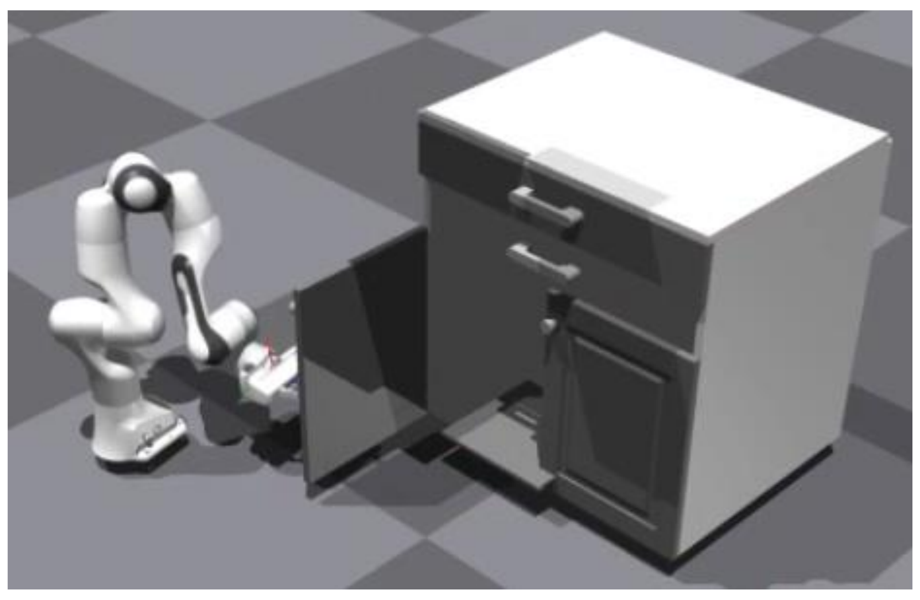
Meta-Policy Learning over Plan Ensembles for Robust Articulated Object Manipulation

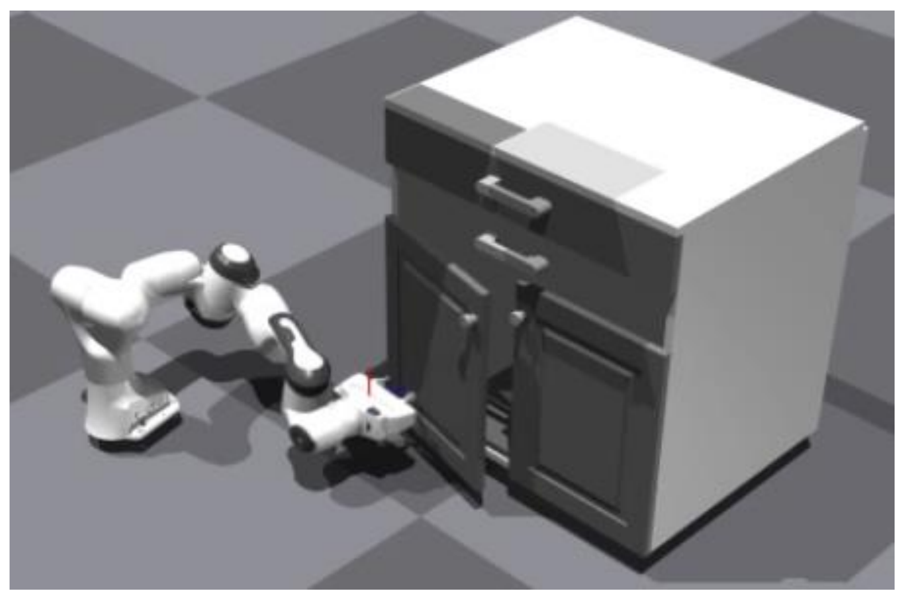
Constantinos Chamzas[12], Caelan Garrett[1], Balakumar Sundaralingam[1], Lydia E. Kavraki[2], Dieter Fox[2]

1.Nvidia 2.Rice University

Articulated Object Manipulation in Uncertain Domains



Start State (Open Door)



Goal State (Closed Door)

