Combining the Robot Operating System with Building Information Modeling for Robotic Applications in Construction Logistics

Camilla Follini[1,2(✉)], Michael Terzer[2], Carmen Marcher[1,2], Andrea Giusti[2], and Dominik Tobias Matt[1,2]

[1] Faculty of Science and Technology, Free University of Bozen-Bolzano, Bolzano, Italy
camilla.follini@natec.unibz.it
[2] Fraunhofer Italia Research s.c.a.r.l., Bolzano, Italy

Abstract. Logistics in construction sites represents one of the main causes of musculoskeletal disorders for workers in the long run. For this reason, logistics represents a task that is highly suitable for automation. However, the application of robotics in construction is hindered by the on-site environment, which is often unstructured and subject to frequent change. To tackle this problem, the system proposed in this paper is a collaborative robotic platform aimed at human-centered applications on-site. The platform follows an operator while it carries heavy loads, such as materials and equipment, and it stops when it is near to the assisted worker. The system does not only avoid obstacles that are detected by its sensors, but it also navigates thanks to its knowledge of geometric and semantic information of the building project. This is achieved through a bridge between the Robot Operating System and the project data contained in the Building Information Model. The proposed system was developed and tested in laboratory, where it followed an operator while avoiding dynamic obstacles and areas marked in the building project file. This system represents a step forward towards the application of on-site collaborative construction robots.

Keywords: Industry 4.0 · Robot Operating System · Building information modeling · Logistics · Collaborative robotic platform · Construction robotics

1 Introduction

The construction industry represents one of the main economies worldwide. In the European Union, it generates about the 9% of the gross domestic product [1]. Nonetheless, construction is often plagued by several problems such as inefficiency, low productivity, a high number of accidents, low attractivity, and a general lack of innovation [2]. This becomes even more evident when compared to other industries

C. Follini and M. Terzer—Contributed equally to this work.

S. Zeghloul et al. (Eds.): RAAD 2020, MMS 84, pp. 245–253, 2020.
https://doi.org/10.1007/978-3-030-48989-2_27

pushing towards Industry 4.0 [3], such as manufacturing, where collaborative robotic solutions are arising [4]. Logistics can be considered one of the harshest jobs on construction sites. In indoor tasks, it consists almost entirely of handling operations of heavy loads, such as lowering, lifting, and carrying. These operations are usually boring, repetitive, fatigue-intensive, and amongst the riskiest factors of musculoskeletal disorders in construction [5]. Moreover, logistics is considered work to be minimized according to lean construction standards. That is, because it represents necessary work that produces no added value in the construction supply chain [6]. For these reasons, automation of logistics operations on-site is a promising research area. To this end, some solutions have explored the use of Mobile Robotic Platforms (MRPs) navigating the construction site while carrying loads. Japanese construction companies, in particular, have been developing systems ranging from one to several hundred kilograms in payload [7]. A notable example is Shimizu's Robo-Carrier, a forklift that autonomously navigates the construction site to transport material to other robotic systems developed by the company in charge of the assembling [8]. The knowledge of the building geometry is particularly important for robotic navigation in construction sites and for localization of relevant logistics areas (such as temporary storage points and site warehouse). Among the different software tools for project design, Building Information Modeling (BIM) is steadily spreading in the construction industry [9]. Its main advantage over more traditional tools is that it enables the linking of information to geometric 3D objects. For instance, a 3D object can store information about its material, cost, producer, and construction scheduling time. When connected to scheduling information, the model is referred to as "4D BIM" [9]. Given this, BIM has been used as basis for more than its original purpose as project planning tool, and it has been researched as enabler for autonomous mobile robots on-site. For instance, research has been carried out on algorithm-based route planning in 2D [10] and 3D [11], selecting the most convenient path in buildings. BIM has also been linked with computer vision in view of object recognition. For example, in [12] it is suggested that recognition from site point clouds is effective with objects with a high Level of Detail (LOD). Based on these advancements, the use of BIM has been extended to robotic navigation. For instance, object-recognition-based navigation has been proposed in [13]. In this research, real-life pictures are segmented and compared with a digital 3D object repository. Upon recognition of the object and its real-world scale, the system updates the BIM file of the building with the library object. The connection of BIM to the Robot Operating System (ROS) [14] is explored in [15]. In that research, a 4D-BIM file is used to retrieve waypoints where the construction progress must be monitored. After planning a route comprising all the waypoints, the MRP navigates the site autonomously to collect data. Obstacle detection has been implemented to avoid collisions due to unforeseen changes or mapping misalignments.

