2021 The 21st International Conference on Control, Automation and Systems (ICCAS 2021)
Ramada Plaza Hotel, Jeju, Korea, Oct. 12∼15, 2021

# Trajectory Optimization with Geometry-Aware Singularity Avoidance for Robot Motion Planning

Luka Petrović[1]*, Filip Marić[1,2], Ivan Marković[1], Jonathan Kelly[2], Ivan Petrović[1]

[1]Laboratory for Autonomous Systems and Mobile Robotics,
University of Zagreb Faculty of Electrical Engineering and Computing,
Zagreb, Croatia
[2]Space & Terrestrial Autonomous Robotic Systems Laboratory,
University of Toronto Institute for Aerospace Studies,
Toronto, Canada
(luka.petrovic@fer.hr) * Corresponding author

**Abstract:** One of the principal challenges in robot arm motion planning is to ensure robot's agility in case of encountering unforeseeable changes during task execution. It is thus crucial to preserve the ability to move in every direction in task space, which is achieved by avoiding singularities, i.e. states of configuration space where a degree of freedom is lost. To aid in singularity avoidance, existing methods mostly rely on manipulability or dexterity indices to provide a measure of proximity to singular configurations. Recently, a novel geometry-aware singularity index was proposed that circumvents some of the failure modes inherent to manipulability and dexterity. In this paper, we propose a cost function based on this index and integrate it within a stochastic trajectory optimization framework for efficient motion planning with singularity avoidance. We compare the proposed method with existing singularity-aware motion planning techniques, demonstrating improvement in common indices such as manipulability and dexterity and showcasing the ability of the proposed method to handle collision avoidance while retaining agility of the robot arm.

**Keywords:** motion planning, singularity avoidance, manipulability, dexterity

## 1. INTRODUCTION

Motion planning is one of the key challenges in robotics. Motion planning algorithms should ensure that robot's motion through an environment is both feasible and optimal with regards to a given performance criterion. Feasibility implies that the produced robot trajectory satisfies constraints (e.g. adhering to the joint limits), while optimality often indicates either minimal execution time, path length or energy consumption. When performing a desired task, articulated robots oftentimes have additional constraints induced by the task (e.g. obstacle avoidance) which may restrict their movement in task space. Such constraints may incur non-optimal mapping of some configurations from the configuration space to the task space, resulting in singular configurations, i.e. *singularities*. Singularities hinder task space mobility, requiring particularly large joint motions which can result in perilous joint movement or task execution failure. It is crucial to preserve the ability to move in every direction in task space in order to ensure robot's agility in case of encountering unforeseeable changes during task execution. Therefore, recognizing and avoiding singular configurations during motion planning has been a topic of robotics research for decades [1], [2], [3], [4].

Singularities can be identified by observing singular values of the robot's Jacobian matrix, which is a function that maps configuration space velocities to task space velocities [5]. There exist several indices that try to capture the kinematic sensitivity by relying on singular values of the Jacobian. The manipulability index [1] measures the volume of the manipulability ellipsoid and is perhaps the most widely employed way of recognizing singular configurations. Another commonly used measure is dexterity index [6] that calculates the ratio between the longest and shortest manipulability ellipsoid axes. These indices have been successfully integrated with inverse kinematics [3], [4] and motion planning [7], [8], [9], [10] by looking to maximize the volume of the manipulability ellipsoid in order to avoid singular configurations. Nevertheless, both the manipulability and dexterity index possess shortcomings due to their inability to fully capture the geometry of the manipulability ellipsoid. Redundancy resolution approaches such as [11] do not explicitly consider manipulability or dexterity indices and have long been used for singularity avoidance. However, they can be prone to algorithmic singularities caused by conflicting objectives and are therefore employed only for simple task hierarchies. Rozo et al. used Stein divergence to derive a geometry-aware similarity measure between two manipulability ellipsoids [12] and employed it in the context of task learning. A recently proposed geometry-aware singularity index [13] relies on a Riemannian metric on the manifold of symmetric positive definite matrices to measure the proximity of a configuration to singularity, bypassing the failure modes inherent to manipulability and dexterity.

