Full-Scale Prototype for Bricklaying Activity

Michele Ambrosino[1], **Fabian Boucher** [2], **Pierre Mengeot** [2], **and Emanuele Garone**[1]

[1]Service d'Automatique et d'Analyse des Systèmes, Université Libre de Bruxelles, Brussels, Belgium.
[2] NV BESIX SA, Brussels, Belgium.

Michele.Ambrosino@ulb.ac.be, fabian.boucher@besix.com, pierre.mengeot@besix.com,
egarone@ulb.ac.be

Abstract -

A major challenge for construction robotization concerns the use of robots for building activities on site. Several innovative ideas have been tried out in recent years. One of these is the idea of a mechanical mason to make the construction of walls as automated as possible. However, most of the solutions proposed so far have not gone beyond the prototype stage. In this work, we propose a new idea of a multi-robot system able to build walls with large and heavy blocks. After a detailed analysis of the current manual masonry process, the proposed mechanical design is analyzed. To test the feasibility of this innovative approach, a full-scale demonstrator and the entire control algorithm are presented and explained in detail. Experimental results show the efficiency of the proposed multi-robotic system.

Keywords -

Construction robotics; Control Architecture; Bricklaying activity; Multi-robotic System

1 Introduction

Among the different construction operations, masonry has always been an excellent candidate for robotization. It is a quite repetitive and almost deterministic activity that requires setting, in the same way, thousands of identical blocks. Moreover, especially with heavy and large blocks, masonry is one of the most dangerous construction activities. For more than 150 years, research groups around the world have been trying to develop innovative solutions to perform this type of construction activity. As a matter of fact, patents for bricklaying machines have already been announced in 1875 [1] and in 1904 [2]. These first attempts were purely mechanical bricklayers that could not sense anything about their environment. They applied a layer of mortar and mechanically placed a brick at regular intervals. However, all of these attempts never made it far beyond the demonstration stage, and never found any sort of commercial success. Beginning in the late 1980s and early 90s, we start to see attempts based on robotic arms. Unlike the previous machines, which were purely mechanical, these machines had an information processing component. These solutions were based on a high degree-of-freedom robotic arm with sensors and control systems to "feel" the construction environment and to interact with blocks. Despite all the efforts, these attempts saw the same level of success as the previous ones. Most did not get past the level of technical descriptions, and a few reached the level of prototypes, but essentially no progress was made beyond that. Over the years, masonry has declined in importance as a construction technology in the developed world, and with it the interest in automating it. Nowadays, there is one commercial machine, SAM100, offering automation of bricklaying for large straight building facades [3]. SAM is a masonry robot built by Construction Robotics, and has been in use on commercial projects since 2015. This machine is based on a standard industrial manipulator with a gripper mounted on a large mobile base. The bricks are stored in the mobile base. A conveyor belt and a mortar dispenser serve the robotic arm with new bricks covered with mortar. A drawback of this product is that the mobile base moves on rails. Therefore the environment for this system must be structured. Moreover, the robot is a typical industrial rigid arm that is able to work only with bricks of small size. Another commercial machine is Hadrian X by Fastbrick Robotics [4]. This system consists of a big truck equipped with a telescopic robotic arm and a conveyor belt that brings the blocks to the tip of the arm. So far, this robot has been tested only to build low-rise detached houses. The adaptability of this solution in high-rise buildings and in dense urban environments is at the current stage doubtful. Other advanced prototypes on the same subject are DimRob [5], In Situ Fabricator [6], ABLR [7], and a parallel-kinematic manipulator [8]. However, to the best of our knowledge, none of them has gone beyond the advanced prototypical state. The reasons why most attempts so far have failed are different [9]. Several of the proposed solutions are designed to work in very well-structured environments and this is hardly compatible with the reality of most construction sites. Moreover, all of the previous robotic attempts require custom or small blocks which leads to increased cost and poor re-usability of the system. In the project, which this paper is part of, we propose a robotic solution that aims at overcoming some of the main limitations of the designs proposed so far (see also [10, 11]). We propose a robotic solution capable of laying large and heavy construction blocks while remaining lightweight and maneuverable. This innovative design is based on a 'non-rigid' robot, such as a crane, in charge of the macro-movement and of holding most of the weight of the block, and a small rigid robot mounted on an aerial work platform to achieve the desired precision during the fine placement of the block.