Compared to the literature above, our solution focuses on a system for site logistics that is aimed at collaborating with human workers. In particular, our proposed approach tackles two relevant barriers for employment of MRPs on-site: safety and acceptance [16]. Concerning the latter, the interface of the system has been developed to be intuitive and useable by workers with little experience of robotic devices. Therefore, the interface requires very little input from the workers' side. The system reads their position, and it follows them autonomously while it carries heavy materials. Safety is ensured by the system stopping when it finally reaches the operator. The robustness of the system is increased through dynamic obstacle avoidance and fusion of data detected by the sensing capabilities of the robot and BIM.

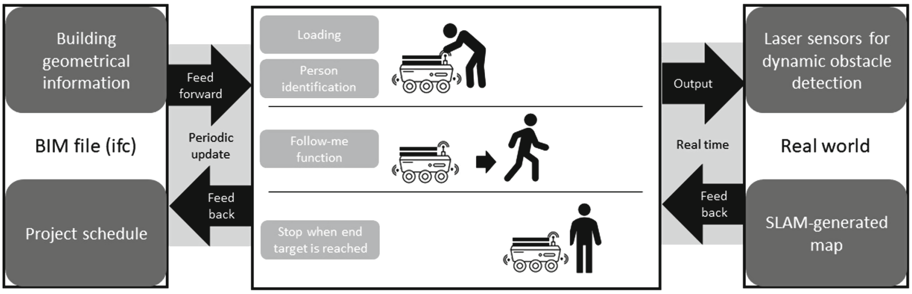

Fig. 1. Conceptual representation of the system main goals

2 Proposed Approach

A barrier for the implementation of MRPs in construction is the site environment, which is often unstructured and subject to continuous change. At the same time, a complete description of the geometry of the building and the scheduling of tasks exists before the start of construction works, sometimes under a common repository that is the BIM file.

We aim at the development of a robotic system that takes advantage of 3D and 4D BIM digital data to navigate the construction site while it follows an operator and it carries heavy materials or equipment (see Fig. 1). The data is overlapped to the map already generated by the system's LIDAR sensors, and it helps the system avoid obstacles that cannot be recognized by them, such as holes in the ground and overhanging objects. The data is transmitted to the platform in the form of a ROS 2D *costmap*. Here, elements are filtered not only by their potential of collision with the system, but also by their scheduled construction time. To simplify the problem, the following assumptions were considered: 1) a BIM file exists which has parameters for scheduling and whose level of detail is high enough for navigation; 2) the MRP has means to determine the current date; 3) the MRP can locate its real-world starting point in the BIM file. The proposed method has been conceptually divided into three main parts: 1) the follow-me function, 2) the dynamic obstacle detection, and 3) the extraction of data from the BIM file to ease navigation.

3 Implementation

The follow-me function on the MRP has been developed using an indoor tracking system. Indoor tracking systems deliver highly accurate measurements of position and orientation of the tracked bodies. Therefore, they are suited for this preliminary setup, and the data obtained from them can represent a ground truth for further development. The desired trajectory of the follow-me function gets calculated by tracking the position of both the MRP and the operator. These two are tracked thanks to spherical reflective markers that have been fixed respectively on the platform and on a construction helmet. The relative positions are then translated to a world coordinate system, which is defined by the tracking system. The implementation of dynamic obstacle avoidance has been realized through two planar LIDAR sensors. These sensors provide the possibility to measure distances on a 2D plane around them. BIM is used to retrieve information on objects unidentifiable or difficult to identify by the LIDAR sensors, and on planned building geometries as well as metadata. The metadata considered concern scheduling information, in particular the date in which construction of objects would start and end. To achieve this, objects are extracted from an Industry Foundation Classes (IFC) file, which is the open format that can be generated by every BIM software tool. The data extracted from this file is translated to a *costmap* that can be used inside of the ROS navigation stack. The map is then transmitted and superimposed to the one generated by the LIDAR sensors. Before the real system was tested, several simulations have been performed. The setup consisted of a PC running ROS and Gazebo [17], a ROS-enabled software for robotic simulation. Since the producer of the MRP delivers CAD models and Unified Robot Description Format (URDF) descriptions for their product, a realistic simulation within Gazebo suited the authors' needs. The system has then been tested in a protected indoor environment, in view of creating a solid basis to further tailor it to the use in unstructured environments.

