



Cartesian coordinate control for teleoperated construction machines

Hyung Joo Lee¹ · Sigrid Brell-Cokcan¹

Received: 15 July 2020 / Accepted: 25 January 2021
© The Author(s) 2021

Abstract

Despite the continuous development of the hardware, most construction machines are exclusively teleoperated limiting the control to a single paradigm. The operators usually have to move different joints of the machine in a coordinated way solely relied on their experiences leading to reduced local accuracy and work efficiency. Automation of construction machinery can open up new possibilities to improve efficiency and safety during the construction process. This work introduces a generic method that can adapt construction machines that have been already used in the field for decades, so that a more intuitive and versatile control paradigm can be allowed. We introduce the system architecture with the necessary hardware extension and the closed-loop inverse kinematic based motion controller implemented in a visual programming environment. In contrast to existing works, which are mostly based on developing entirely new systems, an autonomous machine suited for construction sites and other hazardous environments can be obtained at a reduced effort. Because of its low cost and generality, this approach can be widely utilized in construction industries opening possibilities for a combination of the advanced robotics technology with proven machines from construction sites. We present our first prototype system based on a BROKK 170 demolition machine and highlight its capabilities but also the inherent limitations of the proposed method.

Keywords Automation in construction · Construction robotics · Motion control

1 Introduction

During the course of past decades, the hardware of construction machinery has been continuously developed. For instance, industry companies like Husqvarna and Brokk have successfully introduced teleoperated demolition machines for unstructured and hazardous conditions even such as nuclear power plants (Osumi 2014).

However, the human-machine interface (HMI) is still primitive. Operators use levers to directly control each joint. Since these machines have multiple degree of freedoms (DoF) to extend the reachability, operators need months of training to get familiar with the control system. Even after months of training and experience, the work performance cannot be always guaranteed, since operators on-site are usually not able to preview the results of their joystick maneuvers. The appropriateness of the operation can only be verified after the machine is actually moved. Moreover, the

teleoperated systems can quickly reach their limits, because the work performance heavily depend on the operator's situational awareness which is difficult to obtain with 2D camera images. As a result, teleoperated machines are often lacking local accuracy. Thus, manual work is still mostly preferred on construction sites (Motzko et al. 2016), and laborious, repetitive, and tedious tasks are typically carried out by human workers.

Despite these limitations, as teleoperation can, with relatively little effort, prevent operators from working in dangerous conditions by keeping them in the control loop, teleoperation will remain as an essential element of construction machinery (Chen et al. 2007). Nevertheless, the primitive HMI must be further developed and adaptive to diverse environments, situations, and operator requirements, so that the system can, for instance, accept high-level-commands such as a trajectory tracking in the Cartesian space to reduce the burden of a coordinated joints control.

For addressing these aforementioned issues of teleoperation, different algorithmic control setups are presented in prior works to simplify tasks for operators. Jud et al. (2019) presented an autonomous walking excavator, which has been highly customized from an commercially available

✉ Hyung Joo Lee
lee@ip.rwth-aachen.de

¹ Individualized Production in Architecture, RWTH Aachen University, Schinkelstr. 1, 52062 Aachen, Germany

excavator. Numerous hardware adaptations and additions such as exchanging the hydraulic cylinders and integrating new valve systems were introduced in the work. In Park et al. (2017), an online learning control framework was designed based on echo-state networks to track the excavator motion. The authors used an embedded computer to communicate with the machine control unit. Similarly, a discrete time delay controller was proposed for an excavator in Kim et al. (2019). A personal computer with Controller Area Network (CAN) modules was used as a controller. In Tomatsu et al. (2015), a model predictive controller with retrofitted servo valves was introduced. However, despite the reported high-performance result, the possibility of retrofitting the existing hydraulic components as conducted in the prior works is not always guaranteed due to its complexity and cost.