The aforementioned geometry-aware indices that avoid the failure modes inherent to manipulability and dexterity are yet to be exploited in the context of motion planning. In order to efficiently tackle the robot motion planning problem while considering different objectives and task constraints, a considerable research effort has been dedicated to trajectory optimization methods [14], [15], [16], [17]. These methods start with a naive initial trajectory (e.g. straight line in configuration space) and optimize it by minimizing a given objective criteria that may include a collision avoidance costs and other task-related objectives and constraints.

In this paper, we derive a stochastic trajectory optimization framework with geometry-aware singularity avoidance that circumvents the failure modes of the commonly used indices. We make the following contributions:

(i) we propose a cost function that features a geometry-aware singularity index that uses a Riemannian metric to provide a measure of proximity to singular configurations,

(ii) we integrate the proposed cost function within a stochastic trajectory optimization framework for efficient motion planning, and

(iii) we compare the proposed approach to existing singularity avoidance and manipulability maximization techniques, demonstrating improvement in keeping trajectory states away from singularities.

## 2. BACKGROUND

In this paper we consider serial robotic manipulators consisting of $n$ actuated joints. First we define the robot configuration at a given time instant $t_i$ as a set of joint positions $\mathbf{q_i}$ in the configuration space $\mathbf{q_i} \in \mathcal{Q} \subset \mathbb{R}^n$ which is comprised of all feasible joint positions. Similarly, robot's end-effector poses $\mathbf{x}$ construct the task space $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^p$. Robot's forward kinematics function is defined as the following nonlinear mapping

$$\mathbf{f} : \mathcal{Q} \to \mathcal{X}.$$

Conversely, the robot's inverse kinematics is defined as

$$\mathbf{f^{-1}} : \mathcal{X} \to \mathcal{Q}.$$

The Jacobian matrix is obtained by finding the gradient of $\mathbf{f}$ with respect to joint values $\mathbf{q_i}$,

$$\mathbf{J} = \frac{\partial \mathbf{f}}{\partial \mathbf{q_i}}\,.$$

The Jacobian matrix defines kinematic relationship between configuration and task space velocities [5]

$$\dot{\mathbf{x_i}} = \mathbf{J}\dot{\mathbf{q_i}}, \tag{1}$$

where $\mathbf{J}$ is the Jacobian matrix computed at configuration $\mathbf{q_i}$, while $\dot{\mathbf{q_i}}$ and $\dot{\mathbf{x}}_i$ are the joint and task space velocities.

### 2.1 Manipulability ellipsoid

The mapping of a unit sphere in the space of joint velocities $||\dot{\mathbf{q_i}}||^2 = 1$ to the task velocity space can be defined as

$$||\dot{\mathbf{q_i}}||^2 = \dot{\mathbf{x}}^T \left(\mathbf{J}\mathbf{J^T}\right)^{-1} \dot{\mathbf{x}}\,. \tag{2}$$

The conditioning of the matrix $\mathbf{J}\mathbf{J^T}$ governs the scaling of joint velocities to the task space. Therefore, the eigenvalues of this matrix provide information about directional mobility of the robot arm in some configuration $\mathbf{q_i}$. The pertaining matrix $\mathbf{J}\mathbf{J^T}$ in Eq. (2) is symmetric positive semi-definite (SPD) and is termed the manipulability ellipsoid of the end-effector [1]

$$\mathbf{M}\left(\mathbf{q_i}\right) = \mathbf{J}\mathbf{J^T}. \tag{3}$$

By computing singular value decomposition (SVD) of $\mathbf{J} = \mathbf{U\Sigma V^T}$, we can obtain the principal axes $\sigma_0\mathbf{u_0}, \sigma_\mathbf{1}\mathbf{u_1}, \ldots, \sigma_\mathbf{p}\mathbf{u_p}$. The length and orientation of these axes determine directions in which higher values of task space velocities can be achieved.

The eigenvalues of manipulability ellipsoid $\mathbf{M}$ are equal to the squared singular values $\sigma^2$ of the Jacobian matrix $\mathbf{J}$. This implies that when one or more eigenvalues of $\mathbf{M}$ become zero, the Jacobian matrix is poorly conditioned and thus non-invertible, i.e. the robot is in a singular configuration.

