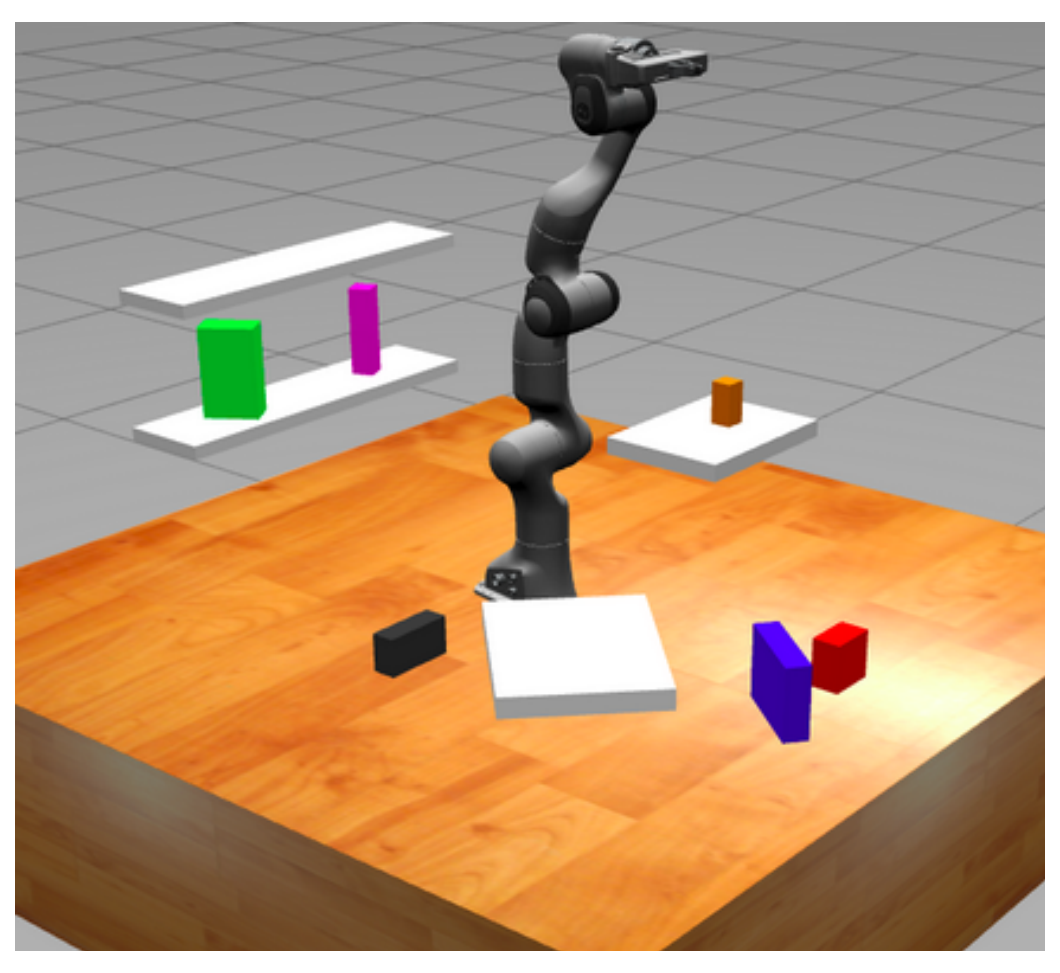
Learning to Predict Action Feasibility for Task and Motion Planning in 3D Environments