To increase the work performance that is often limited in teleoperated machines, the idea of customizing classical industrial robots for the unstructured nature of the construction sites has drawn the attention of many researchers. In Giftthaler et al. (2017), a robot system consisted of a customized platform with ABB IRB 4600 robot arm is introduced. The proposed system is capable of automating multiple tasks while driving on non-flat or soft ground. Keating et al. (2017) presented a large mobile 3D printer using KUKA KR 10 built on an aerial lift system Altec AT40GW. Feng et al. (2015) introduced a robotic system that can support an autonomous assembly process using bricks. KUKA KR 100 is used with fiducial marker-based pose estimation algorithm. However, since the construction tasks often require high payloads, the classical electrically actuated robot arms without compliant elements will always suffer from a low payload/weight ratio and their susceptibility to vibration and damage. Consequently, the usage is limited to certain types of tasks and a laboratory indoor environment.

All the aforementioned works aim to work performance by reducing and/or simplifying the task for the operator. However, they are limited to the specific solution for their considered hardware without having a more general perspective. Since these solutions rely on highly customized hardwares, they are not easily replicable. Hence, the work described in this paper aims to introduce a more generic way to enable more advanced control such as the Cartesian space control of the tool by maximizing the usage of standard components. The Cartesian space control of the tool is considered a valuable feature, since construction machines usually consist of multiple DoF. By allowing the human operator to focus on the tool movement, the task complexity is heavily reduced. We present embedded hardware and software architecture that allow the Cartesian position control for a teleoperated construction machine. The proposed approach is implemented on an off-the-shelf demolition machine BROKK 170. This work builds upon our previous work showing the first concept of adapting existing teleoperated

construction machine (Lee et al. 2019, 2020) and introduces further studies to investigate the adapted emolition machines ability through preliminary experiments and a real-world task, i.e., insertion of the chisel into a borehole (see Fig. 1). We also highlight the challenges due to non-linearity in the actuator system.

The paper is structured as follows. The next section describes the system architecture, where hardware extensions to implement algorithms for the Cartesian position control are introduced. To enable the end-effector in the Cartesian space, the mapping between the given goal in the Cartesian, i.e., operational space and the joint space has been established. Section 3 presents the implementation of the well-known Closed-loop inverse kinematic (CLIK) approach to solve this mapping. In Sect. 4, we highlight the visual programming interface. Section 5 details the experiments to validate the presented approach. The existing issues and conclusion are described in Sects. 5 and 6, respectively.

2 System architecture

To enable coordinated joints control by means of the Cartesian position control, computational capabilities have to be integrated into the machine control unit (MCU). In modern construction machinery including BROKK 170, actuators, sensors, and other electronics are connected to the MCU using a CAN bus. More precisely, if the operator sends commands with the original control device, bus messages are



Fig. 1 The teleoperated demolition machine BROKK 170 and the experimental setup used in this work

generated according to these commands. Since every unit in the machine has its unique identifier (ID), the generated messages are delivered to specific power electronics to drive the mechanical components of the machine.

2.1 Communication link

Thus, an embedded controller is connected to the machine’s bus controller to enable the communication between the existing bus network and the host computer. The communication via the bus system requires the bus protocol containing policies, i.e., message bits, ID, cycle time, and length to generate proper bus messages. This information can be obtained from the manufacturer of the machine. Otherwise, one can empirically estimate it by executing a particular operation while monitoring the incoming bus messages. Since the scope of this work is to implement a Cartesian position controller, we have only identified bus messages required to move each joint.

Figure 2 provides an overview of hardware architecture used in this work. The original control device of BROKK 170 works with two levers (see Fig. 3) which generates PWM signals according to the lever manipulation. This PWM signal with a voltage level is forwarded to the valve system by the MCU and generates joint motions with different speeds according to the PWM signal.

Therefore, in our work, we directly emulate these PWM signals to control the machine from a programming interface. However, since the machine does not provide a general interface to interact with external signals, a communication link is developed based on an embedded control system

