UTBots@Home 2025 Team Description Paper

1st Gustavo Fardo Armênio[1]
gustavofardoarmenio@alunos.utfpr.edu.br

2nd Bruno Bock Silva[1]
brunobock@alunos.utfpr.edu.br

3rd David Segalle[1]
davidsegalle@alunos.utfpr.edu.br

4th João Henrique Cordeiro Pires[1]
joaopires@alunos.utfpr.edu.br

5th Larissa Bialski Lima da Silva[1]
larissabialski8@gmail.com

6th Luiz Guilherme Silvino Greca[1]
luizgreca@alunos.utfpr.edu.br

7th Bruno Emanuel Zenatti[1]
brunozenatti@alunos.utfpr.edu.br

8th Henrique Marcelo de Souza Ramos[1]
henriqueramos@alunos.utfpr.edu.br

9th Rachel Paulino Previtali[1]
rachelprevitali@alunos.utfpr.edu.br

10th Gabriel Tomás Cunha Pereira[1]
gabrielp.2006@alunos.utfpr.edu.br

11th Pedro Eugênio Marin do Nascimento[1]
pedronascimento.2023@alunos.utfpr.edu.br

12th João Victor Mendes de Siqueira[1]
Joaosiqueira.2005@alunos.utfpr.edu.br

13th Pedro Henrique de Freitas Neves[1]
pneves.2022@alunos.utfpr.edu.br

14th Rafaela Ximenes Santos[1]
rafaelaximenes@alunos.utfpr.edu.br

15th Maria Clara Skibinski de Carvalho e Silva[1]
maomaomarii@gmail.com

16th Eduardo Borgio Guimarães Silva[1]
esilva.2024@alunos.utfpr.edu.br

17th Victor de Moura[1]
moura.2024@alunos.utfpr.edu.br

18th Rafael Fulgencio Rosario da Cruz[1]
rcruz@alunos.utfpr.edu.br

19th Camily Cosme de Lima[1]
camilylima@alunos.utfpr.edu.br

20th Michael Moraes[1]
michaelm@alunos.utfpr.edu.br

20th Danielle Cristina Xavier Quintana[1]
Daniellecristina@alunos.utfpr.edu.br

21th Marlon de Oliveira Vaz[2, 4]
marlon.vaz@ifpr.edu.br

22th André Cristiano Cordeiro[1, 3]
ac.cordeyro@outlook.com

23th Arthur Massaru Nonaka[1, 3]
arthurnonaka@alunos.utfpr.edu.br

24th André Schneider de Oliveira[1, 2]
andreoliveira@utfpr.edu.br

25th João Alberto Fabro[1, 3]
fabro@utfpr.edu.br

***Abstract*—This paper presents the current developments and innovations of the UTBots@Home team for the 2025 CBR@Home competition. Building upon a decade-long commitment to advancing service robotics in domestic environments, we detail the integration of ROS2 in our software stack and the resulting improvements in task orchestration, face recognition, voice processing, and robot vision. Notable contributions include a robust multi-frame object detection algorithm, a low-cost high-payload robotic manipulator named THESEUS, and enhanced behavior control using Behavior Trees. Our natural language understanding pipeline combines state-of-the-art tools for speech recognition, noise suppression, and intent detection, enabling more fluid and resilient human-robot interaction. Hardware improvements to our custom mobile base and robot structure have led to significant gains in navigation and manipulation performance. This paper outlines our current solutions, research initiatives, and future directions aimed at tackling the increasing complexity of @Home challenges while promoting open-source and accessible robotics development.**

***Index Terms*—RoboCup Brazil, Robot Manipulation, Task Orchestration, Natural Language Processing, Robot Vision, Robot Voice, State Machine, Behavior Tree, Robotic Arm, Mobile Base, Emotional Interface, Object Detection, Object Position Estimation, Face Recognition, Robotic Manipulation.**

1 - UTFPR (Federal University of Technology – Paraná), Curitiba, PR, Brasil, 2 - CPGEI (Graduate Program on Electric Engineering and Industrial Informatics), UTFPR, Curitiba, PR, Brasil, 3 - PPGCA (Graduate Program on Applied Computing), UTFPR, Curitiba, PR, Brasil, 4 - IFPR (Federal Institute of Paraná), São José dos Pinhais, PR, Brasil

I. Overview

The UTBots@Home[1] team has been dedicated to the ongoing development of service robots designed to assist humans in household tasks since 2014. Driven by the challenges presented by the @Home competition [1], our efforts have advanced in both complexity and abstraction. Initially focused on basic control of robotic systems, we now integrate and orchestrate our hardware and software systems at a level of

[1]HomePage of the Team: https://utfpr.curitiba.br/utbots-at-home/home/

abstraction suited to the ontology of a household environment in line with the competition's characteristics.