The aim of this paper is to show the implementation and preliminary experimental validation of the proposed multi-robot bricklayer system. The experimental validation is performed on a full-scale prototype based on an industrial robotic arm and a custom-made overhead crane. The proposed control scheme will be implemented following a modular development approach, breaking down the entire architecture into several sub-modules, each of which implements a feature of the control system.

The remainder of this paper is organized as follows. In Section 2, an analysis of the process of laying sand lime blocks is proposed. In Section 3, the mechanical design of the proposed robotic solution is described. The current full-scale prototype is explained in detail in Section 4. In Section 5, the whole control architecture that allows us to properly control the system is explained and realistic experiments are shown in Section 6. Section 7 concludes the paper.

2 Bricklaying activity

To deeply understand the building process for the foreseen robotization, in this section we analyse the manual masonry process, focusing on the actions required to build walls and the type of blocks used in this operation. As a case study, we focus on sand-lime block masonry. As this type of block is widely used in the European context, and due to their size and weight, none of the robotic solutions proposed so far could handle them.

2.1 Manual masonry process

The manual masonry process to build walls with sand-lime blocks is characterized by repetitive action steps and the use of a lifting mechanism (i.e. a small crane). Typically, this activity is carried out by a team of two masons (see Fig. 1): one mason controls the crane to move the block from the pallet to its final position, and the second mason guides manually the block toward its final location. Therefore, the entire laying activity can be divided into two crucial phases. The first phase is the macro-movement performed by the crane. In this phase, the crane lifts the block and brings it close to its final position. The main difficulty of this first phase concerns the oscillations of the payload which must be counteracted with proper handling of the crane by the operator. As demonstrated by the authors in [12, 13, 14], these problems can be solved by properly controlling the crane by making use of nonlinear control in combination with constrained control techniques. The second phase is the precision placement, which is mainly performed by the second mason. When the block is close to the final position, the mason guides the block to the final position while the other mason only operates the winch of the crane to gradually lower the load. This second operation is the subject of this work.

Figure 1. Pose of Silka Sand-Lime blocks

3 Mechanical Design

Based on the analysis carried out in the previous sections, we propose a new robotic concept for the bricklaying with large and heavy blocks based on a 'non-rigid' robot, such as a crane, in charge of the macro-movement and of holding most of the weight of the block, and a small rigid robot mounted on an aerial work platform to achieve the desired precision during the fine placement of the block. A schematic of the envisioned solution is reported in Fig.2 where (1) is the crane, (2) the aerial work platform, (3) is the robotic arm, (4) is the block to be placed, and (5) is the existing wall.

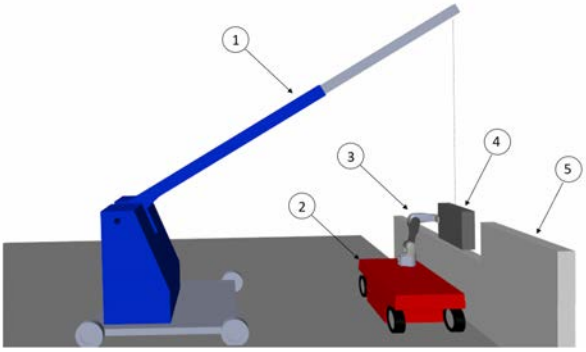

Figure 2. Layout of the robotic solution

There are a number of factors that advocate the effectiveness and feasibility of this solution:

- **R&D Cost Effectiveness:** singularly taken boom cranes, aerial platforms, and industrial manipulators are well-engineered and mature devices that do not require any major research or redesign. Accordingly, the R&D needed to bring to the real world the solution proposed in this paper concerns mainly the integration and the cooperative control of the components.
- **Flexibility:** the proposed solution makes use of standard units that can be possibly used independently also for other building operations.
- **Simplicity**: once connected to the block, the rigidity of the robotic manipulator allows to sense and control the exact position and orientation of the block w.r.t. the basis of the robot. This allows for avoiding an overly complex sensory suite for the fine placement of the block.
- **Extensibility** : the concept based on the cooperation between the crane and the robotic arm can be used not only for masonry work with heavy blocks but also to perform other construction tasks where heavy materials need to be positioned precisely (e.g. steel structures, prefabricated buildings/elements assembly).

4 Experimental Setup

Figure 3. Robotic Prototype.

To test the proposed multi-robotic architecture, we built a full-scale demonstrator (see Fig. 3). The prototype consists of a robotic arm and an overhead crane to mimic the behavior of the lifting mechanism used in manual operations. In this work, the robot is fixed in one position. The idea of using a mobile base (see the mechanical design explained in Section 3), will be the subject of future implementations. The robotic arm used is a KUKA LBR IIWA14 R820. The lifting mechanism is composed of two motors electric motors. The electric motor for the horizontal motion of the cart, subsequently called the “x-motor”, is an LK4ESZ by Holzmann Maschinen. The electric motor for the motion of the cable, from now on called the “z-motor” is an LHM1011 by FERM. The sensor for measuring the position of the x-motor is an ESA02 AH006820 (from now on referred to as the “x-sensor”), and the one keeping track of the position of the wire (from now on the “z-sensor”) is a Hengstler Incremental Encoder 1024 by RS. To be able to send commands to the motors and to read measures from the sensors, the communication is ensured by a NIDAQmx (NI USB-6008), one of National Instruments’ current-generation acquisition drivers. Moreover, a vision system based on an Intel Camera D455 is used to measure the oscillations of the suspended block thanks to the presence of markers on the surface of the block.

5 Control Architecture

This section describes the multi-robot control architecture and its sub-components (see Fig. 4). In this work, we focus on a specific task of the laying activity, namely the positioning of the block in the desired position once the robotic arm has grasped the block. From a control viewpoint, this phase is the most challenging and nonstandard one. In fact, to perform this operation an adequate control scheme should be designed so that the two units collaborate in the correct and safe way in order to guarantee that the block is placed in the final position and during the operation, the robot is never overloaded by the weight of the block since the block to be placed weighs more than the maximum admissible payload that the robot alone can handle. The other steps of the construction activity under investigation will be the subject of future exploitation of the proposed control architecture.

The control architecture in this study was developed on two devices: a Windows-based machine and an Ubuntu-based machine. A Controller Area Network (CAN bus) standard protocol is used to implement the communication between the two devices to allow the exchange of information. Moreover, in the Ubuntu-based machine, we used the Robot Operating System (ROS) framework since it provides an environment where a developer can combine numerous sub-processes called ROS nodes into an application package. In this study, five ROS nodes were developed and linked to each other in the system architecture (see Fig. 4). These nodes and their interactions are described below.

The goal of the proposed control architecture is to send commands to the crane and to the robotic arm in order to move the common payload to a desired final position. As mentioned in Section 4, the communication between the computer and the crane is ensured by a NI USB-6008.

