Human-Robot Assembly of 3D-Printed Building Components Combining Motion Planning and Dynamic Movement Primitives

Arwin Hidding[1,2], Tom Lim[1], Henriette Bier[2], and Luka Peternel[1](✉)

[1] Cognitive Robotics (ME), Delft University of Technology, Delft, The Netherlands
{a.j.hidding,l.peternel}@tudelft.nl
[2] Robotic Building (BK), Delft University of Technology, Delft, The Netherlands

Abstract. Constructing a Martian habitat presents significant challenges due to extreme temperature variations and a low-density and -pressure atmosphere. To address these challenges a habitat constructed from prefabricated, interlocking Voronoi-based components that are assembled by human-robot collaboration has been explored in the Rhizome projects at TU Delft. In this paper, we propose a combined robot motion planning and learning method that can optimize human involvement in assembly tasks in on-site construction. The proposed hybrid approach exploits motion planning to create motion trajectories for aspects of the task where robot autonomy is capable of solving the problem on its own using sensors and intelligence. When the task becomes too difficult for existing planning capabilities, the human can step in and teach motion trajectories via kinaesthetic demonstration using Dynamic Movement Primitives (DMPs). The trajectories are then executed on the low level by an impedance controller to handle the physical interaction with the environment during the assembly. The decision-making process is managed by a behavior tree.

Keywords: Human-Robot Interaction · Robot Learning · Robotic Assembly · Martian Habitat

1 Introduction

Human space exploration with long-term missions in off-Earth environments requires human habitats, which have to be constructed on-site. Mission payload constraints and remoteness pose serious challenges to the habitat construction process, where the availability of specialized tools and large machines is very limited. In this case, collaborative robots present themselves as excellent universal tools/machines that can team up with humans to perform a variety of contribution tasks. to this end, the Rhizome projects[1] at TU Delft, co-funded by the

[1] http://cpa.roboticbuilding.eu/index.php?title=Rhizome2.

K. Jovanović et al. (Eds.): RAAD 2025, MMS 190, pp. 513–520, 2025.
https://doi.org/10.1007/978-3-032-02106-9_57

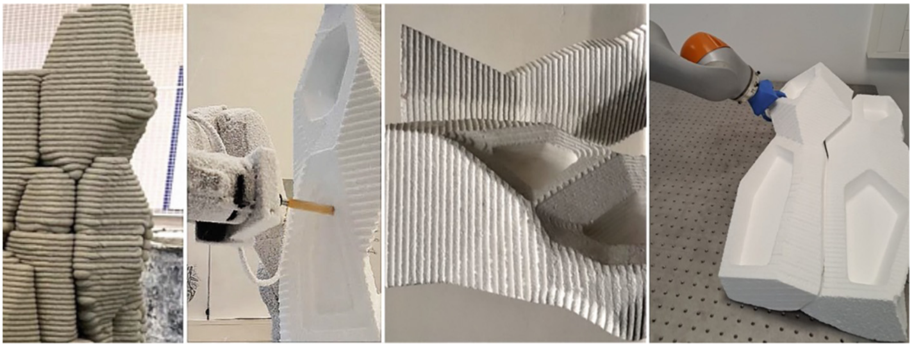

Fig. 1. Robotically milled mock-up components emulating real Martian building components (left and center) that are used in the human-robot collaborative assembly experiments in this study (right).

European Space Agency (ESA) and Vertico, aim to advance the design and construction of a Martian habitat. The habitat is constructed from prefabricated, interlocking Voronoi-based components that are assembled by Human-Robot Interaction (HRI) (Fig. 1).

The Voronoi-based design of building components facilitates the 3D printing using In-Situ Resource Utilization (ISRU) techniques [3] and structural interlocking to create a self-supporting compression structure, where the compressive forces firmly hold the interlocking parts together. All components have unique Voronoi-based shapes and have integrated grabbing holes that allow grippers mounted on the robotic arms of rovers to pick and place the components. For testing the assembly procedure, the components were robotically milled from Expanded PolyStyrene (EPS) thus ensuring that the weight of the components is suitable for the Kuka LBR iiwa robot payload (Fig. 1).

