A Unified Approach to Obstacle Avoidance and Motion Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

A dynamical system based motion representation for obstacle avoidance and motion learning is proposed. The obstacle avoidance problem can be inverted to enforce that the flow remains enclosed within a given volume. A robot arm can be controlled by using the Γ-field in combination with the converging dynamical system. The closed-form model is extended to time-varying environments, i.e., moving, expanding and shrinking obstacles. This is applied to an autonomous robot (QOLO) in a dynamic crowd in the center of Lausanne. Using Gaussian Mixture Regression (GMR) motion can be learned by describing them as a combination of local rotations. The motion can be further refined to create a safe invariant set within the obstacles' hull.

1 Introduction

Robots navigating in human-inhabited, unstructured environments have to plan or learn the initial path in advance, but they encounter disturbances constantly. In milliseconds a flexible, yet safe control scheme must take the right decisions to avoid collisions.

Motion learning and collision avoidance is often regarded as two independent problems, i.e., [1]–[3] Recent approaches try to combine these two. Global planning through probabilistic road map (PRM) was extended to dynamic environments through rapid adaptation in [4], but this results in a loss of convergence. Control barrier functions (CBFs) combined with Lyapunov Functions were united through the use of quadratic programming (QP) to create collision-free paths. The QP methods extend the initial controller but are prone to not finding a feasible solution in real-time [5], or introducing a history dependence [6]. A method to modulate initially learned (nonlinear) motion to avoid collisions was introduced in [7], but the approach cannot ensure the absence of local minima in closed-form.

We propose a novel approach to unifying learning and obstacle avoidance. On the one hand, the unification allows elevating similarities of the two problems by interpreting both as a modulation of a desired motion towards a goal. On the other hand, the combined approach allows combining learning and avoidance for safe navigation in real-world scenarios while still ensuring convergence constraints.

2 Obstacle Avoidance Formulation

Closed-form dynamical systems (DS) have proven suitable for dynamic applications as no re-planning is required. A general dynamical system can be written as a function of the state ξ as:

$$\dot{\xi} = \mathbf{f}(t, \xi) \tag{1}$$

2.1 Obstacle Description

Each obstacle has a continuous distance function $\Gamma(\xi) : \mathbb{R}^d \mapsto \mathbb{R}_{\geq 0}$, which allows to distinguish free points ($\Gamma > 1$), boundary points ($\Gamma = 1$), and interior points ($0 < \Gamma < 1$).

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

Additionally a reference point ξ_i^r is chosen within its boundaries. This allows to define the reference direction towards the obstacle as $\mathbf{r}_o(\xi) = (\xi - \xi_i^r)/\|\xi - \xi_i^r\|$.

2.2 Obstacle Avoidance through Modulation

In [8], real-time obstacle avoidance is obtained by applying a dynamic modulation matrix to a dynamical system $\mathbf{f}(\xi)$:

$$\dot{\xi} = \mathbf{M}(\xi)\mathbf{f}(\xi) \qquad \text{with} \quad \mathbf{f}(\xi) = k(\xi - \xi^a) \tag{2}$$

Describing the obstacle avoidance as a modulation of the initial dynamics ensured that attractors are conserved, i.e. $\mathbf{0} = \mathbf{M}(\xi)\mathbf{0}$. Additionally, no spurious (new) attractors are introduced, as long as the matrix $\mathbf{M}(\xi)$ has full rank.
We will focus on motion with a clearly defined goal (i.e. attractor ξ^a). The scaling parameter k introduces change with respect to time. It is of unit s^{-1}.

2.2.1 Fluid Dynamic Inspired Compression and Stretching

The potential (laminar) flow of an incompressible fluid around a cylinder is a known problem in fluid dynamics with known closed-form description It will serve as a basis for the obstacle avoidance algorithm. Similarly to the potential flow, the vector field is scaled in tangent and radial direction. Hence, the modulation matrix is defined as $\mathbf{M}(\xi) = \mathbf{E}(\xi)\mathbf{D}(\xi)\mathbf{E}(\xi)^{-1}$, a function of the decomposition matrix $\mathbf{E}(\xi)$ and the diagonal scaling matrix $\mathbf{D}(\xi)$.

