CBF-Guided RRT with SDF-Based Spherical Robot Approximation under Noisy Map Estimations

ECE 276C Robot Manipulation and Control Final Project

Pedram Aghazadeh
Department of Electrical and Computer Engineering
University of California San Diego
La Jolla, CA, U.S.
paghazadeh@ucsd.edu

Mani Amani
Department of Electrical and Computer Engineering
University of California San Diego
La Jolla, CA, U.S.
mamani@ucsd.edu

***Abstract*—Sampling-based motion planning is a standard tool for robotic manipulation but often relies on binary collision checking and lacks continuous safety guarantees. Furthermore, linear interpolations for non-linear arm links are inaccurate. In this work, we compare a standard rapidly-exploring random tree (RRT) planner against a control barrier function (CBF) guided RRT for joint-space planning of robotic manipulators. The robot is modeled using a spherical approximation, and safety is enforced using signed distance fields (SDFs) defined in the workspace. We assign a barrier function with probabilistic guarantees given statistics about the noise in the environment. We incorporate the CBF into the RRT steering procedure via a quadratic program with a safety filtering objective that projects nominal controls onto a safe set defined by obstacle clearance. We evaluate the methods across noisy map estimations and provide empirical results in two different settings.**

***Index Terms*—Motion planning, robotic manipulation, rapidly-exploring random trees, control barrier functions, signed distance fields.**

I. Introduction and Background

Motion planning for robot manipulation in cluttered environments remains challenging due to high-dimensional configuration spaces and strict safety requirements. Sampling-based planners such as RRT scale well with dimensionality, but their reliance on discrete collision checking can lead to frequent failures in narrow passages or environments with tight clearance.

Control Barrier Functions (CBFs) provide a continuous-time framework for enforcing state constraints by restricting control actions to remain within a safe set. This design can encompass both kino-dynamic constraints and remove the need for direct collision checking in RRT. One of the assumptions of the RRT algorithm is that the environment is fully known; however, this assumption is quickly violated given any real-world implementation requring map estimation. Given the uncertainty in state estimation, the performance of RRT-like algorithms can be significantly drop. Recent work has shown that CBFs can be integrated with motion planning algorithms to steer trajectories away from unsafe regions while preserving feasibility [1]. Long et al. demonstrated that CBF-based motion planning using distance fields is feasible in configuration space [2]. In their approach, a configuration-space distance field is approximated by a neural network, which is queried by providing the system state as input.

In contrast, one of our contributions is to show that, by spherically approximating each link, we do not need an explicit configuration-space distance field: all sphere–link safety constraints can be enforced directly in the workspace using the workspace SDF instead of having the CDF explicitly. Following the proposal, this project investigates the integration of CBFs into RRT steering for robotic manipulators using:

- A joint-space single-integrator model for steering the tree,
- Workspace signed distance fields (SDFs),
- A spherical approximation of robot links.

Contributions of this work:

1) Formulating a SDF-CBF barrier using workspace SDF instead of configuration space distance fields while providing forward invariance for all sphere constraints.
2) Empirical comparison between vanilla RRT and RRT+CBF across increasingly complex environments.
3) Extension of the comparison to a dual-arm manipulation scenario.

II. Preliminaries

A. System Model

Let the robot joint configuration be $\mathbf{q} \in \mathbb{R}^n$, with control input $u \in \mathbb{R}^n$. We assume first-order single-integrator dynamics:

$$\dot{\mathbf{q}} = u. \tag{1}$$

B. Spherical Robot Approximation

Each robot link is approximated by a set of spheres indexed by $k \in \mathcal{S}$. Each sphere has radius r_k and center position

$$c_k(\mathbf{q}) \in \mathbb{R}^3, \tag{2}$$

obtained via forward kinematics.

C. Signed Distance Field

Let $\Omega \subset \mathbb{R}^3$ be a solid region and $\partial\Omega$ its boundary. The signed distance field $\phi : \mathbb{R}^3 \to \mathbb{R}$ is defined as

$$\phi(\mathbf{x}) = s(\mathbf{x}) \min_{\mathbf{y} \in \partial\Omega} \|\mathbf{x} - \mathbf{y}\|_2, \qquad \mathbf{x} \in \mathbb{R}^3, \tag{3}$$

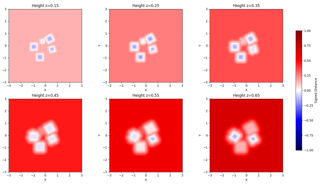


Fig. 1. SDF function at different heights for the simple environment.

