A Structured Approach for Uncertain Transformations Trees

Marco Sewtz, Lukas Burkhard, Xiaozhou Luo, Leon Dorscht, Rudolph Triebel
Institute of Robotics and Mechatronics
German Aerospace Center (DLR)
Wessling, Germany
{firstname.lastname}@dlr.de

***Abstract*—In the field of robotics, ensuring precise representation of spatial transformations is imperative for maintaining reliable system performance. However, conventional approaches often prove inadequate due to their failure to consider internal inaccuracies in the robot and environmental factors. In the context of robotic systems, deviations from nominal transformations arise from various sources such as sensor decalibration, inaccuracies in joint positions, deformations induced by mechanical stress, and gravitational influences, among other contributing factors. The same applies to environmental uncertainties, where the registered poses of objects and landmarks suffer from limitations in the perception methods. This paper advocates for a paradigm shift by introducing a framework that incorporates uncertainty into transformation trees, utilizing Lie Algebra for a consistent computation. Our approach addresses the aforementioned challenges, providing a realistic and robust representation of transformations. We demonstrate the applicability and efficacy of our framework through real-world examples.**

***Index Terms*—robotics, transformation tree, uncertainty modeling, Lie Algebra**

I. Introduction

In the dynamic landscape of robotics, accurately representing spatial transformations is pivotal for reliable system performance. Conventional methods, which treat provided transformations as precise and deterministic, face difficulties in coping with inherent inaccuracies within the system and environmental complexities. This paper underscores the critical need for inaccuracies-aware spatial representations in robotics, often denoted as scene graphs. These representations allow modeling not only the spatial relationships in a robot-environment system but also our missing knowledge about it.

An illustrative instance can be found in the distinction between a robotic arm's repetition accuracy, which signifies its capability to consistently reach the same point in a workspace, and the robot's absolute accuracy. There, the first can be assumed to be "exact" for conventional robotic systems. However, the error of the latter can be higher by several orders of magnitude, motivating the modeling of the error. Position measurements constrained by both physical limitations and environmental influences, frequently fall short of the requisite precision. This constraint becomes especially critical in applications requiring high accuracy, such as surgical robotics.

An additional example is the process of registering a robot with respect to its environment, a task achieved through either an inaugural calibration procedure or by means of the navigation implemented in mobile robotic systems.

Interestingly, various scholarly works have considered robot uncertainty within specific domains, such as the kinematic structure or autonomous navigation components.However, there is limited progress in combining these several domains into one single representation like a scene graph to have a unified consideration of inaccuracy-aware spatial relations. Conventional approaches that disregard uncertainty in scene graphs fall short in capturing the intricacies of real-world scenarios.

This paper advocates for a paradigm shift by introducing a framework that incorporates uncertainty into scene graphs, offering a more realistic and robust representation of transformations. By addressing challenges posed by both robot internal inaccuracies and the uncertainty of the robot's interaction with the environment, our approach aims to enhance the reliability and performance of robotic systems in practical applications.

We use the following terminology in this paper: Robotic systems can be subject to errors that cause *inaccurate* pose calculations, either within the system or with respect to its environment. A common simplification is to model such inaccuracies in a probabilistic way, thus subjecting nominal relative poses to an additional *uncertainty*. For a multitude of robotic applications, such uncertainty is modeled as a *zero-mean normal distribution*, thus an uncertain pose consists of a nominal pose and a covariance matrix. Generally, this simplification trades the exact representation of robotic errors for the availability of powerful mathematical tools and is thus well established in the robotic community. We adopt this error modeling as well, which allows us to immediately integrate the probabilistic pose information from other software components into our scene graph.

II. Related Work

Accurately describing the spatial relationships of a robot and its environment is a key aspect of robotics specifically

This work was supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy (StMWi) by means of the project SMiLE2gether (LABAY102).

and mechanical mechanisms generally.

Commencing with the early explorations in formulating a framework for kinematics in mechanical structures [1], [2], the field witnessed significant strides with one of the pivotal works by Denavit and Hartenberg [3]. In this groundbreaking contribution, the authors devised a structured yet elegant methodology to comprehensively describe the chain of transformations associated with robotic arms. Subsequent endeavors augmented the toolbox of robot kinematics representation, for example by considering the underlying Lie-Algebra of spacial transformations [4]. Our recent work [5][1] provides a kinematic robot description that allows to consider the inaccuracies from joint position measurements, mechanical stress-induced deformations, and gravitational influences in a probabilistic manner.