Over the years, the competition has served as a platform for numerous academic research endeavors at UTFPR. Our research involves, but is not limited to, navigation with fuzzy logic [2] [3], computational intelligence frameworks [4], human-robot emotional interface [5], fast methods for dataset capture for object detection and manipulation [6], methods for estimating an object's representative 3D point in space [7], and new methods for people recognition [8].

In the domain of software, our current work is primarily focused on Natural Language Understanding, Computer Vision, and task orchestration techniques involving state machines and behavior trees. The migration of our whole software system to ROS2 resulted in the upgrade of several robot functionalities such as face recognition, natural language understanding, text-to-speech and task orchestration.

This Team Description Paper (TDP) provides an overview of the team's current solutions and ongoing research for various challenges in the competition. Section 2 provides a description of the innovations the team has made for this year's participation. Section 3 focuses on the current solutions we apply for several components of our robot and the tasks it has to perform. Section 4 concludes and offers insights into future perspectives.

II. Innovation

A. Rapid fine-tuning of vision models with RoboCup@Home object dataset

An ongoing project in the last years has been creating an extensive household objects dataset in the competition setting. In each participation, we collect and annotate a new set of objects inside the arena. The objects' main characteristics remain similar – even more so inside each category of objects –, but the variation of product brands and specific shapes, ambient characteristics and perspectives is expected to improve fine-tuning-based models such as YOLOv11 [9]. For our CBR 2025 participation, we pretrained a YOLOv11 model on our dataset of all objects from all previous @Home editions and fine-tuned on the last set of object images. In the time-constrained period of two warm up days, this method resulted in a test mAP value of 0.786 with a reduced need of new annotations and less training time.

B. Object Batch Detection

Object detection models usually deal with the uncertainty of a prediction with a confidence value as a threshold to rely on for deciding if a given object is present or not in the scene. In RoboCup@Home, having a reliable object detector is essential for almost all tasks, and depending on the confidence value only does not always guarantee the needed robustness. These models can have big changes in prediction between frames of a constantly changing point of view of a mobile robot. Aiming for more robust detections, we developed a multi-image detection procedure that composes the predictions from several frames to filter out incorrect outlier bounding boxes or moving objects that are not going to stay in the same place – this solution takes inspiration from machine learning ensemble models.

Our algorithm has some initial parameters that can be tuned for a more flexible or strict detector, such as IoU threshold, number of frames (batch size) for evaluation and Support threshold inside the batch. Considering a predicted bounding box can change slightly between frames, the IoU (Intersection-Over-Union) is a metric that gives us the information about a bounding box corresponding to another in both frames: if the IoU is bigger than the threshold and the predicted class is the same, they are considered the same object and the bounding box support value is incremented. Finally, after all evaluated frames, all predicted bounding boxes in all frames are filtered by the Support threshold – if it is predicted in a minimal percentage of the total number of frames. A current limitation for this method is that the camera has to be static for all frames, because the bounding boxes' coordinates are not relative to camera orientation in the environment.

C. Task Orchestration

Upon realizing the necessary complexity to implement Stage 2 tasks with state machines, it became clear that Behavior Trees offer a better solution in tasks that require ample reactivity. BehaviorTreeCppv4 [10] offers the tools we need to develop custom trees with ample modularity in how to define nodes and conditions, as well as being the same system used for controlling navigation within the ROS nav2 ecossystem, meaning it is likely to continue being supported for the foreseeable future. A greatly simplified solution to a @Home task is found in Figure 3.