### 2.2 Manipulability index

As a consequence of the manipulability ellipsoid eigenvalues corresponding to the squared singular values of the Jacobian, all configurations for which the Jacobian matrix is non-invertible have a corresponding manipulability ellipsoid with at least one degenarate axis. Movements of the robot end-effector that begin from these singular configurations are susceptible to high joint velocities and unwanted dynamic characteristics. To detect singular configurations, we can observe singular values of the Jacobian matrix, which is a basis of several developed indices that attempt to provide a measure of proximity of a configuration to singularity. One of the indices commonly used for singularity avoidance and improving kinematic sensitivity is the so-called manipulability index, defined as

$$m = \sqrt{\det(\mathbf{J}\mathbf{J^T})}\,. \tag{4}$$

The value of $m$ is in proportion to the volume of the manipulability ellipsoid defined in Eq. (3), providing an intuitive geometric interpretation of the manipulability index. Singularities imply that the corresponding manipulability ellipsoids have at least one degenerate axis, resulting in zero volume manipulability ellipsoids. Consequently, manipulability index of singularities is zero and can thus be used for singularity detection in inverse kinematics [4] and trajectory optimization [9]. However, there exists cases when manipulability index is ineffective in detecting configurations that are close to being singular. This stems form the fact that the volume of the manipulability ellipsoid can remain relatively large near singularities if other axes of the manipulability ellipsoid are particularly elongated.

### 2.3 Dexterity index

Another established index commonly used for singularity detection and avoidance is the dexterity index [6], defined as

$$\kappa = \frac{\sigma_{max}}{\sigma_{min}}\,, \tag{5}$$

where $\sigma_{max}$ and $\sigma_{min}$ denote the maximal and minimal singular values of the Jacobian matrix. It follows that the dexterity index is essentially a measure of difference in relative lengths between the longest and shortest axis of the manipulability ellipsoid, where large values indicate an elongated shape. When the length of one axis of the manipulability ellipsoid is near-zero, the dexterity index may be high and can thus in most cases be used for singularity detection. However, the drawback of the dexterity index is its incapability to inform about the scale of the ellipsoid. This problem manifests when a configuration is far from singularity but has one axis of a particularly large length – the dexterity index cannot distinguish such configuration from a singular one.

## 3. STOCHASTIC TRAJECTORY OPTIMIZATION WITH SINGULARITY AVOIDANCE

Consider a joint space trajectory $\boldsymbol{q}$ as a function that maps time instances $0 \leq t \leq T$ to configurations $\boldsymbol{q}(t)$. We can then formulate the motion planning problem as trajectory optimization

$$\underset{\boldsymbol{q}(t)}{\text{minimize}} \quad F\left[\boldsymbol{q}(t)\right] + \sigma G\left[\boldsymbol{q}(t)\right], \tag{6}$$

where $F\left[\boldsymbol{q}(t)\right]$ is the cost functional that encodes trajectory smoothness, $G\left[\boldsymbol{q}(t)\right]$ is a state-dependent cost functional that can pertain to singularity avoidance, collision avoidance and various task-dependent constraints, and $\sigma$ is a cost weighting parameter. In the following subsections we describe the smoothness cost, propose a novel geometry-aware singularity avoidance cost functional and present the employed trajectory optimization procedure.

### 3.1 Smoothness cost

Closely following [17], we consider robot's trajectory as a sample from a continuous-time Gaussian Process (GP) $\boldsymbol{q}(t) \sim \mathcal{GP}(\boldsymbol{\mu}(t), \mathbf{K}(t,t'))$, with mean $\boldsymbol{\mu}(t)$ and covariance $\mathbf{K}(t,t')$. We consider GPs generated by a linear time-varying stochastic differential equation (LTV-SDE)

$$\dot{\boldsymbol{q}}(t) = \mathbf{A}(t)\boldsymbol{q}(t) + \boldsymbol{u}(t) + \mathbf{F}(t)\boldsymbol{w}(t), \tag{7}$$

