A Robot Task Planner that Merges Symbolic and Geometric Reasoning

Elective in Artificial Intelligence

Reasoning Agents

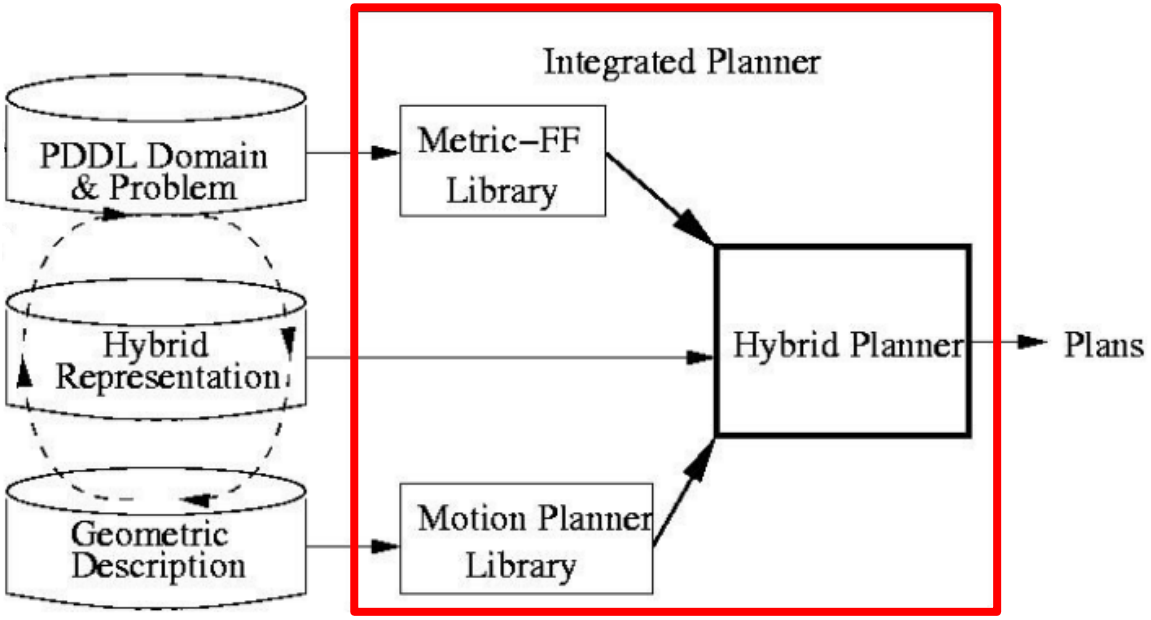
Prof. Fabio Patrizi

Leandro Maglianella - 1792507

SAPIENZA
UNIVERSITÀ DI ROMA

Paper Content

This paper is about **aSyMov**, a planner designed to address intricate robot planning problems where geometric constraints are relevant and influence the symbolic plan.

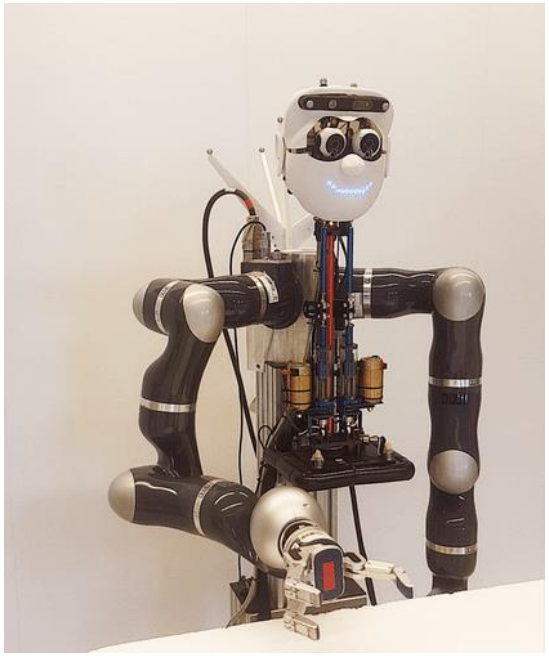


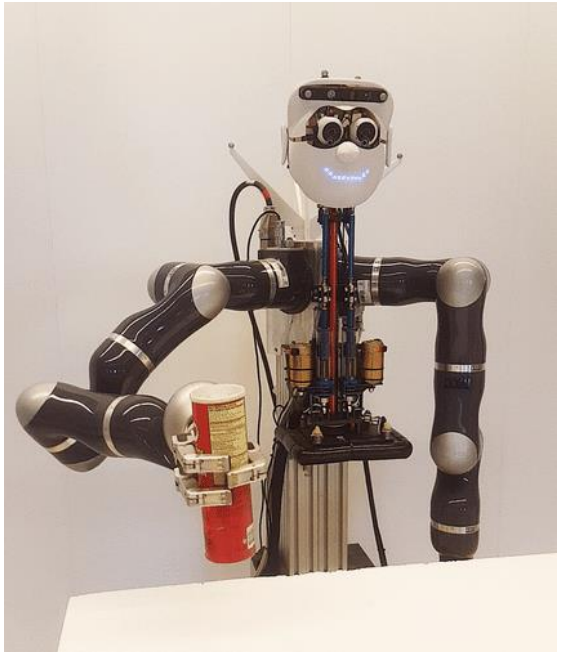
The proposed implementation of aSyMov is tested on an illustrative example.