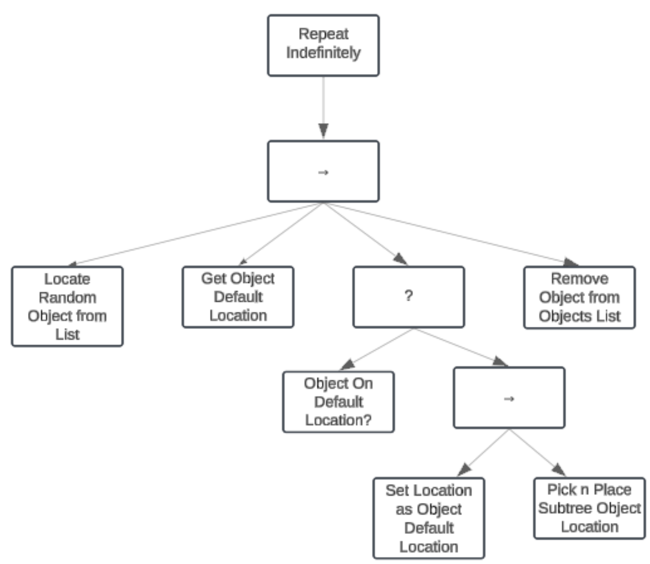

Fig. 1: Simplified Behavior Tree designed to perform pick and place.

D. Robot Scructure

Following last year's redesign of the robot, a few adjustments were made in its structure to increase its effectiveness

Fig. 2: H.E.S.T.I.A. (Household Efficient Service Task-Integrated Assistant)

in important functionalities necessary for @Home tasks, such as autonomous navigation, human-robot interaction and object manipulation. The most important changes was reducing the robot footprint to a radius of 32cm, which increased significantly the quality of autonomous navigation in constrained spaces.

E. THESEUS: Low-cost Open-Source Robotic Manipulator

The most popular solution for robotic manipulators in the Latin American scene is the use of high torque servo motors, more precisely Robotis' Dynamixel servo motors. Some teams buy manipulators and others build considering the servo motors specifications, but even building them is highly expensive because of the cost of the motors and has big limitations of payload. Robotis OpenManipulator-X (RM-X52-TNM) [11] and Trossen Robotics ReactorX 200 [12] are examples of such commercial manipulators. The former has 5 DOF, supports a payload up to 500g and currently is sold for $1,416.60; the latter has 5 DOF, supports a payload of only 150g and is currently sold for $2,099.95.

Our team's proposal with THESEUS was to design a manipulator capable of upholding up to 1.5kg with 5 degrees of freedom (DOF). In comparison with the cited manipulators, it is also built with aluminum extrusions and 3D printing, but, instead of using servo motors, we decided to apply outrunner brushless motors, reduced with custom cycloidal 121:1 gearboxes controlled by magnetic encoders with position control drivers based on the open-source Odrive [13] hardware. With this choice, the idea was to reduce the cost to less than $1000.00 for a bigger payload capacity.

For THESEUS' first version we estimate a cost of around $700.00. Comparing the cost of $269.90 for each Dynamixel MX430 servo-motor [14] used in RM-X52-TNM and ReactorX 200, we were capable of reducing the cost of each motor with controller to around $100.00 – and estimate made with the SEQURE ODESCv3.6 [15] two-motor controller, Odrive's D5312s [16] brushless motor and the as5048a magnetic encoder. The idea is to share THESEUS as an open-source project[2] in order to aid other teams with resource constraints to build a low-cost higher payload capacity manipulator.

Fig. 3: THESEUS robotic manipulator

F. Face Recognition

With the transition from ROS Noetic to ROS Humble, it became clear that some dependencies on old Python libraries needed to be updated to be compatible with the libraries running ROS itself. With this in mind, our old solution based on KNN algorithms [8] had to be abandoned.

We decided to utilize the deepface [17] library with a self-developed wrapper in order to be able to freely choose from many different algorithms for face detection and face recognition. Currently, our solution uses MTCNN [18] for high-quality, yet slow, face detection and FACENET512 [19] for face recognition. Our new system allows us to easily transition to newer models as they get added. The time of

[2]THESEUS Project Source: https://github.com/UtBots-Home/theseus-project

the face registration procedure was significantly reduced with this approach, as the model does not need to be re-trained and the only required step is to register pictures of the face.

III. Current State of Solutions

A. Environment and Task Orchestration

This year, a major software redesign resulted in the migration of all software to ROS2, specifically ROS Humble. The deprecation of old libraries and new available solutions in newer versions of the Linux Ubuntu OS created an opportunity to upgrade several functionalities in our software.

