi_2 \mid \neg\varphi \mid \bigcirc \varphi \mid \varphi_1 \,\mathsf{U}\, \varphi_2$$

- Semantics[5]
- Specified over AP

[5] C. Baier, J.-P Katoen. *Principles of model checking*, 2008.

KTH Electrical Engineering

Temporal Task Specification

How to formally specify the task?

- Language: Linear-time Temporal Logic (LTL) formula
- Syntax:

$$\varphi ::= \texttt{True} \mid a \mid \varphi_1 \vee \varphi_2 \mid \neg\varphi \mid \bigcirc \varphi \mid \varphi_1 \,\mathsf{U}\, \varphi_2$$

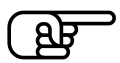
- Semantics[5]
- Specified over AP
- To specify control tasks:
  - Safety: $\Box\neg\,\varphi_1$. Order: $\Diamond(\varphi_1 \wedge \Diamond(\varphi_2 \wedge \Diamond\varphi_3))$.
  - Response: $\varphi_1 \Rightarrow \varphi_2$. Liveness: $\Box\Diamond\varphi_1$.

[5] C. Baier, J.-P Katoen. *Principles of model checking*, 2008.

KTH Electrical Engineering

Task Interpretation

- "go to the kitchen, find an apple and bring it to me"

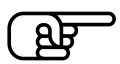


- $\varphi = \Diamond\big((\texttt{Kit} \wedge \texttt{PickApp}) \wedge \Diamond(\Box\texttt{Liv})\big)$

KTH Electrical Engineering

Task Interpretation

- "go to the kitchen, find an apple and bring it to me"
- "keep an eye on the kids in the living room and bedroom"

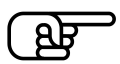


- $\varphi = \Diamond\big((\texttt{Kit} \wedge \texttt{PickApp}) \wedge \Diamond(\Box\texttt{Liv})\big)$
- $\varphi = \Box\Diamond\texttt{Liv} \wedge \Box\Diamond\texttt{Bed}$

KTH Electrical Engineering

Task Interpretation

- "go to the kitchen, find an apple and bring it to me"
- "keep an eye on the kids in the living room and bedroom"
- "put all the toys in the basket"
- ...



- $\varphi = \Diamond\big((\texttt{Kit} \wedge \texttt{PickApp}) \wedge \Diamond(\Box\texttt{Liv})\big)$
- $\varphi = \Box\Diamond\texttt{Liv} \wedge \Box\Diamond\texttt{Bed}$
- $\varphi = \Box\Diamond\Big(\texttt{PickToy} \rightarrow \big(\neg\texttt{PickToy}\,\mathcal{U}\,(\texttt{Bas} \wedge \texttt{DropToy})\big)\Big)$
- ...

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Problem Formulation

Given the FTS $\mathcal{T}_c$ and the LTL task φ

Problem Formulation

Given the FTS $\mathcal{T}_c$ and the LTL task φ

- Synthesize a discrete plan that satisfies φ

$$r1\, c1\, c2\, c3\, r3$$
$$c3\, c2\, r2\, c2\, r5$$
$$c2\, r2\, c2\, c1 (r1)^{\omega}$$

KTH Electrical Engineering

Problem Formulation

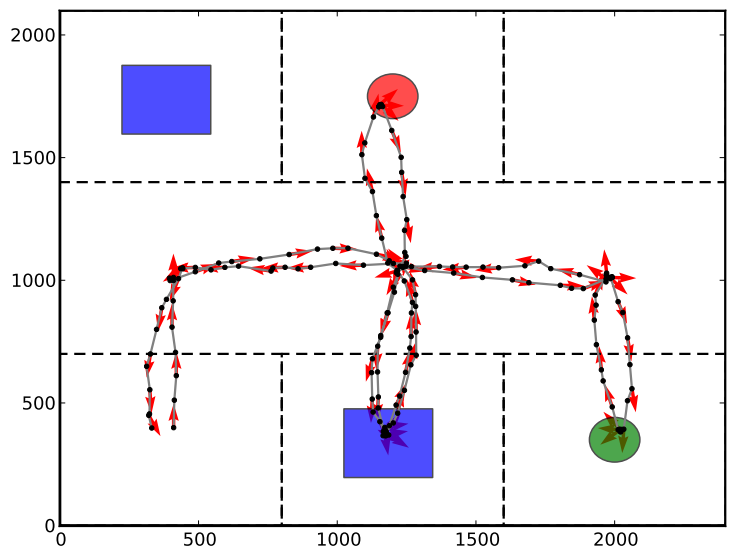
Given the FTS $\mathcal{T}_c$ and the LTL task φ

- Synthesize a discrete plan that satisfies φ
- Construct the hybrid control strategy to execute the derived plan

$$r1\, c1\, c2\, c3\, r3$$

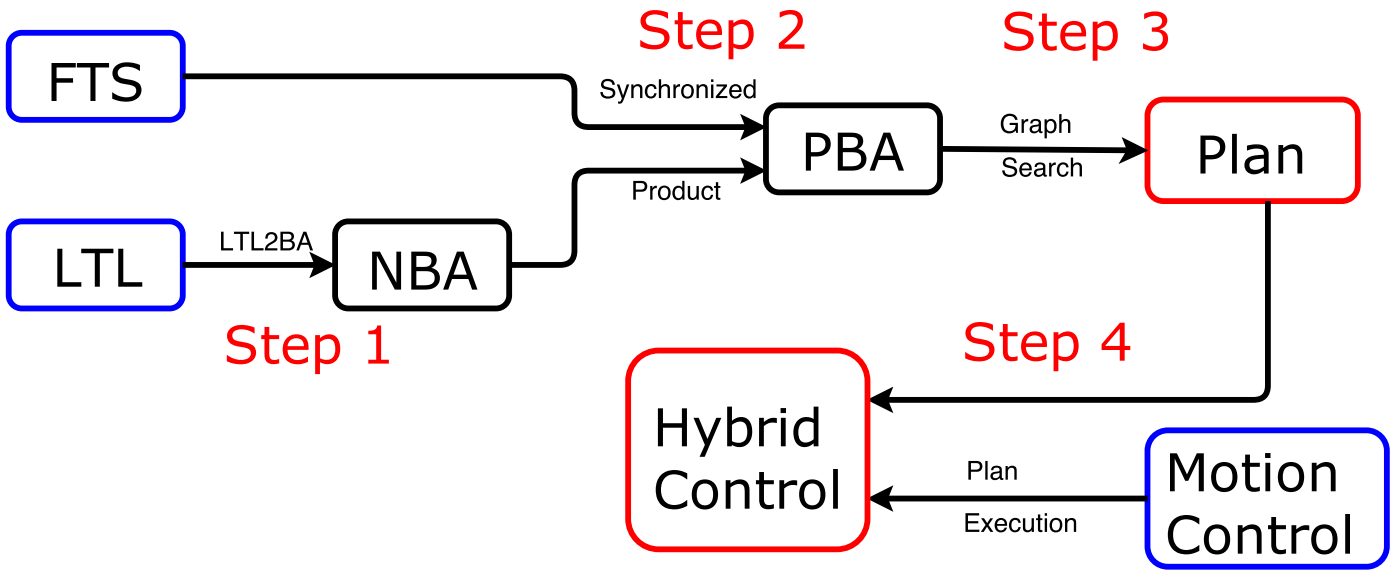
$$c3\, c2\, r2\, c2\, r5$$

$$c2\, r2\, c2\, c1 \left(r1\right)^{\omega}$$



KTH Electrical Engineering

Nominal Solution Outline



- Automata-based model-checking algorithm[6]
- Hybrid controller synthesis[7]

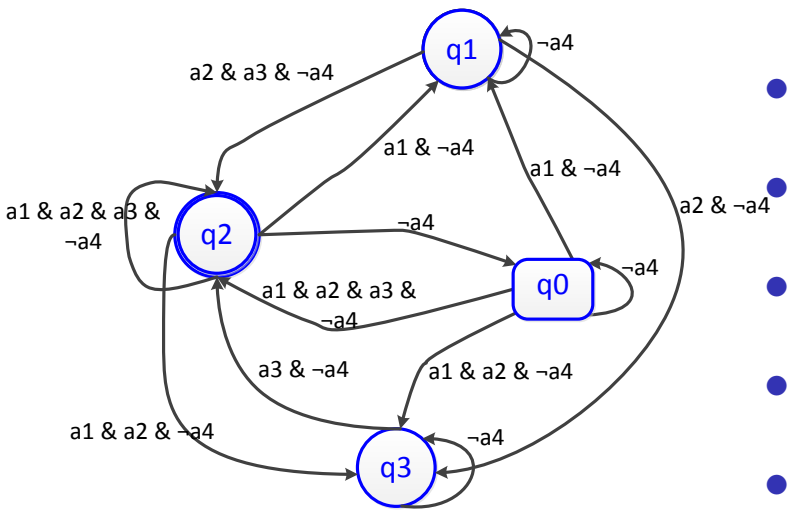
[6] C. Baier, J.-P Katoen. *Principles of model checking*, 2008.

[7] G. E. Fainekos et al., Temporal logic motion planning for dynamic robots. *Automatica*, 2009

Step 1. Translation

- Translate φ into the Nondeterministic Büchi automaton (NBA) $\mathcal{A}_\varphi$ over 2^{AP}:

$$\mathcal{A}_\varphi = (Q, 2^{AP}, \delta, Q_0, \mathcal{F}),$$



- Q is a finite set of states
- 2^{AP} are alphabets.
- $\delta \subseteq Q \times 2^{AP} \times Q$.
- Q_0, $\mathcal{F}$ are initial, accepting states.
- $\chi(q_m, q_n) = \{l \in 2^{AP} | (q_m, l, q_n) \in \delta\}$.