In the field of robotic navigation, many approaches already consider the uncertainty of relative transformations, especially in the area of SLAM where e.g. [6] or [7] use the covariance or information matrix, respectively, to weigh different spatial transformations in a graph optimization.

The interaction of a robot with objects in its environment, specifically the uncertainties inherent in the workspace, has been investigated in [8]. Additionally, notable strides have been made in recent research towards modeling the uncertainty embedded within the perception process of classical [9] and deep-learning-based [10][1] methods.

Finally, the hand-eye-calibration of a robot is nothing else but an additional transformation between the real and the nominal robot geometry, and can thus also be subject to inaccuracies, as discussed by [11].

In the end, all these sub-fields of robotics provide a multitude of different types of spatial transformations, where potentially all of them are subjected to errors which are being modeled as uncertainties.

Systematic approaches to order a multitude of interconnected transformations, particularly within the realm of virtual reality (VR) [12], [13], and robotic simulators [14], [15], considered the utilization of a scene graph to represent relative spatial relationships. This scene graph, akin to a tree structure, comprises multiple nodes arranged in a parent-child manner. This innovative approach enhanced the representation and simulation capabilities in both virtual reality and robotic simulation domains. The current state of the art is *tf* [16], the scene graph framework of ROS (robot operating system).

Interestingly, very little work has been published that considers the uncertainty of spatial information by interconnecting the different realms of robotics. Initial efforts have been directed towards acknowledging uncertainty within the scene graph, for example [17]. However, these early attempts typically fall short in correctly modeling the error propagation using Lie Algebra. Alternatively, some implementations resort to sampling-based approaches to represent the overall uncertainty within the system, such as [18], which however comes with computational costs.

[1]Now known as L. Burkhard, *et al.*

The Lie-Algebra allows to acknowledge the manifold character of spatial relationships and is a powerful tool to compute and propagate uncertainty along chains of spatial transformations. An introduction to it together with the application to robotic navigation is provided by [19]. Similarly, Lie-Algebra-based concepts are provided for the error propagation within robotic manipulators, either for single errors [20] or as our comprehensive kinematic model [5].

Despite the widespread use of Lie Algebra in uncertainty estimation, to the best of our knowledge, no existing approach formulating a scene graph for robotics has integrated Lie Algebra-based uncertainty propagation. In our ongoing work, we aim to address this gap and demonstrate the efficacy of incorporating Lie Algebra into a scene graph framework for a more nuanced and accurate representation of uncertainty in kinematic systems.

III. ROBOTIC AND ENVIRONMENTAL CONFIGURATION STATE

Accurate assessment of the current configuration state in robotic systems holds significant importance across various applications. This is particularly pronounced in scenarios involving non-static components equipped with perception sensors. Registering cameras affixed to robotic manipulators to the robot's origin is imperative for seamlessly integrating spatial information within the correct coordinate framework. Knowledge of the system's distance to the environment is indispensable for collision avoidance, especially when navigating confined spaces. To achieve this, it's crucial to carefully observe and organize the positions of joints into a transformation tree. This tree helps illustrate how the coordinate framework depends on a specified starting point known as the root frame and obtaining an estimate of the robot's spatial volume. However, overlooking the inherent uncertainty in these measurements and the subtle non-static characteristics of certain links—attributable to mechanical stress and gravitational forces—can lead to erroneous state estimations. In the ensuing discussion, we elaborate on representing the robotic and environmental configuration state (RECS) as a transformation tree. Subsequently, we introduce Lie Algebra as a robust solution for modeling uncertainty in this process. Finally, we detail our implementation of a managed and centralized approach for addressing the RECS problem within an inter-process communication (IPC) framework.

A. Transformation Tree

Deriving the transformation between two coordinate frames is a pivotal task in robotics. A widely employed approach involves modeling the system as a hierarchical tree of frame transformations, as seen in the example Figure 1. This facilitates information extraction from the CAD model, allowing for the calculation of spatial offsets between structural points.