Currently, most of our tasks are solved using state machines with the YASMIN library [20]. State machines offer a great system to perform tasks which are linear in nature and don't require lots of reactivity, their simplicity offers a great solution to integrate new members into the ROS ecosystem, by letting them develop tasks in a highly interactive way, they can get accustomed to ROS with a relatively simple top-down approach. Additionally, this means they are great for developing test demos, as well as the tasks present in Stage 1.

B. Voice and Language Processing Pipeline

For complex dynamics and domains of Human-Robot Voice Interaction, we built an extensive pipeline to handle multiple levels of comprehension with robustness to ambient noise, common in the competition environments. The pipeline is built for GPSR and common voice interaction requirements in RoboCup@Home tasks by leveraging intent detection, RAG (Retrieval Augmented Generation) and response generation with both a robust LLM model and a low-resource cost model, RASA [21] for Natural Language Understanding.

- *Voice Activity Detection*: Initiated by Silero VAD [22] to segment the active voice from the background silence in real time.
- *Noise Suppression*: We use RNNoise [23], to further isolate the human voice in noisy environments.
- *Speech-To-Text*: For multilingual transcription, OpenAI's Whisper v3 turbo quantized [24] model offers accuracy and significantly reduced latency and memory usage.
- *Text-To-Speech*: System responses are vocalized using Coqui TTS [25], with natural-sounding speech synthesis.

C. Emotion Interface

Motivated by a desire to make human-robot interactions in the domestic environment more human, our team has two main emotion interfaces integrated. Our package *display_emotions* [26] implements a facial interface with smooth emotion transitions based on Plutchik's Wheel of Emotions [27], selectable by ROS2 topics. An RGB LED "heart", 3D printed, compatible with the former package, changes color according to the emotion selected.

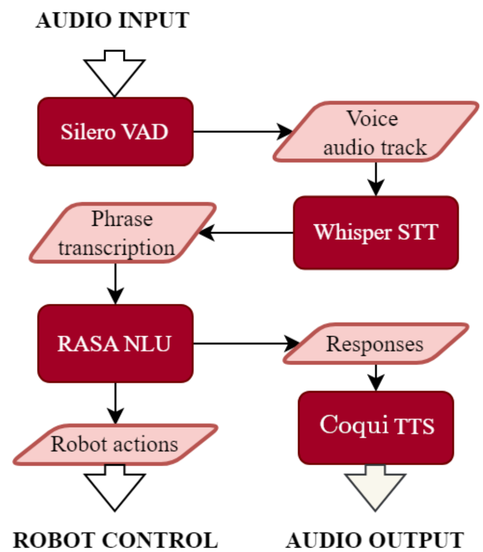

Fig. 4: Voice and language processing systems diagram

D. Robot Vision

An Intel Realsense offers visual and depth information essential for generating point clouds and enabling various functionalities such as object recognition, person detection, tracking, pose estimation, and face recognition.

- *Object Detection*: We apply the YOLOv11 convolutional neural network [28], performing a transfer-learning training with the new competition objects each year. The object detector outputs a bounding box for a certain object in a RGB image, an essential information used in several tasks, as well as by other vision systems.
- *Person Tracking, Gesture and Pose Estimation*: For tasks such as following a person and identifying gestures, the custom *mediapipe_track*[3] package implements MediaPipe Pose [29] with ROS2 messages. MediaPipe Pose is responsible for processing body pose information and tracking the general skeleton pose of a person. The *mediapipe_track* package enhances this functionality by performing tracking and generating 3D position estimates for several body landmarks. It achieves this by subscribing to RGB image messages, estimating the image pixel coordinates for a landmark (such as the right elbow) and a distance value from its perspective. A simple use, for example, is calculating the right and left shoulder and right and left hip landmark 3D values mean for estimating a person's distance from the camera.
- *Face Recognition*: As specified in Section II-F, we currently apply a pipeline with MTCNN [18] for face detection and FACENET512 [19] for face recognition, without the need to retrain the model in runtime, optimizing the registration procedure.

