FAIM/ISCA Workshop on Artificial Intelligence for Multimodal Human Robot Interaction
14-15 July 2018, Stockholm, Sweden

Bridging the Gap Between Supervised Autonomy and Teleoperation

Adrian S. Bauer[1], Peter Birkenkampf[1], Alin Albu-Schäffer[1,2], Daniel Leidner[1]

***Abstract*— Human teleoperation of robots and autonomous operations go hand in hand in many of todays service robots. While robot teleoperation is typically performed on low to medium levels of abstraction, automated planning has to take place on a higher abstraction level, i.e. by means of semantic reasoning. Accordingly, an abstract state of the world has to be maintained in order to enable an operator to switch seamlessly between both operational modes. We propose a novel approach that combines simulation-based geometric tracking and semantic state inference by means of so called State Inference Entities to overcome this issue. The system is demonstrated in real-world experiments conducted with the humanoid robot Rollin' Justin.**

I. Introduction

Space assistant robots such as the humanoid robot Rollin' Justin [1] have become mechanically capable to manipulate their environment. Still, they do not provide full autonomy and need a human operator. Thus, our research on telerobotic manipulation focuses on direct teleoperation (e.g. Kontur-2 [2]) and supervised autonomy (e.g. METERON SUPVIS Justin [3]). In these missions, we learned that both control modalities are necessary to operate a robot efficiently under varying conditions (see Fig. 1).

Traded control is an approach that allows the operator to switch between autonomous task execution and teleoperation [4]. An open research question in traded control is the synchronization of world states while switching from teleoperation to autonomous mode. That is, planning in autonomous mode requires an accurate semantic representation of the world, but during teleoperation robots are not yet able to keep track of the semantic state changes initiated by the operator. Thus, the semantic world state at the end of a teleoperation session is unknown, making it impossible for the robot to operate autonomously afterwards.

We derived an intuitive analogy for this problem by comparing the robot with a sleepwalker whose motor capabilities are intact while at the same time he/she does not perceive the environment consciously [5]. Similarly, robots are active during teleoperation without "perceiving" the changes they exert on the environment and they lack status updates during teleoperation, leaving them "disoriented" afterwards. While *visual servoing* may be used to track geometric state changes during teleoperation [6], and concepts like *anchoring* try to capture symbolic states from senor data [7], there is currently no work known to the authors that employs physical simulations for state inference. Simulation brings the advantage of being able to represent changes of objects that, no matter

[1] German Aerospace Center (DLR), Robotics and Mechatronics Center (RMC), Weßling, Germany
[2] Technische Univ6ersität München (TUM), Sensor-Based Robotic Systems and Intelligent Assistance Systems, Munich, Germany

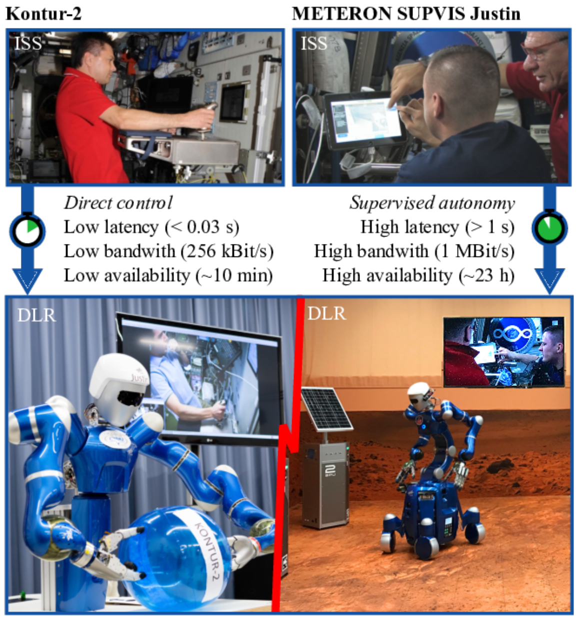

Fig. 1: The humanoid robots Space Justin and Rollin' Justin remotely operated from the International Space Station using direct control (Kontur-2 experiment left) and supervised autonomy (METERON SUPVIS Justin experiment right).

why, are not perceivable by sensors. Accordingly, this paper proposes an approach to derive semantic state transitions from robot telemetry retrieved during teleoperation by means of physics simulations and *State Inference Entities (SIEs)*.

The contributions of this paper include a software architecture to infer semantic state transitions during robotic teleoperation and the concept of SIEs that allow to extract semantic knowledge from physics simulations. The developed methods are validated based on real world robot telemetry recorded during teleoperation of the humanoid robot Rollin' Justin.