ICRA LONDON · 2023

Smail Ait Bouhsain*, Rachid Alami*, Thierry Siméon*

Background and Motivation

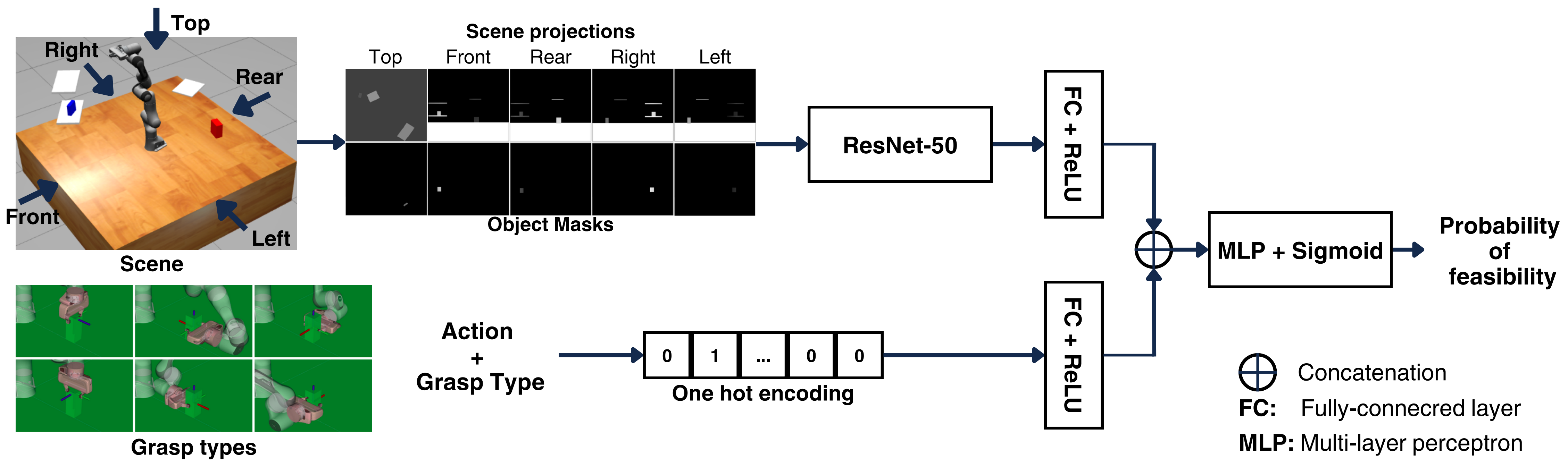
- Due to the **high combinatorial complexity of TAMP**, geometric planning creates a **bottleneck** to the planning process, requiring **numerous calls to motion planners** to verify the feasibility of actions and plan their trajectories.
- Previous works [1,2,3] propose **learning approaches** that aim at **predicting the feasibility of actions** in order to prioritize feasible actions during planning, and reduce the number of calls to the motion planner.
- Our paper aims at **generalizing** these methods (especially [2]) **to 3D environments**, with multiple support surfaces and multiple objects.



Typical 3D scene our method is able to deal with.

Action Feasibility Prediction Network (AFP-Net)

- This paper proposes a deep convolutional neural network which takes as input:
  - **The scene** represented as **5 depth images** corresponding to **different views of the scene**, grouped as a single 5-channel image. This representation is able to encode scenes with **different numbers of objects** while keeping its **dimensionality fixed**.
  - **The object of interest** represented as **masks over the scene views**, showing the object at the pose it should be picked or placed at.
  - **The action** to be tested are Pick or Place actions with one of **6 discrete types of grasp**. They correspond to the side from which the object is approached **in the perspective of the robot.**
- The model outputs the **probability of feasibility** of performing the tested action on the object given the current scene configuration.



Architecture of the proposed neural network AFP-Net

Feasibility-Informed TAMP Algorithm

- The TAMP algorithm is a **best-first tree search**, where a node represents the state of the environment and the robot, and edges represent actions.
- Each time a child is added to the search tree, **AFP-Net** is queried to compute its probability of feasibility.
- The probability of feasibility produced by the neural network is used to **sort the set of nodes to expand.**
- During the search, the task planner **prioritizes actions with a high probability of feasibility.**

```
Algorithm 2 findChildren
Input: $S, E$
1: $children \leftarrow \emptyset$
2: for all $a$ applicable at $S$ do
3:     $child \leftarrow nextState(S, a)$
4:     $child.cost \leftarrow computeCost(child)$
5:     $child.p \leftarrow predictFeasibility(child)$
6:     $children \leftarrow children \cup child$
7: end for
8: return $children$
```