2.2.2 Modulation through Decoupling of Rotation and Stretching

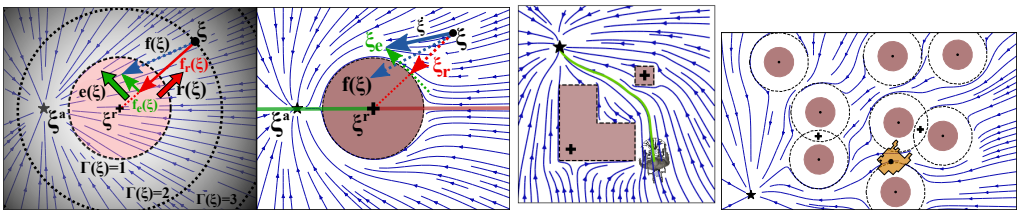
Alternatively, any vector transformation can be interpreted as a rotation with matrix $\mathbf{R}(\xi)$ and a stretching $h(x)$. This concept has been used in two dimensions for local modulation by [9]:

$$\dot{\xi} = h(\xi)\mathbf{R}(\xi)\ \mathbf{f}(\xi) \tag{3}$$

The rotation can alternatively be applied by using the *orientation-space transform* described in [10] (instead of the matrix modulation by $\mathbf{R}(\xi)$).

2.3 Multiple Obstacles

In the presence of multiple obstacles, the velocity is modulated for each obstacle individually. The final velocity is obtained by taking the weighted mean in direction space in [8]. This has been applied in Fig. 1.



(a) Γ-function (b) Avoidance of circle (c) Non-smooth edges (d) Multiple obstacles

Figure 1: Obstacle avoidance around a single and multiple obstacles.

3 Inverted Obstacle Avoidance

An autonomous robot might be in a scenario where it has boundaries which cannot be pass. This might be a wall for a wheeled robot, or it can be safety or joint limits for a robot arm. This is stated as the constraint of staying within an obstacle, where the boundary of the obstacle represents the limits of the free space.

3.1 Distance Inversion

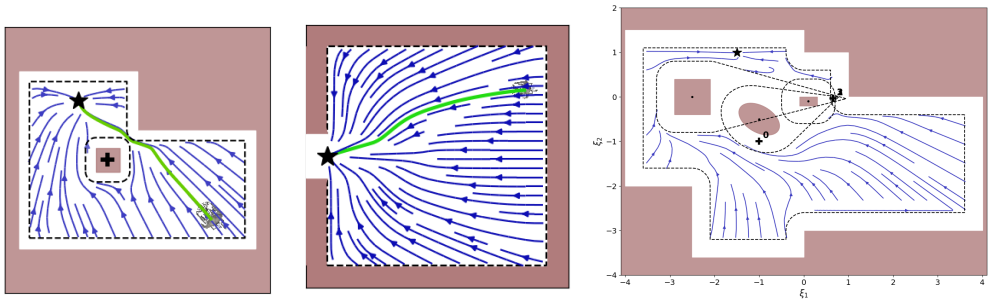
If we use the previously introduced obstacle description to denote enclosing hulls, the interior points of the classical obstacle become points of free space of the enclosing hull and vice versa. For this reason, we introduce the inverted distance function Γ^w as:

$$\Gamma^w(\xi) = 1/\Gamma^o = (R(\xi)/\|\xi - \xi^r\|)^{2p} \qquad \forall\, \mathbb{R}^d \setminus \xi^r \tag{4}$$

This allows to treat boundaries the previously defined algorithm (but the newly introduced Γ^w). It follows that convergence is still ensured.

3.2 Gaps in the Wall

In many practical scenarios a hull entails gaps or holes through which the agent enters or exits the space (e.g., door in a room). The inverted obstacle avoidance slows the agent down to zero while it is trying to approach this exit. For this reason, a *guiding reference point* for boundary obstacles is introduces. It counters this effect and nullifies the avoidance effect close to a gap (see Fig. 2).



(a) Inverted obstacle (b) Exit through gap (c) Mixed scenario

Figure 2: The inverted obstacle description ensures safe navigation within an obstacle (a) and complex environments (c). The introduction of a *guiding reference point* allows exiting through gaps of walls (b).

4 Obstacle Avoidance with a Robot Arm

The algorithm has so far been described for a point-mass. Extending it to a robots which can be encapsulated in a circular shape is done via a constant margin around all obstacles. The application to a multiple degree of freedom arm can be done by describing and evaluating it in joint-space. Alternatively, we introduce a weighted evaluation of the desired dynamics along the links which ensures a collision-free trajectory towards a desired goal.