A key optimization involves consolidating static displacements into a singular transformation, pruning the tree for computational efficiency. Movable connections are represented as rotations or translations centered around joints, contributing

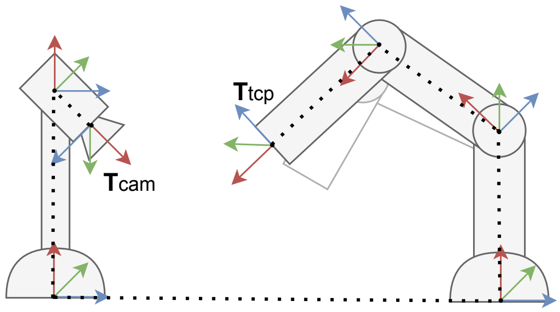


Fig. 1: An illustrated exampled of a robotic manipulator and an external camera. The transformation from the camera to the tool center point of the robot can be calculated concatenating all individual frame transformations.

to a chain of static links and dynamic joints. This approach not only streamlines computational complexity but also provides a comprehensive understanding of a robotic system's kinematic properties, enhancing efficiency and reliability.

Following the comprehensive description of robot kinematics within the previously mentioned tree structure, the process of retrieving the direct transformation between any two arbitrary frames unfolds by traversing the path articulated within this structured tree. This systematic approach ensures a clear and methodical procedure for obtaining the specific transformation information required for precise spatial relationships between frames within the robotic system.

B. Transformations and Uncertainty

Our treatment of uncertainties follows our previous work on probabilistic robot kinematics [5], which in turn builds upon the mathematical foundations provided by [19] and [21].

We briefly introduce the applied methods here, but refer the interested reader to the related works for more thorough insights. For a general introduction to Lie Algebra in the scope of robotics, we recommend the excellent [22], who's notation we mostly follow.

A pose $\boldsymbol{T}_{AB} \in SE(3)$ describes the position and orientation of an object B with respect to a reference frame A. While a pose quantity is generally an element of the manifold $SE(3)$, it can be described *locally* by its linear tangent space representation $\boldsymbol{\xi} = [\boldsymbol{\rho}\,\boldsymbol{\theta}]^T \in \mathbb{R}^6$, related by the exponential map [22]

$$\boldsymbol{T} = \mathrm{Exp}(\boldsymbol{\xi}). \tag{1}$$

There, $\boldsymbol{\rho}$ denotes the translational and $\boldsymbol{\theta}$ the rotational component of the tangent space element. Local tangent space quantities can be mapped between two different local spaces using the *adjoint matrix* $\mathbf{Ad}$ as

$${}^{A}\boldsymbol{\xi} = \mathbf{Ad}(\boldsymbol{T}_{AB})\,{}^{B}\boldsymbol{\xi}, \tag{2}$$

with

$$\mathbf{Ad} = \begin{bmatrix} \boldsymbol{R} & [\boldsymbol{t}]_\times \boldsymbol{R} \\ \boldsymbol{0} & \boldsymbol{R} \end{bmatrix} \in \mathbb{R}^{6\times 6}, \tag{3}$$

where $\boldsymbol{R}$ being the rotation matrix of $\boldsymbol{T}$ an $[\boldsymbol{t}]_\times$ the skew-symmetric matrix formed by the translation. The term $[\boldsymbol{t}]_\times \boldsymbol{R}$ illustrates, how local rotation errors create translation errors further down a chain of transformations, with the magnitude depending on the distance from the original error's location.

Recall that we describe the error of a pose as local deviation $\boldsymbol{\xi}_{\text{B,err}}$ of a nominal pose $\boldsymbol{T}_{AB}$, i.e., in the tangent space of the pose's reference frame B. The corresponding covariance matrix $\boldsymbol{\Sigma}_{AB} = \mathbb{E}\big[\boldsymbol{\xi}_{\text{B,err}}\,\boldsymbol{\xi}_{\text{B,err}}^T\big] \in \mathbb{R}^{6x6}$ is therefore a locally defined tangent space quantity.

The two mathematical operations on poses, which are needed for the scene graph, are thus defined in these terms. The *concatenation* is computed as

$$\boldsymbol{T}_{AC} = \boldsymbol{T}_{AB} * \boldsymbol{T}_{BC} \tag{4}$$

$$\boldsymbol{\Sigma}_{AC} = \mathbf{Ad}_{\boldsymbol{T}_{BC}^{-1}} \boldsymbol{\Sigma}_{AB} \mathbf{Ad}_{\boldsymbol{T}_{BC}^{-1}}^T + \boldsymbol{\Sigma}_{BC}. \tag{5}$$

Note that the two covariance matrices are transported into the common reference frame C using the adjoint matrix, where they can be added due to the linearity of the tangent space. The covariance composition eq. (5) is a first order approximation (called *second* order in some publications) and is discussed in detail in [19].