Problem Description

Problem Description

Often planners lack of effectiveness in robotics task because there is a gap between the representation they are based on and the physical world.



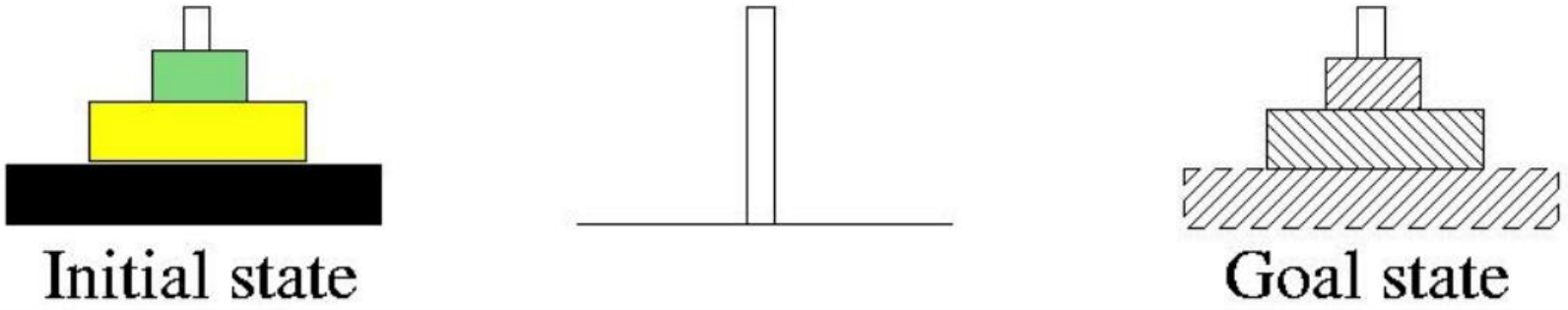


At each step of the planning process **both symbolic and geometric constraints must be considered.**

Problem Description

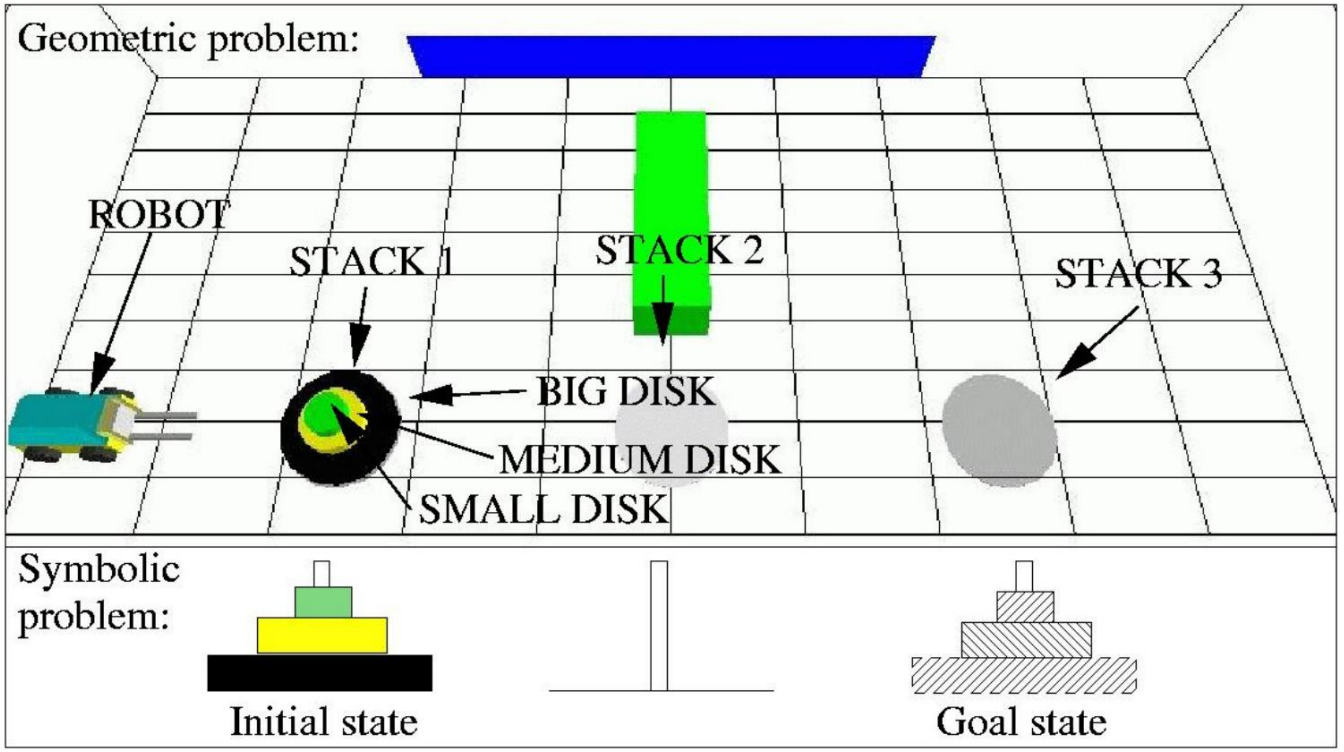
The authors create an example based on the **Hanoi Tower Problem**.

