A Graph-Based Search Approach to Planning and Learning

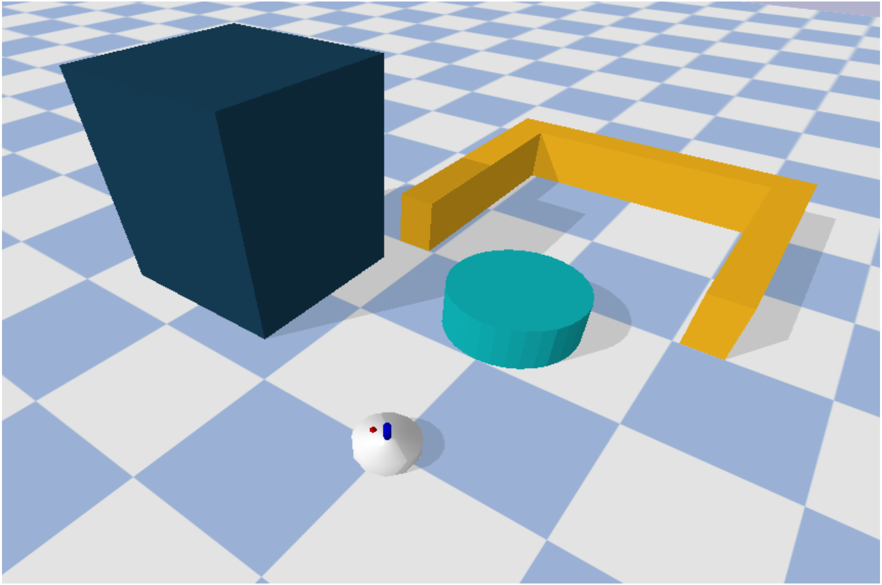
G.S. Groote

Supervisors: C. Pezzato
M. Wisse
C. Smith

Delft University of Technology, The Netherlands

June 28, 2023

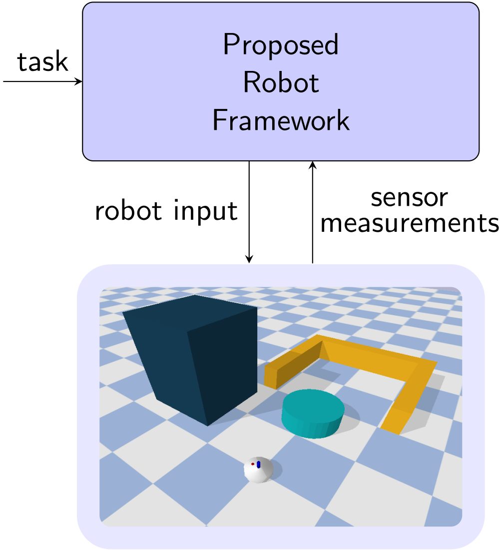
Intro: Robot Environment



Intro: Thesis Goal

- Learning System Models
- Navigation Among Movable Objects (NAMO)
- Nonprehensile Pushing