[3] Available at: https://github.com/UtBotsAtHome-UTFPR/mediapipe_track

E. Navigation

Commercial mobile bases have a relevant disadvantage in the context of long-term and low-resource development: maintenance and support expenses. Our custom mobile differential base uses two brushless motors recycled from a hoverboard. Motor control is done with the original control board, flashed with a firmware that implements Field Oriented Control [30]. This firmware gives access to the control board from ROS2 topics and enables odometry and autonomous navigation. The navigation system that allows access to the new firmware from our ROS2 systems was implemented with ROS2 Humble nav2 packages [31]. With this setup, we utilize SLAM algorithms for the mapping procedure of the arena and AMCL for navigation and auto-localization.

F. Manipulation

Accurate manipulator control is currently the next breakthrough for HESTIA. Our open-source manipulator initiative, THESEUS, aims to be a cost-effective solution that does not need to be replaced when @Home advances to the manipulation of heavier objects. A relevant disadvantage of a custom build manipulator is the intrinsic error that adds up in the first non-optimized hardware versions, which has been a relevant challenge for this mission.

Manipulator libraries are built for robust control. With THESEUS, we implemented the whole hardware interface for joint trajectory control with *ros2_control* [32]. The implementation of direct and inverse kinematics and path planning are all made with MoveIt! library [33] tools. An interface with ROS2 actions makes it simple to send the 3D point of an object as a goal for manipulation.

IV. Conclusion and Future Work

In our 2025 efforts, we have significantly enhanced our robot's autonomy, perception, and interaction capabilities by migrating our entire software stack to ROS2 and incorporating modern frameworks such as Behavior Trees, deep learning-based face and object recognition, and advanced voice processing pipelines. Our hardware developments, particularly the THESEUS robotic manipulator and redesigned mobile base, demonstrate our commitment to building cost-effective yet capable robotic systems.

Looking forward, several areas are targeted for further improvement and validation. One key goal is to continue testing and refining our task orchestration with Behavior Trees under the dynamic and unpredictable conditions of the RoboCup@Home competition environment. Additionally, a complete evaluation of THESEUS's manipulation performance in real-world competition scenarios will inform its next design iterations. We also aim to expand our multi-image object detection method to operate effectively with a moving camera, enhancing detection reliability during robot motion. Finally, a future research direction involves the integration of semantic environment knowledge to provide the robot with contextual awareness, enabling smarter planning and more adaptive behaviors.

Acknowledgment

The authors would like to thank LASER (Advanced Laboratory of Embedded Systems and Robotics), UTFPR (Federal University of Technology – Paraná) and the PPGCA (Graduate Program in Advanced Computing) for their support and resources, which have contributed significantly to the successful completion of this endeavor.

Additionally we would also like to thank OrthoMouse and TRACTIAN for their financial support.

References

[1] Robocup@home official homepage. http://www.robocupathome.org. Accessed Nov. 2023.

[2] J. Cerbaro, D. Martinelli, A. S. Oliveira, and J. A. Fabro. Waiterbot: Comparison of fuzzy logic approaches for obstacle avoidance in dynamic unmapped environments using a laser scanning system (lidar). In *2020 Latin American Robotics Symposium (LARS), 2020 Brazilian Symposium on Robotics (SBR) and 2020 Workshop on Robotics in Education (WRE)*, 2020.

[3] A. L. Sordi Filho, L. P. Oliveira, A. S. Oliveira, J. A. Fabro, and M. A. Wehrmeister. Simultaneous navigation and mapping in an autonomous vehicle based on fuzzy logic. In Nadia Nedjah, Heitor Silvério Lopes, and Luiza de Macedo Mourelle, editors, *Studies in Computational Intelligence*, volume 664, pages 53–68. Springer International Publishing, 1 edition, 2017.

[4] T. Becker, A. S. Oliveira, J. A. Fabro, and R. L. Guimaraes. Lida bridge: A ros interface to the lida (learning intelligent distribution agent) framework. In Anis Koubaa, editor, *Studies in Computational Intelligence*, volume 625, pages 703–728. Springer International Publishing, 2016.

[5] D. Santos, P. Palar, A. S. Oliveira, J. A. Fabro, and T. Becker. Adding conscious aspects and simulated emotions through facial expressions in virtual robot navigation with baars-franklins cognitive architecture. In *2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*, João Pessoa-PB, 2018.

[6] J. Fabro, M. Vaz, and A. Oliveira. Design and development of an automated system for creation of image datasets intended to allow object identification and grasping by service robots. In *January 2020*, pages 1–6.