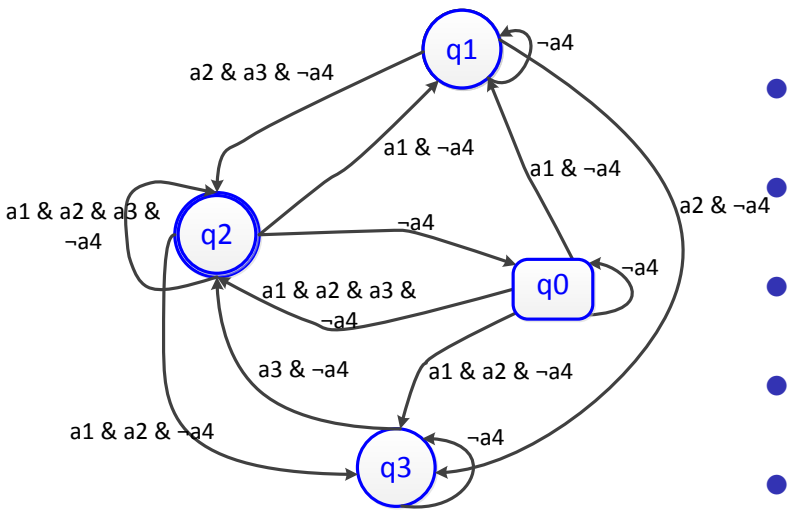
[8] D. Oddoux, P. Gastin. *LTL2BA software*, 2009.

KTH Electrical Engineering

Step 1. Translation

- Translate φ into the Nondeterministic Büchi automaton (NBA) $\mathcal{A}_\varphi$ over 2^{AP}:

$$\mathcal{A}_\varphi = (Q, 2^{AP}, \delta, Q_0, \mathcal{F}),$$



- Q is a finite set of states
- 2^{AP} are alphabets.
- $\delta \subseteq Q \times 2^{AP} \times Q$.
- Q_0, $\mathcal{F}$ are initial, accepting states.
- $\chi(q_m, q_n) = \{l \in 2^{AP} | (q_m, l, q_n) \in \delta\}$.

- Automated
- Fast translation algorithm[8]

[8] D. Oddoux, P. Gastin. *LTL2BA software*, 2009.

KTH Electrical Engineering

Step 2. Product Automaton

- Construct the weighted product automaton $\mathcal{A}_p = \mathcal{T}_c \otimes \mathcal{A}_\varphi$:

$$\mathcal{A}_p = (Q', \delta', Q'_0, \mathcal{F}', W_p)$$

where $Q' = \Pi \times Q$; $Q'_0 = \Pi_0 \times Q_0$; $\mathcal{F}' = \Pi \times \mathcal{F}$

- $\delta' \subseteq Q \times Q$, where $(\langle \pi_i, q_m \rangle, \langle \pi_j, q_n \rangle) \in \delta'$ iff $(\pi_i, \pi_j) \in \longrightarrow_c$ and $(q_m, L_c(\pi_i), q_n) \in \delta$
- $W_p : \delta' \to \mathbb{R}^+$. $W_p((\langle \pi_i, q_m \rangle, \langle \pi_j, q_n \rangle)) = W_c(\pi_i, \pi_j)$

KTH Electrical Engineering

Step 2. Product Automaton

- Construct the weighted product automaton $\mathcal{A}_p = \mathcal{T}_c \otimes \mathcal{A}_\varphi$:

$$\mathcal{A}_p = (Q', \delta', Q'_0, \mathcal{F}', W_p)$$

where $Q' = \Pi \times Q$; $Q'_0 = \Pi_0 \times Q_0$; $\mathcal{F}' = \Pi \times \mathcal{F}$

  - $\delta' \subseteq Q \times Q$, where $(\langle \pi_i, q_m \rangle, \langle \pi_j, q_n \rangle) \in \delta'$ iff $(\pi_i, \pi_j) \in \longrightarrow_c$ and $(q_m, L_c(\pi_i), q_n) \in \delta$
  - $W_p : \delta' \to \mathbb{R}^+$. $W_p((\langle \pi_i, q_m \rangle, \langle \pi_j, q_n \rangle)) = W_c(\pi_i, \pi_j)$

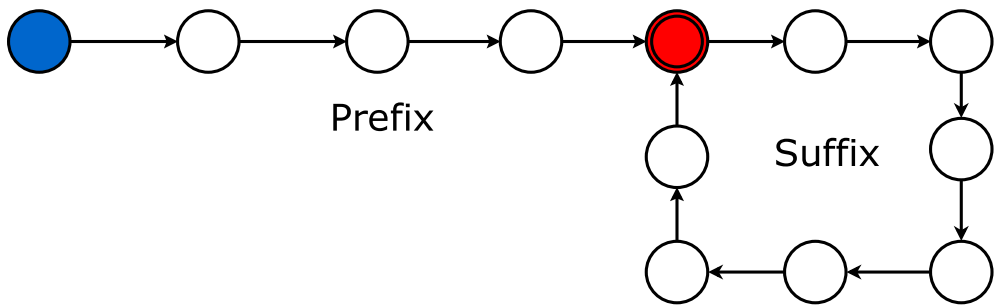
- Algorithms
  - static construction
  - on-the-fly construction

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Plan Structure and Cost

- Accepting run R of $\mathcal{A}_p$ with the prefix-suffix structure

$$R = R_{\text{pre}} \, (R_{\text{suf}})^{\omega} = q'_0 \, q'_1 \cdots q'_{f-1} \, \left[\, q'_f \, q'_{f+1} \cdots q'_n \, \right]^{\omega}$$



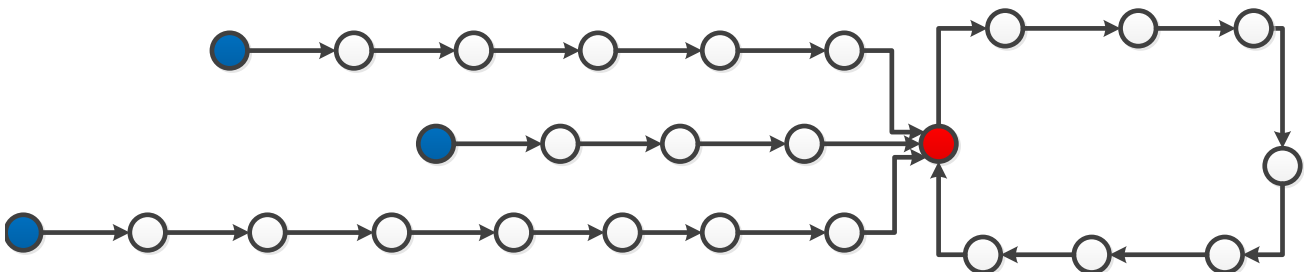
- The total cost:

$$\texttt{Cost}(R, \mathcal{A}_p) = \sum_{i=0}^{f-1} W_p(q'_i, \, q'_{i+1}) + \gamma \sum_{i=f}^{n-1} W_p(q'_i, \, q'_{i+1})$$

KTH Electrical Engineering

Step 3. Graph Search

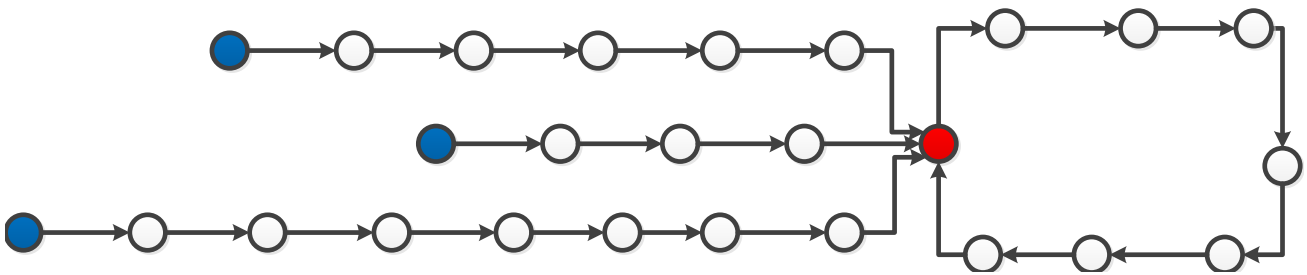
- Algorithm based on Nested-Dijkstra shortest path
  - shortest path from every $q'_0 \in Q'_0$ to every $q'_f \in \mathcal{F}'$
  - shortest cycle from q'_f and back



KTH Electrical Engineering

Step 3. Graph Search

- Algorithm based on Nested-Dijkstra shortest path
  - shortest path from every $q'_0 \in Q'_0$ to every $q'_f \in \mathcal{F}'$
  - shortest cycle from q'_f and back



- The derived accepting run R_{opt} with minimal cost
- Corresponding plan $\tau = R_{\text{opt}}|_{\Pi}$
- Complexity $\mathcal{O}(|\delta'| \cdot \log_2 |Q'| \cdot (|Q'_0| + |\mathcal{F}'|))$

KTH Electrical Engineering

Step 4. Hybrid Control Strategy

To execute $\tau = R_{\text{opt}}|_{\Pi}$ using $\mathcal{U}(x(t), \pi_i, \pi_j)$

- generate an infinite execution with finite representation
- monitor the execution status
- record past motions

$r1\, c1\, c2\, c3\, r3$

$c3\, c2\, r2\, c2\, r5$

$c2\, r2\, c2\, c1 (r1)^{\omega}$

KTH Electrical Engineering

Shortcomings of Nominal Solution

- Reconfiguration
  - plan as sequence of motion (no actions)
  - feasible task
- Real-time adaptation
  - fully-known workspace
  - plan synthesized once off-line
  - executed regardless of the real observation