Intro: Overview Proposed Method



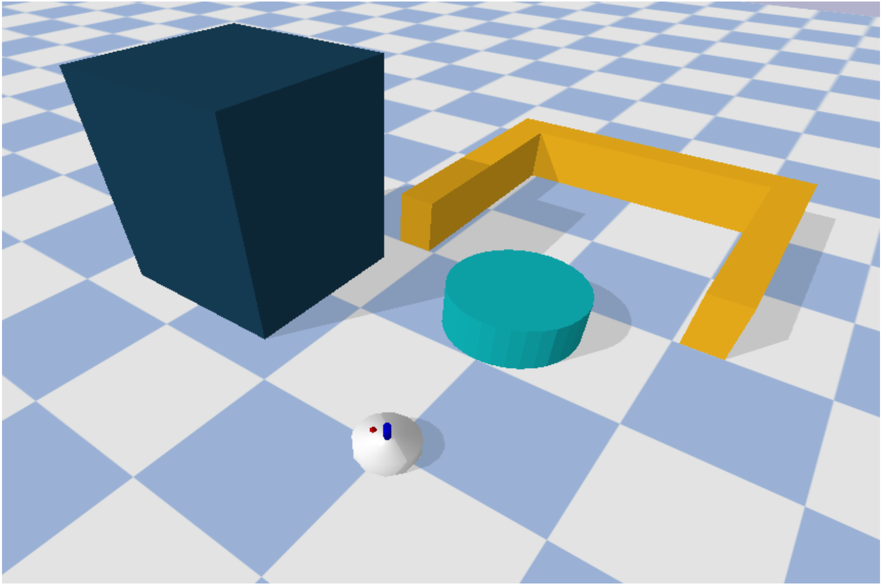
Intro: Research Question

How do learned objects' system models improve global task planning for a robot with nonprehensile push manipulation abilities over time?

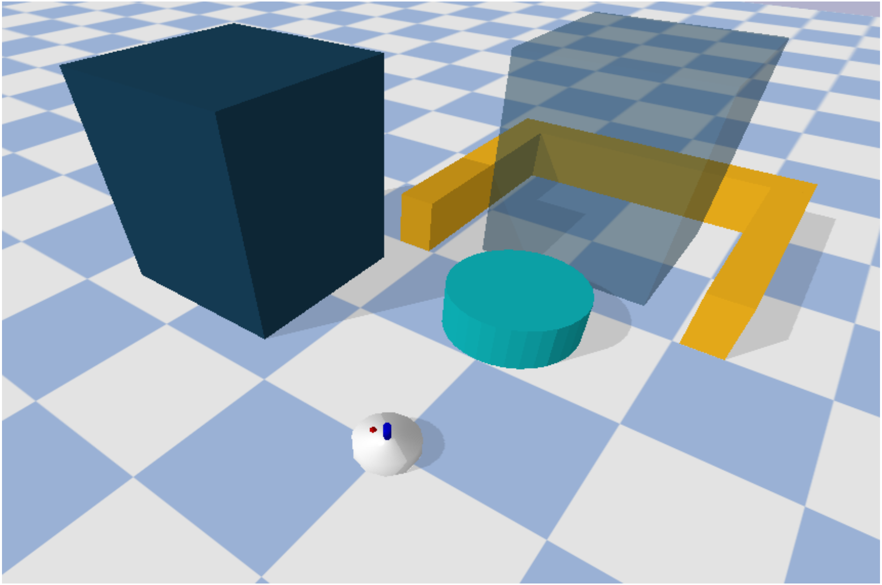
Research Subquestions:

1. How to combine learning and planning for push and drive applications?
2. How does the proposed framework compare against the state-of-the-art?

Intro: Task Specification



Intro: Task Specification



Assumptions

1. **Closed-World Assumption:** Objects are manipulated, directly or indirectly only by the robot. Objects cannot be manipulated by influences from outside the environment.

Assumptions

1. **Closed-World Assumption:** Objects are manipulated, directly or indirectly only by the robot. Objects cannot be manipulated by influences from outside the environment.
2. **Perfect Object Sensor Assumption:** the robot has full access to the poses and geometry of all objects in the environment at all times.

Assumptions

1. **Closed-World Assumption:** Objects are manipulated, directly or indirectly only by the robot. Objects cannot be manipulated by influences from outside the environment.
2. **Perfect Object Sensor Assumption:** the robot has full access to the poses and geometry of all objects in the environment at all times.
3. **Tasks are Commutative Assumption:** Tasks consist of multiple objects with specified target positions. The order in which objects are pushed toward their target position is commutative.