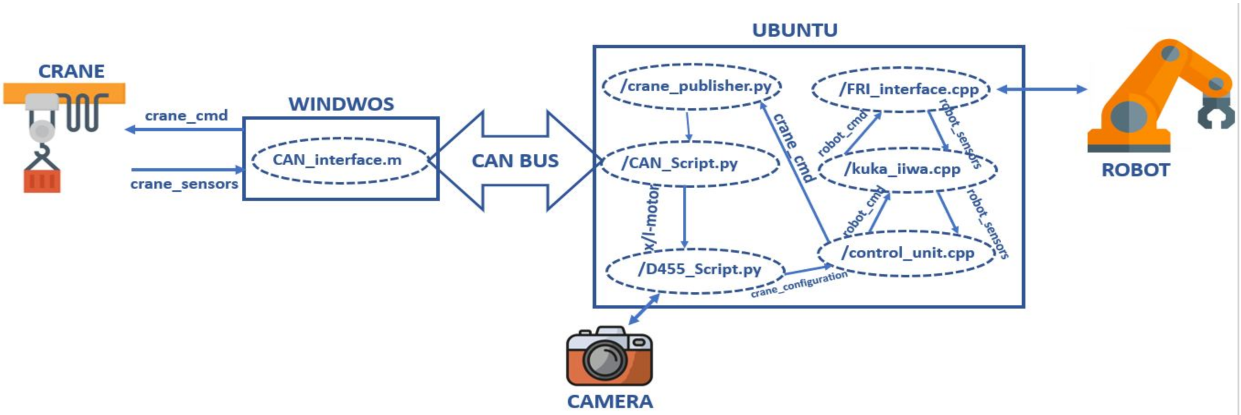

Figure 4. Robot Control Architecture

This device is only compatible with Windows and the Data Acquisition Toolbox of Matlab® allows a quick interface with its driver. However, the robotic arm used in this paper (i.e. a KUKA LBR IIWA14 R820), can be controlled only via an Ubuntu machine. Therefore, CAN-based communication was established between the two computers. The CAN-communication used in this work is a PCAN-USB adapter that enables simple connection to CAN networks. The **CAN_interface.m** is a Matlab® script that sends commands to the x-motor and the l-motor of the crane in order to move the block to the desired position. The script is also used to read measures from the x-sensor and the l-sensor. The desired commands are sent via CAN from the **/control_unit.cpp**, a ROS node. The measures from the sensors are sent instead to the Ubuntu computer. In this communication, each message is characterized by a custom and unique ID: **#ID200** x-motor command, **#ID201** l-motor command, **#ID100** x-sensor measure, and **#ID101** l-sensor measure. These messages are exchanged between **CAN_interface.m** and **/CAN_Script.py**. This ROS node runs on the Ubuntu computer and makes the interface with the CAN Bus. It receives the desired motor commands from the node **/crane_publisher.py** and it sends the measures from the sensor to the node **/D455_Script.py**. The latter node uses camera data and crane sensor data to compute all degrees of freedom of the suspended brick that cannot be measured directly (i.e. the oscillations of the cable and the three angles of the orientation of the block, see Fig. 10). The entire configuration of the crane (i.e. the position of the cart, the length of the cable, and the un-actuated degree of freedom of the block) are then sent to the node **/control_unit.cpp**. This script computes the desired commands (based on a pre-planned trajectory) to be sent to the robot and crane in order to move the common load to its final desired position. The current configuration of the crane is sent by the node **/D455_Script.py**. The current configuration of the robot is sent by the node **/kuka_iiwa.cpp**. Instead, the desired commands are sent from the control node to the node **/crane_publisher.py** to actuate the crane and to the node **/kuka_iiwa.cpp** to move the robot. The last node in the proposed control architecture is the **/FRI_interface.cpp**. This node sends the desired commands to the robotic arm to make it move and receives the measurements from the sensors of the robot that are then sent to the node **/kuka_iiwa.cpp**. The pipeline between the computer and the robotic arm is illustrated in detail in Fig. 5.

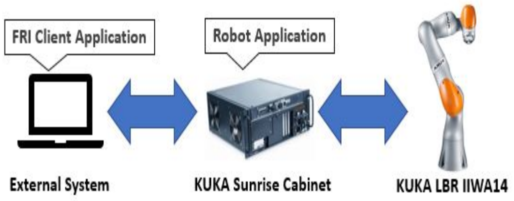

Figure 5. Robotic arm control scheme pipeline

The robot used in the proposed prototype (i.e. a KUKA LBR IIWA14), is especially suited for research in robotics, as it is accessible through a real-time interface named *Fast Robot Interface* (FRI) [15]. FRI is an interface via which data can be exchanged continuously and in real time between a robot application running on the robot controller and an FRI client application running on an external system. The FRI can be switched between position or torque control modes, accepting commands for motor position or joint torque respectively. As shown in Fig. 5, this architecture is comprised of several elements:

- **Robot application.** The Robot application is programmed in Java and executed on the robot controller. In this paper, we used KUKA Sunrise.OS 1.17, KUKA Sunrise.Workbench 1.17 and KUKA Sunrise.FRI 1.17 to program the robot.