KTH Electrical Engineering

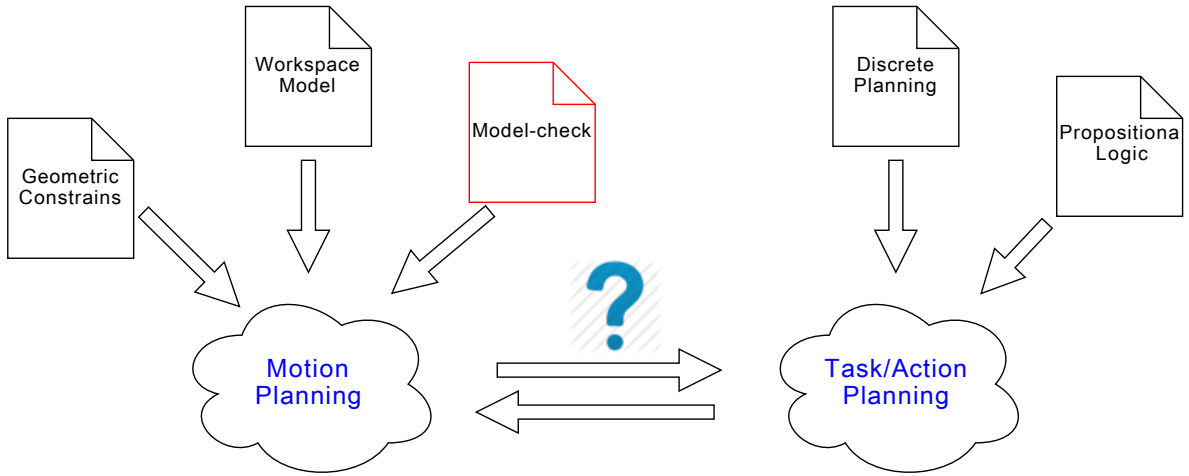
Shortcomings of Nominal Solution

- Reconfiguration
  - plan as sequence of motion (no actions)
  - feasible task
- Real-time adaptation
  - fully-known workspace
  - plan synthesized once off-line
  - executed regardless of the real observation
- Multi-agent system with local tasks
  - independent or dependent tasks
  - communication
  - collaborative tasks

KTH Electrical Engineering

KTH Electrical Engineering

Motion and Action Planning



- Why is it necessary?
  - to automate the choice of actions
  - plan as sequence of motion and actions
- Why plan motion and action together?
  - the purpose of "going somewhere" is to "do something"
- Why model motion and action separately?
  - robot's mobility: depend on the workspace structure
  - robot's capable actions: relatively fixed

KTH Electrical Engineering

Model of Mobility and Action

- Mobility abstraction is given by (similarly as $\mathcal{T}_c$):

$$\mathcal{M} = (\Pi_{\mathcal{M}}, act_{\mathcal{M}}, \longrightarrow_{\mathcal{M}}, \Pi_{\mathcal{M},0}, \Psi_{\mathcal{M}}, L_{\mathcal{M}}, W_{\mathcal{M}}),$$

- Capable action set $Act_{\mathcal{B}} = \{act_{\mathcal{B},0}, act_{\mathcal{B},1}, \cdots, act_{\mathcal{B},K}\}$
- Precondition function:

$$\texttt{Cond} : Act_{\mathcal{B}} \times 2^{\Psi_p} \times 2^{\Psi_s} \longrightarrow \mathsf{True/False}$$

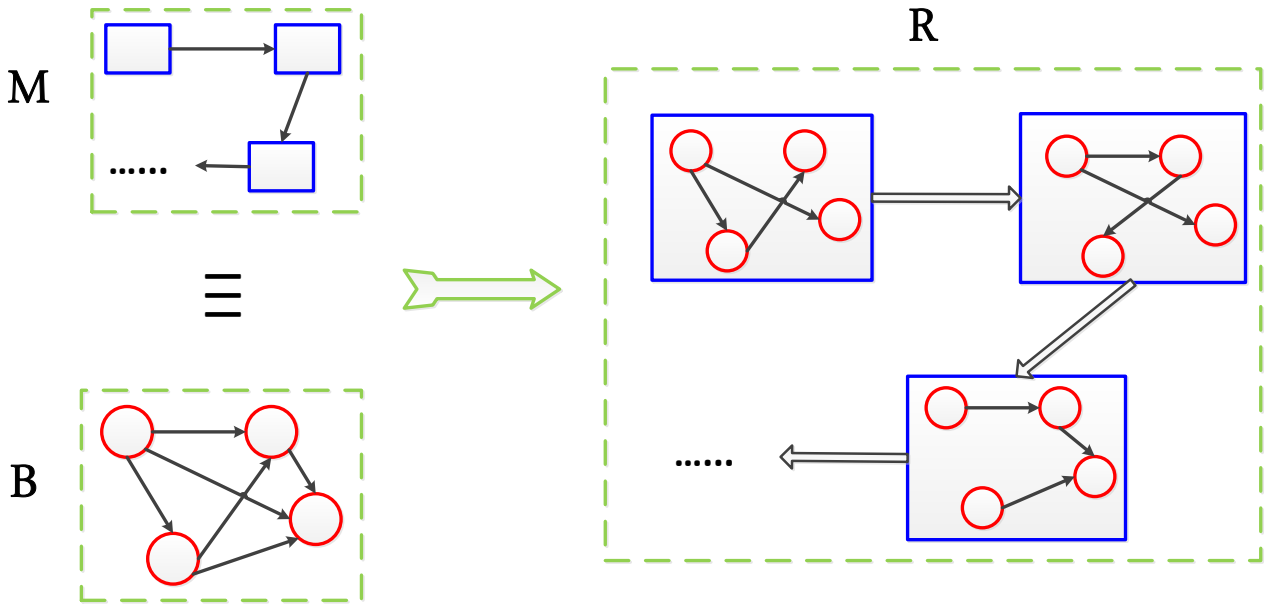
- Effect function:

$$\texttt{Eff} : Act_{\mathcal{B}} \times (2^{\Psi_s} \times \Psi_b) \longrightarrow (2^{\Psi_s} \times \Psi_b),$$

- Mimic action description language (ADL)

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Model of Complete Functionalities



- Complete functionalities by composing $\mathcal{M}$ and $\mathcal{B}$

$$\mathcal{R} = (\Pi_{\mathcal{R}}, Act_{\mathcal{R}}, \longrightarrow_{\mathcal{R}}, \Pi_{\mathcal{R},0}, \Psi_{\mathcal{R}}, L_{\mathcal{R}}, W_{\mathcal{R}}),$$

- Automated algorithm

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

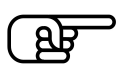
Results

- Much richer task specifications
- Given φ over robot motion, action and internal states
- Accepting run with prefix-suffix structure with similar definition for cost

Results

- Much richer task specifications
- Given φ over robot motion, action and internal states
- Accepting run with prefix-suffix structure with similar definition for cost
- Plan as a sequence of motion and action, minimal cost
- Construct the hybrid controller accordingly

$r1\, c1\, c2\, c3\, r3$
$c3\, c2\, r2\, c2\, r5$
$c2\, r2\, c2\, c1 (r1)^{\omega}$



$r1\, c1\, c2\, c3\, r3$ PickG
$c3\, c2\, r2$ Drop $c2\, r5$ PickR
$c2\, r2$ Drop $c2\, c1 (r1)^{\omega}$

KTH Electrical Engineering

Potentially Infeasible Task

- Def. 2.6: φ is infeasible for $\mathcal{T}_c$ if no accepting run of $\mathcal{A}_p$ can be found.
- Closely related to partially-known workspace
- Example:
  - "go to the kitchen, find an apple and bring it to me"
    $\Rightarrow$ infeasible if no apple
  - "keep an eye on the kids in the living room and bedroom"
    $\Rightarrow$ infeasible if not known where the bedroom is

KTH Electrical Engineering

Potentially Infeasible Task

- Def. 2.6: φ is infeasible for $\mathcal{T}_c$ if no accepting run of $\mathcal{A}_p$ can be found.
- Closely related to partially-known workspace
- Example:
  - "go to the kitchen, find an apple and bring it to me"
    $\Rightarrow$ infeasible if no apple
  - "keep an eye on the kids in the living room and bedroom"
    $\Rightarrow$ infeasible if not known where the bedroom is
- Nominal solution fails
- Our goal

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Potentially Infeasible Task

- Def. 2.6: φ is infeasible for $\mathcal{T}_c$ if no accepting run of $\mathcal{A}_p$ can be found.
- Closely related to partially-known workspace
- Example:
  - "go to the kitchen, find an apple and bring it to me"
    $\Rightarrow$ infeasible if no apple
  - "keep an eye on the kids in the living room and bedroom"
    $\Rightarrow$ infeasible if not known where the bedroom is
- Nominal solution fails
- Our goal
  - synthesize the plan satisfying the task partially
  - user-defined balance on satisfiability and cost of the plan

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Solution

- Relaxed product automaton $\mathcal{A}_r = \mathcal{T}_c \times \mathcal{A}_\varphi$:

$$\mathcal{A}_r = (Q', 2^{AP}, \delta', Q'_0, \mathcal{F}', W_r)$$

where $Q' = \Pi \times Q$; $Q'_0 = \Pi_0 \times Q_0$; $\mathcal{F}' = \Pi \times \mathcal{F}$;

KTH Electrical Engineering

Solution

- Relaxed product automaton $\mathcal{A}_r = \mathcal{T}_c \times \mathcal{A}_\varphi$:

$$\mathcal{A}_r = (Q', 2^{AP}, \delta', Q'_0, \mathcal{F}', W_r)$$

where $Q' = \Pi \times Q$; $Q'_0 = \Pi_0 \times Q_0$; $\mathcal{F}' = \Pi \times \mathcal{F}$;