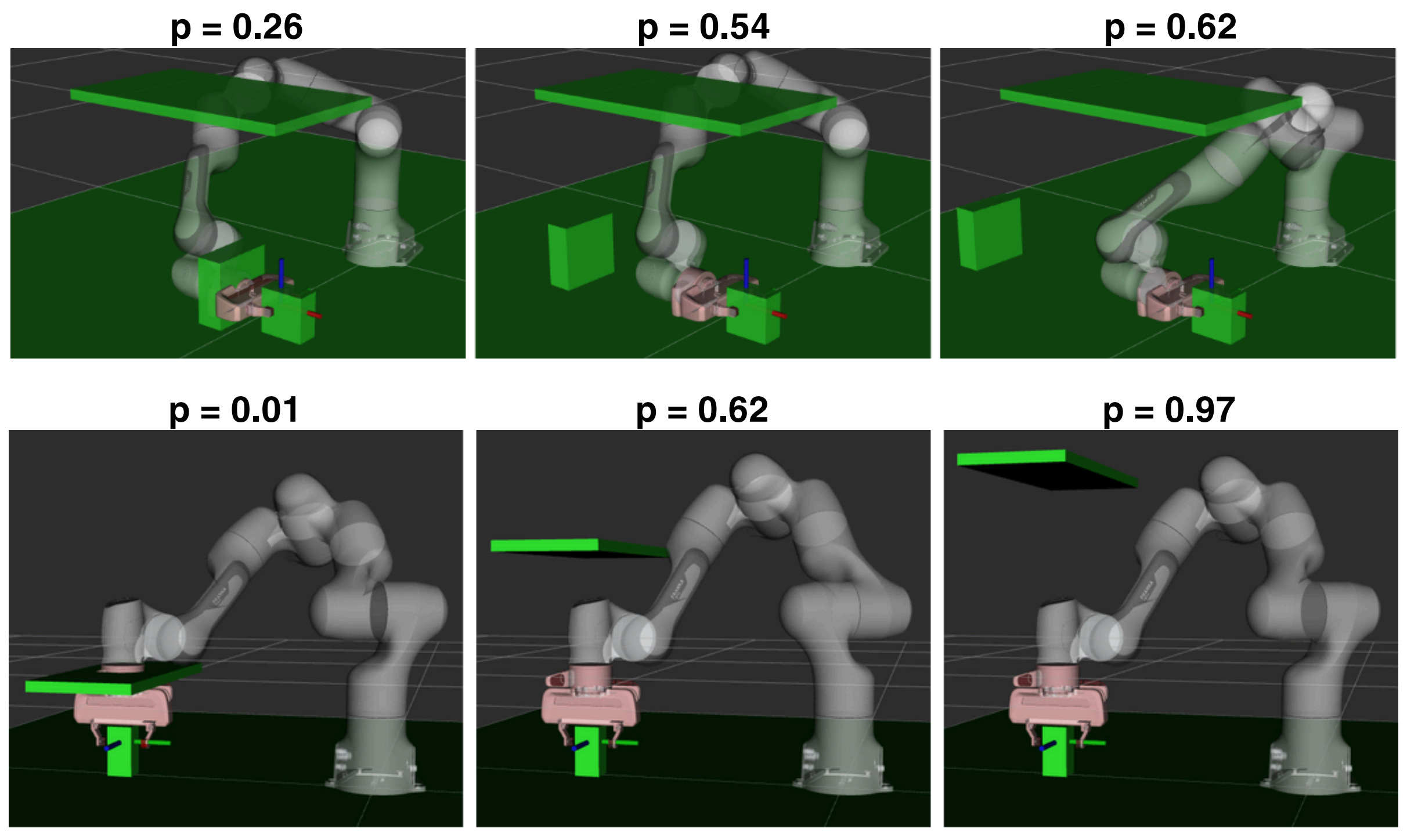
```
Algorithm 1 Task and motion planner
Input: $E, S_0, S_{goal}$                        ▷ Initial and goal states
1: $Q \leftarrow \{S_0\}$                        ▷ Set of nodes to expand
2: while Solution not found and $Q$ not empty do
3:     $S \leftarrow argmax_p(argmin_{cost} \quad Q)$
4:     if $S = S_{goal}$ then
5:         $\tau \leftarrow retrieveTaskPlan(S)$
6:         $\Pi \leftarrow constructMotionSequence(\tau, E)$
7:         if $\Pi$ is feasible then
8:             solution found
9:             break
10:        else
11:            $S_{infeasible} \leftarrow getFirstInfeasibleNode(\Pi)$
12:            $Q \leftarrow Q \setminus pruneChildren(S_{infeasible})$
13:            continue
14:        end if
15:    end if
16:    $Q \leftarrow Q \cup findChildren(S, E)$
17: end while
```

AFP-Net Performance

- **AFP-Net** is trained on a dataset extracted from **3000 randomly generated scenes**, which contain 2 objects and up to 5 stable support surfaces.
- We also generate a test dataset extracted from 1000 randomly generated scenes using a different random seed.