“Classical” Hanoi Tower Problem



Problem Description

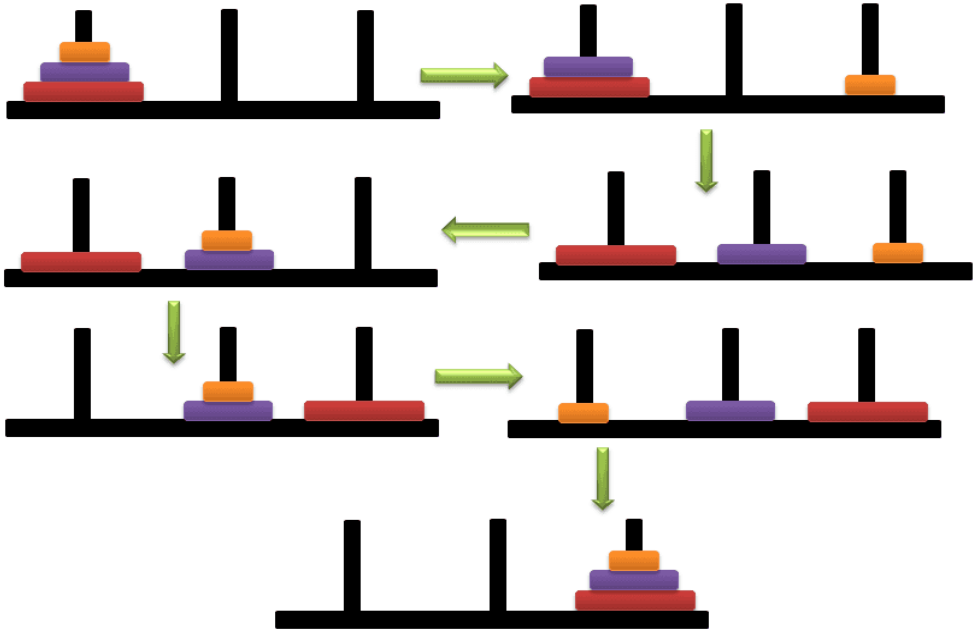
“Geometric Hanoi Tower Problem with 3 disks” (GHTP-3)



Disks are moved by a non-holonomic robotized fork-lift.

Problem Description

A good symbolic planner would be able to find the optimal solution:

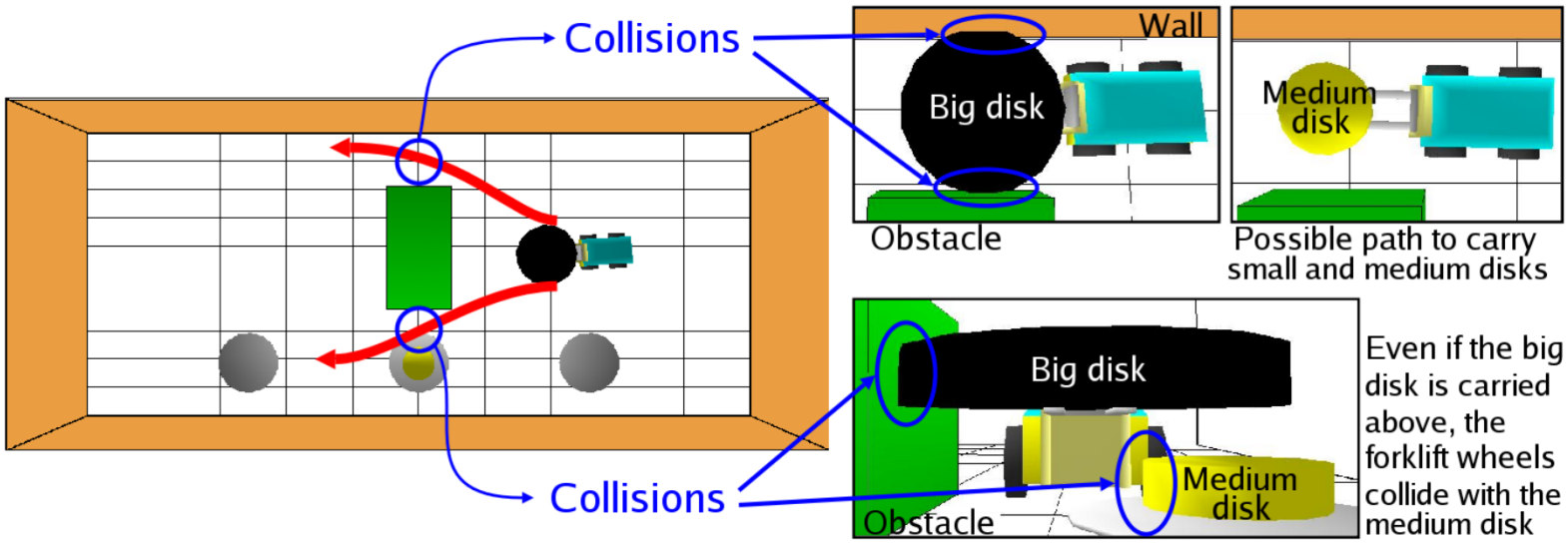


BUT this solution is actually not achievable in our case because the environment is also **geometrically constrained**.

Problem Description

All possible robot motions depend not only on the static environment but on the context (robot and objects positions).

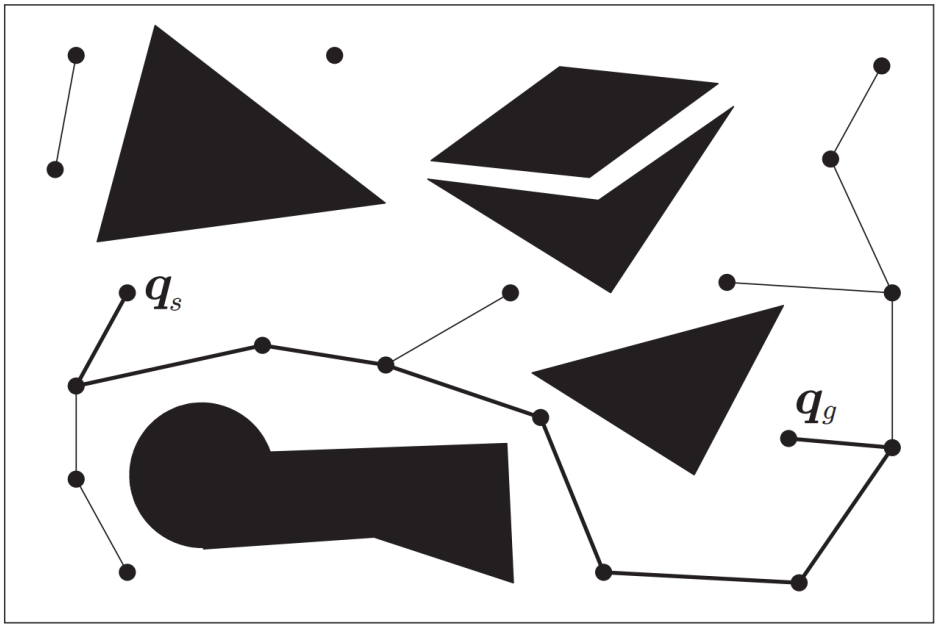
Each context may possibly create a different free-space topology and consequently **create specific constraints on robot motions**.



Implementation

Geometric Reasoning

- **Environment**: 3D-world with static and dynamic components.
- **Roadmap**: The motion planner uses a Probabilistic Roadmap method.



In the constructed graph, a node represents a collision-free configuration and the edges are valid robot motions.