- **FRI client application.** FRI client application can be created in C++ and is executed on an external system, in our case, on a Laptop with Intel(R) Core(TM) i7-6500U CPU 2.50GHz 2.60 GHz. In the proposed control architecture, the client application is implemented in the node **/FRI_interface.cpp**.

The code explained in this section is made available by the authors. Please send an e-mail to one of the authors to receive all information on how to download it.

6 Experimental Results

In this section, we will demonstrate via experiments the feasibility of the proposed approach. In particular, we will show that the full-scale demonstrator presented in Section 3 and the control scheme in Section 5 allows a lightweight robotic arm to manipulate a large and heavy block. The dimensions of the block used in the experiments are reported in Tab.1.

Table 1. Block Dimension

Length [m]	Height [m]	Thickness [m]
0.8	0.6	0.12

The experiment we performed is the following: the robotic arm has to move the suspended object with a weight of **30kg**, along the x-axis of $18cm$ and along the z-axis of $13cm$, in order to place the object in its final position, Fig. 6. The robotic arm used in the simulations, a KUKA IIWA14 R820, can handle a payload of up to **14kg**, [16]. However, due to the fact that the mass of the block is sustained almost entirely by the cable, we can still use a lightweight robotic arm to perform the foreseen operations. A video of the experiments can be found at `https://youtu.be/XTEwk9j-W7A`.

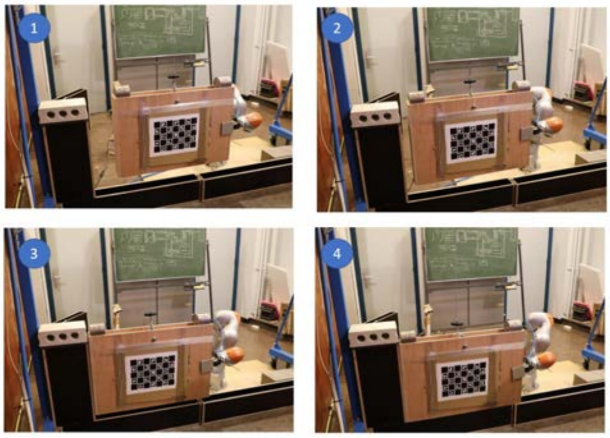

Figure 6. Experiments, pose of block

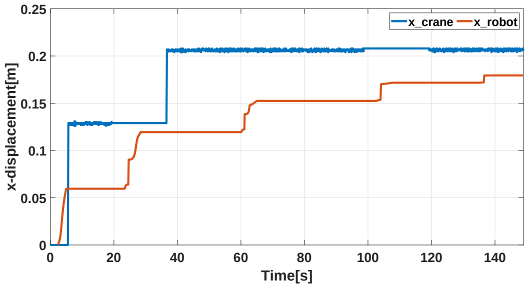

Figure 7. Time evolution of along the x direction. Red Line. Robot end-effector. Blue Line. Crane x-motor.

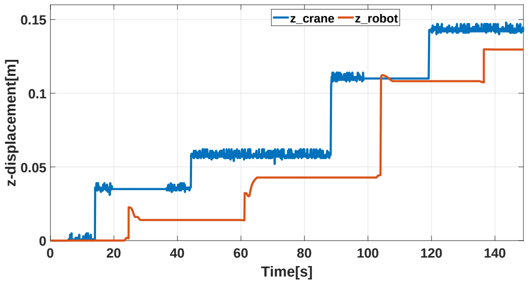

Figure 8. Time evolution of along the z direction. Red Line. Robot end-effector. Blue Line. Crane z-motor.