These components must then be assembled together to create structures for human habitats. We approach this challenge by using collaboration between humans and robots, leveraging their respective strengths, where humans better handle cognitively intensive tasks, while robots are more suited for high effort, precision, and speed. We explored HRI-supported approaches in the assembly of several mock-up components [2]. The robot used Computer Vision (CV) to detect the position and orientation of the grasping holes on the components and moved into the vicinity. The human then haptically guided the robotic gripper to perform a cognitively complex grasping operation after which the robot could autonomously move the component. While effective, the limitation of this approach was that the robot's autonomy was limited to simple point-to-point trajectory movements and the robot could not learn from human guidance.

More complex trajectories can be generated using motion planning methods [6]. State-of-the-art robotic motion planning frameworks, such as MoveIt! [4], can generate high degree-of-freedom trajectories to achieve optimal robot motion through cluttered environments. On the other hand, learning from demonstra-

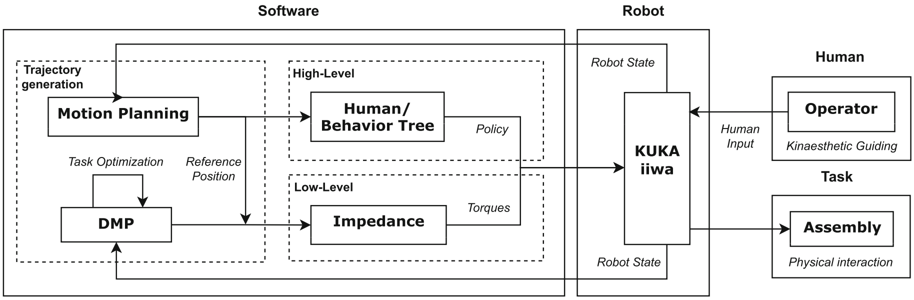

Fig. 2. System overview. The human operator can provide key pre-set configurations related to the task to the motion planning algorithm. The human operator can also use kinesthetic demonstration to create DMPs. The behavior tree component governs the use of the motion planning algorithm and DMPs in different phases of the task.

tion (LfD) methods enable robots to learn the motion directly from human guidance [11]. One of the most common methods employed in LfD manipulation tasks is Dynamic Movement Primitives (DMPs) [12], where trajectories can be learned, adjusted, and generalized in real-time. DMPs have been also used to solve various robotic assembly tasks [1,7,8,10]. A key advantage of LfD is that it offers an easy way for humans to provide corrections to the robot. However, demonstrations can be time-consuming and human involvement can be costly. On the other hand, motion planning methods require little to no human involvement, however, they may fail in cognitively more complex aspects of the task. Based on the recent survey [12], a method combination motion planning method with DMP-based LfD is missing.

To address this gap, this paper presents a novel framework combination motion planning method with DMP-based LfD for automating the assembly of extraterrestrial habitats constructed from Voronoi-based interlocking components. Motion planning method to create trajectories for aspects of the task where robot autonomy is capable of solving the problem on its own using sensors and intelligence. When the task becomes too difficult for existing planning capabilities, the human can step in and teach motion trajectories via kinaesthetic demonstration using DMPs. The trajectories are then executed on the low level by an impedance controller to handle the physical interaction with the environment during the assembly. The decision-making process is managed by a behavior tree [5]. This approach reduces the amount of direct human involvement during assembly tasks in the hazardous extraterrestrial environment, enhancing both safety and efficiency.

2 Methods

The overview of the proposed framework is shown in Fig. 2. The operator can provide key pre-set configurations related to the task to the motion planning

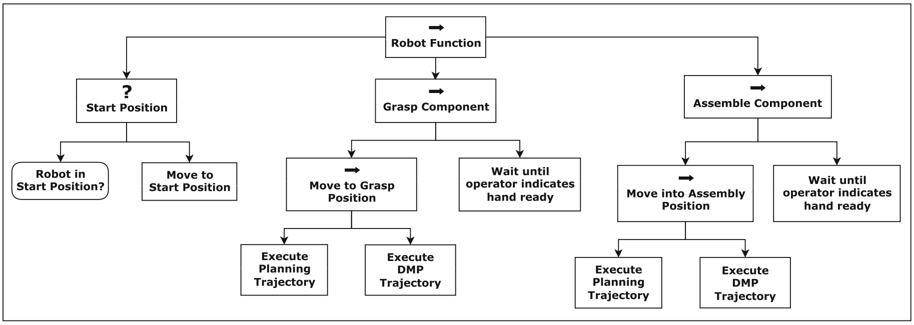

Fig. 3. Visual representation of the designed behavior tree for the assembly task.