Geometric Reasoning

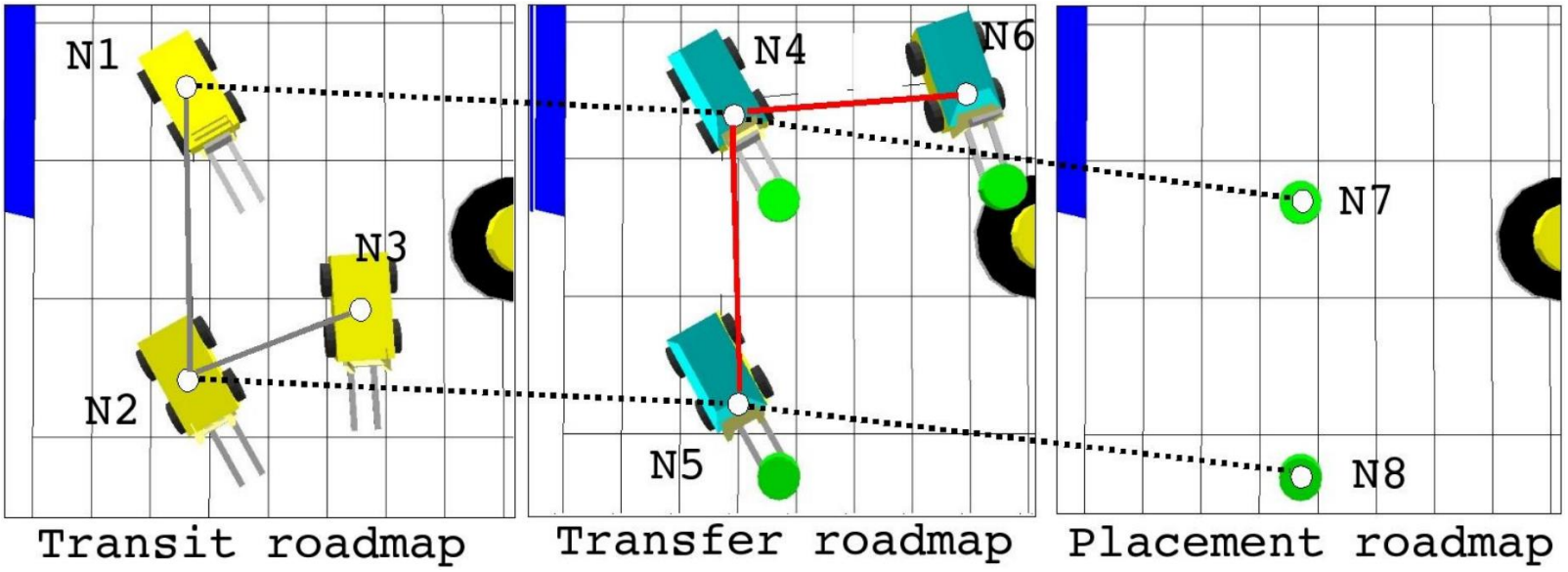
- **Transit / Transfer / Placement**:

A solution to a manipulation problem is a sequence of **transit** and **transfer** motions.

The **transit** motion of the robot between two grasping positions.

The **transfer** motion of the robot with a taken object.

The set of all stable object positions is called **placement**.



Symbolic Reasoning

A **symbolic position** → A subset of the configuration nodes of one roadmap that satisfy a given property.

P_{*robot name*}_{*roadmap name*}_{*property*}

- P_R_TI_TA_O: the configurations where a robot R can grasp an object O.

In the figure = {N1, N2}.

- P_R-O_TA_TI: the configurations where a robot R can drop an object O.

In the figure = {N4, N5}.

- P_R_TI_RELOAD_BATTERY: the configurations where a robot R can apply a RELOAD_BATTERY action.

Symbolic Reasoning

The authors chose to use PDDL 2.1 to specify the described symbolic representation.

Basic predicate set:

- (composed ?r1 ?r2 ?r3)
- (belongs-to ?p ?r ?roadmap-type)
- (has-purpose ?p ?pos-type)
- (connection ?p1 ?p2)
- (on ?r ?p)

Basic action set:

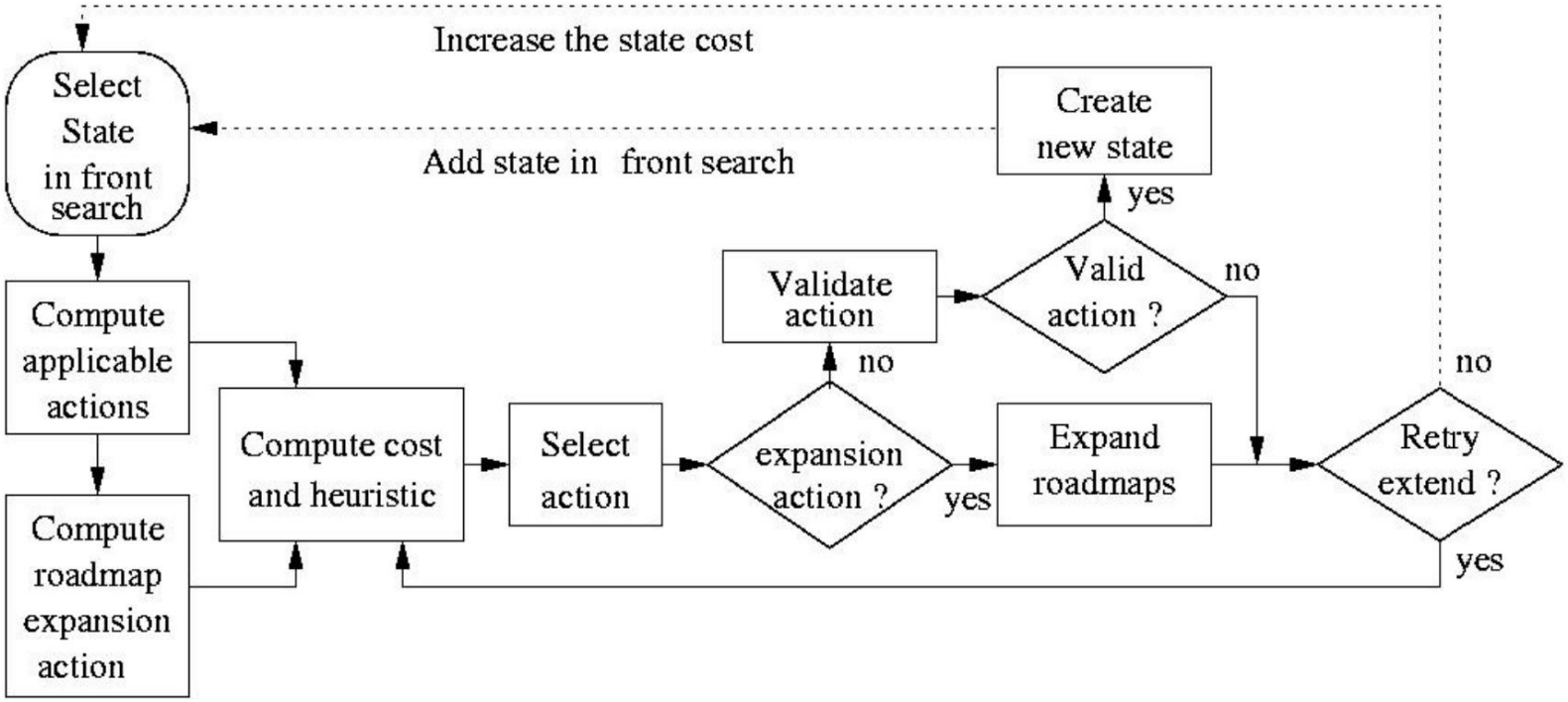
- (goto ?r ?p1 ?p2)
- (grasp ?r ?p1 ?o ?p2 ?new-r ?p3)
- (release ?old-r ?p1 ?o ?p2 ?r ?p3)
- (switch-motion ?r ?p1 ?p2)

```
(:action STACK_ON
 :parameters
     (?r_plus_disk  - robot  ?p3 - position
      ?r - robot  ?p1 - position
      ?disk1 - obj ?p2 - position
      ?stack - position_type ?disk2 - obj
      ?height ?height_plus - position_type)
 :precondition (and
      ;; relaxed representation of geometry
      (composed ?r_plus_disk ?r ?disk1)
      (on ?r_plus_disk ?p3)
      (belongs-to ?p3 ?r_plus_disk TA)
      (belongs-to ?p1 ?r TI)
      (belongs-to ?p2 ?disk1 PL)
      (connection ?p3 ?p1)
      (connection ?p3 ?p2)
      ;; constraints
      (has_purpose ?p2 ?height_plus)
      (has_purpose ?p2 ?stack)
      ;; purely symbolic part
      (bigger_than ?d2 ?d1)
      (on_top ?stack ?d2)
      (height ?stack ?height)
      (minus_height ?height_plus ?height))

 :effect (and
      ;; geometric effects
      (not (on ?r_plus_disk ?p3))
      (on ?r ?p1)
      (on ?disk1 ?p2)
      ;; symbolic effects
      (not (on_top ?stack ?disk2))
      (on_top ?stack ?disk1)
      (not (height ?stack ?height))
      (height ?stack ?height_plus))
)
```