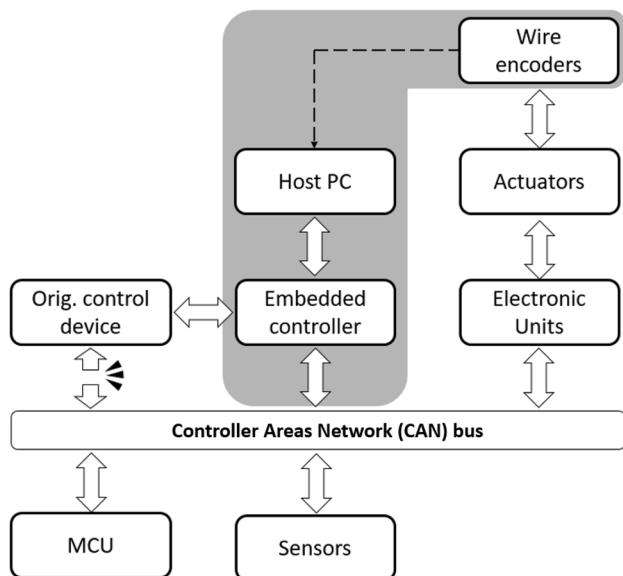


Fig. 2 Overall hardware architecture. The gray box represents the components added to the original machine design

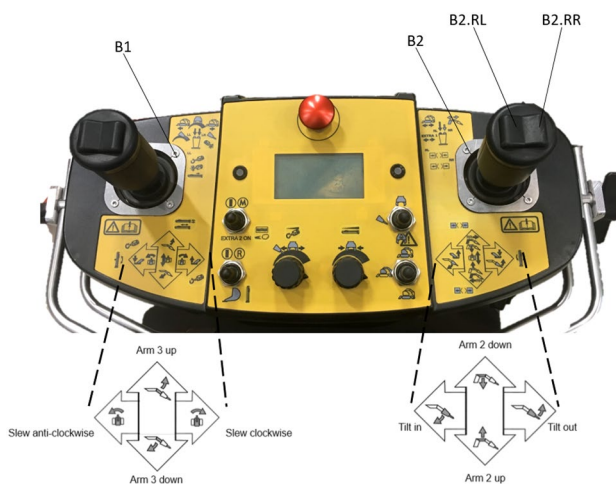


Fig. 3 Control device of Brokk 170 emulated in this work. The right lever B2 has two buttons, B2.RL and B2.RR. Motion of the second joint can be controlled using the right lever B2 and the B2.RR, whereas to control other joints, the activation button B2.RL has to be used (BROKK 2018)

relying on the CAN bus of the machine. To deploy a control algorithm to the MCU, the machine must be able to communicate with the host PC. While one can also completely replace the MCU to maximize the flexibility in programming the machine, as shown in Hutter et al. (2015), we want to maximize the utilization of existing components and communication network. We believe that the utilization of standard components improves replicability of this work, since the readers can more easily apply the presented approach to their own machine. For this purpose, a communication gateway between the host PC and the existing CAN bus of the machine is first built. Within this gateway, the signals from the host PC, i.e., the output of our algorithm are combined with the signals from the original control device by means of replacing CAN bus messages. This way, it is not necessary to emulate the whole control device, but only some parts in the bus messages such as relevant for joints motion can be emulated by preserving other functionalities of the original control device.

2.2 Sensing

In this work, the kinematics of the machine and the shape of each axes are assumed to be known, so that the joint angles are enough to describe the full configuration of the machine, in particular the end-effector pose. Yet, like most other construction machines, BROKK 170 used in this work is not equipped with any sensors to measure the actual joint motion and to estimate the discrepancy between the actual and desired motion. As depicted in Fig. 4, five joint angles are required to determine the configuration of the machine’s end-effector at any instant. We use wire-type encoders

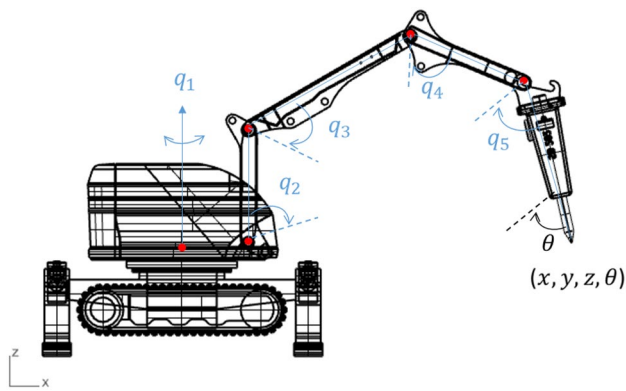


Fig. 4 Joint geometric representation of the 5 degree of freedom BROKK 170. The joint space coordinates are noted in blue, whereas the operational space coordinates are in black. The joint positions are noted in red

