# Safe Human-Robot Interactions with Continuous and Differentiable Distance Fields

Usama Ali[1], Adrian Müller[1], Fouad Sukkar[2], Lan Wu[2], Tobias Kaupp[1], Teresa Vidal-Calleja[2]

*Abstract*— Human-robot collaboration applications require safe and reactive planning. Euclidean distance fields (EDF) are a promising representation of such dynamic scenes due to their ability to reason about free space and the readily available distance to collision costs. A key challenge for the commonly used discrete EDF representations, however, is the need for differentiable distance fields to produce smooth collision costs and efficient updates of dynamic objects. In this paper, we propose to use a Gaussian Process (GP) distance field-based framework that enables both, differentiable distance fields and fast dynamic scene updates. Moreover, we combine this framework with the Riemannian Motion Policies as a local reactive planner to enable safe human-robot interactions. We design a collision avoidance policy that models the repulsive motion using the distance and gradient fields from our GP. We show our reactive planner in an experiment with a UR5e interacting safely and smoothly with a human.

## I. INTRODUCTION

In order for robots to collaborate in close proximity to humans, we require them to be reactive and safe. In this work we leverage the Interactive Distance Field Mapping and Planning (IDMP) framework [1] which utilizes Gaussian Processes (GP) to represent the continuous distance and gradient fields. For avoiding collisions with moving objects we use a local reactive motion generator called Riemannian motion polices (RMP). RMPs remain to be demonstrated in combination with EDFs in online and dynamic settings [2] and have mainly been used in conjunction with voxel-based occupancy representations [3], [4].

In this paper we formulate motion policies that utilise our interactive GP distance and gradient information. We demonstrate the use of RMPs in combination with GP distance fields in a human-robot collaboration setting where a UR5 robot arm must carry out a task in a shared workspace as shown in Figure 1. We show that our method achieves smoother and more natural reactive behaviour compared to an occupancy-based method.

## II. PRELIMINARIES

### A. Interactive Distance Field Mapping

For mapping dynamic scenes we utilise the IDMP framework [1] which implicitly models a continuous Euclidean distance field via a reverting GP function [5]. Using a GP with the covariance kernel $k(\mathbf{x}, \mathbf{x}')$ we can recover the occupancy at any point $\mathbf{x}_*$ by

$$o(\mathbf{x}_*) = \mathbf{k}(\mathbf{x}_*, X) \left( \mathbf{K}(X, X) + \sigma_y^2 \mathbf{I} \right)^{-1} \mathbf{1}, \quad (1)$$

[1]Authors are with the Center for Robotics (CERI) at the Technical University of Applied Sciences Würzburg-Schweinfurt (THWS), Germany.
[2]Authors are with the Robotics Institute, Faculty of Engineering and IT, University of Technology Sydney (UTS), Australia.
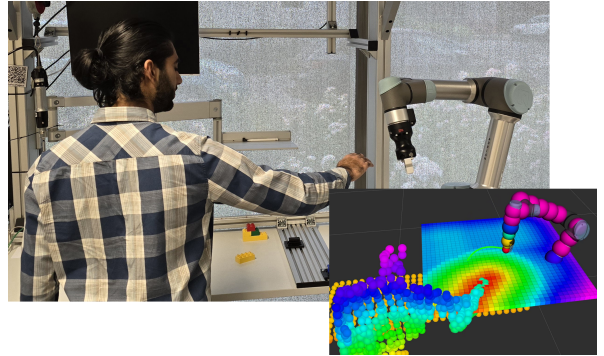
Fig. 1. Dynamic obstacle avoidance using our combined RMP and interactive GP distance field mapping framework. (inset) Robot with collision spheres and trajectory traced out in green. The sensor pointcloud is depicted as spheres, which are colored depending on their height. A slice of the distance field is visualized as colored boxes with a color spectrum from red, meaning small distances, to purple, meaning large distances.

where $\mathbf{k}(\mathbf{x}_*, X) = [k(\mathbf{x}_*, \mathbf{x}_1), ..., k(\mathbf{x}_*, \mathbf{x}_Q)]$ represents the covariance vector between the input points and the query point and $\mathbf{K}(X, X) = \left[ \mathbf{k}(\mathbf{x}_1, X)^\top, ..., \mathbf{k}(\mathbf{x}_Q, X)^\top \right]$ the covariance matrix between the input points.