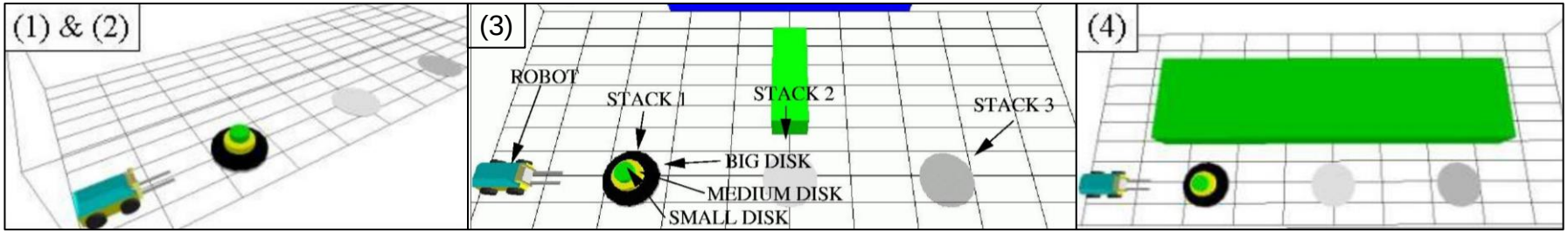
Algorithms and Strategies

Extend state algorithm



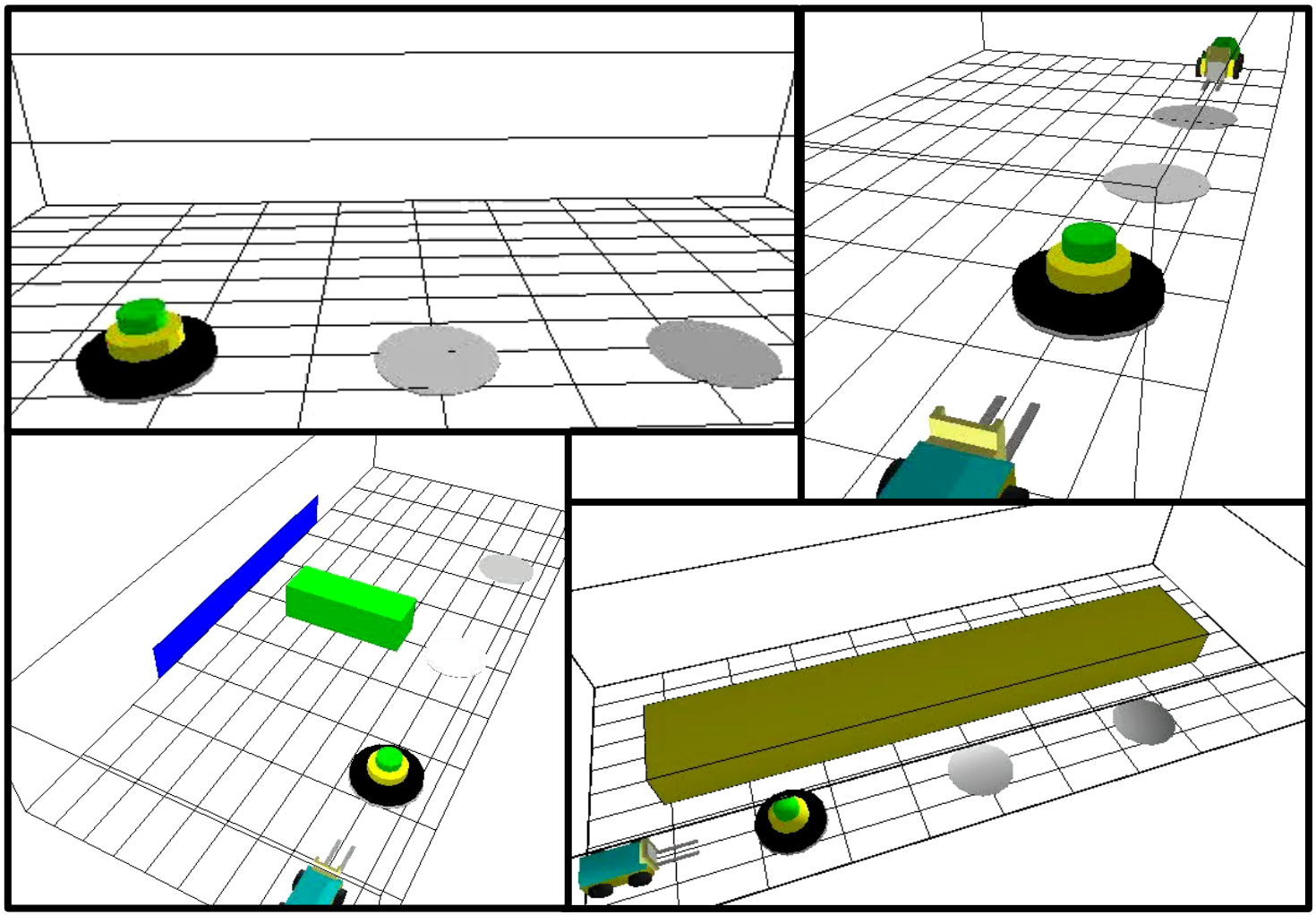
Experiments

Experimental Instances and Results



	CPU time	Nb step	Connectivities	Plan length
(1)	0.266	38.5	17.5	28
(2)	9.97	63.7	25.4	28.8
(3)	271.6	243.8	71.2	44
(4)	298.9	213.6	62.8	40.16

Experimental Instances and Results



Final Personal Comment

Final Personal Comment

- Remarkable research and results, especially being it from 2004.