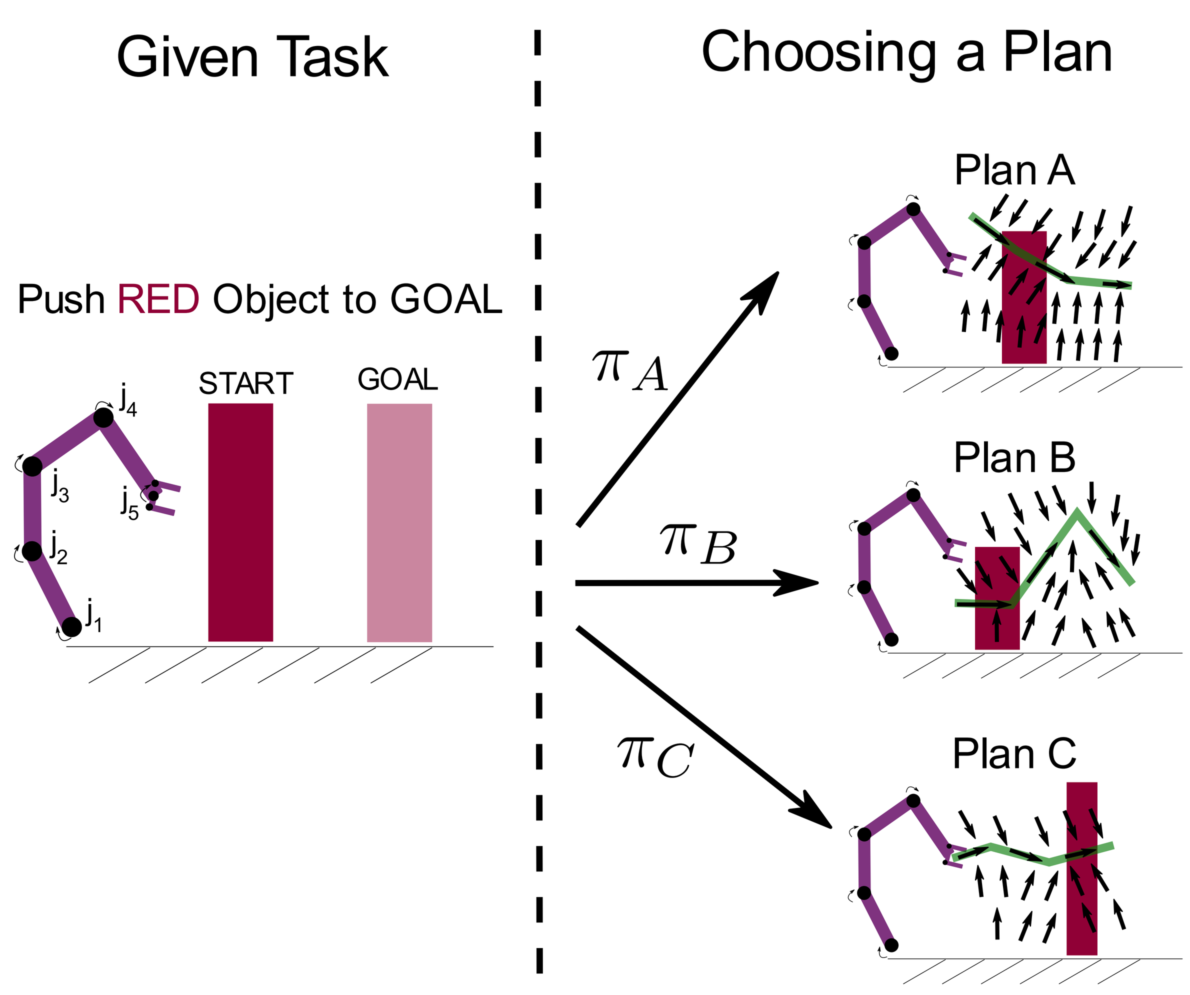
Problem Settings

- Geometry of the cabinet and robot are **known**
- Position of the robot relative to the cabinet is **uncertain**
- Kinematic and Geometric model is available but dynamics **not modeled**

Limitations of Existing Methods

- Model-based planners fail when the world model is not perfectly known
- Pure learning-based methods require a lot of training data and don't leverage prior constraints knowledge

Learning a Policy over Path Ensembles



Key Insights

- Use model-based planners to create candidate paths that satisfy known constraints
- Use learning to choose the most promising path based on past proprioceptive observations