To derive the distance $d(\mathbf{x}_*)$ from the occupancy $o(\mathbf{x}_*)$, the kernel function has to be reverted, resulting in

$$d(\mathbf{x}_*) = \sqrt{-2l^2 * \log\left(\frac{o(\mathbf{x}_*)}{\sigma^2}\right)}. \quad (2)$$

As 1 and 2 are continuous and differentiable, we can retrieve the gradient analytically [5] by

$$\nabla d(\mathbf{x}_*) = \nabla \mathbf{k}(\mathbf{x}_*, X) \left( \mathbf{K}(X, X) + \sigma_y^2 \mathbf{I} \right)^{-1} \mathbf{1}. \quad (3)$$

### B. Riemannian Motion Policies

The Riemannian Motion Policies framework [6] is a local reactive motion generator that combines multiple simple task-based policies to achieve complex high-level behaviours.

A policy $\mathcal{P}$ is modeled on a Riemann Manifold $\mathcal{M}$ and is defined by the tuple $(f, \mathcal{A})^\mathcal{M}$, where $f(\mathbf{x}, \dot{\mathbf{x}})$ is an acceleration and $\mathcal{A}(\mathbf{x}, \dot{\mathbf{x}})$ is a Riemann Metric for $\mathcal{M}$ that both vary with the state. See Ratliff et al. [6] for more details.

## III. REACTIVE PLANNING IN INTERACTIVE DISTANCE FIELDS

We propose to use the output of the GP distance field as described in Section II-A in combination with a custom-designed collision avoidance policy for RMP (Section II-B) to safely navigate in dynamic scenes.

Figure 2 shows the proposed system architecture, where the IDMP framework takes as input the depth sensor's data
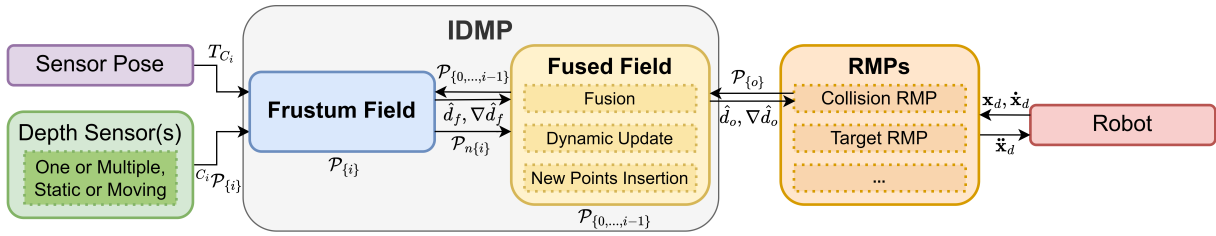
Fig. 2. System diagram of the proposed framework. IDMP processes the sensor data and fuses it into the GP distance and gradient field which is then queried by the reactive planner (RMPs).

and poses. Our reactive planner queries distance and gradient information from the GP to generate accelerations via RMPs which are passed to a controller for execution on the robot.

We can then query distances for each collision point $\mathbf{x}_c$ using Eq. 2 as follows:

$$d(\mathbf{x}_c) = \sqrt{-2l^2 * \log\left(\frac{\mathbf{k}(\mathbf{x}_c, X)\left(\mathbf{K}(X, X) + \sigma_y^2 \mathbf{I}\right)^{-1}}{\sigma^2}\right)}. \quad (4)$$

The gradient $\nabla d(\mathbf{x}_c)$ can be calculated analytically via Eq. 3.

We adapt the collision avoidance policy proposed by Ratliff et al. [6] consisting of a repulsion and a damping component. We model the repulsive motion as a function of the gradient $\nabla d$ and distance $d$ from the GP distance field,

$$\ddot{\mathbf{x}}_{rep} = \eta_{rep} \nabla d(\mathbf{x}) \exp\left(-\frac{d(\mathbf{x})}{\nu_{rep}}\right), \quad (5)$$

with the repulsion gain $\eta_{rep}$ and the length scale $\nu_{rep}$. The damping term reduces oscillations and is a function of the velocity $\dot{x}$ and distance $d$ described by,

$$\ddot{\mathbf{x}}_{damp} = -\eta_{damp}\left(1 - S\left(\frac{\dot{\mathbf{x}}}{\nu_{damp}}\right)\right) * \frac{\nu_{rep}\dot{\mathbf{x}}}{d(\mathbf{x})}, \quad (6)$$

where $S(\mathbf{x}) = \dfrac{1}{1 + e^{-\mathbf{x}}}$, $\eta_{damp}$ is the damping gain and $\nu_{damp}$ the damping length scale.

For the metric we propose a diagonal matrix with values from a smooth activation gate which depends on the distance $d$ and the activation parameter $d_a$ as

$$\mathcal{A} = \mathbf{I} * \begin{cases} \dfrac{d^2}{d_a^2} - 2 * \dfrac{d}{d_a} + 1 & \text{for } d < d_a \\ 0 & \text{for } d > d_a \end{cases} \quad (7)$$

In addition to our collision policy, we use a target attractor policy, a joint limit policy and a velocity limit policy as described in [6], [7].