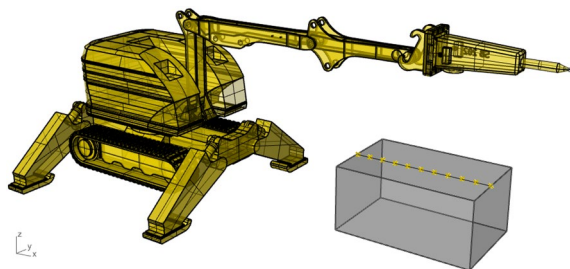


Fig. 5 View panel in Rhino

(BCG05, SICK) with analog interfaces to measure the joint angle values. The encoder outputs a voltage signal in the range of 0–10 V that corresponds to a displacement range of 0–1250 mm. Resolution is 0.1 cm and the calibration process is performed with the integrated functionality of the sensor to program the encoder to its starting value at the home position visualized in Fig. 5.

One advantage of using the proposed communication link from Sect. 2.1 is that it can read any sensor values that are integrated in the machine and communicating with the existing CAN bus network. Although construction machinery is often not equipped with any axis sensors, since they are directly controlled by operators, these hydraulically driven machines often have a pressure sensor in the main pump to monitor the pressure in the system. For our work, we use this pressure sensor to interact with the environment. The sensor is executed at 2 Hz.

2.3 Brokk 170

BROKK 170 is a type of demolition machine commonly used in deconstruction and tunneling for drilling, demolition, cutting, trenching, and soil mixing dependant on the

attached end-effector tool. BROKK 170 is essentially a serial link manipulator with 5 degrees-of-freedom (DOFs). The linkage and actuator systems are attached to a mobile base, but throughout this work, the base is assumed locally fixed. Figure 4 shows the major components of a Brokk 170 with a breaker.

The linkage system is actuated by a hydraulic manipulator system comprised of one main pump supplying volumetric flow for the whole system. This flow is used in each control valves to move actuators and returned into the tank. Originally, the BROKK 170 demolition machine is designed to be controlled with a remote controller shown in Fig. 3. According to the combination of lever displacement and activation button, the command is directly mapped to the corresponding joints. Since this focuses on the Cartesian position control, the following levers commands are analyzed by means of corresponding CAN bus messages and emulated with the host PC:

- q_1 : B1 to left or right + B2.RL
- q_2 : B2 to fore or aft + B2.RR
- q_3 : B2 to fore or aft + B2.RL
- q_4 : B1 to fore or aft + B2.RL
- q_5 : B2 to left or right + B2.RL.

3 Control methods

The aim of this work is to enable high-level control of a teleoperated machine. The hardware extension from Sect. 2 allows the teleoperated machine' MCU to communicate with a host PC. Accordingly, a mapping algorithm must be first developed and deployed to the host PC, which identifies the physical relationship between the operational and the joint space. In other words, for the defined end-effector trajectory in the Cartesian, i.e., operational space, the corresponding joint configuration must be estimated.

3.1 Closed-loop inverse kinematics

The chosen mapping algorithm is based on a control method known as closed-loop inverse kinematics (CLIK). The manipulator control can be formulated as finding proper joint space configurations $q \in \mathcal{R}^N$ given an operational space description $x \in \mathcal{R}^M$, where N is the number of degree-of-freedom (DoF) of the machine and M is the operation dimension. As depicted in Fig. 4, the operational space \mathbf{x} and joint configuration \mathbf{q} of the demolition machine can be described as:

$$\mathbf{x} = [x \quad y \quad z \quad \Theta]^T \in \mathcal{R}^4 \quad (1)$$

$$\mathbf{q} = [q_1 \quad q_2 \quad q_3 \quad q_4 \quad q_5]^T \in \mathcal{R}^5. \tag{2}$$

The mapping between the joint and task space velocities can be formulated as:

$$\Delta \mathbf{x} = \mathbf{J}(q)\Delta \mathbf{q}, \tag{3}$$

where $\mathbf{J} \in \mathcal{R}^{M \times N}$ denotes the Jacobian matrix. $\Delta \mathbf{x}$ and $\Delta \mathbf{q}$ are the changes in the task and joint space, respectively. To obtain the $\Delta \mathbf{q}$ out of the relation in (4), we use the closed loop inverse kinematic (CLIK) approach introduced in Dariusz et al. (2008). For manipulators with $N > M$, the inverted Jacobian matrix can be obtained as follows:

$$\mathbf{J}^+ = \mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}. \tag{4}$$