II. Related Work

The lack of ability to operate fully autonomously lead to the emergence of approaches that aim at blending human and robot intelligence to ease human-robot interaction on different levels of autonomy (see [8, Tab. 5.2]). Their main idea is to ease human-robot interaction by integrating autonomous features supporting the user. While, *supervised autonomy* allows the operator to initiate tasks on a high level of abstraction that are carried out by the robot autonomously [8], *shared control* provides support for the user who controls the robot via continuous input on a *Human-Robot Interface (HRI)* [8], e.g. by means of safequarding. Robots can also provide a mixture of supervised autonomy

10.21437/AI-MHRI.2018-11

and shared control, called *traded control*, as in [9].

Autonomously achieving predefined goals requires the robot to be able to generate and execute plans. Since it is common to human communication to define goals in terms of symbolic states and it is easier to plan on the symbolic level, robots usually employ symbolic planning, potentially with refinement on the geometrical domain (e.g. *hybrid planning* for example in [10], [11]).

While humans are able to employ commonsense, robots require a detailed specification of a problem. The task of predicting the symbolic world state after teleoperation includes lots of physical knowledge that is not specified explicitly. Humans presumably possess an inherent ability to predict and assess physical phenomena, called *naive physics* [12], an idea that was adopted to AI by Hayes [13] by formalizing everyday physics knowledge. Instead of creating a formalization, todays physic simulations enable us to exploit their rich intrinsic physical knowledge in order to solve robotics tasks. Multiple approaches have succeeded at employing physics simulations to planning problems in robotics, allowing for physical commonsense reasoning [14], [15], [16], [17].

Coradeschi and Saffioti proposed a system of *anchoring* sensory stimuli to symbolic representations [7]. This is in line with work from Lemaignan et al. [18] that allow to extract simple symbolic predicates from visual percepts. Since our focus is on the interface between teleoperation and supervised autonomy, goal inference and plan recognition are not considered.

III. System Concept

The proposed framework, running on Rollin' Justin [1], augments the current hybrid planning system that is based on *action templates* [19]. Action templates describe robot actions by means of a symbolic header defined in Planning Domain Definition Language (PDDL), and a geometric body grounding the action geometrically to the robot.

Center to the current system is the world representation module that holds the geometric and symbolic world state, both being essential for hybrid planning. Before performing expensive geometrical planning, the semantic planner generates a high level plan based on the symbolic world state which is finally refined on a geometric level. A deviation in the symbolic world state might, thus, result in an incorrect plan that cannot be executed by the physical robot.

While being teleoperated, the robot is commanded on a low level and does not possess a symbolic representation of the action the user executes. Thus, the symbolic state of the world cannot be updated automatically. The resulting deviation in the symbolic world state prevents the robot from further autonomous operation. In order to resolve this issue it is necessary to update the geometric world representation during teleoperation via a subsequent mapping of the geometric state to a symbolic state as it is described in the following section.

A. Inference Framework

The proposed framework is mainly designed to be active during teleoperation of the robot, thus, modules that cope

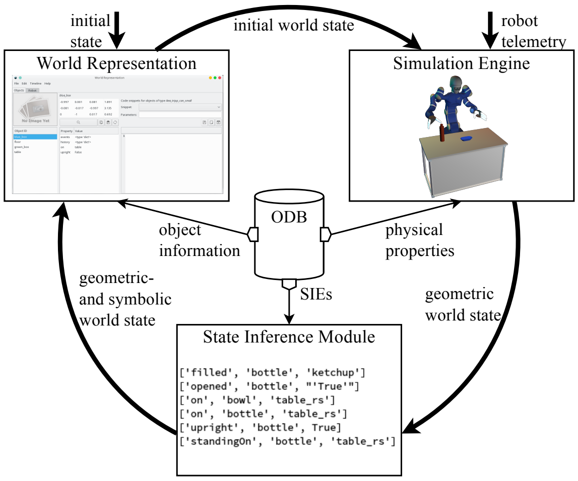

Fig. 2: Overview of the framework consistin of world representation, simulation engine, state inference module and object database.

with planning do not need to be considered. A schematic overview is provided in Fig. 2 showing the interaction between world representation, simulation engine, inference module, and the Object Database (ODB). Forming a closed loop, the framework starts from a known world state, and finishes in a state where the world state is known again. This corresponds to the situation of switching from autonomous control to teleoperation and back again.

Switching on teleoperation mode, the world representation initializes the simulation environment with the current world state. Next, the simulation continuously receives real-robot telemetry and mirrors it on the simulated robot, computing interactions between the robot and the environment. Thus, the simulation maps the initial world state world and a time series of telemetry to a resulting geometrical world state.