Analogously, the *inverse* is computed as

$$\boldsymbol{T}_{BA} = \boldsymbol{T}_{AB}^{-1} \tag{6}$$

$$\boldsymbol{\Sigma}_{BA} = \mathbf{Ad}_{\boldsymbol{T}_{AB}} \boldsymbol{\Sigma}_{AB} \mathbf{Ad}_{\boldsymbol{T}_{AB}}^T, \tag{7}$$

shifting the uncertainty from the tangent space of B in the tangent space A. We omit the discussion on the specific modeling of probabilistic rover kinematics here and refer the reader to our previous publication [5]. Note that this representation can implicitly also consider *exact* transformations, as zero-covariances simply vanish in eq. (5) and eq. (7).

C. Implementation

The presented methodology has been implemented within a C++ library, and the corresponding source code is accessible online[2]. Further, a wrapper for the scripting language Python is provided. Each coordinate frame is characterized by a node-element. A frame is precisely defined by its pose matrix $\boldsymbol{T}$ and an accompanying covariance matrix $\boldsymbol{\Sigma}$ which may be set to zero for precisely known transformations. Distinctive identification of each frame is facilitated through the application of a unique character string. Furthermore, the mathematical operations of *concatenation* and *inverse* for each frame are executed leveraging the computational capabilities provided by the manif library [22] augmented by the uncertainty propagation.

The hierarchical structure is implemented using the Boost.Graph data structure.Each vertex encapsulates a frame as its payload, and the edges define the direction of transformations. To determine a path between two nodes within the tree, a breadth-first search (BFS) routing algorithm is employed. The cumulative transformation along the identified path is computed based on the direction specified by the graph's edges, facilitating a comprehensive understanding of

[2]https://rmc.dlr.de/rm/en/staff/marco.sewtz/software

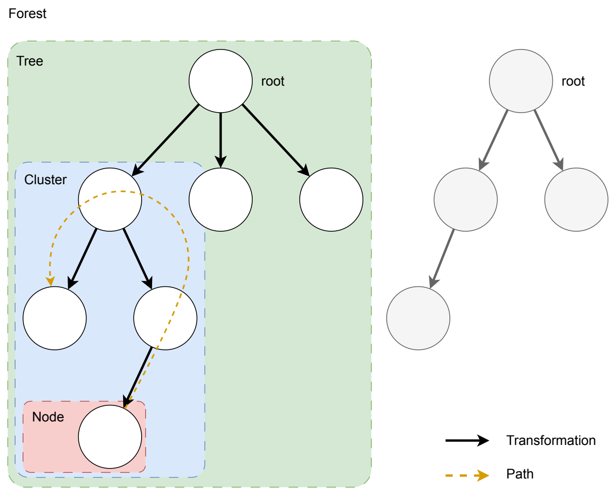


Fig. 2: A schematic overview of the tree structure holding all transformation information. The whole system is consisting of separate trees that do not share any connection. Each tree is constructed by child nodes that are added by directed transformations to their parent node. Further, neighboring nodes can be grouped to a cluster. A transformation between non-neighboring nodes is described by a path.

the transformations between the starting and ending points of the path.

The system allows for the addition of additional root nodes, thereby declaring new trees that remain disconnected from preceding ones. It is imperative to underscore that the establishment of a path between nodes situated on distinct trees within the forest is not feasible. Each root node initiates an independent tree structure, and inter-tree connectivity is explicitly precluded within the system's framework.

The default operational paradigm involves centralized control over all trees, nodes, and computations via a central server. A connected client possesses the capability to perform operations such as creation, retrieval, updating, or deletion of nodes. Additionally, the client can request the cumulative transformation of a specific path. An added feature allows the definition of a local cluster within a tree, enabling the transfer of ownership from the server to a designated client. Consequently, the client gains the ability to locally compute a path within this cluster without necessitating network calls for information retrieval, thereby enhancing computational speed for that particular client. Other clients will be still able to access this information however it must be routed through the server. An illustration of this architecture is given in Figure 2.

IV. Application

To demonstrate the practical utility of the proposed framework, two examples of application will be illustrated in the following. An in-depth analysis of the applying Lie Algebra to the configuration modeling problem has been presented in [5], therefore we want to focus on the scene-graph implementation. At first, the initial application showcases the integration on a robotic arm affected by bending introduced by gravitational pull of the Earth. The second instance will illustrate a mapping application on a system featuring an uncertain RECS, formulated as a graph optimization problem.