The pseudo-inverse has the property to provide the best possible solution according to equation $\mathbf{J}\Delta \mathbf{q} = \Delta \mathbf{x}$. If $\Delta \mathbf{x}$ is not in the range of \mathbf{J} , an exact value of $\Delta \mathbf{x}$ is not available. However, the provided $\Delta \mathbf{q}$ still minimizes the magnitude difference of $\mathbf{J}\Delta \mathbf{q} = \Delta \mathbf{x}$ (Buss 2004). Using this property and closed-loop behavior in CLIK, the convergence to the desired $\Delta \mathbf{x}$ can be ensured. The problem from (4) can be formulated as follows:

$$\Delta \mathbf{q} = \mathbf{J}^+(\Delta \mathbf{x} + \mathbf{K}\mathbf{e}), \tag{5}$$

where $\mathbf{K} \in \mathcal{R}^{M \times M}$ is a positive definite gain matrix and $\mathbf{e} \in \mathcal{R}^{M \times 1}$ is the remained difference between the desired and actual motion. The CLIK controller is further extended with a weighted least squares method to introduced in Chan and Dubey (1995) to handle the joint limit problem. For achieving robustness in the vicinity of singularities, a damped least square method introduced in Nakamura and Hansafusa (1986) is used. Accordingly, the Jacobian matrix from (5) can be reformulated as:

$$\mathbf{J}^+ = \mathbf{W}^{-1}\mathbf{J}^T(\mathbf{J}\mathbf{W}\mathbf{J}^T + \lambda^2\mathbf{I})^{-1}, \tag{6}$$

where \mathbf{W} denotes the $n \times n$ diagonal matrix penalizing the motion of joints, whenever a joint is near its hardware limit and λ refers to the damping factor restricting the robot's motion near the singularity.

3.2 Joint limit problem

The Weighted Least-Norm (WLN) method proposed in Chan and Dubey (1995) is used to prevent the machine moving towards its hardware limit. The principle of WLN follows the idea to penalize the motion of a joint, whenever the joint is near its limit. To penalize the motion of a joint over the others, the gradient of the performance criterion $H(q)$ is defined as follows:

$$\frac{\partial H(q)}{\partial q_i} = \frac{(q_{i,\max} - q_{i,\min})^2(2q_i - q_{i,\max} - q_{i,\min})}{4(q_{i,\max} - q_i)^2(q_i - q_{i,\min})^2}, \tag{7}$$

where q_i is the i th joint angle, and $q_{i,\max}$ and $q_{i,\min}$ are the upper and lower limits, respectively. Correspondingly, the elements of the $n \times n$ diagonal matrix \mathbf{W} are defined by:

$$w_i = \begin{cases} 1 + \left| \frac{\partial H(q)}{\partial q_i} \right|, & \text{if } \Delta \left| \frac{\partial H(q)}{\partial q_i} \right| \geq 0 \\ 1, & \text{elsewhere.} \end{cases} \tag{8}$$

3.3 Kinematic singularity problem

The damped least-squares (DLS) approach is used for the robust usage of a pseudo-inverse matrix, which is introduced in Nakamura and Hansafusa (1986). The basic idea of DLS is similar to the above WLN method. Robot's movement is damped using a damping factor λ , whenever the robot is near the singularity. The factor λ can be dynamically estimated using the manipulability measure m which depends on the current configuration :

$$\lambda = \begin{cases} k(1 - \frac{m}{m_t}), & \text{if } m < m_t \\ 0, & \text{elsewhere,} \end{cases} \tag{9}$$

where k is a constant, $m = \sqrt{\det(\mathbf{J}\mathbf{J}^T)}$, and m_t is a threshold value.

4 Visual programming interface

The essential idea of visual programming is that it allows users to create programs by manipulating program components such as blocks and wires graphically rather than using textual programming languages, as visualized in Fig. 6. Essentially, the user can create a software program for applications, electronics, and even robots just by logically connecting the blocks and wires. Due to its intuitive interface, it lowers barriers in programming and is broadly used in the architecture (Stumm and Brell-Çokcan 2019) and construction industry (Kerber et al. 2020).