F1-Score	Area Under ROC-Curve	True Positive Rate	True Negative Rate
89.21%	95.56%	91.41%	84.30%

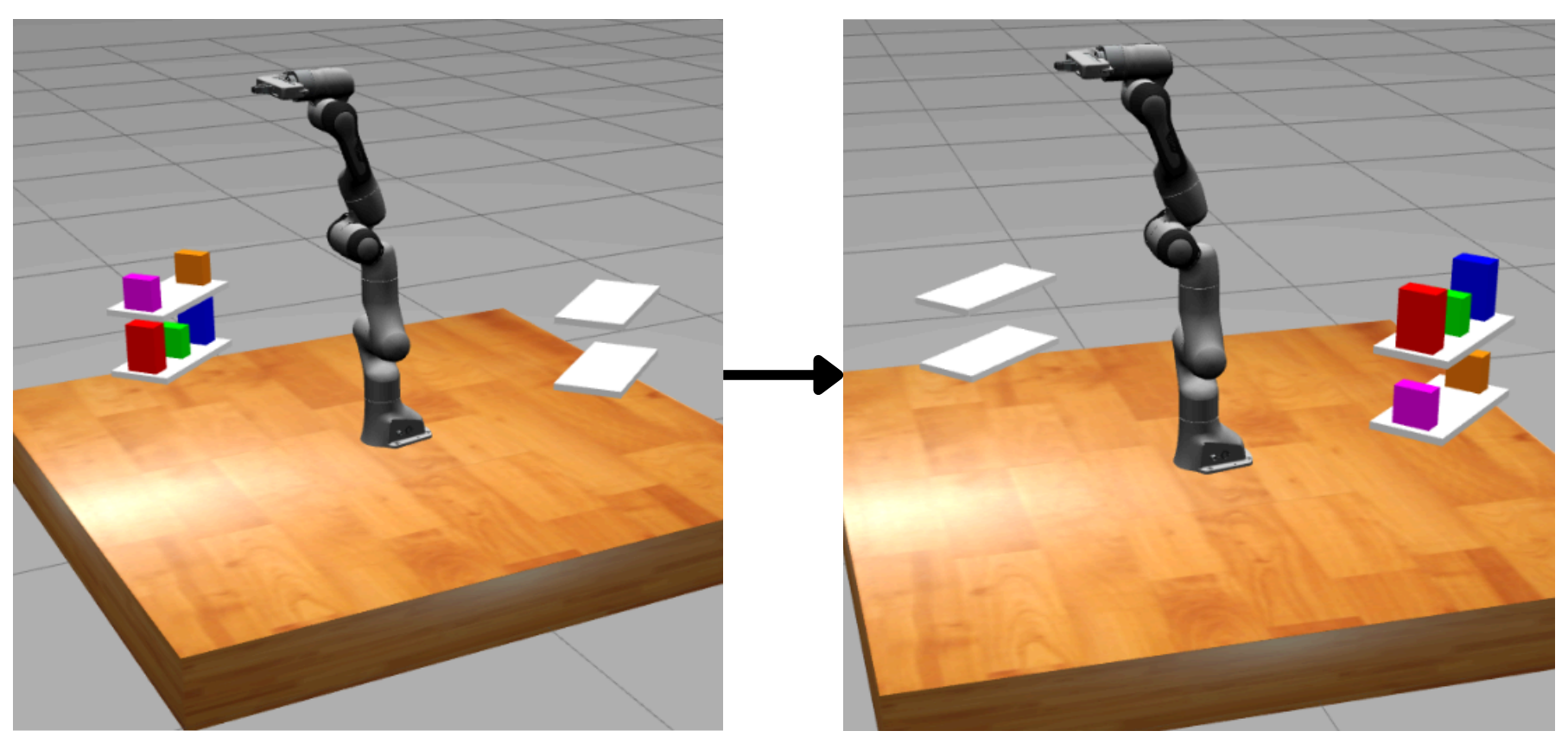
- Quantitative results show that the model is able to accurately predict the feasibility of action in 3D environments, with a **F1 score of 89%.**
- The performance of **AFP-Net** is comparable to the one obtained by [2], even though our test data is not limited to tabletop scenarios.
- **The predicted probability of feasibility is dependent on geometric properties** of the scene (height of a shelf, distance to other objects).