where $\boldsymbol{A}$ and $\boldsymbol{F}$ are system matrices, $\mathbf{u}$ is a known control input and $\boldsymbol{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q_c})$ is a white noise process. While the defined GP is continuous, we can parameterize it with a sparse set of *support states* at discrete time instances $\boldsymbol{q} = \left[\boldsymbol{q}_0 \ldots \boldsymbol{q}_N\right]^T$. An exponential prior distribution resulting from the system in Eq. (7) can be defined, with the mean $\boldsymbol{\mu}$ and kernel $\mathbf{K}$

$$p(\boldsymbol{q}) \propto \exp\{-\frac{1}{2}\|\boldsymbol{q} - \boldsymbol{\mu}\|^2_{\mathbf{K}}\}. \tag{8}$$

The GP prior defined in Eq. (8) penalizes the deviation of the trajectory from the mean and thus encourages trajectory smoothness. The negative logarithm of this prior distribution serves as the smoothness cost functional in the trajectory optimization objective defined in Eq. (6)

$$F\left[\boldsymbol{q}(t)\right] = \|\boldsymbol{q} - \boldsymbol{\mu}\|^2_{\mathbf{K}}. \tag{9}$$

We use a constant-velocity motion model prior, and the mean $\boldsymbol{\mu}$ is then calculated as a temporally equidistant straight line, while derivation of the covariance matrix $\mathbf{K}$ is given in [17].

### 3.2 A geometry-aware singularity avoidance cost

The manipulability and dexterity indices are commonly employed for singularity detection and avoidance. However, as described in Section 2, both of them have drawbacks that make them inapplicable in certain cases. In this paper, we will therefore utilize a recently developed geometry-aware singularity index [13] that is defined as

$$\xi := \left\| \log\left(\boldsymbol{\Sigma}^{-\frac{1}{2}}\mathbf{M}\boldsymbol{\Sigma}^{-\frac{1}{2}}\right)\right\|^2_F\,, \tag{10}$$

where $\mathbf{M}$ is the manipulability ellipsoid of a given configuration, $\boldsymbol{\Sigma}$ is an arbitrary reference ellipsoid and $\|\cdot\|_F$ is the Frobenius norm. The index defined in Eq. (10) is essentially squared Riemannian distance between a reference ellipsoid $\boldsymbol{\Sigma}$ and manipulability ellipsoid $M$. For a more in-depth treatment of Riemannian geometry in the context of manipulability ellipsoids and singularity avoidance the reader is referred to [18], [13]. By carefully choosing the reference ellipsoid $\boldsymbol{\Sigma}$ in Eq. (10), the singularity index gains the notion of measuring proximity of a given configuration to a singular one.

We select the reference ellipsoid $\boldsymbol{\Sigma}$ to be a sphere with the radius greater than the length of the longest manipulability ellipsoid axis, thus enclosing the manipulability ellipsoid

$$\boldsymbol{\Sigma} = k\boldsymbol{I},\; k \geq \sigma^2_{\max}\,. \tag{11}$$

This choice of $k$ results in $\xi$ decreasing with the increase of the singular values of the Jacobian, since the manipulability ellipsoid is always inside the reference ellipsoid $\boldsymbol{\Sigma}$. Due to the symmetry of the sphere, the geometry-aware singularity index is invariant to different orientation of the manipulability ellipsoid [13]. This trait is advantageous in singularity avoidance since the orientation of the manipulability ellipsoid does result in changes to singular values of the Jacobian. When a robot arm changes its configuration, the corresponding manipulability ellipsoid obtains a different size, shape and orientation. In contrast to the manipulability and dexterity indices, the geometry-aware singularity index encodes the ellipsoid's axis lengths, shape and orientation. This property allows for circumventing issues that arise when applying the conventional indices.

To integrate singularity avoidance with trajectory optimization, we propose a cost function that takes into account the value of the geometry-aware singularity index. Combining Eq. (10) and Eq. (11) and summing over every support state $\boldsymbol{q}_i$, we obtain

$$G\left[\boldsymbol{q}(t)\right] = \sum_{i=0}^{N} \left\| \log\left(k^{-\frac{1}{4}}M(\boldsymbol{q}_i)\right)\right\|^2_F\,. \tag{12}$$

The cost function in Eq. (12) should guide the trajectory optimization away from regions of configuration space containing singularities without being prone to failure modes of manipulability and dexterity indices.