algorithm. The human operator can also use kinesthetic demonstration to create DMPs to handle cognitively more complex aspects of the tasks. The segments of trajectories (either generated by planning or learned via DMPs) are then executed by an impedance controller, ensuring compliant interaction for the assembly. We designed a behavior tree to manage the decision-making process of when to do what and when to use planning or DMPs. The whole framework was implemented in ROS.

Assembly tasks can usually be broken down into two types of movements: the first is moving the parts between grasping and assembly locations and the second is the more precise grasping and interlocking of the parts themselves. A motion planning approach is most suitable for the first types of movements, while a learning approach suits the second types of movements.

2.1 State Management Using Behavior Trees

To control the task at a high level we employed a behavior tree. Figure 3 shows the behavior tree that we designed to handle state management for the proposed human-robot collaborative assembly task. The tree starts with a sequence node with three main children. The first main child is a fallback node with a condition that checks whether the robot is already in the start position, otherwise, it executes an action to move it into the initial position. The second main child is another sequence node related to moving into a grasping position and performing grasping. The movement from the start position to the grasping position is executed with motion planning, and if successful the strategy is switched to DMP that handles the grasping. The third main child is also a sequence node and is dedicated to placing a component. The motion planning handles the movement from the grasping position to the placing position and if successful the strategy is switched to DMP, which handles the component inter-locking.

In this context, the behavior tree is key in coordinating when to use motion planning for the larger movements and when to switch to DMPs for the grasping and placing tasks. This ensures that the appropriate method is used at the right time during the assembly process. Our design was implemented using the BehaviorTree. CPP library[2], which also supports ROS integration.

2.2 Motion Planning

The goal of motion planning is to find an optimal robot trajectory between the desired task poses while accounting for various constraints in the environment. In a given assembly task, motion planning is best suited to handle robot motion for the actions of reaching for and carrying of components. Compared to grasping and interlocking, these actions are more predictable and the planning can rely on sensory information such as computer vision.

For robot motion planning we employed MoveIt!, which allows for the generation of highly efficient trajectories, making it well-suited for tasks involving large, reaching motions. It is also integrated with the Robot Operating System (ROS), which makes for a relatively easy development process. We also created an interface that allows the operator to store/select key poses important for the task for the planner to consider.

2.3 Dynamic Movement Primitives

Unlike moving from the starting position to the grasping position, and from the grasping position to the placing position where planning is relatively straightforward with good visual sensory information, grasping and interlocking are much more variable and difficult to predict. Thus, LfD offers a good solution for these challenging phases of the tasks. To this end, we used DMPs to learn human demonstrations via kinesthetic guiding of the end-effector for complex motion related to grasping and interlocking. The DMPs encoded the robot end-effector pose in Cartesian space.

Since our system works in ROS, we had to integrate DMPs with ROS. We created another ROS package that is a ROS wrapper around the DMP repository described in [14]. For the execution of the DMPs, a ROS node was implemented, which was then directly published to the ROS-integrated impedance controller.

3 Experiments

The goal of the experiments was to validate the proposed method that combines motion planning and DMPs for a component assembly task. The experiment setup included a Kuka LBR iiwa robot and two milled mock-up components. The task involved moving the milled components into the assembly location and then interlocking them together using HRI. The robot control system had to switch between motion planning and DMPs based on the phases of the task.

Initially, both components were in the location on the left side of the robot. Figure 4 shows the sequence of photos from the experiments after the first component was already placed at the assembly location. The graph in Fig. 5 shows

[2] https://github.com/BehaviorTree/BehaviorTree.CPP.