  - $\delta' \subseteq Q' \times Q'$. $(\langle \pi_i, q_m \rangle, \langle \pi_j, q_n \rangle) \in \delta'$ iff $(\pi_i, \pi_j) \in \longrightarrow_c$ and $\exists\, l \in 2^{AP}$ such that $(q_m, l, q_n) \in \delta$.
  - $W_r : \delta' \to \mathbb{R}^+$ is the weight function:

$$\begin{aligned} &W_r(\langle \pi_i, q_m \rangle, \langle \pi_j, q_n \rangle) \\ &= W_c(\pi_i, \pi_j) + \alpha \cdot \texttt{Dist}(L(\pi_i), \chi(q_m, q_n)). \end{aligned}$$

- $\alpha \geq 0$: user-defined balance

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Balanced Cost

- Accepting run R with prefix-suffix structure:

$$\begin{aligned} R &= q'_0\, q'_1 \cdots [\, q'_f\, q'_{f+1} \cdots q'_n \,]^\omega \\ &= \langle \pi_0, q_0 \rangle \langle \pi_1, q_1 \rangle \cdots [\, \langle \pi_f, q_f \rangle \cdots \langle \pi_n, q_n \rangle \,]^\omega \ , \end{aligned}$$

- The balanced cost of an accepting run:

$$\begin{aligned} \texttt{Cost}(R, \mathcal{A}_r) &= \sum_{i=0}^{f-1} W_r(q'_i, q'_{i+1}) + \gamma \sum_{i=f}^{n-1} W_r(q'_i, q'_{i+1}) \\ &= \texttt{cost}_\tau + \alpha \cdot \texttt{dist}_\varphi \end{aligned}$$

- $\texttt{cost}_\tau$: implementation cost; $\texttt{dist}_\varphi$: distance to φ
- $\texttt{dist}_\varphi = 0 \Rightarrow R|_\Pi \models \varphi$

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Results

- The balanced accepting run

$$R_{\text{bal}} = \min_{R} \ \texttt{Cost}(R, \mathcal{A}_r)$$

- The balanced plan $\tau_{\text{bal}} = R_{\text{bal}}|_{\Pi}$

KTH Electrical Engineering

Results

- The balanced accepting run

$$R_{\text{bal}} = \min_{R} \texttt{Cost}(R, \mathcal{A}_r)$$

- The balanced plan $\tau_{\text{bal}} = R_{\text{bal}}|_{\Pi}$
- Proposed algorithms:
  - synthesize R_{bal}, given γ and α
  - computes the associated $\texttt{cost}_\tau$ and $\texttt{dist}_\varphi$ for τ_{bal}
- Theorem 3.1: If φ is feasible over $\mathcal{T}_c$, the balanced plan τ_{bal} satisfies φ if $\alpha > \underline{\alpha}$.

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

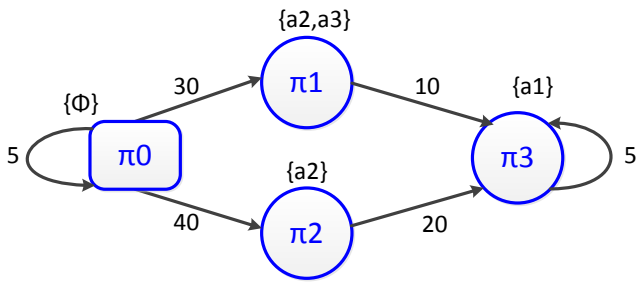
Feedback by Tuning α

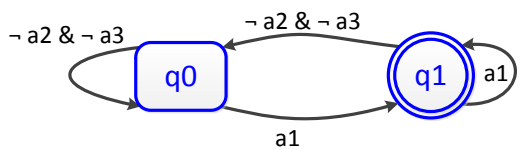
- α: tunable balance between $\texttt{cost}_\tau$ and $\texttt{dist}_\varphi$
  - $\alpha \uparrow \quad \Rightarrow \quad \texttt{dist}_\varphi \downarrow \quad \Rightarrow \quad$ satisfy φ more
  - $\alpha \downarrow \quad \Rightarrow \quad \texttt{cost}_\tau \downarrow \quad \Rightarrow \quad$ cheaper

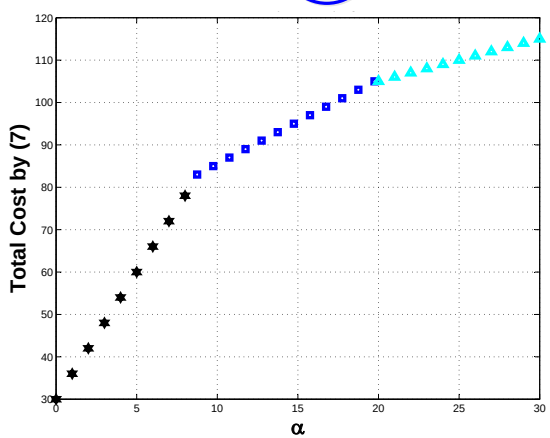
KTH VETENSKAP OCH KONST
KTH Electrical Engineering

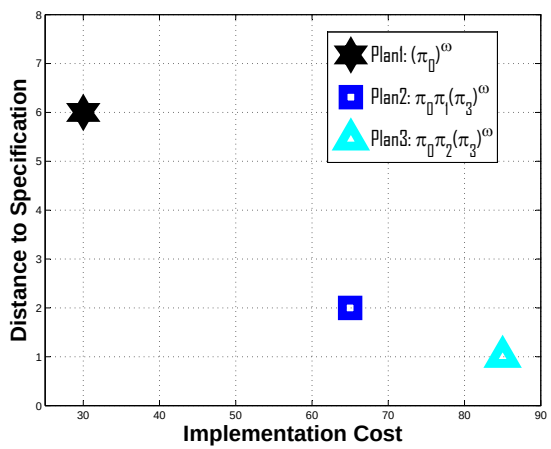
Feedback by Tuning α

- α: tunable balance between $\texttt{cost}_\tau$ and $\texttt{dist}_\varphi$
  - $\alpha \uparrow \quad \Rightarrow \quad \texttt{dist}_\varphi \downarrow \quad \Rightarrow$ satisfy φ more
  - $\alpha \downarrow \quad \Rightarrow \quad \texttt{cost}_\tau \downarrow \quad \Rightarrow$ cheaper
- Example: $\varphi = \Box\Diamond a_1 \wedge \Box\neg(a_2 \wedge a_3)$









KTH Electrical Engineering

Soft and Hard Specification

- Specification with two distinctive parts:

$$\varphi = \varphi^{\text{hard}} \wedge \varphi^{\text{soft}}$$

- φ^{hard}, for safety or security
  - "do not go to the balcony", "always alarm if see fire"
- φ^{soft}, for additional achievement (maybe infeasible)
  - "collect the toys in all rooms"

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Soft and Hard Specification

- Specification with two distinctive parts:

$$\varphi = \varphi^{\text{hard}} \wedge \varphi^{\text{soft}}$$

- φ^{hard}, for safety or security
  - "do not go to the balcony", "always alarm if see fire"
- φ^{soft}, for additional achievement (maybe infeasible)
  - "collect the toys in all rooms"
- Nominal solution fails
- Our goal

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Soft and Hard Specification

- Specification with two distinctive parts:

$$\varphi = \varphi^{\text{hard}} \wedge \varphi^{\text{soft}}$$

- φ^{hard}, for safety or security
  - "do not go to the balcony", "always alarm if see fire"
- φ^{soft}, for additional achievement (maybe infeasible)
  - "collect the toys in all rooms"
- Nominal solution fails
- Our goal
  - synthesize the plan satisfies φ^{hard} completely and φ^{soft} partially
  - user-defined balance on satisfiability and cost of the plan

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Solution

- Relaxed intersection of $\mathcal{A}^{\text{soft}}$ and $\mathcal{A}^{\text{hard}}$ as $\tilde{\mathcal{A}}_\varphi$, by Alg. 9

$$\tilde{\mathcal{A}}_\varphi = (Q,\, 2^{AP},\, \delta,\, Q_0,\, \mathcal{F})$$

- Safety-ensured and relaxed product automaton

$$\tilde{\mathcal{A}}_r = \mathcal{T}_c \times \tilde{\mathcal{A}}_\varphi = (Q',\, \delta',\, Q'_0,\, \mathcal{F}',\, W_r)$$

where $Q' = \Pi \times Q$, $Q'_0 = \Pi_0 \times Q_0$, $\mathcal{F}' = \Pi \times \mathcal{F}$

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Solution

- Relaxed intersection of $\mathcal{A}^{\text{soft}}$ and $\mathcal{A}^{\text{hard}}$ as $\tilde{\mathcal{A}}_\varphi$, by Alg. 9

$$\tilde{\mathcal{A}}_\varphi = (Q,\, 2^{AP},\, \delta,\, Q_0,\, \mathcal{F})$$

- Safety-ensured and relaxed product automaton

$$\tilde{\mathcal{A}}_r = \mathcal{T}_c \times \tilde{\mathcal{A}}_\varphi = (Q',\, \delta',\, Q'_0,\, \mathcal{F}',\, W_r)$$

where $Q' = \Pi \times Q$, $Q'_0 = \Pi_0 \times Q_0$, $\mathcal{F}' = \Pi \times \mathcal{F}$

- $\delta' : Q' \to 2^{Q'}$. $\langle \pi_j,\, q_n \rangle \in \delta'(\langle \pi_i,\, q_m \rangle)$ iff $(\pi_i,\, \pi_j) \in \longrightarrow_c$ and $q_n \in \delta(q_m,\, L_c(\pi_i))$
- $W_r : \delta' \to \mathbb{R}^+$ is the weight function.