4.1 Nominal Velocity

We assume the successful completion of the task, when the end-effector reaches the attractor, while avoiding any collisions of the robot arm with the surrounding on the way. The nominal joint velocity is therefore obtained through inverse-kinematics:

$$\dot{\mathbf{q}}^g = \mathbf{J}^{-1}(\mathbf{q})\dot{\xi}^g \tag{5}$$

with Jacobian J, joint state $\mathbf{q}$ and $\dot{\xi}^g = \mathbf{M}(\xi^{ee})\mathbf{f}(\xi^{ee})$ the velocity towards the goal at the end effector ξ^{ee}.

4.2 Danger Field and Weights

For the evaluation on the robot arm, the Γ-field is interpreted as a danger-field (similar to [11]). The danger-field is evaluated at multiple evaluation points along each link, and the corresponding weight is calculated. The weights are designed to sum up to one in order to balance converging and avoiding.

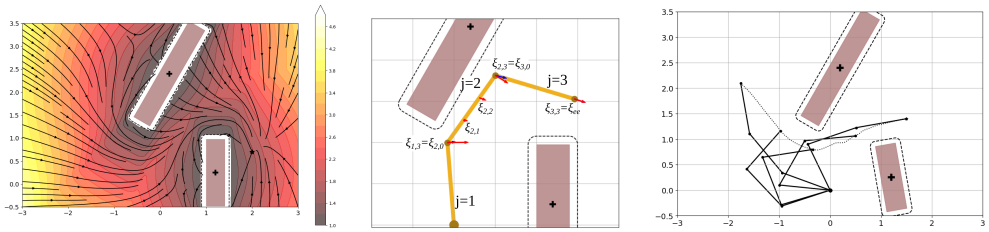
4.3 Robot Kinematics

The velocity of each joint is evaluated starting at the joint closest to the base of the robot and continuing *joint-by-joint* towards the end-effector. This allows joints higher up the chain to potentially compensate the avoidance-motion from joints which are lower in the chain (see Fig. 3).

5 Dynamic Environment and Application to Crowd

In dynamic environments with moving or deforming obstacles, the system is modulated with respect to a relative velocity:

$$\dot{\xi} = \mathbf{M}(\xi)\left(\mathbf{f}(\xi) - \dot{\xi}^o\right) + \dot{\xi}^o \qquad \text{with} \quad \dot{\xi}^o = \dot{\tilde{\xi}}^v + \dot{\tilde{\xi}}^d \tag{6}$$



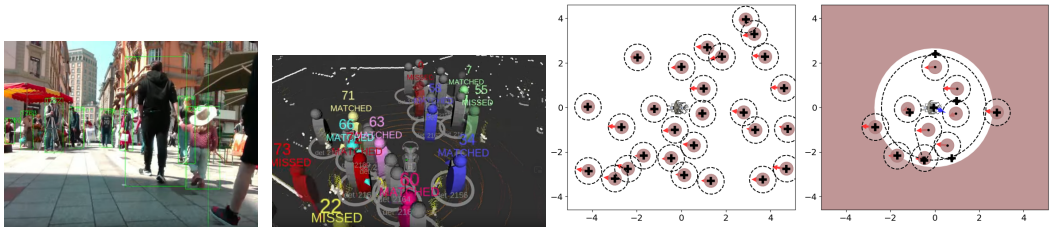
(a) Γ-danger field (b) Weighted evaluation (c) Avoidance of robot arm

Figure 3: The danger-field (a) is evaluated along the arm of the robot (b). This allows for safe navigation of multiple degree of freedom robot arm in clustered environments (c).

The relative velocity consists of the obstacle's velocity $\dot{\tilde{\xi}}^v$ and deformation $\dot{\tilde{\xi}}^d$. Impenetrability can be ensured with this approach.

5.1 Evaluation in Outdoor Crowds

A qualitative proof of concept was performed in an outdoor environment with the QOLO robot (see [12]) in the center of Lausanne, Switzerland. The location of the experiment is a small market place with large diversity in both the pedestrian's velocities and directions of movement. The robot's controller is initialized with a linear DS to reach a goal $20\,\mathrm{m}$ away from the onset position. Pedestrians are detected with a camera and LIDAR-based tracker described in [13] (see Fig. 4).
We see this as a successful test of the obstacle avoidance algorithm in areal crowd scenario. The robot managed to pass the crowd without collision during all five trials.[1]



(a) Camera view (b) Detector interpretation (c) Circular representation (d) Local environment

Figure 4: The camera and Lidar (a) are interpreted by the detector to obtain a crowd representation (b). The crowd (c) is then further reduced to a local environment (d) to increase computational speed while ensuring local convergence.