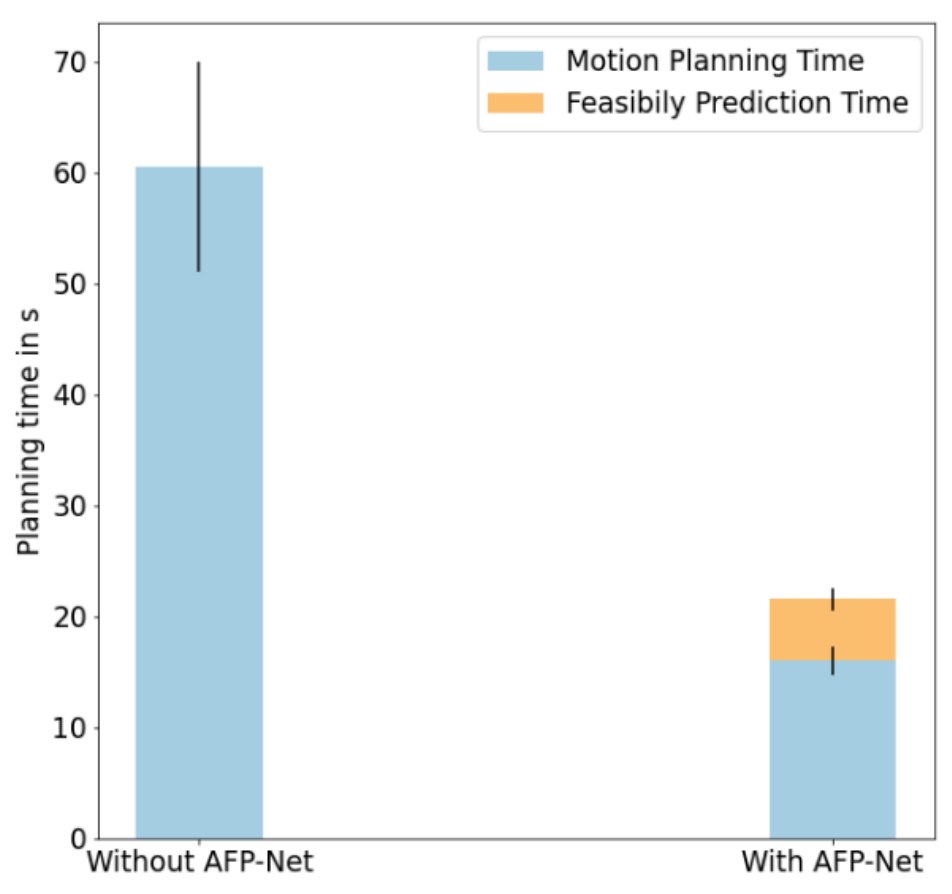


Predicted probability of feasibility depending on (top) the distance to another object, (bottom) the height of a shelf above.