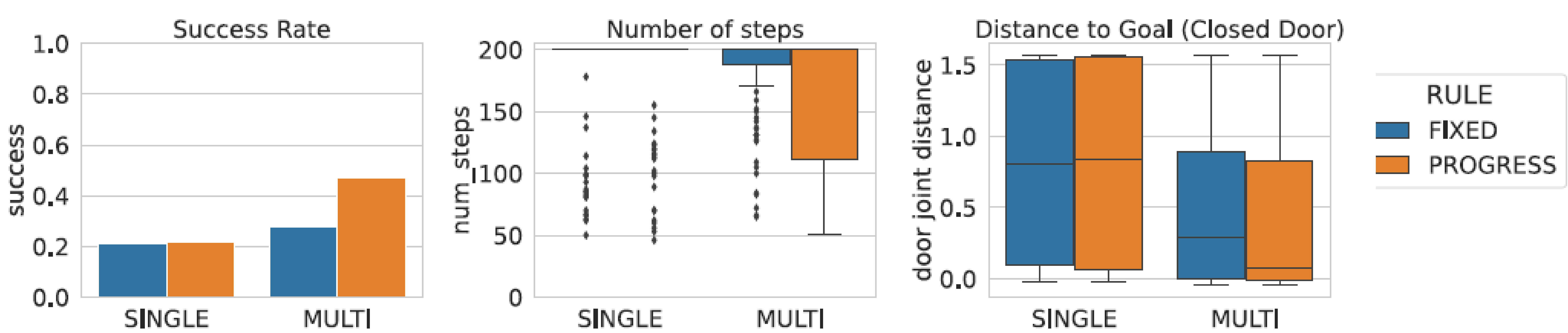
Contributions:

- Ensemble approach for manipulation which outperforms single-trajectory planners
- Learning strategy to select online among the ensemble
- Experiments in simulation that the proposed method outperforms pure model-based planning resulting in up to 40% higher success rate

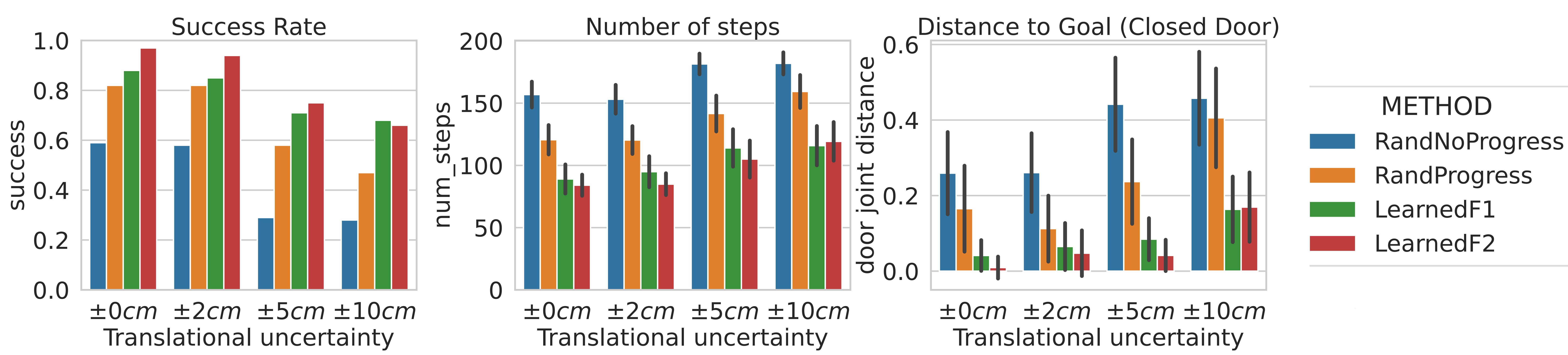
Algorithm 1 Plan Ensemble Pseudocode

```
procedure PLAN-ENSEMBLE(G, Π_p, K)
    Π_p ← ∅                                   ▷ Plan-Ensembles
    while i ≤ N do                            ▷ Create Plan-Ensembles
        ŝ ← SAMPLE-WORLD()
        p ← PLAN(ŝ, G)                        ▷ Plan Geometric Path
        π_p ← MAKE-POLICY(p)                  ▷ Create Plan
        Π_p ← Π_p ∪ π_p
    while t ≤ T do                            ▷ Iteratively Reselect Plan
        o ← OBSERVE()
        π_p^chosen ← argmax_{π_p ∈ Π_p} BEST(o, π_p)
        while PROGRESS(o, p) do
            o ← OBSERVE()                     ▷ Observe State
            if o ∈ G then
                return True                   ▷ Success!
            r ← EXECUTE(π_p^chosen)
    return False                              ▷ Failure
```

Simulation Results



Exp1: Using Plan Ensembles and Progress Rule: Performance of method when using plan ensembles (*MULTI*) or not (SINGLE) and monitoring progress (**Progress**) or not (**Fixed**). All configurations were tested on a 100 different environments, with a 200 timesteps limit



Exp2: Learning the MetaPolicy: Performance of method when using random strategies with (**RandProgress)** and without (**RandNoProgress)** the progress monitor rule, and when using learned strategies with different feature sets (**LearnedF1** / **LearnedF2**). The x-axis denotes the translational estimation error on the cabinet position.

LearnedF1: Learning strategy using only geometric features such as the current and target state for the robot, and the door.
LearnedF2: Learning strategy using the F1 geometric features and additionally 5 observations from the last 5 timesteps.