### 3.3 Optimizing the trajectory

Combining the initial trajectory optimization formulation in Eq. (6) with cost functions for smoothness in Eq. (9) and singularity avoidance in Eq. (12) we arrive to the following optimization problem

$$\boldsymbol{q}^* = \underset{\boldsymbol{q}}{\operatorname{argmin}} \|\boldsymbol{q} - \boldsymbol{\mu}\|^2_{\mathbf{K}} + \sigma \sum_{i=0}^{N} \left\| \log\left(k^{-\frac{1}{4}} M(\boldsymbol{q}_i)\right) \right\|^2_F , \tag{13}$$

Since our objective function is highly nonlinear, we opt for stochastic optimization procedure that may be able to overcome the local minima problem encountered by gradient-based approaches. Similarly to STOMP [15], we estimate the gradient of our objective function by sampling $K$ noisy trajectories $\boldsymbol{q}^k$ from the prior distribution and forming a probability-weighted convex combination $\delta\hat{\boldsymbol{q}}$ that updates the trajectory in a cost-minimizing manner

$$\delta\hat{\boldsymbol{q}} = \sum_{k=1}^{K} P(\boldsymbol{q}^k)(\boldsymbol{q}^k - \boldsymbol{\mu}), \tag{14}$$

where $P(\boldsymbol{q}^k)$ is a probability metric

$$P(\boldsymbol{q}^k) = \exp\{-\frac{1}{\lambda} G\left[\boldsymbol{q^k}\right]\} \tag{15}$$

with $\lambda$ being sensitivity parameter that we choose as a constant $\lambda = 0.1$. The update rule is applied iteratively until convergence and is given as

$$\boldsymbol{q} \leftarrow \boldsymbol{q} + \mathbf{K}\delta\hat{\boldsymbol{q}}. \tag{16}$$

Multiplication with covariance matrix $\mathbf{K}$ ensures that the update is smooth due to the underlying constant-velocity motion prior. For a more in-depth treatment, the reader is referred to [15]. Note that many different costs can be summed with the singularity avoidance cost function to form $G\left[\boldsymbol{q^k}\right]$ and the update rule would work in the same manner.

## 4. EXPERIMENTAL RESULTS

We evaluated the proposed method in simulations on two different motion planning scenarios. First scenario featured a reaching task in an empty environment, where the aim was to reach a desired goal state while successfully avoiding singularities. In Section 4.1 we provide a quantitative performance analysis of the proposed method on the reaching task scenario and comparison with two state-of-the-art approaches to singularity avoidance, namely [9] and [11]. Second scenario featured a collision avoidance task, where the aim was to reach the desired goal state without colliding with the obstacle while avoiding singularities. In Section 4.2 we provide a qualitative example of the second scenario that demonstrated the ability of the proposed method to find a collision free trajectory while staying far from singularities at every trajectory state. Experimental evaluation was performed on a laptop with a 2.8-GHz Intel Core i7-7700HQ CPU and 16 GB of RAM.

### 4.1 Reaching task

The reaching task benchmark was performed using three common robot arms: the six DOF Universal Robots UR10, the seven DOF Kinova Jaco 2 with spherical joints and the seven DOF Kuka LBR IIWA 14. For each arm, the evaluation consisted of 100 different tasks with start and goal configurations randomly sampled from the robot's configuration space. We solved each task with the method proposed in this paper, the trajectory optimization method with manipulability optimization proposed in [9] and the singularity-robust kinematic control method proposed in [11].