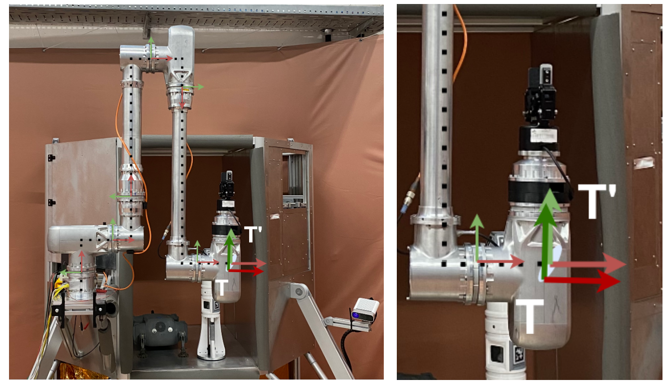


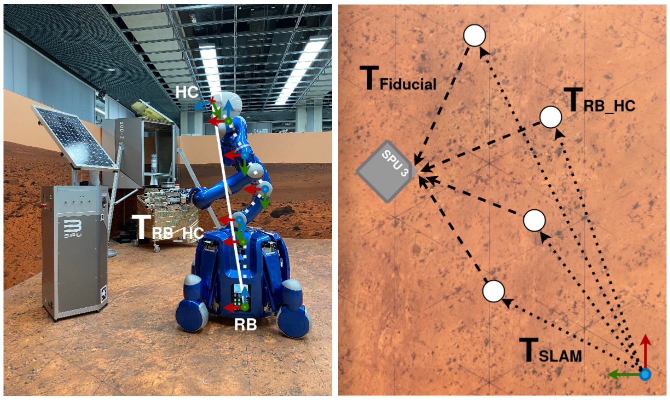
Fig. 3: TINA arm bending due to gravitation. The computed position, designated as $\mathbf{T}'$, represents the theoretical location without accounting for uncertainties.

A. Uncertain robotic and environmental configuration state

As an integral component of the European Space Agency (ESA) project for a Sample Transfer Arm breadboard study, the German Aerospace Center (DLR) developed the TINA manipulator [23] as a compact, modular, and torque-controlled robotic system designed to adhere to the requirements of the Mars Sample Return mission. Figure 3 illustrates the robotic arm in its initial position mounted on a lander. Upon closer inspection, it becomes evident that the manipulator, even in its initial configuration, experiences moderate deformations attributable to its own weight and joint play, particularly in the axial direction. As a result, the pose of the end effector is subjected to several uncertainties, which can be modeled with the proposed framework. By incorporating the expected variance parameters into the transformation tree, the state of the robot configuration can be predicted probabilistically, and the position of the end effector is constrained to an anticipated uncertainty region. Consequently, the consideration of uncertainties provides a more realistic depiction of the arm's pose, acknowledging the impact of various factors, including gravitational forces, and enhances the accuracy of the positional assessment, enabling more precise manipulations. The selection of adequate probabilistic parameters heavily depends on the associated system's specific characteristics and requires specialized technical knowledge. If necessary, an experimental evaluation has to be conducted to validate and fine-tune these parameters.

B. Environmental Mapping

To enable more intricate manipulations and interactions between the robot and its environment, a significant challenge lies in achieving precise registration of the robot relative to its surroundings. This entails aligning various world representations generated for different types of tasks to ensure coherence and accuracy in the robot's perception of its environment.



(a) (b)

Fig. 4: Rollin' Justin mapping a SPU in a Martian environment (a) and the associated optimization graph is represented in (b). The uncertainty-ridden transformation is summarized as T_{RB_HC} from robot base (RB) to the head camera (HC), from which fiducials associated to the SPU are registered.

As depicted in Figure 4a, Rollin' Justin [24] is mapping a Smart Payload Unit (SPU) in a Martian surroundings. In addition to the unknown state of the environmental configuration, a further challenge arises from within the robot. Although the upper body assembly is rigidly connected to the base platform, the wire rope construction in different parts of the torso is inherently less precise than the rigid joints of the arms, introducing uncertainties into the robot's configuration state. Effectively managing and mitigating this uncertainty is crucial since information for navigation purposes is collected from sensors in the base, while other higher-level tasks, e.g., object recognition and manipulation, rely on information from the camera mounted in Justin's head. Therefore, modeling the spatial relations of the robot configuration state, including uncertainties, is essential and can be addressed by the proposed framework. It is further capable of simplifying the handling of transformations and their associated uncertainties by summarizing them into one single step.