Motivated by this, we use Grasshopper which is a plug-in for a CAD software Rhino to deploy the algorithm introduced in Sect. 3 and also to interact with the user. As a plugin for the CAD software Rhino, Grasshopper is widely used for 3D modeling tool in the industrial and design professions. Thus, a major advantage of using Rhino/Grasshopper is its ability to simplify the creation of complex models by combining the design process, simulation capabilities, and robot control components within one type of software based on the visual programming environment (Brell-Cokcan et al. 2012; Steinhagen et al. 2016). Moreover, as Rhino and Grasshopper are

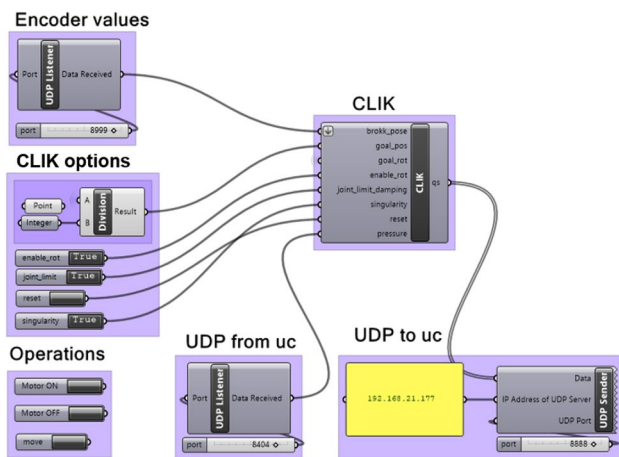


Fig. 6 The main user interface in Grasshopper. The visual programming environment in Grasshopper allows the users to create a software program for different applications by connecting the blocks and wires

open source, there is a strong community actively developing third-party plugins and add-ons for instance for structural analysis (Jamison Gerbo and Saliklisa 2014; Naboni 2014) or environment analysis (Roudsari and Pak 2013).

In Fig. 5, we see a snapshot of the view panel in Rhino, where the home configuration of BROKK 170 is visualized. The operator can give another goal pose simply by interacting with a widget. Since Rhino and Grasshopper are originally developed as a design tool for architects, the operator can also create various geometries for instance to visualize the working environment of the machine. According to the parameters that the user specified, a tool path with waypoints can be created along the given geometry, as visualized in Fig. 5. The user can apply the generated motion to the real BROKK machine or adjust the motion by changing parameter values.

The Grasshopper script is shown in Fig. 6 where the control algorithm CLIK is deployed. In this work, the algorithm is implemented with the built-in python script editor called Ghython which can be interacted with simple buttons or number sliders. Moreover, the operator can build UDP communication to interact with other systems or applications. In the shown example, three different ports are used to acquire the encoder values and pressure value, and to send the computed joint motion to BROKK 170 via the embedded controller.

5 Experimental evaluation

In this work, the CAD-tool Rhino/Grasshopper is exclusively used both for the user interface and the implementation of the control algorithm. For the communication link

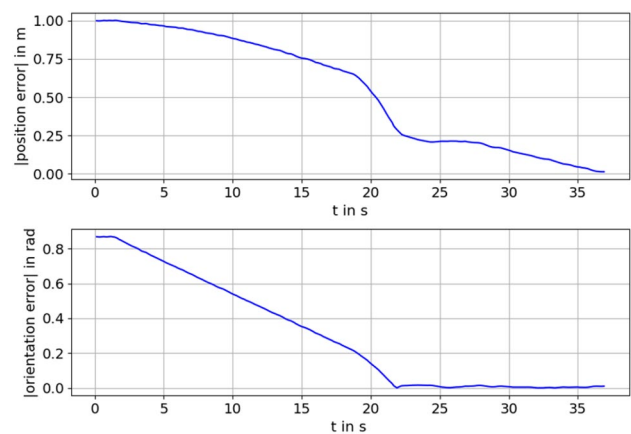


Fig. 7 Illustration of TCP deviation from the goal $\Delta \mathbf{x} = [-0.7, 0, -0.7, 0.87]$

to the machine, Arduino Mega2560 and MCP2515 are used as the embedded and CAN bus controller, respectively. The script is developed in C++ on the host PC, and then compiled and deployed to the embedded controller via Arduino IDE. The analog output of joint angle sensors are read in with the ADC pins of the embedded controller at a rate of 30 Hz. UDP packets are used for data exchange between Rhino/Grasshopper and Arduino script. The experiments are conducted with BROKK 170, which is adapted with the presented method. The realisation of the experiments, where the controllability of a teleoperated machine via the visual programming interface is investigated, is significant to enable automated deconstruction processes, since they involve highly repetitive processes, and, in some cases, are dangerous for human operators.