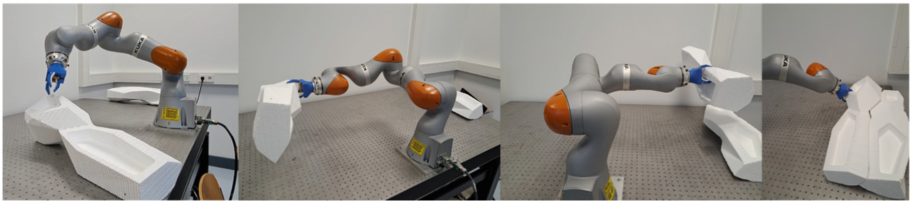

Fig. 4. Sequence of photos during the experiments showing key stages of the assembly task. The first photo shows the scene just prior to grasping. The second and third photos show the beginning and the end of component-carrying motion, respectively. Finally, the fourth photo shows the scene after the two components were interlocked. The actual measured robot motion is shown in the corresponding graph in Fig. 5.

the same sequence in terms of robot end-effector motion, where magenta dots with numbers indicate the corresponding photo in Fig. 4. The first photo was taken just after the motion planner moved the robot from the start position to the grasping position.

Between points 1 and 2, the DMPs were used to facilitate human-robot collaboration during the grasping phase. We can observe that the trajectory between points 1 and 2 (grasping phase) is more complex in shape than between the start and point 1 (reaching phase). After a firm grasp was secured, the motion planning algorithm derived an optimal movement trajectory from the grasping position (point 2) to the placing position (point 3). This trajectory then brought the robot end-effector and the grasped component into the vicinity of the assembly location.

In the final phase, the control system switched to DMPs again to facilitate human-robot collaboration during the interlocking action of the two components. Again, we can observe that the trajectory between points 3 and 4 (interlocking phase) is more complex in shape than between points 2 and 3 (carrying phase). When the two components were assembled, the task was completed and the experiment was successful.

4 Discussion

The experiments demonstrated that the combination of motion planning and LfD with DMPs provides favorable results for component assembly tasks. Motion planning is effective for smooth large-scale movements such as reaching and carrying, while DMPs are effective for small refined complex movements that include human-robot collaboration. In motion planning, the positional and orientational data received from the CV can help with creating optimal smooth large-scape movements. However, information from CV is not accurate enough to create fine motions that involve unpredictable physical interaction to adequately facilitate the grasping and placing of components. Whereas the strength of DMPs is learning to fine-tune movements that are otherwise hard to plan. Since DMPs do not consider joint and workspace limitations, they facilitate human adjustments

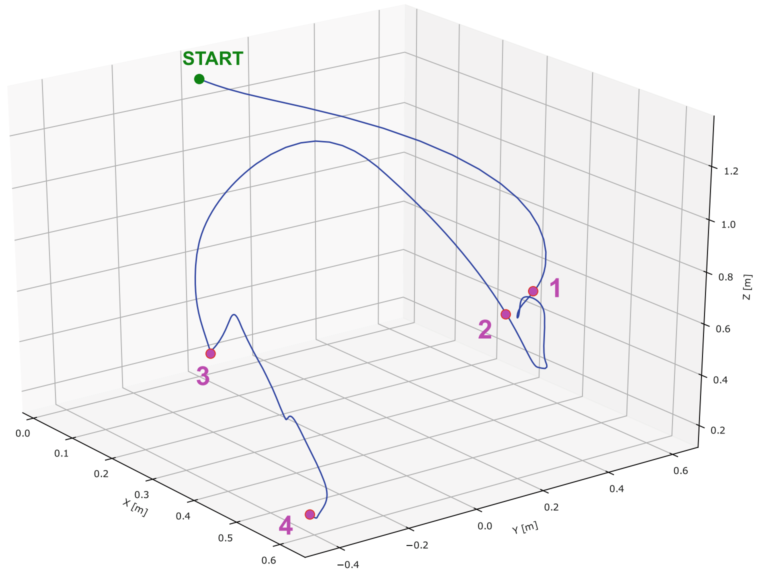

Fig. 5. Position of the end-effector in 3D space during the assembly task. The magenta points indicate the snapshots of the experiment as shown in Fig. 4.

during grasping and interlocking. However, the lack of joint and workspace limitations makes optimal large-scale movements difficult. By combining the two methodologies the framework becomes flexible and efficient for complex assembly tasks.