Experimental Results

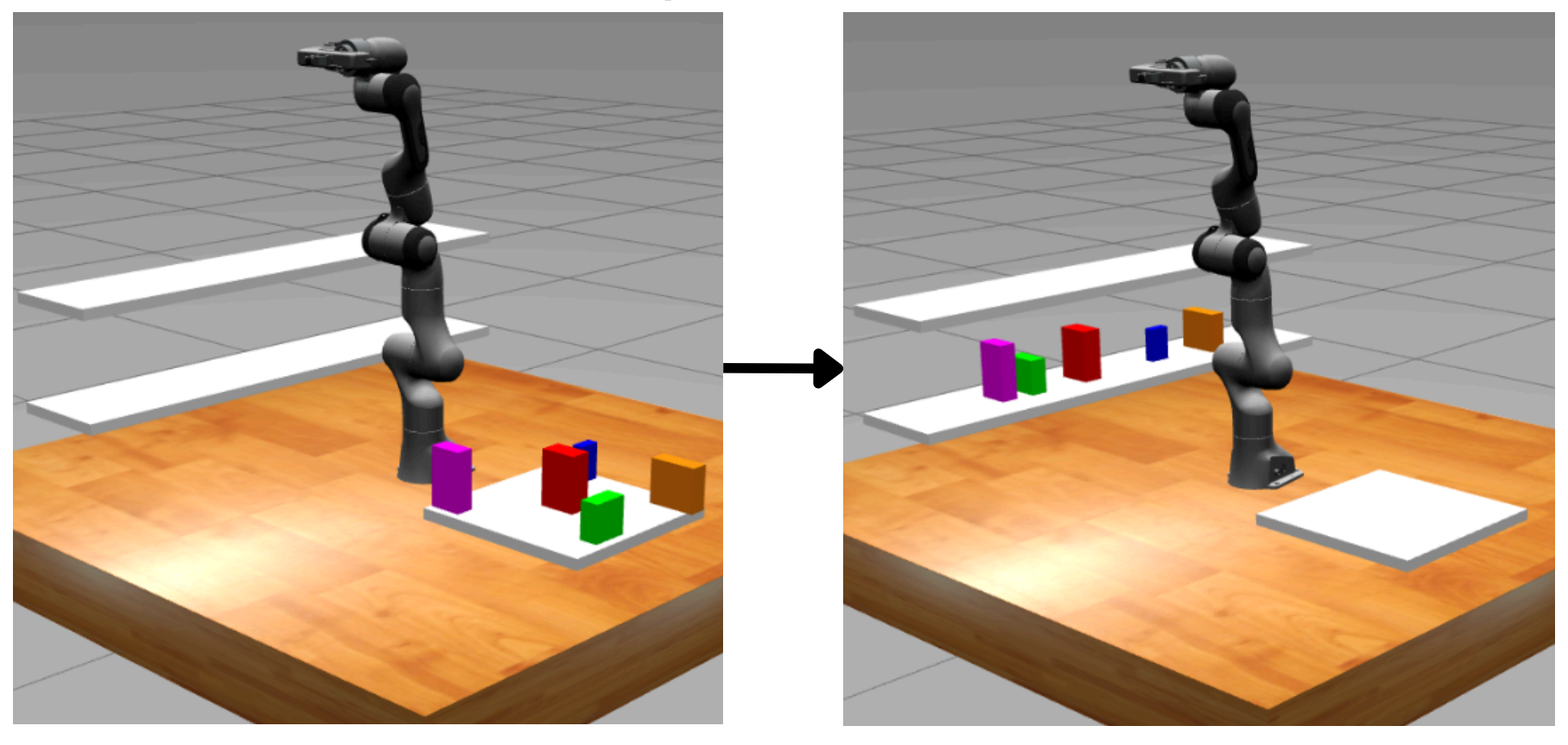
Reorder Problem

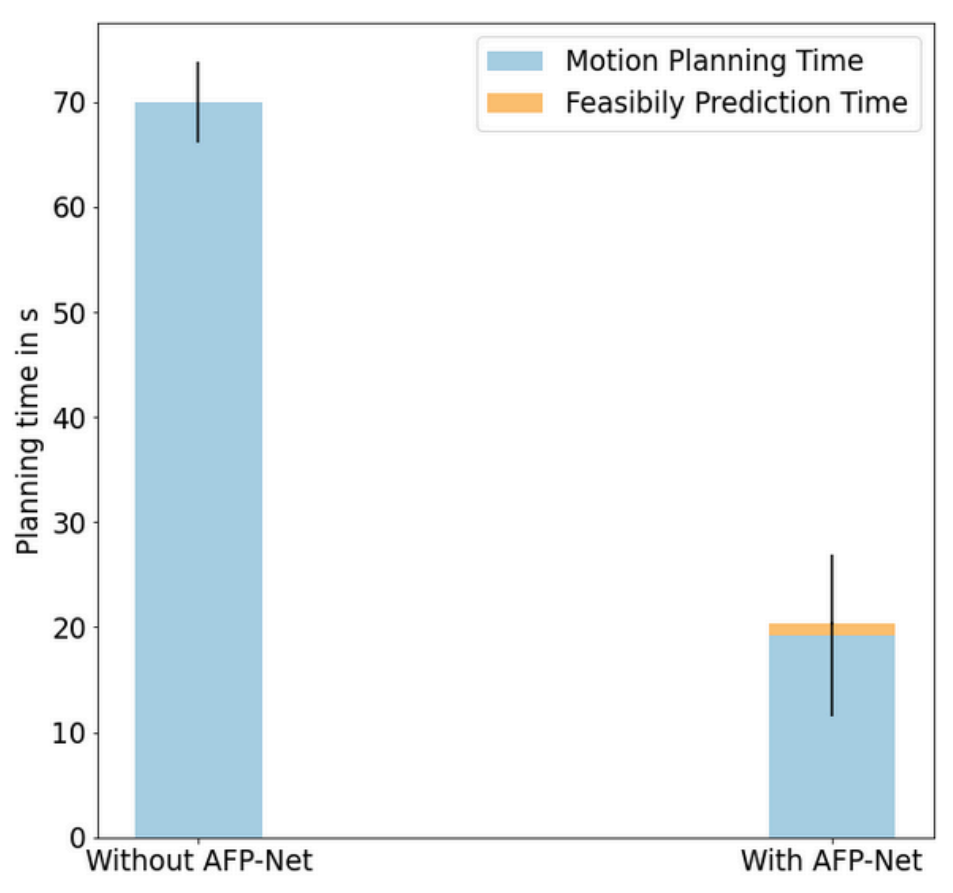




Heuristic	AFP-Net	None
# Infeasible Task Plans	**0.4**	5.0
# Motion Planning Calls	**10.5**	19.8
# Infeasible Motion Plannings	**0.5**	9.8