We compared performance of these methods on three common singularity-related indices which served as performance metrics: the minimum eigenvalue $\sigma_{min}$ of the manipulability ellipsoid $\mathbf{M}(\mathbf{q})$ defined in Eq. (3), the manipulability index $m$ defined in Eq. (4) and the dexterity index $\kappa$ defined in Eq. (5). For every trajectory, we evaluated the performance metric at every discrete time instant and calculated the average value during the whole trajectory. Additionally, we provide performance analysis for a simple straight-line trajectory in configuration space as the baseline. We also compare execution times of evaluated methods, except for the baseline, as the straight-line trajectory can be straightforwardly calculated by linear interpolation in sub-milisecond time. Examples of trajectories obtained by all the methods for each of the employed robot arms are depicted in Fig. 1. For these example trajectories, Fig. 1 also shows the corresponding minimum eigenvalue $\sigma_{min}$ of the manipulability ellipsoid $\mathbf{M}(\mathbf{q})$ during trajectory execution. Results of the performance evaluation are shown in Table 1. Note that for minimum eigenvalue $\sigma_{min}$ and manipulability index $m$ higher values imply better singularity avoidance, while for dexterity index $\kappa$ lower values are better.

As expected, each of the evaluated methods compared favorably to the baseline in every performance metric. The trajectory optimization method with manipulability optimization proposed in [9] attained the lowest average execution time for each of the utilized robot arms. This was expected due to its underlying gradient-based optimization. However, it is clear that thr proposed method consistently achieved the best overall performance on the conducted reaching tasks. The coupling of the Riemannian-based geometry-aware singularity index and a stochastic trajectory optimization method suitable for optimizing nonlinear cost functions proved to work notably well for singularity avoidance.

### 4.2 Collision avoidance

The collision avoidance task was performed using the seven DOF Kuka LBR IIWA 14 robot arm in an envi-

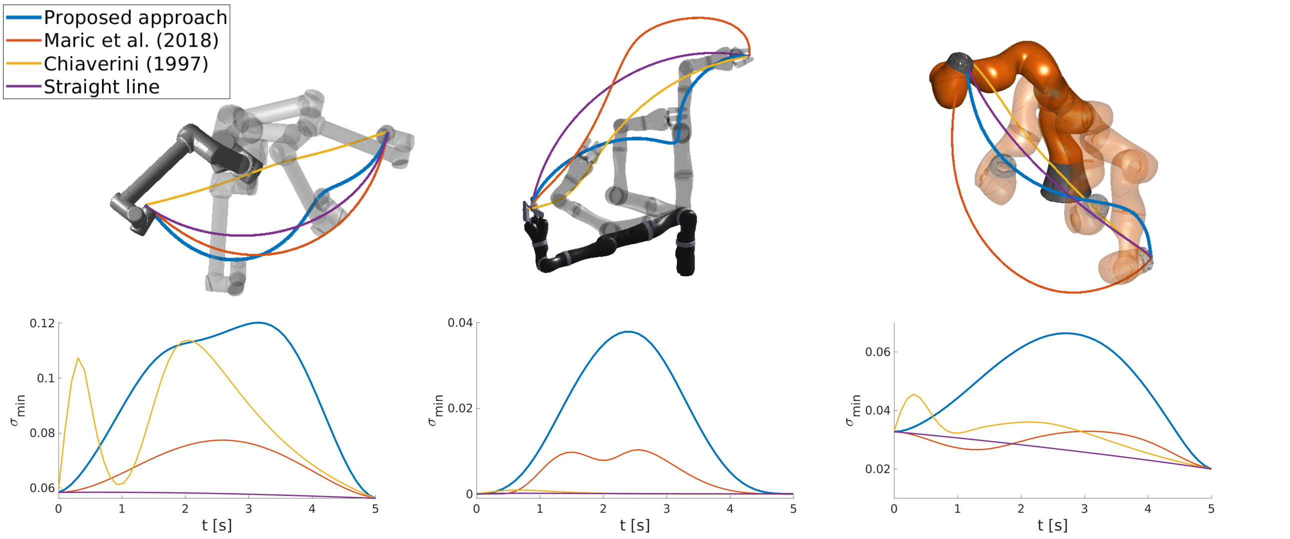

Fig. 1.: Example trajectories obtained by each method on the reaching task. The top figures show the end effector trajectories in robot's task space, while bottom figures show the corresponding minimum eigenvalue of the manipulability ellipsoid during trajectory execution.