$$\begin{aligned} &W_r(\langle \pi_i,\, q_m \rangle,\, \langle \pi_j,\, q_n \rangle) \\ =&W_c(\pi_i,\, \pi_j) + \alpha \cdot \texttt{Dist}(L_c(\pi_i),\, \chi_{\text{soft}}(q_2,\, \check{q}_2)) \end{aligned}$$

KTH Electrical Engineering

Results

- Theorem 3.3: Assume R is an accepting run of $\tilde{\mathcal{A}}_r$.
  $\tau = R|_\Pi$ is safe for $\mathcal{T}_c$ and φ.
- Balanced cost of accepting runs:

$$\texttt{Cost}(R, \tilde{\mathcal{A}}_r) = \texttt{cost}_\tau + \alpha \cdot \texttt{dist}_{\varphi^{\text{soft}}}$$

KTH Electrical Engineering

Results

- Theorem 3.3: Assume R is an accepting run of $\tilde{\mathcal{A}}_r$.
  $\tau = R|_\Pi$ is safe for $\mathcal{T}_c$ and φ.
- Balanced cost of accepting runs:

$$\texttt{Cost}(R, \tilde{\mathcal{A}}_r) = \texttt{cost}_\tau + \alpha \cdot \texttt{dist}_{\varphi^{\text{soft}}}$$

- The safe accepting run

$$R_{\text{safe}} = \min_R \ \texttt{Cost}(R, \tilde{\mathcal{A}}_r)$$

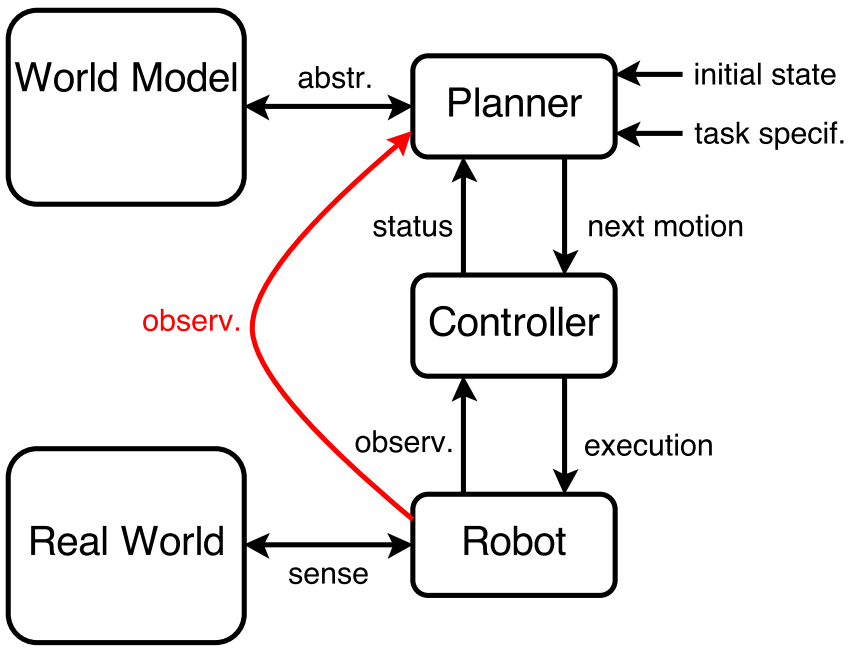
- The safe plan $\tau_{\text{safe}} = R_{\text{safe}}|_\Pi$
- Similar algorithms to synthesize R_{safe}
- Lemma 3.2: $\texttt{dist}_{\varphi^{\text{soft}}} = 0 \Rightarrow \tau_{\text{safe}} \models \varphi$

KTH Electrical Engineering

On-line Planning

Why put planner on-line?

- To handle partially-known workspace
- Plan may not be executed as expected
- Plan could be improved
- Feed real-time observation back to planner



KTH Electrical Engineering

Problem Formulation

- The agent's FTS at time $t \geq 0$:

$$\mathcal{T}_c^t = (\Pi, \longrightarrow_c^t, \Pi_0, AP, L_c^t, W_c^t)$$

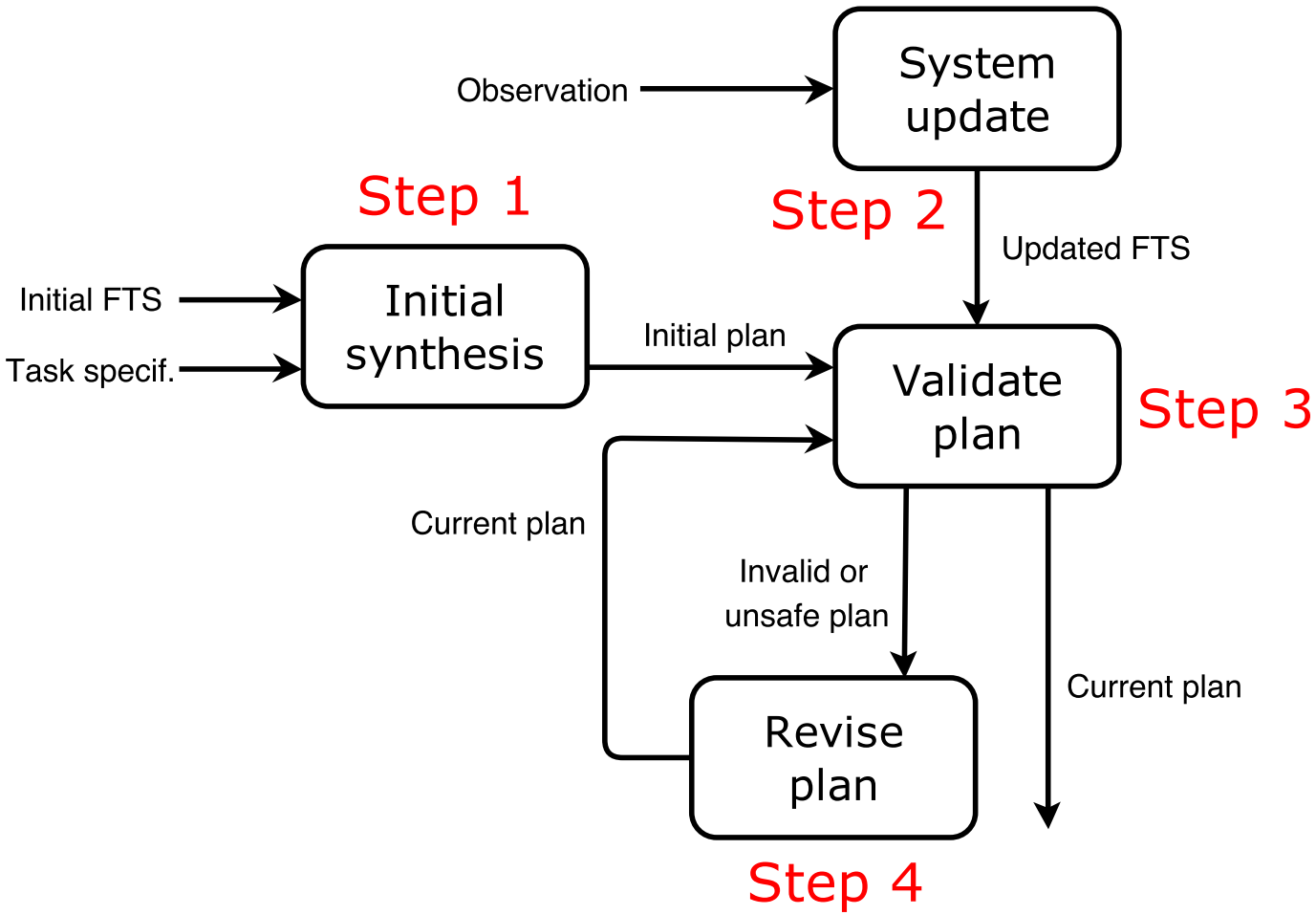
- Task specification

$$\varphi = \varphi^{\text{soft}} \wedge \varphi^{\text{hard}}$$

- Our goal:
  - model the robot's sensing
  - update the system model
  - guarantee the plan is always valid and safe

KTH Electrical Engineering

Solution Framework



KTH Electrical Engineering

Initial Synthesis and System Update

- Step 1. initial synthesis at $t = 0$
  - τ^0 obtained for feasible or infeasible task
  - starts executing τ^0

Initial Synthesis and System Update

- Step 1. initial synthesis at $t = 0$
  - τ^0 obtained for feasible or infeasible task
  - starts executing τ^0
- Step 2. knowledge update at $t \geq 0$
  - sensing information obtained

$$\mathbf{Sense}^t = \{((\pi,\, S,\, S_\neg),\, E,\, E_\neg)\}$$

KTH Electrical Engineering

Initial Synthesis and System Update

- Step 1. initial synthesis at $t = 0$
  - τ^0 obtained for feasible or infeasible task
  - starts executing τ^0
- Step 2. knowledge update at $t \geq 0$
  - sensing information obtained

$$\mathbf{Sense}^t = \{((\pi,\, S,\, S_\neg),\, E,\, E_\neg)\}$$

- Step 2. update $\mathcal{T}_c^t$ based on $\mathbf{Sense}^t$

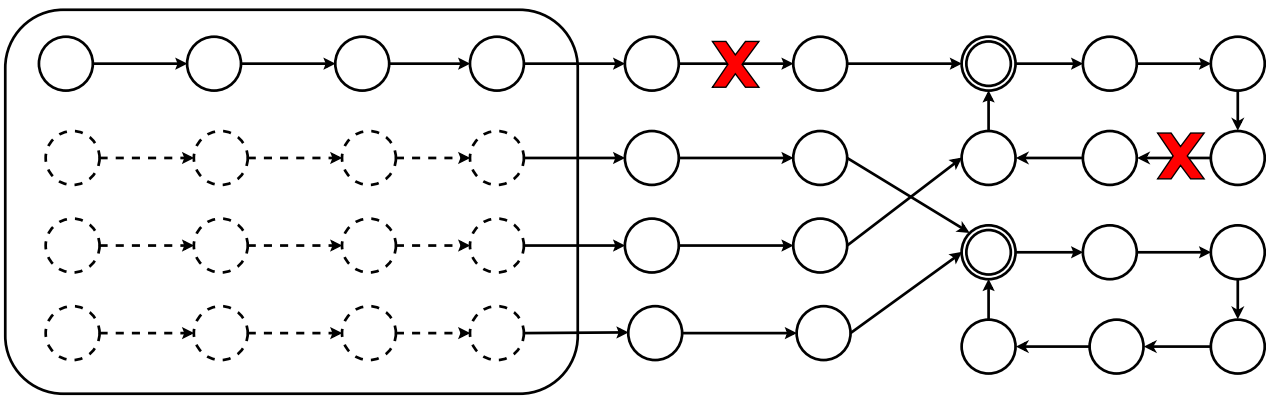
KTH Electrical Engineering

Plan Verification and Revision

- Step 3. validate the current plan
  - validity $\Longleftrightarrow$ invalid transitions
  - safety $\Longleftrightarrow$ unsafe transitions