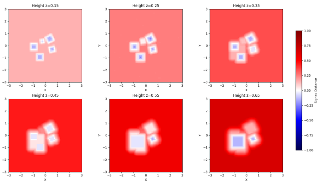


Fig. 2. SDF fuuction at different heights for the complex environment.

where

$$s(\mathbf{x}) = \begin{cases} -1, & \mathbf{x} \in \Omega, \\ 0, & \mathbf{x} \in \partial\Omega, \\ +1, & \mathbf{x} \in \mathbb{R}^3 \setminus \overline{\Omega}. \end{cases} \tag{4}$$

In coordinates, this can be written as

$$\phi(\mathbf{x}) = s(\mathbf{x}) \min_{\mathbf{y} \in \partial\Omega} \sqrt{||\mathbf{x} - \mathbf{y}||}. \tag{5}$$

With a safety margin $d_{\text{min}} \geq 0$, we define CBFs for each sphere:

$$h_k(\mathbf{q}) = \phi(c_k(\mathbf{q})) - r_k - d_{\text{min}}. \tag{6}$$

In this project, we consider two sets of environments, simple and complex shown in Fig. 1 and Fig. 2.

Proposition 1 (Probabilistically safe SDF via Cantelli tightening)**.** *Let $\hat{\phi}(\mathbf{x})$ be an estimated SDF and*

$$\phi(\mathbf{x}) = \hat{\phi}(\mathbf{x}) + e(\mathbf{x}),$$

where $e(\mathbf{x})$ has mean $\mu_e(\mathbf{x})$ and variance $\sigma_e^2(\mathbf{x}) < \infty$. For a risk level $\delta \in (0, 1)$, define:

$$\beta_\delta(\mathbf{x}) := \sigma_e(\mathbf{x}) \sqrt{\frac{1-\delta}{\delta}}$$

$$\phi_{\text{safe}}(\mathbf{x}) := \hat{\phi}(\mathbf{x}) + \mu_e(\mathbf{x}) - \beta_\delta(\mathbf{x}).$$

Then, for all $\mathbf{x}$,

$$\mathbb{P}\big(\phi(\mathbf{x}) \geq \phi_{\text{safe}}(\mathbf{x})\big) \geq 1 - \delta.$$

In particular, for a robot sphere i of radius r_i with center $f_i(\boldsymbol{q})$, define:

$$h_i(\boldsymbol{q}) := \phi\big(f_i(\boldsymbol{q})\big) - r_i$$

$$h_i^{\text{safe}}(\boldsymbol{q}) := \phi_{\text{safe}}\big(f_i(\boldsymbol{q})\big) - r_i.$$

If $h_i^{\text{safe}}(\boldsymbol{q}) \geq 0$, then

$$\mathbb{P}\big(h_i(\boldsymbol{q}) \geq 0\big) \;=\; \mathbb{P}\big(\phi(f_i(\boldsymbol{q})) \geq r_i\big) \;\geq\; 1 - \delta.$$

Proof. Fix $\mathbf{x}$ and write $X := e(\mathbf{x})$ with mean $\mu := \mu_e(\mathbf{x})$ and variance $\sigma^2 := \sigma_e^2(\mathbf{x})$. Cantelli's inequality states that for any $a > 0$,

$$\mathbb{P}(X - \mu \leq -a) \;\leq\; \frac{\sigma^2}{\sigma^2 + a^2}.$$

Choose $a = \sigma\sqrt{(1-\delta)/\delta}$. Then the right-hand side equals δ, so

$$\mathbb{P}(X \geq \mu - a) \;=\; 1 - \mathbb{P}(X - \mu \leq -a) \;\geq\; 1 - \delta.$$

By the definition of $\beta_\delta(\mathbf{x})$, this is

$$\mathbb{P}\big(X \geq \mu_e(\mathbf{x}) - \beta_\delta(\mathbf{x})\big) \;\geq\; 1 - \delta.$$

Since $\phi(\mathbf{x}) = \hat{\phi}(\mathbf{x}) + X$, we obtain

$$\mathbb{P}\big(\phi(\mathbf{x}) \geq \hat{\phi}(\mathbf{x}) + \mu_e(\mathbf{x}) - \beta_\delta(\mathbf{x})\big) = \mathbb{P}\big(\phi(\mathbf{x}) \geq \phi_{\text{safe}}(\mathbf{x})\big) \;\geq\; 1 - \delta.$$