3.1 Hardware

The tracking system employed is produced by the company ART[1]. It consists of four tracking cameras, a computing unit in form of a rack, and several marker configurations. The MRP is a Husky A200 from the company Clearpath[2]. This vehicle is specifically designed for research activities in harsh outdoor environments. The user bay of the MRP is equipped with a computing unit running ROS and a network switch. The MRP is equipped with two LIDAR Sensors, respectively pointing forward and backwards, with a 270° planar field of view each. In addition, the MRP is equipped with an Inertial Measurement Unit (IMU) that provides acceleration and angular velocity measurements in the cartesian coordinate system. The control PC is a Dell Latitude E6420 running Linux Ubuntu 16.04 and ROS Kinetic. This PC is needed to launch the overall control architecture, as well as the bridge to the tracking system. The LIDAR sensors are connected over an ethernet interface to the computer of the MRP. The MRP includes a local network, that has a wireless connection to the external network. The external network consists of a router that connects the computing unit of the tracking system and the ROS-led PC. Since the system includes two computer running ROS, a time synchronization of the two systems is required. This

[1] ART company website: https://ar-tracking.com/.

[2] Clearpath company website: https://clearpathrobotics.com/.

synchronization is realized over the NTP *deamon chrony*[3], running both on the MRP and the control PC. An overview of the different hardware components can be seen in the set-up shown in Fig. 2.

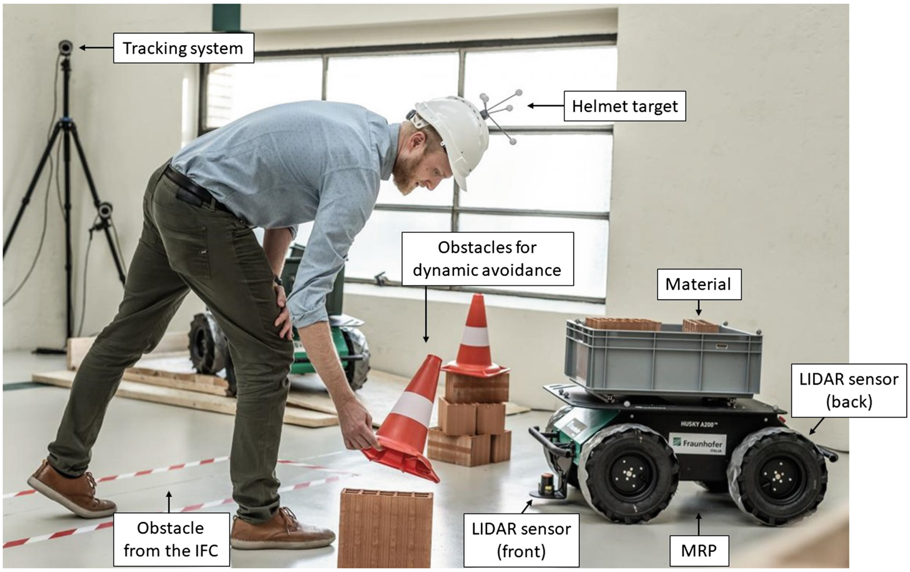

Fig. 2. Picture of the laboratory set-up for the testing phase (Image adapted from: live-style.it)

3.2 System Architecture in ROS

The overall software architecture of the system is shown in Fig. 3. Here, nodes have been distinguished between those that have been developed by the authors and those available as open-source nodes or ROS packages. The PC running on the MRP runs one of the master nodes named *roscore*. This node interfaces the LIDAR-scanner-generated messages (*/laser*) and a node that controls the MRP (*husky_ctrl*). A second *roscore* is run by the control PC. It enables: 1) the *art_bridge* node, which interfaces the data coming from the tracking system to *tf*-messages [18] in ROS, 2) a controller node for the follow-me algorithm, 3) the *nav2d*[4] package, visualized as Operator-node in Fig. 2, that includes nodes for various purposes: (such as mapping managing, trajectory planning, and obstacle avoidance), and 4) a node for the map server, which feeds the static map coming from the *BIM-to-ROS* toolchain into the global costmap.