Once the teleoperation finishes, the results of the simulation are forwarded to the state inference framework where they are evaluated based on SIEs. Sending the evaluated predicates to the world representation ensures that the world state is updated accordingly whenever the robot switches back to autonomous mode. The knowledge necessary for inferring symbolic states from geometrical information is attached to the objects in the ODB.

The loop-structure further allows the system to be employed in order to generate on-line estimates of the world state *during* teleoperation. Intermediate world states can be inferred by invoking the inference during teleoperation. Executing this process recursively, any temporal resolution, only limited by the temporal resolution of the sensors and the real-time capability of the resulting system, can be achieved. Experiences with the framework indicate that the simulation engine forms a bottleneck for the overall speed.

B. Simulation Environment

For simulation we use the robot simulation environment Gazebo [20] in conjunction with the open dynamics engine

(ODE)[1]. At the beginning of a teleoperation session, information from the ODB and the world representation is used to initialize the simulation with the current state of the world.

Once initialized, the robot in the simulation starts to mirror the movements of the real world robot by following the transmitted telemetry (here joint angles). As soon as the teleoperation finishes, the results of the simulation are forwarded to the inference module. For the sake of modularity, the simulation module can easily be replaced by another simulation, only requiring a new adapter for communication with the other modules.

C. Inference Module

After receiving data from the simulation, the inference process starts with the goal to extract semantic predicates from the simulated geometric state of the world. The inference process is based on the position of objects, collisions between objects, and the corresponding forces. The predicates (unary or n-ary) that are to be evaluated are bound to the objects in the ODB and so is the information used for the evaluation of predicates. We implement the inference knowledge in terms of SIEs stored per object in the ODB.

In order to reduce the computational costs of the inference process, inference is only performed on objects that have either been manipulated during teleoperation or that are in collision with a manipulated object. The results from the inference are collected and used to update the world state.

D. State Inference Entities (SIEs)

The SIEs form a central aspect of the state inference process. They consist of executable Python code and implement a common method `executeSnippet()` that is called from the state inference module. Following the principle of modularity, this method can be implemented freely. In case of space applications, it is preferable to have hand-coded snippets that produce reproducible results, but the snippets might also execute a probabilistic classifier in other setups.

The `executeSnippet` method returns a list of predicates that have been evaluated. In case a predicate does not hold, its corresponding SIE returns an empty list.

IV. Evaluation

In order to achieve a proof-of-concept, the system was used to infer the semantic state in the *boxworld* environment, which consists of two boxes of different size placed on a table. Teleoperation was simulated by moving the robot manually in compliance mode (see Fig. 3). Ground truth data was generated by localizing objects before and after teleoperation by means of APRIL-tags [21].

A. Experiment

The inference capabilities of the framework were tested by implementing two predicates, namely `on` and `upright` for the two boxes. A common problem of robotic simulation is the *reality gap*, that describes the phenomenon of divergence between real world and simulated world. The parameters

[1]`http://ode.org/`, last retrieved on May 8, 2018

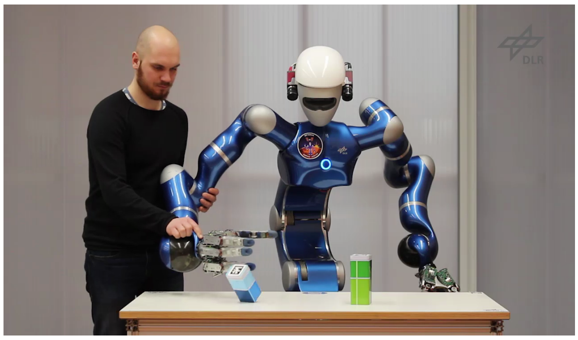

Fig. 3: Overview of the experiment setup.

mainly responsible for the reality gap in our simulation were the friction coefficients of the boxes (μ_1, μ_2), the *constraint force mixture (CFM)*, and the *error reduction parameter (ERP)*, the last two are specific parameters of the simulation engine ODE. Parameters were estimated by employing an evolutionary strategy, similar to Laue and Hebbel [22].

Fig. 4 shows the simulation of the recorded telemetry of pushing over the blue box and the resulting change in the world representation, both for a simulation with good and bad parameterization. In both cases the transition from the simulated geometrical world state to the symbolic world state is performed correctly but the geometric world state is incorrect in the upper row.

B. Discussion

The proposed framework enables us to keep track of the symbolic world state during teleoperation. Geometric and symbolic reasoning are split up into two modules, allowing us to employ a physics simulation and its inherent physical commonsense knowledge to infer geometric state changes. Using a physic simulation enabled us to use highly advanced physical knowledge without the need to re-implement it. The modular design of the framework allows for integrating a new simulation engine with minimum effort.