For the sphere constraint, set $\mathbf{x} = f_i(\mathbf{q})$. If $h_i^{\text{safe}}(\mathbf{q}) \geq 0$, then $\phi_{\text{safe}}(f_i(\mathbf{q})) \geq r_i$, hence

$$\{\phi(f_i(\mathbf{q})) \geq \phi_{\text{safe}}(f_i(\mathbf{q}))\} \subseteq \{\phi(f_i(\mathbf{q})) \geq r_i\},$$

. □

Given that adding a random variable in the form of $X \sim \mathcal{N}(0, \sigma^2)$ such that $\phi(x.y) + X$, we can assume that the measured SDF is noisy with the SDF measurement being a Gaussian random variable such that $\hat{\phi}(x, y) \sim \mathcal{N}(\phi(x, y), \sigma^2)$. This allows us to utilize the Cantelli inequality to mimic real world scenarios using online SDF methods such as Gaussian Processes.

III. Problem Definition

Given

- An initial configuration $q_s \in \mathcal{C}$,
- A goal region $\mathcal{G} \subset \mathcal{C}$,
- Joint and control limits,

the objective is to compute a collision-free joint-space path from q_s to $\mathcal{G}$ where C represents the safe set.

We compare two planners:

1) **Vanilla RRT:** Collision checking only.
2) **CBF-guided RRT:** Continuous safety enforced during steering without collision checking during the step.

A. Goal definition

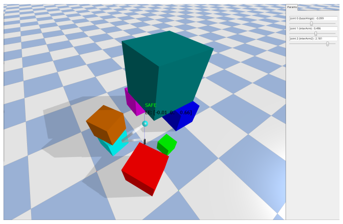
We developed two scripts to assure that the goals are different from the assignments and that they're still feasible in both simple and complex environments as well as single and dual arm robots. Figures 3 and 4 demonstrate how the goals are set for each environment and robot, with simple sliders to move the robot arms around and observing the end-effector's position in real-time.

IV. Methods

A. Baseline: Vanilla RRT

Standard RRT expands a tree $\mathcal{T}$ as shown in Alg. 1.

Collision checking is binary and performed after candidate extensions.



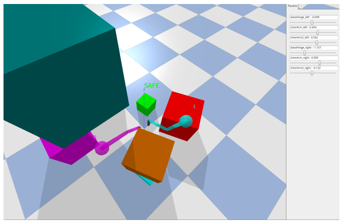
Params

Joint 0 (baseHinge) : -0.099

Joint 1 (interArm) : 0.496

Joint 2 (interArm2) : 2.181

Fig. 3. Finding feasible goals in the complex environment.



Params

baseHinge_left : -0.099

interArm_left : 0.694

interArm2_left : 0.562

baseHinge_right : -1.157

interArm_right : 0.959

interArm2_right : -0.132

Fig. 4. For dual arm robots the goals are defined in a way to avoid interlocking arms with each other.

B. CBF-Guided RRT (Proposed)

Instead of naive steering, we compute a nominal control

$$u_{\text{nom}} = K\big(\mathbf{q}_{\text{rand}} - \mathbf{q}_{\text{near}}\big), \tag{7}$$

and then project it onto the safe set using CBF constraints.

We model the local robot dynamics as a single integrator

$$\dot{\mathbf{q}} = u, \tag{8}$$

and use the (probabilistically) safe barrier functions

$$h_k(\mathbf{q}) := \phi_{\text{safe}}\big(f_k(\mathbf{q})\big)$$

for each robot sphere k, where $f_k(\mathbf{q})$ is the center of sphere k and r_k its radius.

For each active sphere constraint k, the CBF condition is

$$\nabla h_k(\mathbf{q})^\top u + \alpha\, h_k(\mathbf{q}) \;\geq\; 0, \tag{9}$$

with $\alpha > 0$. At each RRT extension step, given q_{near} we solve the quadratic program

$$\begin{aligned} u^\star(q) = \;& \arg\min_{u \in \mathbb{R}^n} \; \big\| u - u_{\text{nom}} \big\|^2 \\ & \text{s.t.} \quad \nabla h_k(\mathbf{q})^\top u + \alpha\, h_k(\mathbf{q}) \geq 0, \quad \forall k \in \mathcal{K}(\mathbf{q}), \end{aligned} \tag{10}$$

where $\mathcal{K}(\mathbf{q})$ is the set of active sphere constraints at configuration q.

Defining the safe set

$$\mathcal{C}_k := \{\, \mathbf{q} \in \mathcal{Q} \mid h_k(\mathbf{q}) > 0 \,\}, \qquad \mathcal{C} := \bigcap_k \mathcal{C}_k, \tag{11}$$

we obtain the following invariance guarantee.