With the addition of LfD, the framework becomes more suitable for extraterrestrial on-site assembly where many unpredictable situations need to be addressed. Therefore, in the proposed approach, humans remain involved when the complexity of the assembly is too difficult for the AI. Nevertheless, human on-site involvement in hazardous off-Earth environments should ideally be minimized. Future work will investigate the possibility of using teleoperation [9,13], so HRI is available for certain complex assembly tasks while keeping humans off the construction site.

In future work, the assembly of the components will be computationally simulated and take into account Martian physics. This method will validate the assembly logic in Martian conditions that cannot be replicated on Earth due to the differences in gravity. This approach also allows the entire building process to be tested digitally in advance of any physical testing. Based on these simulations, the gripping methodology will be adjusted to meet multi-rover and co-assembly requirements.

Acknowledgments. The work was supported by the Rhizome 2.0 project co-funded by ESA (contract number: 4000141650) and Vertico.

References

1. Abu-Dakka, F.J., Nemec, B., Kramberger, A., Buch, A.G., Krüger, N., Ude, A.: Solving peg-in-hole tasks by human demonstration and exception strategies. Ind. Robot: Int. J. **41**(6), 575–584 (2014)
2. Bier, H., Hidding, A., Prendergast, J., Peternel, L.: Computer vision- and human-robot interaction-supported assembly for collaborative off-earth habitat construction. In: Bhateja, V., Dey, M., Simic, M. (eds.) FICTA 2024 2024. Smart Innovation, Systems and Technologies, vol. 421, pp. 103–114. Srpinger, Singapore (2025). https://doi.org/10.1007/978-981-96-0143-1_9
3. Calabrese, G., Hidding, A., Bier, H.: Review of cementless materials for 3d printing of on-and off-earth habitats. Adapt. On-and Off-Earth Environ. 39–58 (2024)
4. Chitta, S., Sucan, I., Cousins, S.: Moveit![ros topics]. IEEE Robot. Autom. Mag. **19**(1), 18–19 (2012)
5. Colledanchise, M., Ögren, P.: Behavior Trees in Aobotics and AI: An Introduction. CRC Press (2018)
6. Kavraki, L.E., LaValle, S.M.: Motion planning. In: Springer Handbook of Robotics, pp. 139–162. Springer, Cham (2016)
7. Kramberger, A., Kunic, A., Iturrate, I., Sloth, C., Naboni, R., Schlette, C.: Robotic assembly of timber structures in a human-robot collaboration setup. Front. Robot. AI **8**, 768038 (2022)
8. Nemec, B., Simonič, M., Ude, A.: Learning of exception strategies in assembly tasks. In: 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 6521–6527. IEEE (2020)
9. Peternel, L., Ajoudani, A.: After a decade of teleimpedance: a survey. IEEE Trans. Hum.-Mach. Syst. **53**(2), 401–416 (2022)
10. Peternel, L., Petrič, T., Babič, J.: Robotic assembly solution by human-in-the-loop teaching method based on real-time stiffness modulation. Auton. Robot. **42**, 1–17 (2018)
11. Ravichandar, H., Polydoros, A.S., Chernova, S., Billard, A.: Recent advances in robot learning from demonstration. Ann. Rev. Control, Robot. Auton. Syst. **3**(1), 297–330 (2020)
12. Saveriano, M., Abu-Dakka, F.J., Kramberger, A., Peternel, L.: Dynamic movement primitives in robotics: a tutorial survey. Int. J. Robot. Res. **42**(13), 1133–1184 (2023)
13. Si, W., Wang, N., Yang, C.: A review on manipulation skill acquisition through teleoperation-based learning from demonstration. Cogn. Comput. Syst. **3**(1), 1–16 (2021)
14. Stulp, F., Raiola, G.: Dmpbbo: a versatile python/c++ library for function approximation, dynamical movement primitives, and black-box optimization. J. Open Source Softw. **4**(37), 1225 (2019)