[3] Chrony official website https://chrony.tuxfamily.org/.

[4] *Nav2d* page on ROS wiki: http://wiki.ros.org/nav2d.

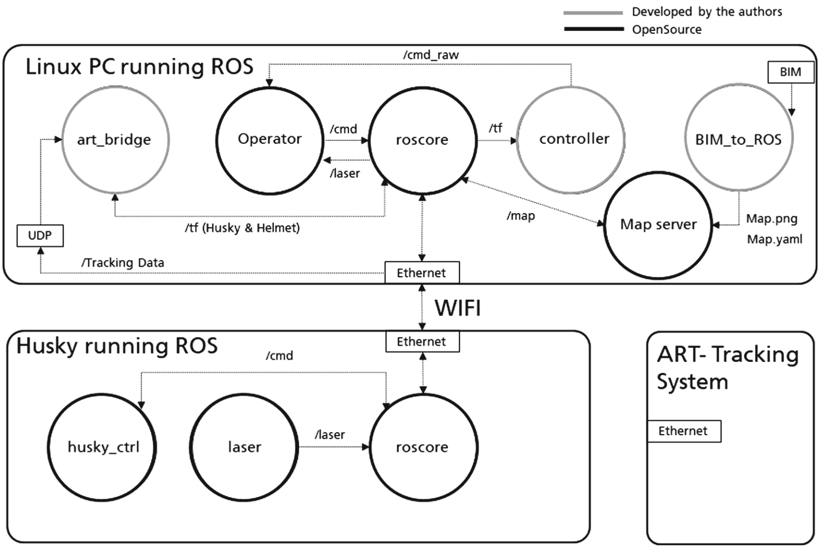

Fig. 3. Software architecture based on ROS

3.3 Navigation and Integration of IFC Data

The *art_bridge* node receives the data from the ART tracking system over a User Datagram Protocol (UDP) interface, and it composes a *tf* message according to each coordinate frame received. This use of *tf*-frames within ROS enables their visualization (e.g. in RViz [19]), as well as the use of related coordinate system transformation tools.

The tracking system uses a custom data format for each tracked coordinate frame. The *follow_me_ctrl* node enables the follow-me algorithm. The inputs of this node are the *art_bridge tf* messages, while the output is a custom message type *cmd*, defined by the nav2d package. The node registers the current position of the vehicle, and it computes the vector to the target, which is defined by the coordinate frame of the tracked helmet. Thanks to that, the desired turn value and speed get calculated. When the MRP is close to the target, the system sets its velocity to 0 to stop the MRP in the near proximity of the operator. The *BIM_to_ROS* node is the one used to superimpose the BIM-generated map to the one based on LIDAR information. The node is based on the *IFCOpenShell*[5] library. It first parses an IFC file to read and store the geometry and the metadata of BIM objects, grouped under their building floor. It then filters objects relevant to the navigation based on their IFC type (e.g. *IfcDoor*, *IfcWall*, etc.). Filtering is also dependent on objects geometry (i.e. whether it imposes a risk of collision with the MRP), and their planned starting and ending construction date. Thereafter, the objects that are not filtered out are sectioned by a plane. The result is a series of meshes representing the intersection between the 3D objects and the sectioning plane. This section is finally printed as a picture, where white pixels represent the empty space

[5] IFCOpenShell library website: http://ifcopenshell.org/.

whereas black pixels the obstacles. This picture is translated to a ROS *costmap* by the creation of a *yaml* file, which contains further mapping information, such as the map scale and origin. The *costmap* is sent to the navigation control node through the *map_server*[6], and it can be activated by enabling the static map on RViz. The BIM-based map is superimposed to the map dynamically generated by the LIDAR sensors.