In the context of environmental mapping, the transformation from the robot base to the head camera becomes particularly critical as it serves as the foundation for registering fiducials linked to the SPU. Combined with the spatial relationship to the registered fiducials and information regarding the global reference provided by MROSLAM [25], an optimization graph can be constructed, as illustrated in Figure 4b. The optimization problem can be effectively addressed using GTSAM [26] or comparable algorithms, leading to an optimized estimation of the SPU's pose. This comprehensive approach significantly improves the reliability and quality of environmental mapping outcomes in the robot's operational context.

V. Conclusion

We present a Lie Algebra-based framework for uncertainty estimation, realized as a transformation tree. Our work develops a scene-graph-like structure and details the library implementation. Real-world examples demonstrate practical applicability, and comparative analysis highlights method superiority. This contribution enhances robotic transformations, offering a versatile tool for improved reliability and performance.

Future work includes temporal deviation modeling for enhanced capabilities, enabling configuration retrieval from previous timesteps. We aim to align the interface with ROS's *tf* implementation for seamless integration.

References

[1] Kennedy, *The Kinematics of Machinery*. New York: D. Van Nostrand, 1881.
[2] Calvert, *Developing problem-solving skills in engineering*, 1953.
[3] Denavit and Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices," 1955.
[4] Richard, *A Mathematical Introduction to Robotic Manipulation*, 1994.
[5] Meyer et al., "The Probabilistic Robot Kinematics Model and its Application to Sensor Fusion," 2022.
[6] Kaess et al., "isam2: Incremental smoothing and mapping using the bayes tree," *The International Journal of Robotics Research*, 2012.
[7] Kummerle et al., "g2o0: A general framework for graph optimization," in *2011 IEEE International Conference on Robotics and Automation*. IEEE.
[8] Su et al., "Manipulation and propagation of uncertainty and verification of applicability of actions in assembly tasks," *IEEE Transactions on Systems, Man, and Cybernetics*, 1992.
[9] Stoiber et al., "A sparse gaussian approach to region-based 6dof object tracking," in *Proceedings of the Asian Conference on Computer Vision*, 2020.
[10] Meyer et al., "Robust probabilistic robot arm keypoint detection exploiting kinematic knowledge," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems, Workshop on Probabilistic Robotics in the Age of Deep Learning*, 2022.
[11] Nguyen et al., "On the covariance of X in AX = XB," *IEEE Transactions on Robotics*, 2018.
[12] Carlsson et al., "Dive—a platform for multi-user virtual environments," *Computers & graphics*, vol. 17, no. 6, pp. 663–669, 1993.
[13] Tramberend, "Avocado: A distributed virtual reality framework," in *Proceedings IEEE Virtual Reality (Cat. No. 99CB36316)*. IEEE, 1999, pp. 14–21.
[14] Browning et al., "Übersim: a multi-robot simulator for robot soccer," in *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, 2003, pp. 948–949.
[15] Drumwright et al., "Extending open dynamics engine for robotics simulation," in *Simulation, Modeling, and Programming for Autonomous Robots: Second International Conference*. Springer, 2010.
[16] Foote, "tf: The transform library," in *IEEE Conference on Technologies for Practical Robot Applications (TePRA)*. IEEE, 2013.
[17] Coelho et al., "Osgar: A scene graph with uncertain transformations," in *IEEE and ACM International Symposium on Mixed and Augmented Reality*. IEEE, 2004.
[18] T. Ruehr, "uncertain tf," https://github.com/ruehr/uncertain_tf, 2013, last accessed 2023-11-30.
[19] Barfoot et al., "Associating uncertainty with three-dimensional poses for use in estimation problems," 2014.
[20] Yunfeng et al., "Error propagation on the euclidean group with applications to manipulator kinematics," *IEEE Transactions on Robotics*, 2006.
[21] ——, "Nonparametric second-order theory of error propagation on motion groups," *The International Journal of Robotics Research*, 2008.
[22] Sol et al., "A micro Lie theory for state estimation in robotics," *CoRR*, 2018.
[23] Maier et al., "Tina: The modular torque controlled robotic arm - a study for mars sample return," in *2021 IEEE Aerospace Conference (50100)*, 2021.
[24] Fuchs et al., "Rollin' justin - design considerations and realization of a mobile platform for a humanoid upper body," in *2009 IEEE International Conference on Robotics and Automation*, 2009.
[25] Sewtz et al., "Robust approaches for localization on multi-camera systems in dynamic environments," in *2021 7th International Conference on Automation, Robotics and Applications (ICARA)*, 2021.
[26] Dellaert et al., "borglab/gtsam," May 2022. [Online]. Available: https://github.com/borglab/gtsam)