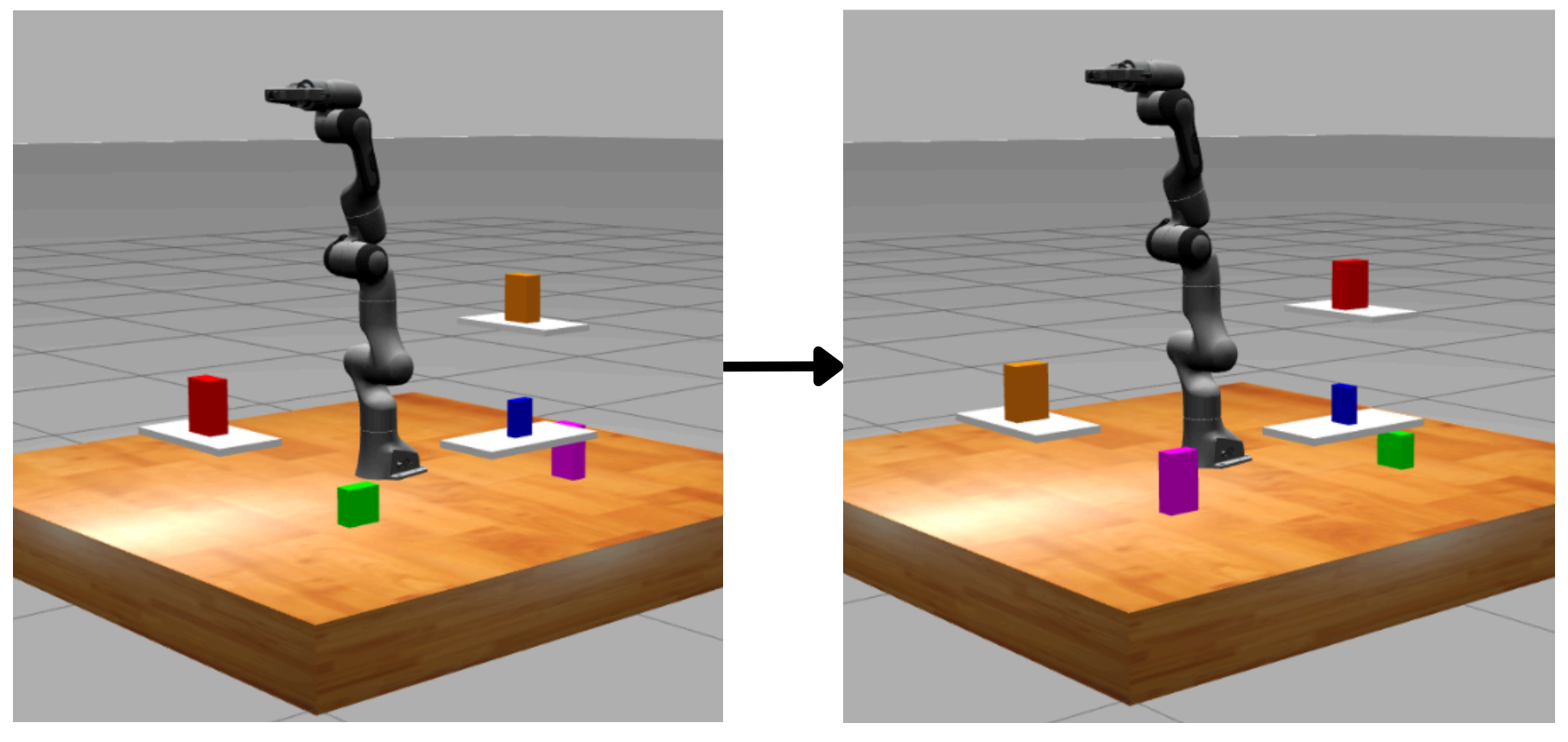
Unpack Problem

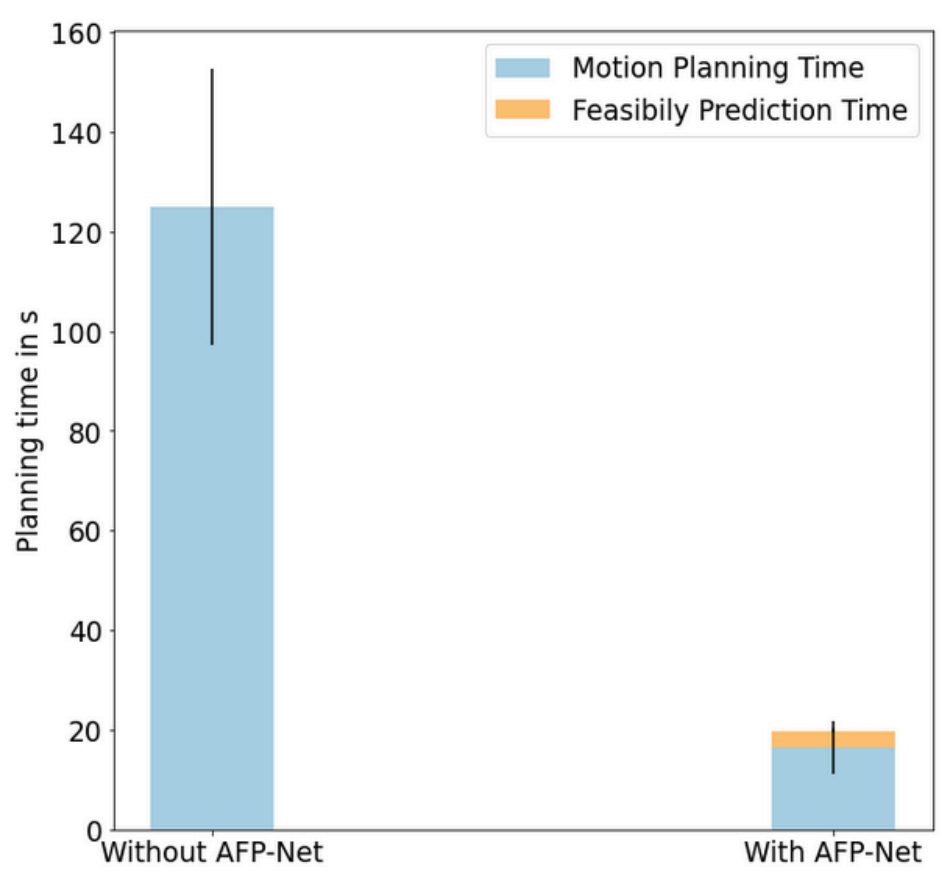




Heuristic	AFP-Net	None
# Infeasible Task Plans	**3.3**	11.8
# Motion Planning Calls	**17.1**	29.5
# Infeasible Motion Plannings	**6.3**	19.5

Swap Problem





Heuristic	AFP-Net	None
# Infeasible Task Plans	**3.9**	30.6
# Motion Planning Calls	**22.0**	67.7
# Infeasible Motion Plannings	**7.8**	53.7

- We test our approach on 3 problem domains with different types of difficulties.
- Results show that **AFP-Net** is able to help the TAMP planner **reduce the number of calls to the motion planner**, and thus **reduce the overall planning time by up to 87%.**
- Even in the case of misclassifications, the proposed method still performs better than without any feasibility heuristic.
- **The harder the problem, the more significant the performance gain** from using our approach will be.

Conclusion

- This paper presents a method for accelerating task and motion planning using a learned feasibility heuristic.
- Although **AFP-Net** can generalize to scenes with a (moderately) higher number of objects, it is still limited to simple environments.
- Future work:
  - Generalize this approach to **more realistic environments** including obstacles and objects with complex shapes.
  - Extend our method to **non-prehensile actions** such as pushing or pulling, as well as **multi-robot TAMP problems.**

References

[1] Wells, Andrew M., et al. "Learning feasibility for task and motion planning in tabletop environments." IEEE robotics and automation letters 4.2 (2019): 1255-1262.

[2] Driess, Danny, et al. "Deep visual heuristics: Learning feasibility of mixed-integer programs for manipulation planning." 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2020.

[3] Driess, Danny, Jung-Su Ha, and Marc Toussaint. "Deep Visual Reasoning: Learning to Predict Action Sequences for Task and Motion Planning from an Initial Scene Image." Robotics: Science and Systems 2020 (RSS 2020). RSS Foundation, 2020.

LAAS-CNRS
\ Laboratoire d'analyse et d'architecture des systèmes du CNRS

Contacts:
*LAAS-CNRS, {saitbouhsa, alami, simeon}@laas.fr

Laboratoire conventionné avec
Université de Toulouse