## IV. EXPERIMENTS

We evaluate our method in a mock human-robot interaction scene where the robot is tasked to cycle between two waypoints. During the execution, a human enters the workspace and places their arm in the way of the robot.

We compare the behaviour of our framework against an occupancy-based reactive method implemented in the ROS package MoveIt. This baseline method builds an Octomap [8]
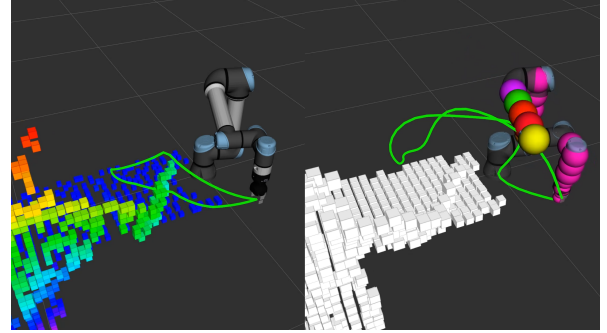


Fig. 3. Comparison of trajectories for the baseline (left) and our proposed method (right). The robot is tasked to move between two waypoints, as the human moves their arm in the way.

which is continuously updated with the sensor input. A trajectory is planned using the Bi-directional Fast Marching Tree (BFMT*) algorithm [9]. During execution the trajectory is checked for possible collisions which then triggers replanning.

As can be seen in Table I the trajectories produced by our method result in much smoother trajectories. Notably the mean squared jerk and change in curvature were approximately 2x and 3x lower, respectively, than the baseline.

TABLE I
COMPARISON OF SMOOTHNESS METRICS.

| Metric | IDMP-RMP | Baseline | Difference |
|---|---|---|---|
| Variance of Acceleration (X) | **2.30e-05** | 3.08e-05 | -25.32% |
| Variance of Acceleration (Y) | **3.07e-05** | 5.47e-05 | -43.88% |
| Variance of Acceleration (Z) | **1.03e-05** | 4.13e-05 | -75.06% |
| Total Jerk | **0.602** | 0.832 | -27.64% |
| Mean Squared Jerk | **2.63e-05** | 5.28e-05 | -50.19% |
| Mean Curvature Change | **0.302** | 0.925 | -67.35% |

## V. CONCLUSION

In this paper we presented a framework that utilises interactive distance and gradient fields based on GPs in combination with RMPs to reactively plan in dynamic scenes. We demonstrated this in a human-robot collaboration scenario where a robot arm smoothly avoids colliding with a person entering its workspace while continuing to carry out its task.

Future work aims to use semantics such as distinguishing between human and non-human moving objects. Exploring how this can be incorporated into the RMP formulation to achieve more natural and nuanced avoidance behaviours in more complex collaborative settings is a promising avenue.

## REFERENCES

[1] U. Ali, L. Wu, A. Mueller, F. Sukkar, T. Kaupp, and T. Vidal-Calleja, "Interactive Distance Field Mapping and Planning to Enable Human-Robot Collaboration," *arXiv preprint arXiv:2403.09988*, 2024.

[2] M. Pantic, C. Cadena, R. Siegwart, and L. Ott, "Sampling-free obstacle gradients and reactive planning in neural radiance fields," in *Proc. of Workshop on" Motion Planning with Implicit Neural Representations of Geometry" at ICRA*, 2022.

[3] M. Pantic, I. Meijer, R. Bähnemann, N. Alatur, O. Andersson, C. Cadena, R. Siegwart, and L. Ott, "Obstacle avoidance using Raycasting and Riemannian Motion Policies at kHz rates for MAVs," in *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 1666–1672.

[4] V. Reijgwart, M. Pantic, R. Siegwart, and L. Ott, "Waverider: Leveraging hierarchical, multi-resolution maps for efficient and reactive obstacle avoidance," *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[5] C. Le Gentil, O.-L. Ouabi, L. Wu, C. Pradalier, and T. Vidal-Calleja, "Accurate Gaussian-Process-based Distance Fields with applications to Echolocation and Mapping," *IEEE Robotics and Automation Letters*, 2023.

[6] N. D. Ratliff, J. Issac, D. Kappler, S. Birchfield, and D. Fox, "Riemannian motion policies," *arXiv preprint arXiv:1801.02854*, 2018.

[7] C.-A. Cheng, M. Mukadam, J. Issac, S. Birchfield, D. Fox, B. Boots, and N. Ratliff, "Rmpflow: A geometric framework for generation of multitask motion policies," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, pp. 968–987, 2021.

[8] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Autonomous robots*, vol. 34, pp. 189–206, 2013.

[9] J. A. Starek, J. V. Gomez, E. Schmerling, L. Janson, L. Moreno, and M. Pavone, "An asymptotically-optimal sampling-based algorithm for bi-directional motion planning," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 2072–2078.