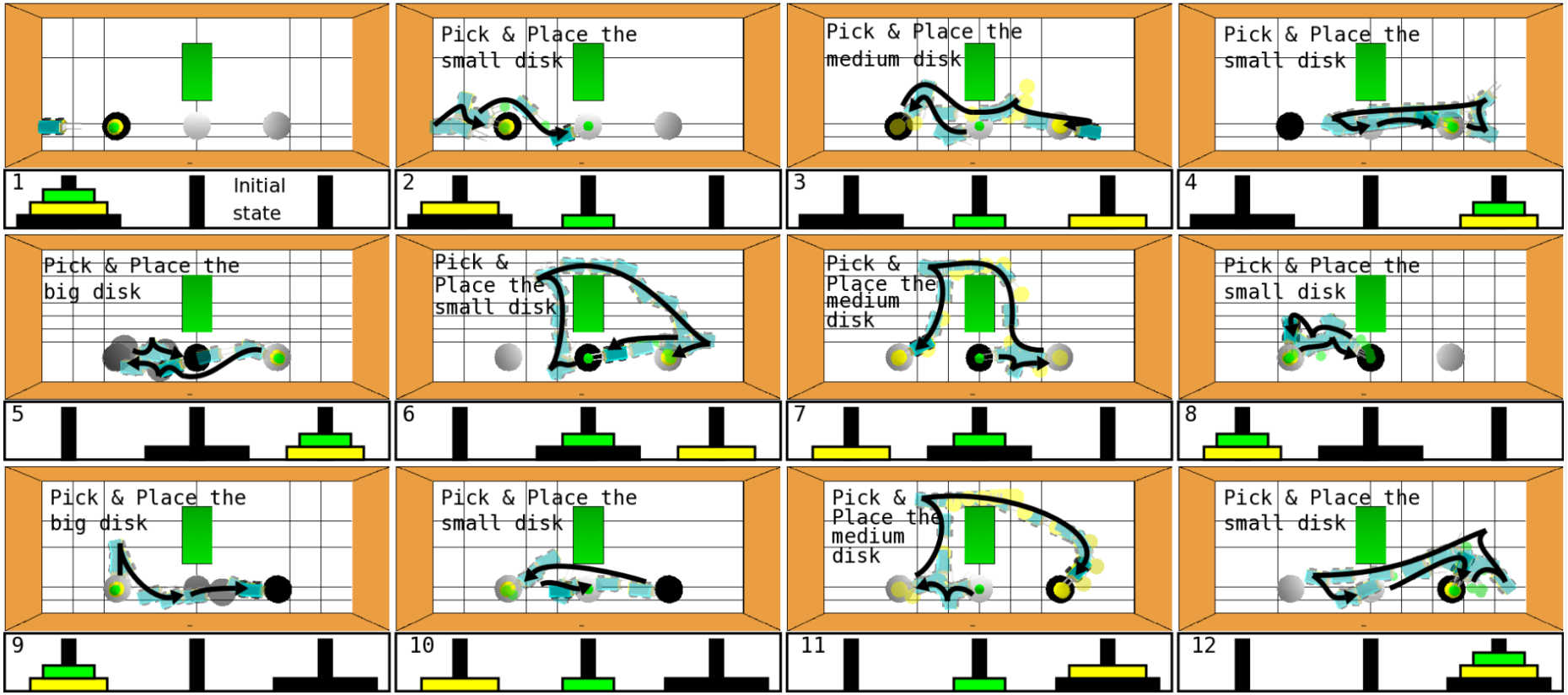
- Generated plans take into account the geometric consequences of robots' actions. $\Longrightarrow$ Plans have a better chance to be valid in the real world.

- We intend to make a project about a robot that act as a teacher, therefore not colliding with children while moving an object is a plus.

References

- **Stephane Cambon, Fabien Gravot and Rachid Alami**
  "A Robot Task Planner that Merges Symbolic and Geometric Reasoning", 2004
  Proceedings of the 16th Eureopean Conference on Artificial Intelligence, ECAI'2004

Appendix


1
Initial state
2
Pick & Place the small disk
3
Pick & Place the medium disk
4
Pick & Place the small disk
5
Pick & Place the big disk
6
Pick & Place the small disk
7
Pick & Place the medium disk
8
Pick & Place the small disk
9
Pick & Place the big disk
10
Pick & Place the small disk
11
Pick & Place the medium disk
12
Pick & Place the small disk