Table 1.: Comparison of the proposed approach and several other methods on the reaching task benchmark for three different robot arms. We compare the methods with three different measures, namely the average minimal singular value, the average manipulability and the average dexterity.

| | UR-10 | | | | Kinova Jaco 2 | | | | Kuka IIWA 7 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\sigma_{min}$ | $m$ | $\kappa$ | Time | $\sigma_{min}$ | $m$ | $\kappa$ | Time | $\sigma_{min}$ | $m$ | $\kappa$ | Time |
| Proposed approach | **0.044** | **0.130** | **2.339** | 114 | **0.027** | **0.111** | **2.364** | 163 | 0.025 | **0.076** | **2.404** | 126 |
| Maric et al. (2019), [9] | 0.034 | 0.111 | 2.518 | **66** | 0.025 | 0.099 | 2.380 | **72** | 0.017 | 0.057 | 2.621 | **69** |
| Chiaverini (1997), [11] | 0.030 | 0.072 | 2.801 | 129 | 0.023 | 0.073 | 2.440 | 182 | **0.026** | 0.061 | 2.421 | 138 |
| Straight line | 0.031 | 0.096 | 2.716 | / | 0.021 | 0.090 | 2.583 | / | 0.013 | 0.049 | 2.851 | / |

ronment featuring a table. We picked a start state under the table and the goal state above the table in order to require collision avoidance to successfully solve the task. We ran trajectory optimization twice: first, including both collision avoidance and our geometry-aware singularity avoidance in the cost function, and second, only considering collision avoidance. For collision avoidance cost, we utilized precomputed signed distance field and hinge loss, similarly to [17]. Examples of trajectories obtained by these two runs are depicted in Fig. 2, as well as the corresponding minimum eigenvalue $\sigma_{min}$ of the manipulability ellipsoid $\mathbf{M}(\mathbf{q})$, manipulability index $m$ and dexterity index $\kappa$ during trajectory execution.

It can be seen that the inclusion of the singularity avoidance cost led to the robot arm staying far away from singularities at every trajectory state, while retaining the ability of the motion planning method to find a collision-free trajectory. Better values of manipulability and dexterity indices also suggest that robot's agility would be ensured in case of unpredictable changes.

## 5. CONCLUSION AND FUTURE WORK

In this paper we have presented a trajectory optimization method for robot arm motion planning that successfully avoids singularities. First we derived a cost function

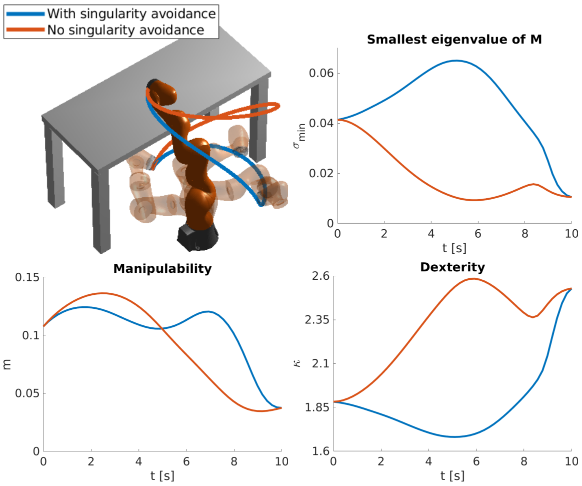

Fig. 2.: Trajectories obtained on the collision avoidance task. The top left figure shows the end effector trajectories in the robot's task space, while other figures correspond to the minimum eigenvalue of the manipulability ellipsoid, manipulability index and dexterity index during trajectory execution for trajectories obtained by the proposed stochastic trajectory optimization method with and without singularity avoidance optimization.

that penalizes proximity to singular configurations by relying on a recently proposed geometry-aware singularity index based on a Riemannian metric. Then we coupled the proposed cost function with a stochastic trajectory optimization method to efficiently obtain robot trajectories that maximize the distance from singular regions while reaching the desired task. We compared the proposed method to existing singularity avoidance and manipulability maximization techniques, demonstrating improvement in common singularity indices such as manipulability and dexterity. We also demonstrated the ability of the proposed method to produce collision-free motion plans that stay far from singularities.