5.1 Experiment 1: Programming movements

The illustration in Fig. 7 shows the progression of a point-to-point movement (PTP movement) to reach the given goal $\Delta \mathbf{x} = [-0.7, 0, -0.7, 0.87]$. For the PTP movement, a step sizes of 2 cm for the position and 0.015 rad for the orientation are selected, respectively. The same values are also used for as threshold values for the movement. The illustrated TCP motion is obtained by computing the forward kinematics relationship with the joint angle values. As Fig. 7 shows, the TCP achieves the desired position and orientation after around 37 s.

Next, we define a path for the TCP in the CAD-tool Rhino and this Cartesian path is interpolated at a resolution of 4 cm using the component *Divide Distance* in Grasshopper. Throughout the tracking task, the orientation is neglected. The pre-defined path and the corresponding joint motions in the Grasshopper are used as the reference values. Figures 8 and 9 show the tracking results in the Joint and Cartesian Space, respectively. Although, we can achieve reasonable

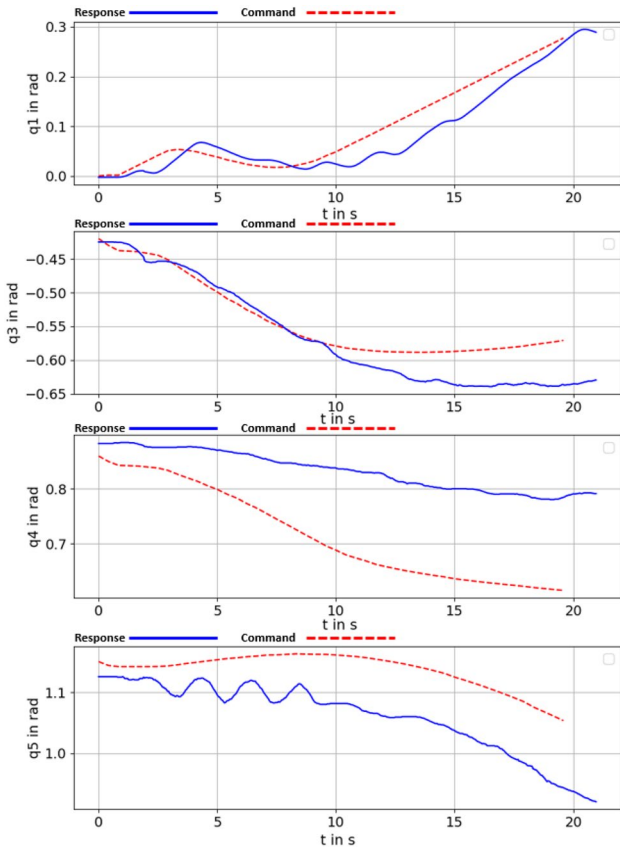


Fig. 8 Tracking results using the CLIK based control framework for arm motion in joint space. Dashed line: reference trajectory. Solid line: response

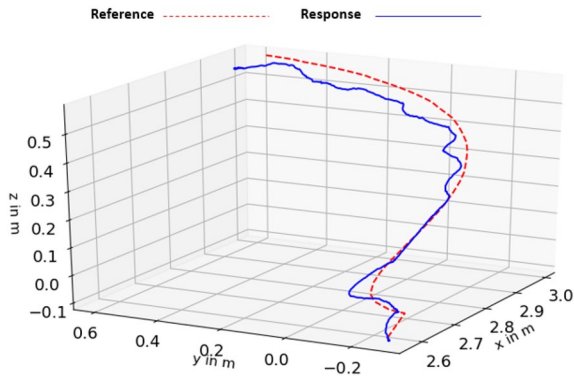


Fig. 9 Tracking results of TCP motion in the Cartesian space. Dashed line: reference trajectory. Solid line: response

tracking results, as depicted in Table 1, severe error sources can be observed.