Assumptions

1. **Closed-World Assumption:** Objects are manipulated, directly or indirectly only by the robot. Objects cannot be manipulated by influences from outside the environment.
2. **Perfect Object Sensor Assumption:** the robot has full access to the poses and geometry of all objects in the environment at all times.
3. **Tasks are Commutative Assumption:** Tasks consist of multiple objects with specified target positions. The order in which objects are pushed toward their target position is commutative.
4. **Objects do not tip over Assumption:** Movable objects slide if pushed.

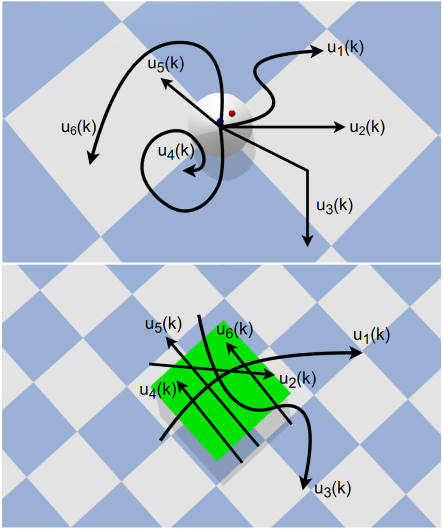
Assumptions

1. **Closed-World Assumption:** Objects are manipulated, directly or indirectly only by the robot. Objects cannot be manipulated by influences from outside the environment.
2. **Perfect Object Sensor Assumption:** the robot has full access to the poses and geometry of all objects in the environment at all times.
3. **Tasks are Commutative Assumption:** Tasks consist of multiple objects with specified target positions. The order in which objects are pushed toward their target position is commutative.
4. **Objects do not tip over Assumption:** Movable objects slide if pushed.

State-of-The-Art

		NAMO		Specify object target poses	
Author	Learns object dynamics	prehensile	nonprehesile	prehensile	nonprehesile
Ellis et al.	✓	✗	✓	✗	✗
Sabbagh Novin et al.	✓	✓	✗	✓	✗
Scholz et al.	✓	✓	✗	✗	✗
Vega-Brown and Roy	✗	✓	✗	✓	✗
Wang et al.	✓	✗	✓	✗	✗
Groote	✓	✗	✓	✗	✓

Required Background: System identification



Required Background: System identification

gif that shows going toward pushing pose, then push

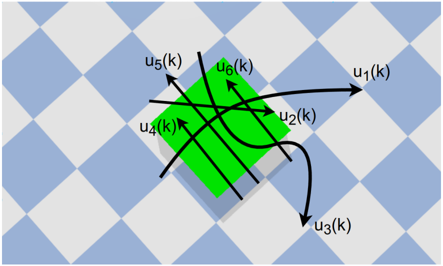
Required Background: System identification

$$x_{lti-drive-model}(k+1) = \begin{bmatrix} x_{robot}(k+1) \\ y_{robot}(k+1) \end{bmatrix} = \begin{bmatrix} x_{robot}(k) + DTu_x(k) \\ y_{robot}(k) + DTu_y(k) \end{bmatrix}$$

$$x_{lti-push-model}(k+1) = \begin{bmatrix} x_{robot}(k+1) \\ y_{robot}(k+1) \\ x_{obj}(k+1) \\ y_{obj}(k+1) \end{bmatrix} = \begin{bmatrix} x_{robot}(k+1) + DTu_x(k) \\ y_{robot}(k+1) + DTu_y(k) \\ x_{obj}(k+1) + \frac{1}{2}DTu_x(k) \\ y_{obj}(k+1) + \frac{1}{2}DTu_y(k) \end{bmatrix}$$

Required Background: Control Methods

Both MPC and MPPI use a system model and an objective function. The main difference lies in MPC uses a mathematical solver to obtain the best input, whilst MPPI samples random rollouts into the future to then take the weighted average the results into the lowest cost function