Proposition 2 (Forward invariance under multi-sphere CBF constraints)**.** *Consider the system $\dot{\boldsymbol{q}} = u$ with differentiable $h_k : \mathcal{Q} \to \mathbb{R}$. Assume that for all $\boldsymbol{q}$ in a neighborhood of $\mathcal{C}$ the applied control $u(\boldsymbol{q})$ satisfies*

$$\nabla h_k(\boldsymbol{q})^\top u(\boldsymbol{q}) + \alpha\, h_k(\boldsymbol{q}) \;\geq\; 0 \quad \textit{for all } k. \tag{12}$$

Algorithm 1 Standard RRT

```
Initialize tree $\mathcal{T} \leftarrow \{q_s\}$
for $i = 1$ to $N$ do
    Sample $q_{\text{rand}}$
    Find nearest node $q_{\text{near}} \in \mathcal{T}$
    Steer from $q_{\text{near}}$ toward $q_{\text{rand}}$ to obtain $q_{\text{new}}$
    if segment$(q_{\text{near}}, q_{\text{new}})$ is collision-free then
        Add $q_{\text{new}}$ to $\mathcal{T}$
    end if
    if $q_{\text{new}} \in \mathcal{G}$ then
        return path from $q_s$ to $q_{\text{new}}$
    end if
end for
```

Algorithm 2 CBF-Guided RRT

```
Initialize tree $\mathcal{T} \leftarrow \{q_s\}$
for $i = 1$ to $N$ do
    Sample $q_{\text{rand}}$
    Find nearest node $q_{\text{near}} \in \mathcal{T}$
    Compute nominal control $u_{\text{nom}} \leftarrow K\big(q_{\text{rand}} - q_{\text{near}}\big)$
    Solve CBF-QP to obtain $u^\star$
    Update configuration $q \leftarrow q_{\text{near}} + \Delta t\, u^\star$
    if $q \in \mathcal{C}$ then
        Add $q$ to $\mathcal{T}$
    end if
    if $q \in \mathcal{G}$ then
        return path from $q_s$ to $q$
    end if
end for
```

If $q(0) \in \mathcal{C}$, then $q(t) \in \mathcal{C}$ for all $t \geq 0$; i.e., the set $\mathcal{C} = \bigcap_k \mathcal{C}_k$ is forward invariant.

Proof. Fix any k and consider the evolution of $h_k(q(t))$ along system trajectories. By the chain rule and the dynamics $\dot{q} = u(\mathbf{q})$,

$$\dot{h}_k(\mathbf{q}) = \nabla h_k(\mathbf{q})^\top \dot{q} = \nabla h_k(\mathbf{q})^\top u(\mathbf{q}).$$

By the assumption (12),

$$\dot{h}_k(\mathbf{q}) \;\geq\; -\alpha\, h_k(\mathbf{q}).$$

Thus h_k satisfies the scalar differential inequality

$$\dot{h}_k(t) \;\geq\; -\alpha\, h_k(t).$$

By the comparison lemma, if $h_k(0) \geq 0$, then $h_k(t) \geq 0$ for all $t \geq 0$. Since this holds for every k, we have

$$\mathbf{q}(0) \in \mathcal{C} \;\Rightarrow\; h_k(\mathbf{q}(t)) \geq 0 \,\forall k,\ \forall t \geq 0 \;\Rightarrow\; \mathbf{q}(t) \in \mathcal{C},\ \forall t \geq 0,$$

which proves forward invariance of $\mathcal{C}$. □

The algorithm for RRT-CBF is explained in Alg. 2.

C. Dual-Arm Extension

For two arms with configurations $\mathbf{q}^{(1)}, \mathbf{q}^{(2)}$, constraints are enforced independently per arm and jointly for inter-arm collision avoidance by adding cross-arm sphere constraints into the CBF set.

TABLE I
SIMPLE ENVIRONMENT: SUCCESS RATE AND NODE COUNT (MEAN ± STD) OVER 5 SEEDS.

σ	Succ. RRT-CBF	Succ. RRT	Nodes RRTCBF	Nodes RRT
0.000	1.00	1.00	**72.7** ±13.7	86.0 ± 30.8
0.005	1.00	1.00	**72.6** ±12.3	86.0 ± 30.8
0.010	1.00	1.00	**72.6** ±12.3	86.0 ± 30.8
0.020	1.00	1.00	**73.9** ±11.4	86.0 ± 30.8

TABLE II
COMPLEX ENVIRONMENT: SUCCESS RATE AND NODE COUNT (MEAN ± STD) OVER 5 SEEDS.