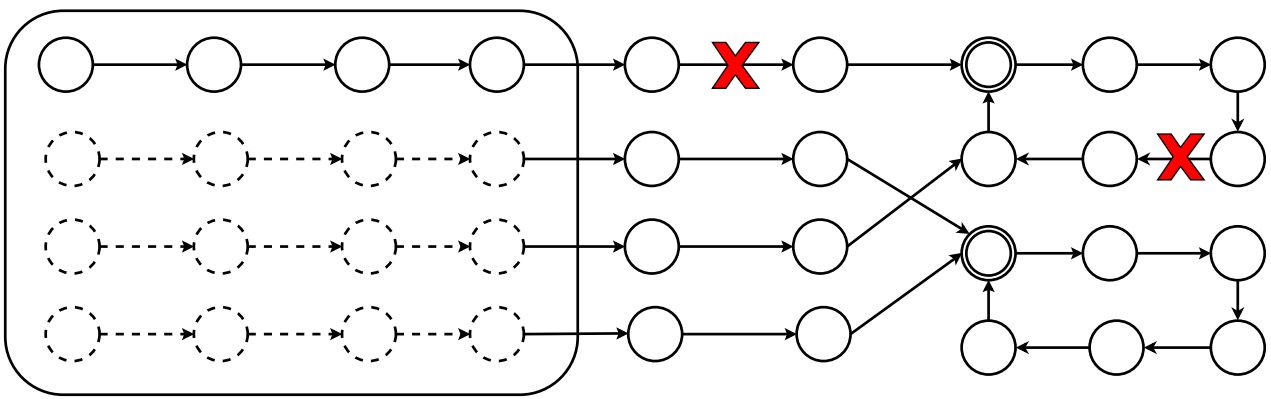
Plan Verification and Revision

- Step 3. validate the current plan
  - validity $\Longleftrightarrow$ invalid transitions
  - safety $\Longleftrightarrow$ unsafe transitions
- Step 4. local revision, instead of full synthesis



Plan Verification and Revision

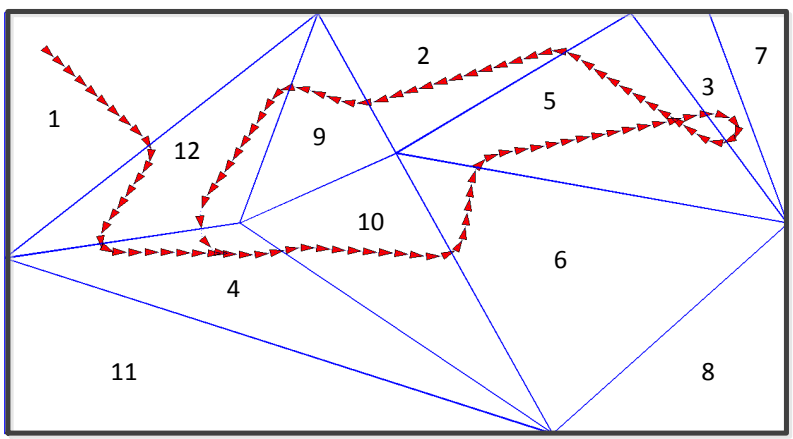
- Step 3. validate the current plan
  - validity $\Longleftrightarrow$ invalid transitions
  - safety $\Longleftrightarrow$ unsafe transitions
- Step 4. local revision, instead of full synthesis
- Low complexity, suitable for real-time applications
- Theorem. 3.5: validity and safety of the revised plan guaranteed

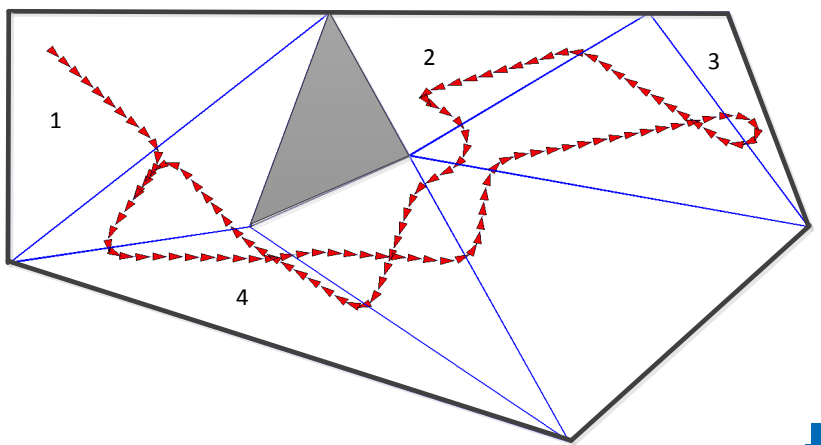


KTH Electrical Engineering

Simulation Example

- Nonholonomic ground vehicle
- Potential filed-based navigation controller[9]
- Surveillance over regions 2, 3, 4
- Detect walls and obstacles in real-time





[9] S. R. Lindemann, I. I. Hussein, S. M. LaValle. Real time feedback control for nonholonomic mobile robots with obstacles. CDC, 2006

KTH Electrical Engineering

KTH Electrical Engineering

Dependent Local Tasks

- System of N agents, $i = 1, \cdots, N$:

$$\mathcal{T}_i = (\Pi_i, \longrightarrow_i, \Pi_{i,0}, AP_i, L_i, W_i)$$

- Locally-assigned LTL specification by φ_i
- φ_i may contain requirements on other agents $j \neq i$
  - constraints, e.g., "do not be in the same room with agent 1"
  - collaborations, e.g., "move the desk together with agent 1"

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Dependent Local Tasks

- System of N agents, $i = 1, \cdots, N$:

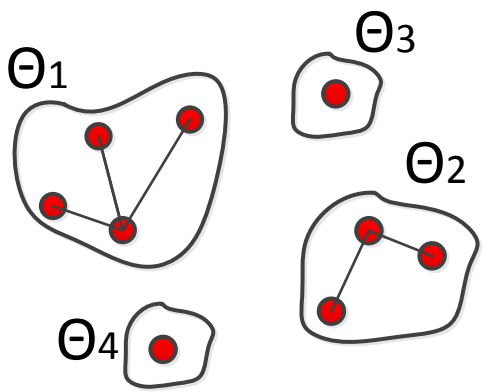
$$\mathcal{T}_i = (\Pi_i, \longrightarrow_i, \Pi_{i,0}, AP_i, L_i, W_i)$$

- Locally-assigned LTL specification by φ_i
- φ_i may contain requirements on other agents $j \neq i$
  - constraints, e.g., "do not be in the same room with agent 1"
  - collaborations, e.g., "move the desk together with agent 1"
- Difficulties:
  - the joined execution may not be mutually feasible even though the individual one is.
  - the priority of each agent plays an important role.

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Dependency Cluster

- Dependency: Agents i and j are called dependent if:



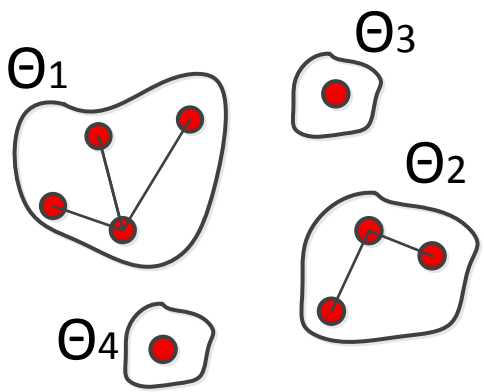
(1) agent i depends on agent j if $AP_{\varphi_i} \wedge AP_j \neq \emptyset$, or

(2) agent j depends on agent i if $AP_{\varphi_j} \wedge AP_i \neq \emptyset$.

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Dependency Cluster

- Dependency: Agents i and j are called dependent if:



(1) agent i depends on agent j if $AP_{\varphi_i} \wedge AP_j \neq \emptyset$, or

(2) agent j depends on agent i if $AP_{\varphi_j} \wedge AP_i \neq \emptyset$.

- Dependency graph: $G_d = (V, E)$, $V = 1, 2 \cdots, N$ and $E \subseteq V \times V$ is the dependency relation
- Dependency cluster: $\Theta \subseteq V$, $\forall i, j \in \Theta$ there is a path from i to j in G_d.

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Mutual Infeasible

Within one cluster $\Theta = \{1, 2, \cdots, M\}$

- The composed FTS $\mathcal{T}_\Theta = \mathcal{T}_1 \times \cdots \mathcal{T}_M$ is:

$$\mathcal{T}_\Theta = (\Pi_\Theta, \longrightarrow_\Theta, \Pi_{\Theta,0}, AP_\Theta, L_\Theta, W_\Theta)$$

- The mutual specification is

$$\varphi_\Theta = \varphi_1 \wedge \varphi_2 \cdots \wedge \varphi_M$$

- Mutually infeasible if φ_Θ is infeasible over $\mathcal{T}_\Theta$.