Required Background: Planning a Path

image that has start and target for the robot in an interesting environment

Required Background: Path Estimation

1. Path Estimation

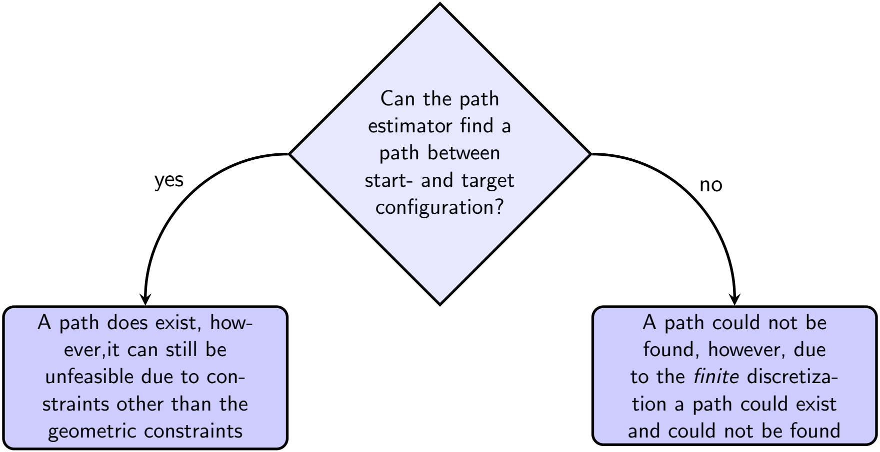
Required Background: Path Estimation

1. Path Estimation
2. Path Planning

Required Background: Path Estimation

with the image above, that interesting environment, create an occupancy grid

Required Background: Path Estimation



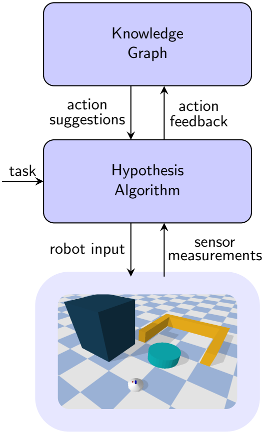
Required Background: Planning

recap for RRT*

Required Background: Planning

recap for RRT* image of environment and make a video please

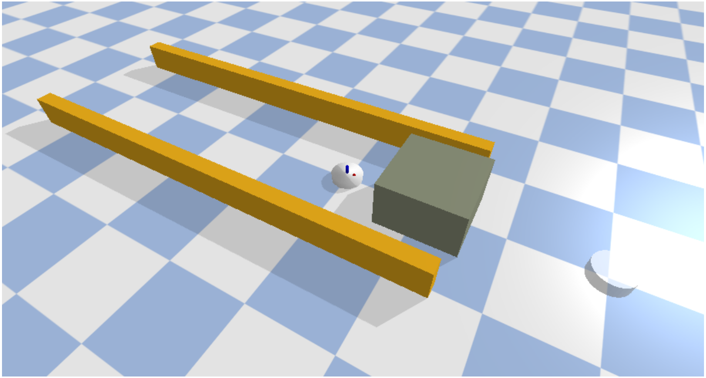
Overview Proposed Method



Required Background: Extension to 4 subspaces

Add the unknown and movable space redefine the cost in RRT*

Required Background: Extension to 4 subspaces

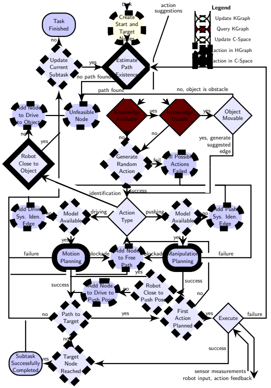


(a) First subfigure

Hypothesis Graph

make hgraph definition

make example hgraph

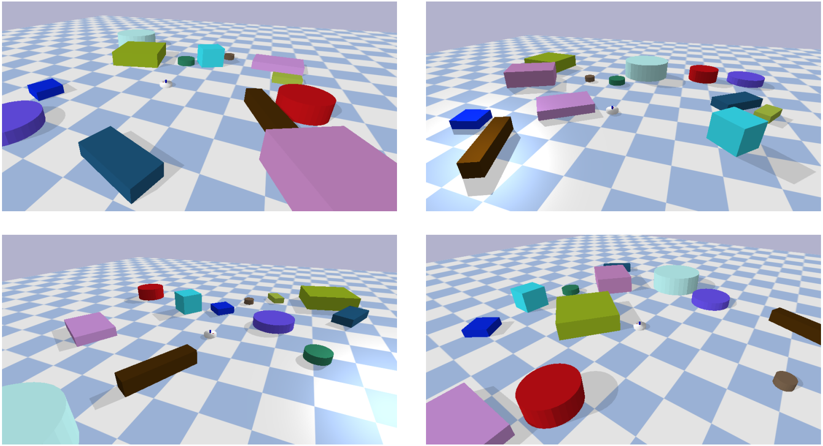


task
action suggestions
Legend
Update KGraph
Query KGraph
Update C-Space
Task Finished
Create Start and Target Nodes
Estimate Path Existence
Update Current Subtask
no
yes
no path found
path found
no, object is obstacle
Add Node to Drive to Object
Unfeasible Node
Knowledge Available
Knowledge Usable
Object Movable
yes
no
Robot Close to Object
Generate Random Action
All Possible Actions Failed
fail
yes, generate suggested edge
identification
success
Add Drive Sys. Iden. Edge
Model Available
driving
Action Type
pushing
Model Available
Add Push Sys. Iden. Edge
failure
Motion Planning
blockade
Add Node to Free Path
blockade
Manipulation Planning
failure
success
Add Node to Drive to Push Pose
Robot Close to Push Pose
success
Path to Target
First Action Planned
Execute
failure
Subtask Successfully Completed
Target Node Reached
success
sensor measurements
robot input, action feedback