σ	Succ. RRT-CBF	Succ. RRT	Nodes RRT-CBF	Nodes RRT
0.000	1.00	1.00	**80.4** ±13.8	81.2 ± 19.1
0.005	1.00	1.00	**73.4** ±15.2	81.2 ± 19.1
0.010	**1.00**	0.80	**73.4** ±15.2	81.2 ± 19.1
0.020	**1.00**	0.60	**74.4** ±14.0	81.2 ± 19.1

V. EXPERIMENTS AND RESULTS

A. Scenarios

We evaluate both planners in:

1) **Simple Environment:** Sparse obstacles, wide clearance.
2) **Cluttered Environment:** Narrow passages and tight clearance.
3) **Dual-Arm Environment:** Two arms operating in shared workspace. The ablation study evaluates a wide range of parameter variations.

B. Metrics

We chose to use the following metrics

- Collision free plan success rate
- Number of nodes used

All the results from these experiments in different scenarios are shown in Tables I, II, and III. An example of the search for the path in the dual arm robot is shown in Fig 5 with the end effector's goals shown as spheres.

VI. DISCUSSION AND CONCLUSIONS

Results indicate that CBF-guided RRT substantially improves planning success in cluttered and dual-arm environments.

We chose to not reduce the probabilistic safe bound from the RRT configuration space given the fact that we wanted to maintain the most basic and classical method as our baseline with all of it's original assumptions. While the baseline RRT struggles with frequent rejected extensions and narrow passages, the CBF-guided planner actively steers expansions within the safe set.

The computational overhead of solving small QPs is modest compared to the cost of repeated collision checking failures. Overall, integrating CBFs into RRT provides a principled, geometry-aware improvement for safe manipulation.

We see one big outlier in convergence in the dual arm scenario for RRT given a certain random seed. This shifts the statistics by a large margin. Without the outlier the statistics are closer to that of the RRT-CBF, however, they still use on average about 351 nodes. This is more than twice of that of the RRT-CBF which is expected given our formulation.

TABLE III
SIMPLE ENVIRONMENT: SUCCESS RATE AND NODE COUNT (MEAN ± STD) OVER 5 SEEDS FOR THE **DUAL ARM ROBOT**.

σ	Succ. RRT-CBF	Succ. RRT	Nodes RRTCBF	Nodes RRT
0.000	1.00	1.00	**159.2** ±34.16	1288.8 ± 2101.59
0.005	**1.00**	0.80	**157.8** ±25.46	1288.8 ± 2101.5
0.010	**1.00**	0.60	**166.6** ±26.82	1288.8 ± 2101.5
0.020	**0.80**	0.60	**180** ±22.91	1288.8 ± 2101.5

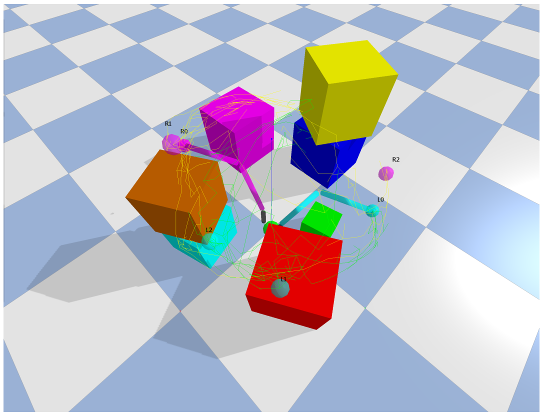


Fig. 5. Trees found for each arm in a dual arm setting ensuring that the robot does not collide with itself nor the obstacles.

VII. FUTURE WORK

For future work, we propose to use time parameterized paths that would also avoid paths colliding at times. The current methodology also does not account for self collisions with the robot itself since we are operating in workspace SDF. Finally, we observed inaccuracies due to the execution not satisfying the dynamics of steering.

In the current implementation, we check if the nodes are in collision between each arm, however, this does guarantee that the overlapping paths will never collide. In order to account for that, future work needs to analyze either incorporating such constraints into the CBF or time parametrize the trajectories as mentioned above.

ACKNOWLEDGMENTS

This project was completed as part of ECE 276C at UC San Diego. We thank the teaching staff for guidance and infrastructure support.

REFERENCES

[1] L. Liu, Y. Zhang, L. Zhang, and M. Kermanshabi, "Rrt-cbf based motion planning," 2024. [Online]. Available: https://arxiv.org/abs/2410.00343

[2] K. Long, K. M. B. Lee, N. Raicevic, N. Attasseri, M. Leok, and N. Atanasov, "Neural configuration-space barriers for manipulation planning and control," 2025. [Online]. Available: https://arxiv.org/abs/2503.04929