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Mutual Infeasible

Within one cluster $\Theta = \{1, 2, \cdots, M\}$

- The composed FTS $\mathcal{T}_\Theta = \mathcal{T}_1 \times \cdots \mathcal{T}_M$ is:

$$\mathcal{T}_\Theta = (\Pi_\Theta, \longrightarrow_\Theta, \Pi_{\Theta,0}, AP_\Theta, L_\Theta, W_\Theta)$$

- The mutual specification is

$$\varphi_\Theta = \varphi_1 \wedge \varphi_2 \cdots \wedge \varphi_M$$

- Mutually infeasible if φ_Θ is infeasible over $\mathcal{T}_\Theta$.
- The relaxed intersection of $\mathcal{A}_{\varphi_i}$ of $\{\varphi_i, i \in \Theta\}$:

$$\tilde{\mathcal{A}}_{\varphi_\Theta} = (Q, 2^{AP_{\varphi_\Theta}}, \delta, Q_0, \mathcal{F}),$$

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Solution

- The relaxed product automaton

$$\mathcal{A}_{r,\Theta} = \mathcal{T}_\Theta \times \tilde{\mathcal{A}}_{\varphi_\Theta} = (Q', \delta', Q'_0, \mathcal{F}', W_r)$$

where $Q' = \Pi_\Theta \times Q$, $Q'_0 = \Pi_{\Theta,0} \times Q_0$, $\mathcal{F}' = \Pi_\Theta \times \mathcal{F}$

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Solution

- The relaxed product automaton

$$\mathcal{A}_{r,\Theta} = \mathcal{T}_\Theta \times \tilde{\mathcal{A}}_{\varphi_\Theta} = (Q', \delta', Q'_0, \mathcal{F}', W_r)$$

where $Q' = \Pi_\Theta \times Q$, $Q'_0 = \Pi_{\Theta,0} \times Q_0$, $\mathcal{F}' = \Pi_\Theta \times \mathcal{F}$

- $\delta' \subseteq Q' \times Q'$. $(\langle \pi_\Theta, q_a \rangle, \langle \pi'_\Theta, q_b \rangle) \in \delta'$ iff $(\pi_\Theta, \pi'_\Theta) \in \longrightarrow_\Theta$ and $(q_a, q_b) \in \delta$.
- $W_r : \delta' \to \mathbb{R}^+$ is the weight function:

$$W_r(\langle \pi_\Theta, q_1, \cdots, q_M, t \rangle, \langle \pi'_\Theta, q'_1, \cdots, q'_M, t' \rangle)$$
$$= W_\Theta(\pi_\Theta, \pi'_\Theta) + \alpha \sum_{i=1}^{M} \beta_i \, \texttt{Dist}(L_\Theta(\pi_\Theta), \chi_i(q_i, q'_i))$$

- α: penalty on violating φ_Θ
- β_i: priority of agent i' task

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

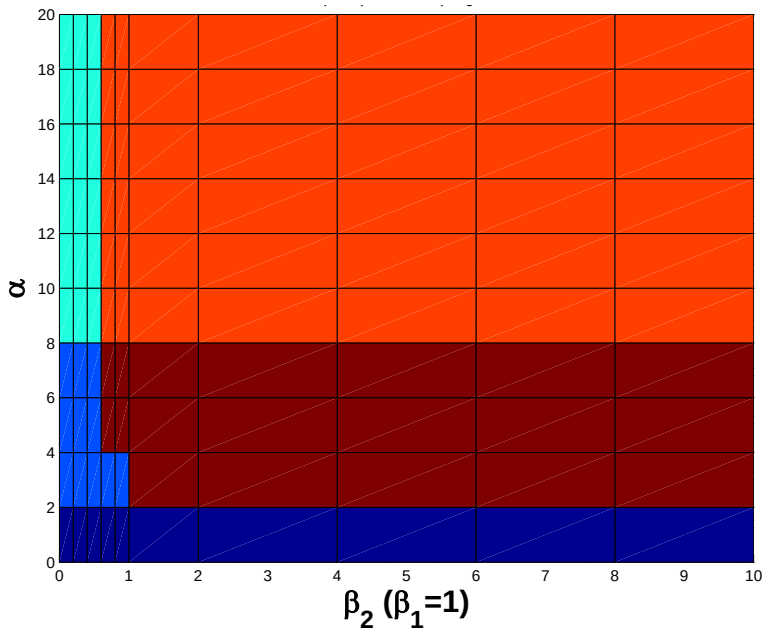
Results

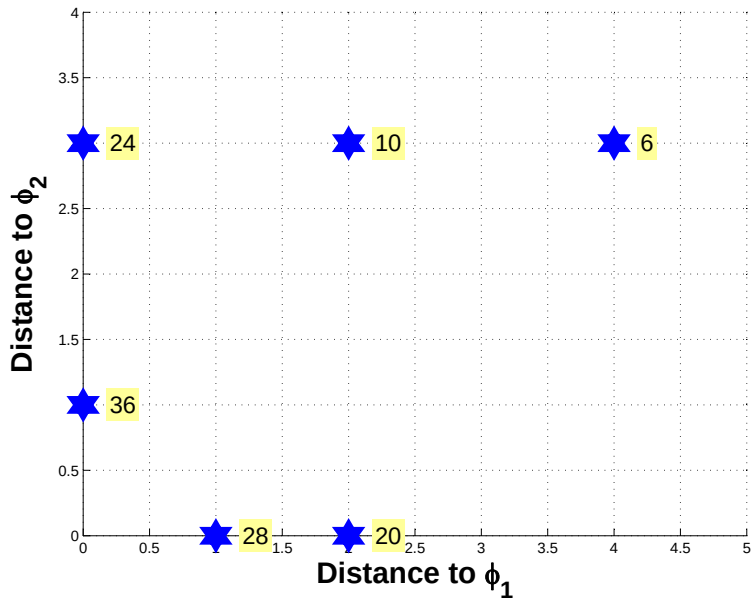
- Synthesize the balanced accepting run of $\mathcal{A}_{r,\Theta}$
- Projection onto $\mathcal{T}_i$ as the individual plan, $i \in \Theta$

KTH Electrical Engineering

Results

- Synthesize the balanced accepting run of $\mathcal{A}_{r,\Theta}$
- Projection onto $\mathcal{T}_i$ as the individual plan, $i \in \Theta$
- Change α and β
- Example





KTH Electrical Engineering

Independent Local Tasks

- System of N agents coexisting within a partially-known workspace:

$$\mathcal{T}_i^t = (\Pi_i, \longrightarrow_i^t, \Pi_{i,0}, AP_i, L_i^t, W_i^t)$$

- Locally-assigned task specification and independent

$$\varphi_i = \varphi_i^{\text{soft}} \wedge \varphi_i^{\text{hard}}$$

KTH Electrical Engineering

Independent Local Tasks

- System of N agents coexisting within a partially-known workspace:

$$\mathcal{T}_i^t = (\Pi_i, \longrightarrow_i^t, \Pi_{i,0}, AP_i, L_i^t, W_i^t)$$

- Locally-assigned task specification and independent

$$\varphi_i = \varphi_i^{\text{soft}} \wedge \varphi_i^{\text{hard}}$$

- Motivation:
  - agents located at various locations within the workspace
  - observe up-to-date information
  - beneficial to communicate

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Knowledge Update and Transfer

- Knowledge update by
  - own sensing ability $\mathbf{Sense}_k^t = \{(\pi, S, S_\neg),\ E,\ E_\neg\}$
  - communication with others