The result of this superimposition is shown in Fig. 4. Here, a simple IFC file was used, which contained only the lab boundaries (i.e. walls and access door) and an element in the center of the room representing a hole in the ground. This element has been modeled in the BIM file as a skylight that was planned to remain uncovered in a certain timeframe, which contained the experiment date. The hole on the floor level has been chosen because it is an obstacle that cannot be detected by the 2D LIDAR sensors. The same process can be used to highlight small obstacles as well as to create off-limits areas, where undetectable elements have been posed on the ground and should not be moved by the MRP (e.g. electrical cables, measuring tapes). We consider other short obstacles that can be commonly found on construction sites, such as debris or unlevelled floor parts, to be negligible. That is, because we expect that the MRP can drive over them, since the LIDAR is positioned under the plane passing through the center of the wheels. As shown in Fig. 4, there is no distinction between the BIM-generated obstacles and the ones dynamically detected by the LIDARs. Therefore, the MRP adjusts its trajectory to avoid both.

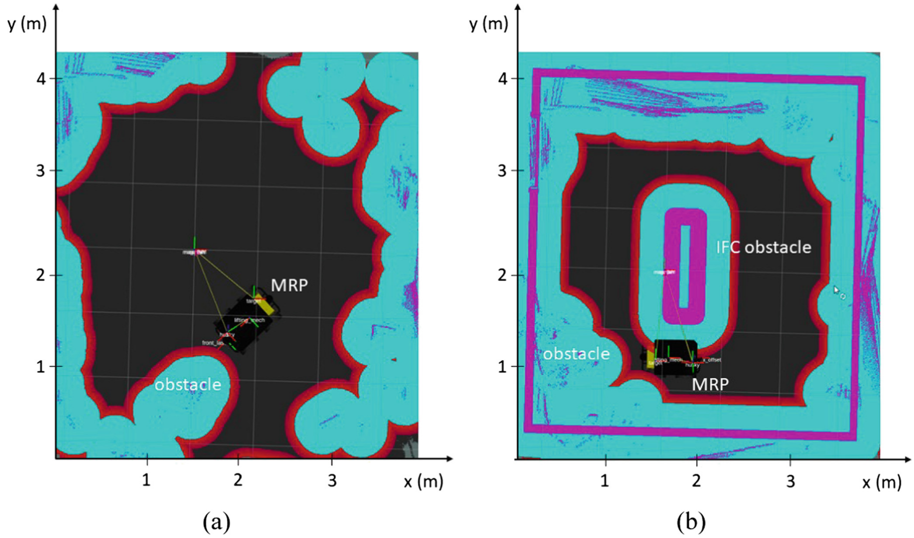

Fig. 4. Visualization on RViz of the laboratory *costmap* without (a) and with (b) the static IFC map overlapping. As shown in Fig. 2, the obstacle in the center of the room is strictly virtual, and its position on the lab floor has been marked with tape.

[6] *Map_server* page on ROS wiki: http://wiki.ros.org/map_server.

4 Discussion and Conclusion

The paper presents the development of a human-centered assistive system to support with logistics on the construction site. It enables an outdoor-capable MRP to follow an operator while avoiding obstacles both detected by its sensors and extracted from a BIM file of the building. Further development will foresee the use of sensors that can be mounted on the MRP, without requiring a fixed tracking system to identify the position of the user. In this work we started from a bottom-up research strategy. Indoor tracking systems deliver highly accurate measurements of position and orientation of the tracked bodies. Therefore, they are suited for a preliminary setup, and the data obtained from them can represent a ground truth for further development. However, this will introduce uncertainty in the detection of the relative position between operator and MRP and requires further development of the follow-me function that assists the system in path planning in case the operator disappears from the MRPs field of view. Additionally, future work will evaluate the feasibility of a bi-directional connection to IFC files, where information on geometries and metadata can be updated to be readable on a BIM software tool.

Further work will also address the evaluation of the actual applicability of the system on a real construction site. This environment is characterized by many more challenges for robotic systems compared to manufacturing factories. Among these, the most relevant are the unstructured space subject to weather conditions, the very rough tolerance of operation (often on a scale of centimeters compared to the plan), the collaboration with workers not used to robotic systems, and finally the economic competition with cheap manual labor. To this end, the applicability potential and the actual decision of adopting this type of advanced system on a construction site can be supported with a dedicated decision support system (DSS) for equipment selection as described in [20].