Overall, the framework is able to extract the predicates `on` and `upright` successfully provided only with robot telemetry and the initial world state state.

V. Conclusion and Outlook

The proposed framework brings us a step closer to “awake” the sleepwalking robot and enable it to keep track of the changes it induces in its environment, thus, allowing for transmissions between teleoperation and (supervised) autonomous behavior. The framework can be used for teleoperation scenarios where the objects in the scene are well known and where the physics of the overall scene are rather stable. It does not rely on constantly monitoring object positions visually but predicts their behavior based on a physical simulation. The strength of this concept lies in the robustness against failure of sensors, however, it requires physically accurate models of the objects in the scene and its accuracy is limited by the precision of the simulation engine.

Following this first step we see some opportunities and open questions that can be investigated further. Firstly, the

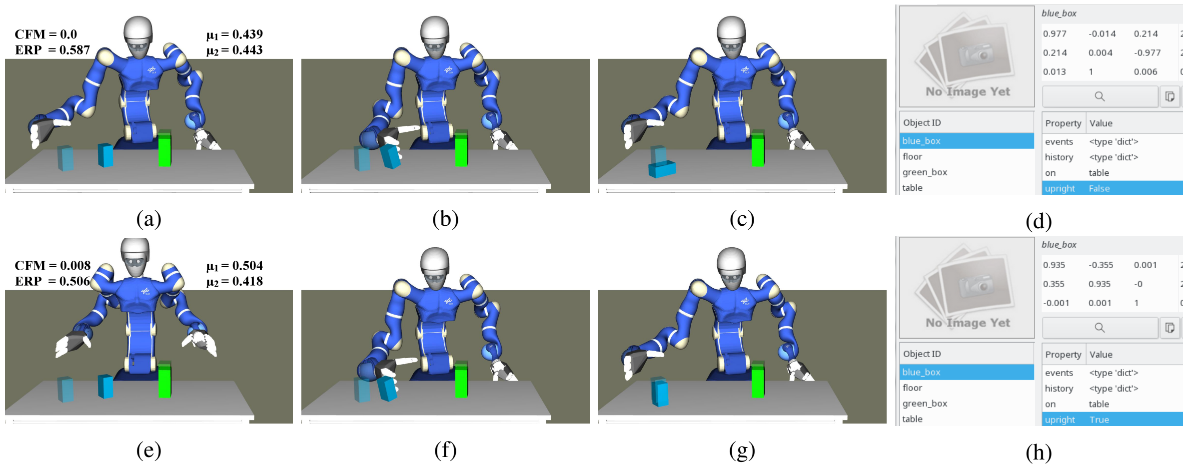

Fig. 4: Results from two runs with different parametrization with bad parameterization (first line) and good parameterization (second line): (a) and (e) before the recorded telemetry is replayed, (b) and (f) during replay, (c) and (g) after the replay has finished, and (d) and (h) the extracted symbolic world state. Generally the transparent blue box marks the ground truth position of the blue box after the experiment.

open simulation should be augmented by feedback from sensors to minimize the prediction error. Secondly, instead of assuming absolute discrete states, the framework should be extended to being able to cope with noise and uncertainty.

ACKNOWLEDGMENT

This work was partially supported by the Bavarian Ministry of Economic Affairs and Media, Energy and Technology, and the project SMiLE.

REFERENCES

[1] C. Borst, T. Wimbock, F. Schmidt, M. Fuchs, B. Brunner, F. Zacharias, P. R. Giordano, R. Konietschke, W. Sepp, S. Fuchs, C. Rink, A. Albu-Schaffer, and G. Hirzinger, "Rollin'justin-mobile platform with variable base," in *Proc. 2009 IEEE Int. Conf. Robotics and Automation (ICRA)*, Kobe, Japan, May 2009, pp. 1597–1598.

[2] J. Artigas, R. Balachandran, C. Riecke, M. Stelzer, B. Weber, J. H. Ryu, and A. Albu-Schaeffer, "Kontur-2: Force-feedback teleoperation from the international space station," in *Proc. 2016 IEEE Int. Conf. Robotics and Automation (ICRA)*, Stockholm, Sweden, May 2016, pp. 1166–1173.

[3] N. Y. Lii, D. Leidner, P. Birkenkampf, B. Pleintinger, R. Bayer, and T. Krueger, "Toward scalable intuitive telecommand of robots for space deployment with meteron supvis justin," in *Proc. 14th Symp. Advanced Space Technologies in Robotics and Automation (ASTRA)*. Leiden, The Netherlands: European Space Agency (ESA), Jun 2017.