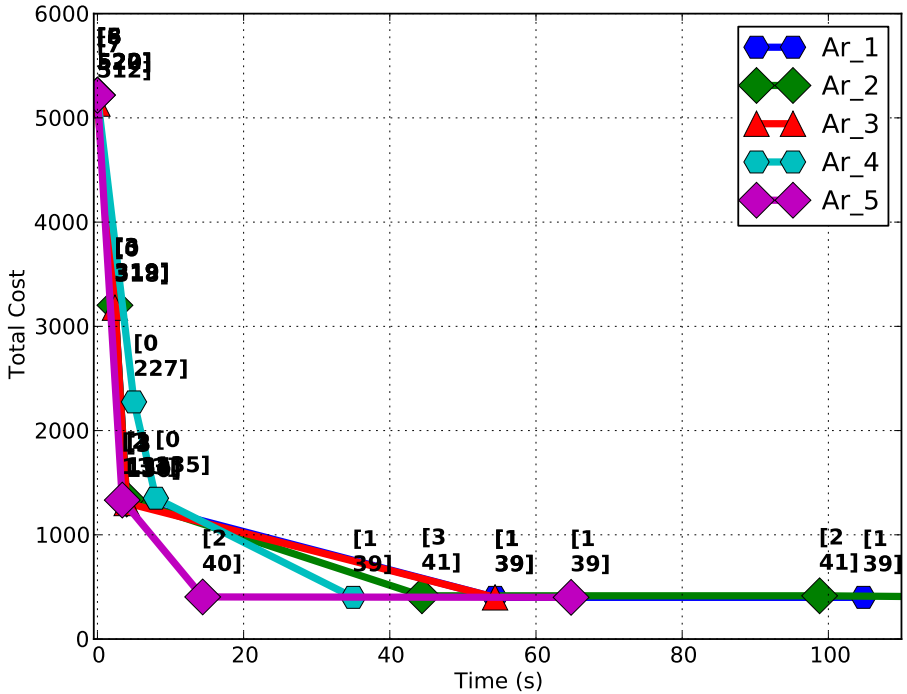
Knowledge Update and Transfer

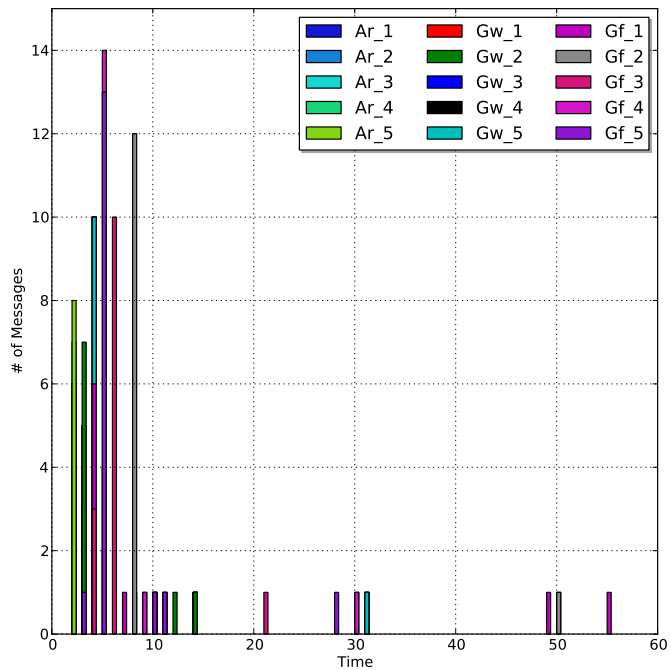
- Knowledge update by
  - own sensing ability $\mathbf{Sense}_k^t = \{(\pi, S, S_\neg),\ E,\ E_\neg\}$
  - communication with others
- Communication network: $\mathcal{N}_k \in \mathcal{N}$ (static or dynamic)
- Transfer knowledge:
  - request once: $\mathbf{Request}_{k,g}^t = (k,\ \varphi_k|_{AP_k})$
  - event-based reply: $\mathbf{Reply}_{h,k}^t = (\pi,\ S',\ S'_\neg)$, where $S' = S \cap (\varphi_h|_{AP_h})$ and $S'_\neg = S_\neg \cap (\varphi_h|_{AP_h})$.
- Update $\mathcal{T}_k^t$ based on $\mathbf{Sense}_k^t$ and $\mathbf{Reply}_{g,k}^t$
- Validate and revise the current plan

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Results

- Full synthesis or local revision
- Event-based trigger to re-synthesize the plan

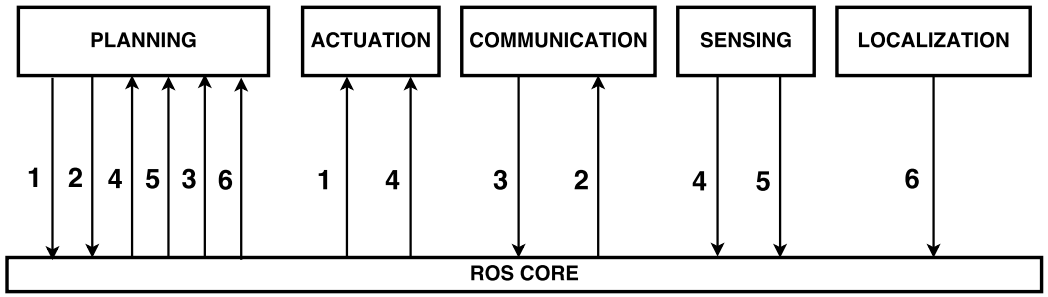




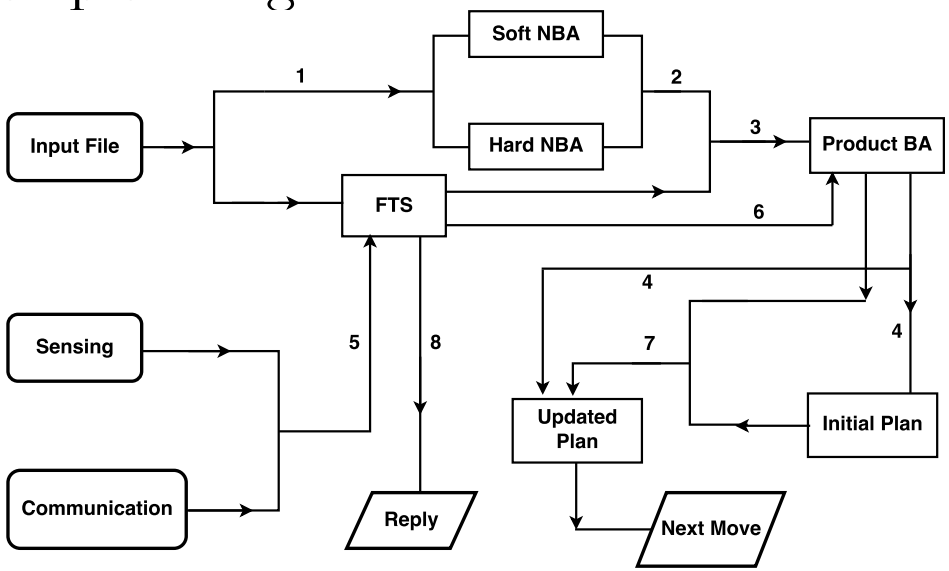
KTH Electrical Engineering

Software Implementation

- Robot operating system(ROS)-based
- ROS core + ROS nodes



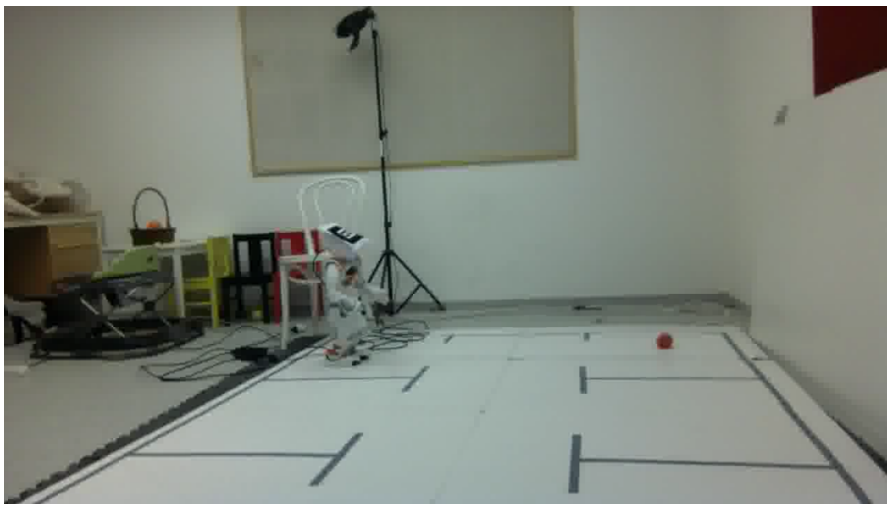
- ROS node for planning



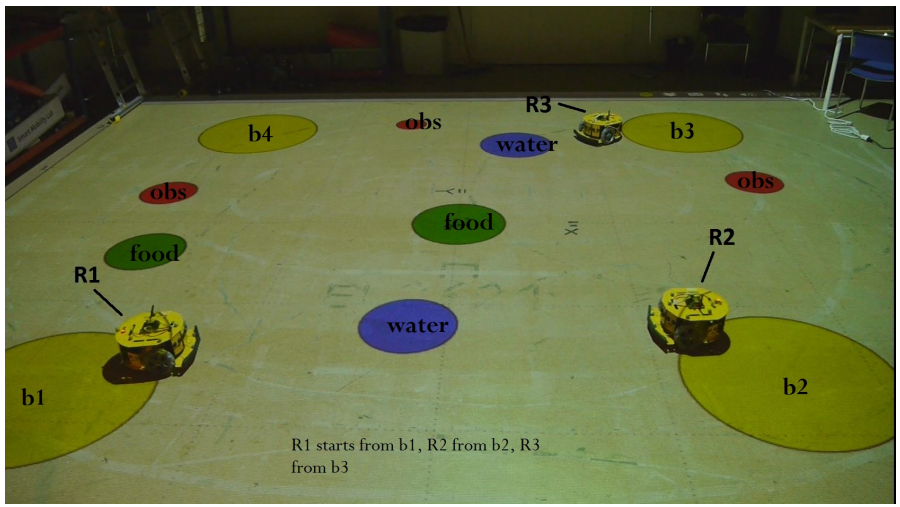
KTH Electrical Engineering

Experiments

- NAO humanoid
- MAS Lab @CVAP



- NEXUS ground vehicle
- Smart Mobility Lab @ACL



KTH Electrical Engineering

KTH Electrical Engineering

Summary

- Motion and task planning
  - Discrete motion and task plan with minimal cost
  - Hybrid control strategy

KTH Electrical Engineering

Summary

- Motion and task planning
  - Discrete motion and task plan with minimal cost
  - Hybrid control strategy
- Reconfiguration and real-time adaptation
  - Potentially infeasible task
  - Soft and hard specifications
  - Partially-known workspace
  - Motion and action planning

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Summary

- Motion and task planning
  - Discrete motion and task plan with minimal cost
  - Hybrid control strategy
- Reconfiguration and real-time adaptation
  - Potentially infeasible task
  - Soft and hard specifications
  - Partially-known workspace
  - Motion and action planning
- Multi-agent systems with local tasks
  - Dependent tasks
  - Independent tasks
  - Software implementation

KTH VETENSKAP OCH KONST
KTH Electrical Engineering

Future Work

- Automated abstraction
- Natural language to LTL, graphic interface
- Trade-off between computational complexity and optimality
- Robustness and fault tolerance (both motion and action)
- Continuous constraints, coupled dynamics

KTH Electrical Engineering

Thank you!

KTH Electrical Engineering