6 Extension to Motion Learning

The dynamical system frame work has shown to be suitable for motion learning. In combination with Gaussian Mixture Regression (GMR) complex, yet reactive motion patterns can be learned [2].

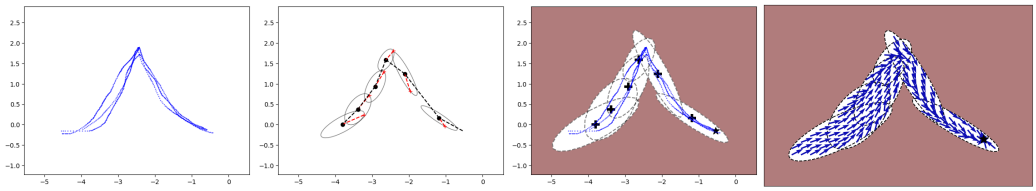
6.1 Learning the Motion as Local Rotation

A motion can be described as a rotation of an initial dynamical system (see Sec. 2). Learning a regression from recorded data using GMR, the optimal rotation at each position can be predicted. Using the directional summing described in [10], it can be ensured that the resulting vector field does not have any spurious attractors.

6.2 Learning Boundaries to Create an Invariant Set

The Gaussian Mixture Model (which was obtained by the GMR) can be used as the base of a motion boundary. Ellipses are created with the center aligned with the Gaussians, and the axes lengths are proportional to the variances of the Gaussians (see Fig. 5). The union of the set of ellipses can be interpreted as the hull of our learned environment. In combination with the learned motion, this can now ensure that the motion stays within the invariant set hull. Hence, create a safer and more reactive model which stays close to the known data.

[1] The video can be found under `https://youtu.be/3nbfwcTw8G4`



(a) Demonstration data (b) GMM (c) Learned boundary (d) Learned DS

Figure 5: The A-shape of the LASA-dataset (a) is used to obtain corresponding GMM-model (b). This allows the creation of an invariant set (c) to evaluated the dynamics (d).

References

[1] P. Long, W. Liu, and J. Pan, "Deep-learned collision avoidance policy for distributed multiagent navigation," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 656–663, 2017.

[2] S. M. Khansari-Zadeh and A. Billard, "Learning stable nonlinear dynamical systems with gaussian mixture models," *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 943–957, 2011.

[3] K. Neumann and J. J. Steil, "Learning robot motions with stable dynamical systems under diffeomorphic transformations," *Robotics and Autonomous Systems*, vol. 70, pp. 1–15, 2015.

[4] J. Vannoy and J. Xiao, "Real-time adaptive motion planning (ramp) of mobile manipulators in dynamic environments with unforeseen changes," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1199–1212, 2008.

[5] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *53rd IEEE Conference on Decision and Control*, IEEE, 2014, pp. 6271–6278.

[6] M. F. Reis, A. P. Aguiar, and P. Tabuada, "Control barrier function-based quadratic programs introduce undesirable asymptotically stable equilibria," *IEEE Control Systems Letters*, vol. 5, no. 2, pp. 731–736, 2020.

[7] S. M. Khansari-Zadeh and A. Billard, "A dynamical system approach to realtime obstacle avoidance," *Autonomous Robots*, vol. 32, no. 4, pp. 433–454, 2012.

[8] L. Huber, A. Billard, and J.-J. Slotine, "Avoidance of convex and concave obstacles with convergence ensured through contraction," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1462–1469, 2019.

[9] K. Kronander, M. Khansari, and A. Billard, "Incremental motion learning with locally modulated dynamical systems," *Robotics and Autonomous Systems*, vol. 70, pp. 52–62, 2015.

[10] L. Huber, J.-J. Slotine, and A. Billard, "Avoiding dense and dynamic obstacles in enclosed spaces: Application to moving in a simulated crowd," *arXiv e-prints*, arXiv–2105, 2021.

[11] B. Lacevic, P. Rocco, and A. M. Zanchettin, "Safety assessment and control of robotic manipulators using danger field," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1257–1270, 2013.

[12] D. F. P. Granados, H. Kadone, and K. Suzuki, "Unpowered lower-body exoskeleton with torso lifting mechanism for supporting sit-to-stand transitions," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 2755–2761.

[13] D. Jia, A. Hermans, and B. Leibe, "Dr-spaam: A spatial-attention and auto-regressive model for person detection in 2d range data," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 10 270–10 277. DOI: 10.1109/IROS45743.2020.9341689.