References

1. European Commission: Construction. https://ec.europa.eu/growth/sectors/construction_en. Accessed 06 Dec 2019
2. European Construction Sector Observatory: Analytical Report - Improving the human capital basis (2017)
3. Kagermann, H., Wahlster, W., Helbig, J.: Recommendations for implementing the strategic initiative INDUSTRIE 4.0 (2013). https://www.din.de/blob/76902/e8cac883f42bf28536e7e8165993f1fd/recommendations-for-implementing-industry-4-0-data.pdf
4. Di Cosmo, V., Giusti, A., Vidoni, R., Riedl, M., Matt, D.T.: Collaborative robotics safety control application using dynamic safety zones based on the ISO/TS 15066:2016. In: Berns, K., Görges, D. (eds.) Advances in Service and Industrial Robotics, pp. 430–437. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-19648-6_49
5. CPWR: Chart Book (6th edition): Fatal and Nonfatal Injuries - Musculoskeletal Disorders in Construction and Other Industries. https://www.cpwr.com/chart-book-6th-edition-fatal-and-nonfatal-injuries-musculoskeletal-disorders-construction-and-other. Accessed 09 Dec 2019

6. Pérez, C.T., Fernandes, L.L.A., Costa, D.B.: A Literature Review on 4D Bim for Logistics Operations and Workspace Management. Presented at the 24th Annual Conference of the International Group for Lean Construction, Boston (2016)
7. Bock, T., Linner, T.: Site logistics robots. In: Construction Robots. Elementary technologies and Single-Task Construction Robots (2016)
8. Shimizu Corporation: A Production System for a New Era. https://www.shimz.co.jp/en/topics/construction/item12/. Accessed 07 Apr 2018
9. Harris, B.N.: Building Information Modeling: A Report from the Field. Building Information Modeling (2016)
10. Schlette, C., Roßmann, J.: Sampling-based floor plan analysis on BIMs. Presented at the 33rd International Symposium on Automation and Robotics in Construction, 21 July (2016). https://doi.org/10.22260/ISARC2016/0004
11. Lin, W.Y., Lin, P.H., Tserng, H.P.: Automating the generation of indoor space topology for 3D route planning using BIM and 3D-GIS techniques. Presented at the 34th International Symposium on Automation and Robotics in Construction, 1 July (2017). https://doi.org/10.22260/ISARC2017/0060
12. Chen, J., Fang, Y., Cho, Y.K.: Performance evaluation of 3D descriptors for object recognition in construction applications. Automat. Constr. **86**, 44–52 (2018). https://doi.org/10.1016/j.autcon.2017.10.033
13. Ferguson, M., Law, K.: A 2D-3D object detection system for updating building information models with mobile robots. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, pp. 1357–1365. IEEE (2019). https://doi.org/10.1109/WACV.2019.00149
14. Quigley, M., Conley, K., Gerkey, B.P., Faust, J., Foote, T., Leibs, J., Wheeler, R.C., Ng, A. Y.: ROS: an open-source robot operating system. In: ICRA 2009 (2009)
15. Ibrahim, A., Sabet, A., Golparvar-Fard, M.: BIM-driven mission planning and navigation for automatic indoor construction progress detection using robotic ground platform. Presented at the 2019 European Conference on Computing in Construction, 10 July (2019). https://doi.org/10.35490/EC3.2019.195
16. Czarnowski, J., Dąbrowski, A., Maciaś, M., Główka, J., Wrona, J.: Technology gaps in human-machine interfaces for autonomous construction robots. Automat. Constr. **94**, 179–190 (2018). https://doi.org/10.1016/j.autcon.2018.06.014
17. Koenig, N., Howard, A.: Design and use paradigms for Gazebo, an open-source multi-robot simulator. In: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566), vol. 3, pp. 2149–2154 (2004). https://doi.org/10.1109/IROS.2004.1389727
18. Foote, T.: tf: The transform library. In: 2013 IEEE Conference on Technologies for Practical Robot Applications (TePRA), Woburn, MA, USA, pp. 1–6. IEEE (2013). https://doi.org/10.1109/TePRA.2013.6556373
19. Kam, H.R., Lee, S.-H., Park, T., Kim, C.-H.: RViz: a toolkit for real domain data visualization. Telecommun. Syst. **60**, 337–345 (2015). https://doi.org/10.1007/s11235-015-0034-5
20. Marcher, C., Giusti, A., Schimanski, C.P., Matt, D.T.: Application of decision support systems for advanced equipment selection in construction. In: Cooperative Design, Visualization, and Engineering, pp. 229–235. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30949-7_26