The desired operation is performed by combining two movements. The first movement is along the horizontal-axis. This part of the trajectory is planned so that: *i*) the robot end-effector reaches the final desired pose in the horizontal plane (see Fig. 7); *ii*) the crane follows the movement of the payload in the horizontal plane (i.e. it keeps as low as possible the angle of the cable w.r.t. to gravity) while keeping the altitude of the block constant. From Fig. 7, one can clearly see that while the cart reaches its limit after only two steps, the robot continues to move (and push) the load for the last few centimeters in order to reach the desired position and alignment with the pre-existing wall. During this operation, the parameters of the internal controller of the robotic arm are tuned so as to ensure sufficient compliance when the block enters in contact with the existing wall.

The second part of the trajectory consists of movements along the vertical-axis. Once the block is sufficiently well aligned above its horizontal desired position, the block is lowered toward the vertical destination as shown in Fig. 8. During this operation, the robot is controlled aiming to keep the horizontal position and the alignment of the block, while along the vertical axis, it follows the block in a compliant way. Moreover, as one can see in Fig. 9, during

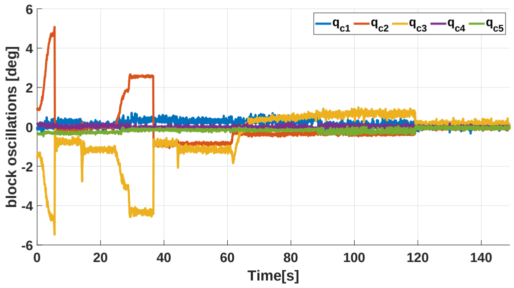

Figure 9. Time evolution of block oscillations.

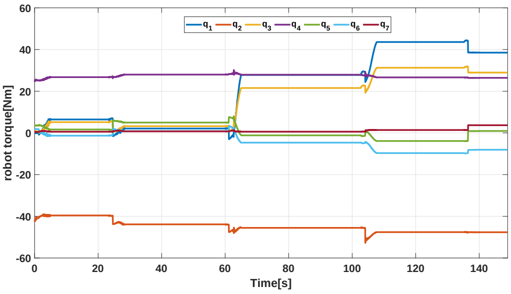

Figure 11. Time evolution of robot torques.

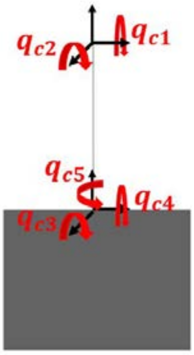

Figure 10. Block configuration

the operations, the robotic arm is able to damp almost all the oscillations of the block in order to place it in the correct position with a high level of precision. In fact, there are no residual oscillations once the block has been positioned, and all the oscillation angles of the block are almost zero degrees. This allows us to conclude that the block has been positioned correctly. The two first oscillations that can be seen in the behavior of q_{c2} and q_{c3} in Fig. 9 are due to the first two movements of the block with respect to the horizontal axis. Please, refer to Fig. 10 to understand the meaning of each oscillation of the block.

It is important to notice that, despite the limitations of the payload that can be managed by the robot, the cooperation between the two robotic units ensures that the robot is never overloaded. In fact, as one can see in Fig.11, the torques required to the robotic actuators are well within the joint torques limits.

7 Conclusion

This paper proposes a novel concept for the bricklaying of large sand-lime blocks and provides a possible embodiment to perform the construction activity. The main idea is to use the crane currently used in manual bricklaying operations in conjunction with a lightweight robotic arm. The role of the crane is to move the payload and sustain most of the weight of the block. The role of the robot is to place the block within the expected precision levels. The control architecture that allows the multi-robot system to perform brick-laying operations is explained in detail, highlighting its modularity that can be easily modified and extended for other technologies. We demonstrated via experimental analysis performed with a KUKA LBR IIWA14 R820 and with two industrial electric motors the feasibility of this novel approach. The preliminary results shown in this work will be exploited in the future for two activities. Firstly, the control law will be modified to be able to perform the block-grabbing activity by the robot. Secondly, the mobile platform will be implemented and tested, so that the robot can be moved around and increase its working space.