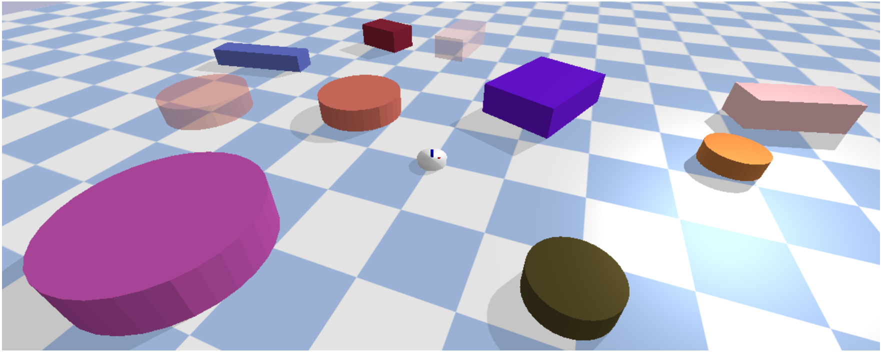
Knowledge Graph

kgraph definition and example

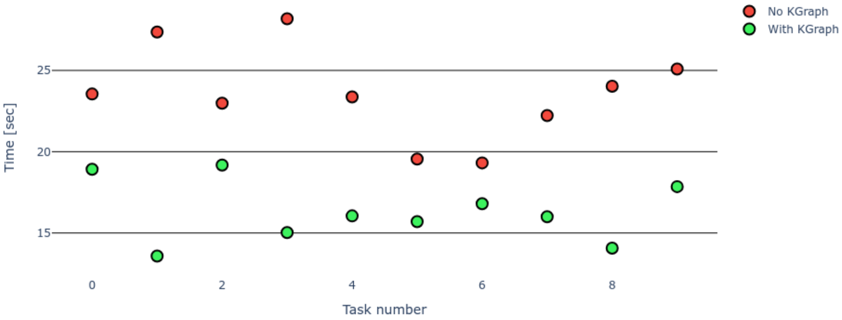
Results: Randomisation Drive Task



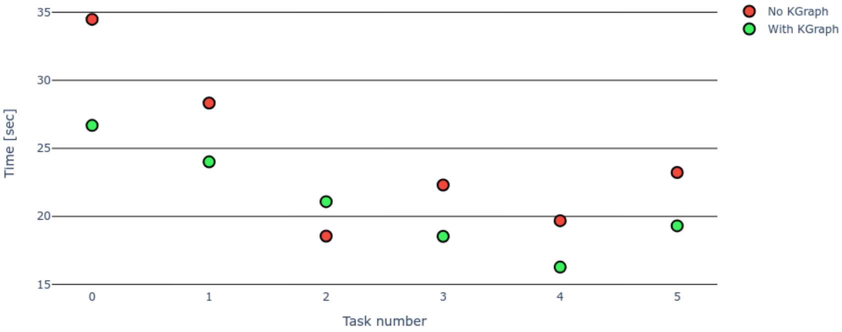
Results: Randomisation Push Task



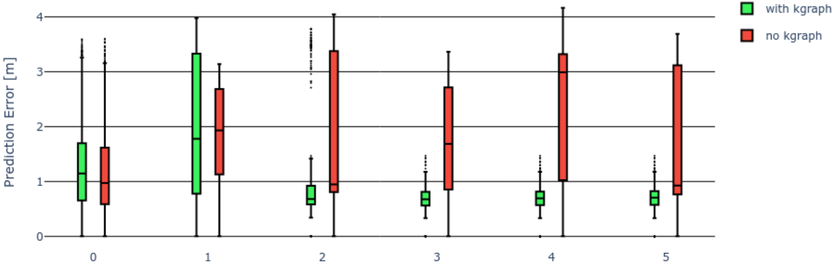
Results: Randomisation Drive Task Execution Times



Results: Randomisation Push Task Execution Times



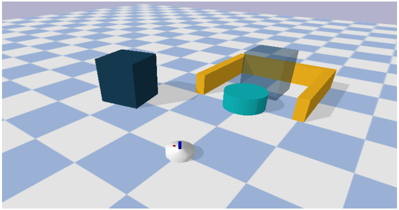
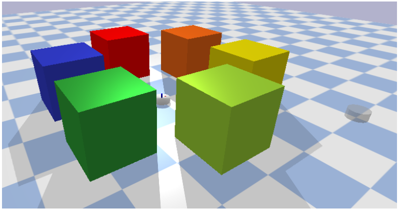
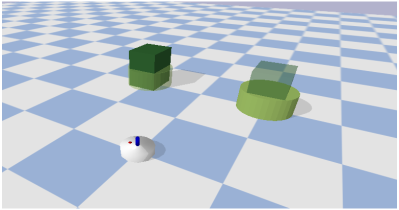
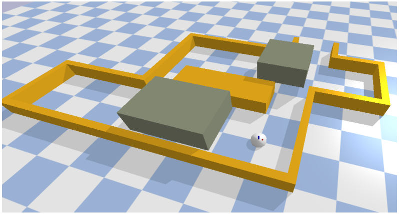
Results: Randomisation Push Task Prediction Error



Results

Author	Learning	NAMO	Object to Target	Manipulation
Ellis et al.	✓	✓	✗	pushing
Sabbagh	✓	✗	✓	grasp-push
Novin et al.				grasp-pull
Scholz et al.	✓	✓	✗	graph-push
				grasp-pull
Vega-Brown et al.	✗	✓	✓	gripping
Wang et al.	✓	✓	✗	pushing
Groote	✗/✓	✓	✓	pushing

More convincing environments

4 Week Planning

Good Scenario

- 50 % Report
- 50 % Presentation

4 Week Planning

Good Scenario

- 50 % Report
- 50 % Presentation

Better Scenario

- 40 % Report
- 40 % Presentation
- 20 % Benchmark Environments