[7] G. F. Armênio, J. A. Fabro, R. R. Tognella, F. P. Conter, M. V. Oliveira, and E. S. Silva. Estimating the 3d center of an object with kinect sensor rgb-d images. In *Latin American Robotics Symposium (LARS), XX Simpósio Latino Americano de Robótica (LARS 2023)*, Salvador, Bahia, Brazil, 2023.

[8] D. Segalle, G. F. Armênio, G. F Lewin, E. S. Silva, and J. A. Fabro. Identifying new persons in the context of the robocup@home competition using knn. In *Latin American Robotics Symposium (LARS), XX Simpósio Latino Americano de Robótica (LARS 2023)*, Salvador, Bahia, Brazil, 2023.

[9] Ultralytics Jocher and contributors. Yolov11: You only look once, 2024.

[10] Davide Faconti. BehaviorTree.CPP: A C++ library to build Behavior Trees. https://github.com/BehaviorTree/BehaviorTree.CPP, 2018. Version 4.x.

[11] ROBOTIS. Openmanipulator-x rm-x52-tnm. https://www.robotis.us/openmanipulator-x-rm-x52-tnm/. Accessed: 2024-11-21.

[12] Trossen Robotics. Reactorx 200 robotic arm. https://www.trossenrobotics.com/reactorx-200. Accessed: 2024-11-21.

[13] Oskar Weigl and Joseph Morris. Odrivehardware. https://github.com/odriverobotics/ODriveHardware, 2018.

[14] ROBOTIS. Dynamixel xm430-w350-t. https://www.robotis.us/dynamixel-xm430-w350-t/. Accessed: 2024-11-21.

[15] Sequre Mall. Odesc dual drive. https://sequremall.com/collections/odesc/products/odesc-dual-drive. Accessed: 2024-11-21.

[16] ODrive Robotics. Dual shaft motor d5212s 300kv. https://shop.odriverobotics.com/products/dual-shaft-motor-d5212s-300kv. Accessed: 2024-11-21.

[17] Sefik Serengil and Alper Ozpinar. A benchmark of facial recognition pipelines and co-usability performances of modules. *Journal of Information Technologies*, 17(2):95–107, 2024.

[18] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE signal processing letters*, 23(10):1499–1503, 2016.

[19] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[20] Miguel Ángel González-Santamarta, Francisco Javier Rodríguez-Lera, Camino Fernández Llamas, Francisco Martín Rico, and Vicente Matellán Olivera. Yasmin: Yet another state machine library for ros 2, 2022.

[21] Tom et al. Bocklisch. Rasa nlu: Open-source natural language understanding, 2017.

[22] Snakers4 Team. Silero vad: A high-performance voice activity detection model, 2023.

[23] Jean-Marc Valin and etc. Maxwell. Rnnoise: A noise suppression library based on deep learning, 2018.

[24] Alec et al. Radford. Whisper: Robust speech recognition via large-scale weak supervision. OpenAI Research, 2023.

[25] Coqui Team. Coqui tts: Open-source text-to-speech, 2024.

[26] P. S. Palar. Ros package ros_display_emotions. https://github.com/kpiatan/ros_display_emotions. Accessed Jun., 2023.

[27] R. Plutchik and H. Kellerman. *Emotion: Theory, Research and Experience*. Academic Press, New York, 1986.

[28] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics YOLO, January 2023.

[29] Camillo Lugaresi et al. Mediapipe: A framework for building perception pipelines, 2019.

[30] E Feru. hoverboard-firmware-hack-foc. https://github.com/EFeru/hoverboard-firmware-hack-FOC, 2019.

[31] Steve Macenski et al. The marathon 2: A navigation system. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[32] Sachin Chitta, Eitan Marder-Eppstein, Wim Meeussen, Vijay Pradeep, Adolfo Rodríguez Tsouroukdissian, Jonathan Bohren, David Coleman, Bence Magyar, Gennaro Raiola, Mathias Lüdtke, and Enrique Fernández Perdomo. ros_control: A generic and simple control framework for ros. *The Journal of Open Source Software*, 2(20):456, 2017.

[33] David Coleman, Ioan A. Șucan, Sachin Chitta, and Nikolaus Correll. Reducing the barrier to entry of complex robotic software: a moveit! case study. *Journal of Software Engineering for Robotics*, 5(1):3–16, May 2014.