[4] D. Kortenkamp, R. P. Bonasso, D. Ryan, and D. Schreckenghost, "Traded control with autonomous robots as mixed initiative interaction," in *AAAI Spring Symp. Mixed Initiative Interaction*, Stanford, CA, USA, 1997, pp. 89–94.

[5] M. Jouvet, *The Paradox of Sleep: The Story of Dreaming*, ser. Bradford book. MIT Press, 2000.

[6] T. Schmidt, K. Hertkorn, R. Newcombe, Z. Marton, M. Suppa, and D. Fox, "Depth-based tracking with physical constraints for robot manipulation," in *Proc. 2015 IEEE Int. Conf. Robotics and Automation (ICRA)*, Seattle, USA, May 2015, pp. 119–126.

[7] S. Coradeschi and A. Saffiotti, "An introduction to the anchoring problem," *Robotics and Autonomous Systems*, vol. 43, no. 2, pp. 85 – 96, 2003.

[8] M. A. Goodrich, J. W. Crandall, and E. Barakova, "Teleoperation and beyond for assistive humanoid robots," *Reviews of Human factors and ergonomics*, vol. 9, no. 1, pp. 175–226, 2013.

[9] S. Hayati and S. T. Venkataraman, "Design and implementation of a robot control system with traded and shared control capability," in *Proc. 1989 Int. Conf. Robotics and Automation*, May 1989, pp. 1310–1315 vol.3.

[10] J. A. Wolfe, B. Marthi, and S. J. Russell, "Combined task and motion planning for mobile manipulation." in *Proc. 20th Int. Conf. Automated Planning and Scheduling (ICAPS)*, 2010, pp. 254–258.

[11] D. Leidner, A. Dietrich, F. Schmidt, C. Borst, and A. Albu-Schffer, "Object-centered hybrid reasoning for whole-body mobile manipulation," in *2014 IEEE Int. Conf. Robotics and Automation (ICRA)*, May 2014, pp. 1828–1835.

[12] O. Lipmann and H. Bogen, *Naive Physik: Arbeiten aus dem Institut für Angewandte Psychologie in Berlin; theoretische und experimentelle Untersuchungen über die Fähigkeit zu intelligentem Handeln*. JA Barth, 1923.

[13] P. J. Hayes, "The naive physics manifesto," in *Expert Systems in the Microelectronic Age*, D. Michie, Ed. Edinburgh University Press Edinburgh, 1979, vol. 220, pp. 242–270.

[14] B. Johnston and M.-A. Williams, "A generic framework for approximate simulation in commonsense reasoning systems." in *Proc. 2007 AAAI Spring Symp.: Logical Formalizations of Commonsense Reasoning*, Stanford, USA, Mar 2007, pp. 71–76.

[15] L. Mösenlechner and M. Beetz, "Using physics- and sensor-based simulation for high-fidelity temporal projection of realistic robot behavior," in *Proc 19th Int. Conf. Automated Planning and Scheduling*, Thessaloniki, Greece, Sep 2009, pp. 249–256.

[16] ——, "Parameterizing Actions to have the Appropriate Effects," in *Proc. 2011 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, San Francisco, CA, USA, Sep 2011.

[17] M. Pomarlan, D. Nyga, M. Picklum, S. Koralewski, and M. Beetz, "Deeper Understanding of Vague Instructions through Simulated Execution (Extended Abstract)," in *Proc. 2017 Int. Conf. Autonomous Agents & Multiagent Systems*, Sao Paulo, Brazil, May 2017, pp. 1694–1696.

[18] S. Lemaignan, R. Ros, E. A. Sisbot, R. Alami, and M. Beetz, "Grounding the interaction: Anchoring situated discourse in everyday human-robot interaction," *International Journal of Social Robotics*, vol. 4, no. 2, pp. 181–199, 2012.

[19] D. Leidner, C. Borst, and G. Hirzinger, "Things are made for what they are: Solving manipulation tasks by using functional object classes," in *Proc. 2012 12th IEEE-RAS Int. Conf. Humanoid Robots (Humanoids)*, Nov 2012, pp. 429–435.

[20] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *Proc. 2004 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Sendai, Japan, Sep 2004, pp. 2149–2154.

[21] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proc. 2011 IEEE Int. Conf. Robotics and Automation*, Shanghai, China, May 2011, pp. 3400–3407.

[22] T. Laue and M. Hebbel, "Automatic parameter optimization for a dynamic robot simulation." in *RoboCup*. Springer-Verlag Berlin Heidelberg, 2008, pp. 121–132.