8 Acknowledgments

This research has been funded by The Brussels Institute for Research and Innovation (INNOVIRIS) of the Brussels Region through the Applied PHD grant: Brickiebots - Robotic Bricklayer: a multi-robot system for sand-lime blocks masonry (réf : 19-PHD-12).

References

[1] C. Franke. Improvement in brick-laying machines. 1875.

[2] J. Thomson. Brick-laying machine. 1904.

[3] Sam100. https://www.construction-robotics.com/sam100/.

[4] Hadrian x. https://www.fbr.com.au/view/hadrian-x.

[5] V. Helm, S. Ercan, F. Gramazio, and M. Kohler. Mobile robotic fabrication on construction sites: Dimrob. pages 4335–4341, 2012. doi:10.1109/IROS.2012.6385617.

[6] Markus Giftthaler, Timothy Sandy, Kathrin Dörfler, Ian Brooks, Mark Buckingham, Gonzalo Rey, Matthias Kohler, Fabio Gramazio, and Jonas Buchli. Mobile robotic fabrication at 1: 1 scale: the in situ fabricator. *Construction Robotics*, 1(1-4):3–14, 2017.

[7] Ablr. `https://constructionautomation.co.uk/ablr/`.

[8] Maike Klöckner, Mathias Haage, Helena Eriksson, Henrik Malm, Klas Nilsson, Anders Robertsson, and Ronny Andersson. Insights into automation of construction process using parallel-kinematic manipulators. In *Proceedings of the 39th International Symposium on Automation and Robotics in Construction*, July 2022. doi:10.22260/ISARC2022/0006.

[9] Balaguer Carlos, Abderrahim Mohamed, Balaguer Carlos, and Abderrahim Mohamed. Trends in robotics and automation in construction. 2008.

[10] Michele Ambrosino, Philippe Delens, and Emanuele Garone. Control of a multirobot bricklaying system. *Advanced Control for Applications*, 3(4):e90, 2021. doi:https://doi.org/10.1002/adc2.90.

[11] Michele Ambrosino, Fabian Boucher, Pierre Mengeot, and Emanuele Garone. Constrained control scheme for the manipulation of heavy pre-fabricated elements with lightweight robotic arm. In *Proceedings of the 39th International Symposium on Automation and Robotics in Construction*, July 2022. doi:10.22260/ISARC2022/0053.

[12] Michele Ambrosino, Arnaud Dawans, and Emanuele Garone. Constraint control of a boom crane system. In *Proceedings of the 37th International Symposium on Automation and Robotics in Construction (ISARC)*, October 2020. doi:10.22260/ISARC2020/0069.

[13] Michele Ambrosino, Marc Berneman, Gianluca Carbone, Rémi Crépin, Arnaud Dawans, and Emanuele Garone. Modeling and control of 5-dof boom crane. In *Proceedings of the 37th International Symposium on Automation and Robotics in Construction (ISARC)*, October 2020. doi:10.22260/ISARC2020/0071.

[14] Michele Ambrosino, Arnaud Dawans, Brent Thierens, and Emanuele Garone. Oscillation reduction for knuckle cranes. In *Proceedings of the 37th International Symposium on Automation and Robotics in Construction (ISARC)*, October 2020. doi:10.22260/ISARC2020/0221.

[15] Günter Schreiber, Andreas Stemmer, and Rainer Bischoff. The fast research interface for the kuka lightweight robot. In *IEEE workshop on innovative robot control architectures for demanding (Research) applications how to modify and enhance commercial controllers (ICRA 2010)*, pages 15–21. Citeseer, 2010.

[16] Kuka. `https://www.kuka.com/`.