In future work, it would be interesting to test the proposed approach in real environments and to exploit the robot's mobility gained by staying far from singularities in order to adapt and replan the robot motion in case of unexpected changes during the task execution.

## ACKNOWLEDGEMENT

This research has been supported by the European Regional Development Fund under the grant KK.01.1.1.01.0009 (DATACROSS).

## REFERENCES

[1] T. Yoshikawa, "Manipulability of robotic mechanisms," *The international journal of Robotics Research*, vol. 4, no. 2, pp.3-9, 1985.

[2] V.D. Tourassis and M.H. Ang Jr., "Identification and analysis of robot manipulator singularities," *The International Journal of Robotics Research*, vol. 11, no. 3, pp.248-259, 1992.

[3] G. Marani, J. Kim, J. Yuh and W.K. Chung, "A real-time approach for singularity avoidance in resolved motion rate control of robotic manipulators," in *Proc. of IEEE International Conference on Robotics and Automation*, pp. 1973-1978, 2002.

[4] K. Dufour and W. Suleiman, "On integrating manipulability index into inverse kinematics solver," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 6967-6972, 2017.

[5] L. Sciavicco and B. Siciliano, "Modelling and Control of Robot Manipulators", *Springer Science & Business Media*, 2012.

[6] J. K. Salisbury and J. J. Craig, "Articulated hands: Force control and kinematic issues," *The International Journal of Robotics Research*, vol. 1, no. 1, pp. 4-17, 1982.

[7] L. Guilamo, J. Kuffner, K. Nishiwaki and S. Kagami, "Manipulability optimization for trajectory generation," in *Proc. of IEEE International Conference on Robotics and Automation*, pp. 2107-2022, 2006.

[8] Y. Zhang, D. Guo, K. Li and J. Li, "Manipulability-maximizing self-motion planning and control of redundant manipulators with experimental validation," in *Proc. of IEEE International Conference on Mechatronics and Automation*, pp. 1829-1834, 2012.

[9] F. Marić, O. Limoyo, L. Petrović, T. Ablett, I. Petrović and J. Kelly, "Fast manipulability maximization using continuous-time trajectory optimization," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 8258-8264, 2019.

[10] J. Haviland and P. Corke, "A purely-reactive manipulability-maximising motion controller," *arXiv preprint arXiv:2002.11901*, 2020.

[11] S. Chiaverini, "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 3, pp. 398-410, 1997.

[12] L. Rozo, N. Jaquier, S. Calinon and D. G. Caldwell, "Learning manipulability ellipsoids for task compatibility in robot manipulation," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3183-3189, 2017.

[13] F. Marić, L. Petrović, M. Guberina, J. Kelly and I. Petrović, "A Riemannian Metric for Geometry-Aware Singularity Avoidance by Articulated Robots," *arXiv preprint arXiv:2103.05362*, 2021.

[14] N. Ratliff, M. Zucker, J. A. Bagnell and S. Srinivasa, "CHOMP: Gradient optimization techniques for efficient motion planning," in *Proc. of IEEE International Conference on Robotics and Automation*, pp. 489-494, 2009.

[15] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor and S. Schaal, "STOMP: Stochastic trajectory optimization for motion planning," in *Proc. of IEEE International Conference on Robotics and Automation*, pp. 4569-4574, 2011.

[16] J. Schulman, Y. Duan, J. Ho, A. Lee, I. Awwal, H. Bradlow, J. Pan, S. Patil, K. Goldberg and P. Abbeel, "Motion planning with sequential convex optimization and convex collision checking," *The International Journal of Robotics Research*, vol. 33, no. 9, pp. 1251-1270, 2014.

[17] M. Mukadam, J. Dong, Y. Xinyan, F. Dellaert and B. Boots, "Continuous-time Gaussian process motion planning via probabilistic inference," *The International Journal of Robotics Research*, vol. 37, no. 11, pp. 1319-1340, 2018.

[18] N. Jaquier, L. Rozo, D. G. Caldwell and S. Calinon, "Geometry-aware manipulability learning, tracking, and transfer," *The International Journal of Robotics Research*, vol. 40, no. 2-3, pp. 624-650, 2021.