A hydraulic machine such as BROKK 170 is a typical nonlinear system. This is due to the dynamic of the hydraulic actuation that is affected by flow losses, various frictions, and deadzones. For instance, the hydraulic machine BROKK

Table 1 Performance result (RMSE) in tracking TCP path task

Dimension	RMSE
q_1	0.02 rad
q_3	0.03 rad
q_4	0.12 rad
q_5	0.08 rad
TCP motion	0.067 m

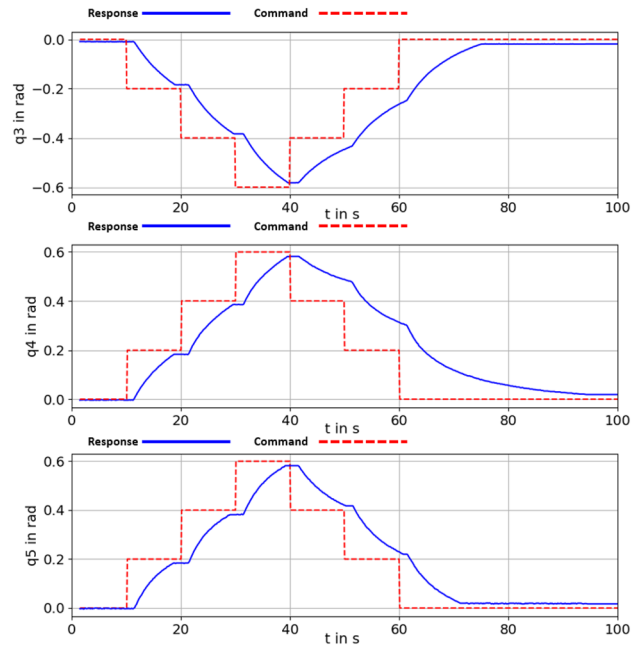


Fig. 10 Step responses of the last three joints across different actuation range

170 used in our work consists of one main pump supplying volumetric flow for the whole system. This flow is used in each control valves to move actuators and returned into the tank. During this circulation, flow losses are raised by partially closed or opened valves, bent pipes, expansions, or contractions. Therefore, the theory of linear controller (i.e., generic pseudo-inverse Jacobian controller) is limited to keep up the performance, since it typically takes the desired (x, y, z) values as input and outputs the corresponding joint configuration without considering the nonlinearities in a hydraulic system. As a result, the discrepancy in Fig. 8 occurred in the real system.

The effects of nonlinearities can be clearly seen in Fig. 10, wherein step responses are shown for a command that is shifted by 0.2 rads gradually after every 5 s. For joint angle regulation, a simple PI controller is implemented:

$$u_{PWM} = PI(\Delta x). \tag{10}$$

The closed-loop responses differ from each joints. They also depend on the actuation direction and range. Moreover, the number of actuated joints also affects the responses. Figure 10 clearly presents the effect of internal delay which also can be seen in Experiments with the PTP movement and path tracking. The internal delay can be explained as the required time for the pressure within cylinders to be built up. Such a delay has been experimentally estimated around 1.3 s for all the joints.

5.2 Experiment 2: Inserting the chisel into a borehole

To further demonstrate the capability of our method, we show the demolition machine BROKK 170 semi-autonomously finding and inserting the chisel into a borehole. In a drill-and-split method, rocks are split by pressures exerted on boreholes (Volden 2015). However, moving the tool center point (TCP) into a hole in a teleoperated paradigm is a challenging task for operators, since operators usually have to control the multiple joints in a very coordinated way solely relied on a 2D camera system.

The carried task is to semi-autonomously insert the chisel with a 6 cm diameter into a hole with an 8 cm diameter without using visual system. In this experiment, we restricted the workspace to xy -plane. The area of the concrete block is measured and used as the boundary of xy -motion. For the finding task, we utilize the pressure sensor which is placed in the hydraulic pump.

For the given task, we designed three unit motions. With the first unit motion, the tool is moved to the test point along the xy -direction. Using the second unit motion, the tool is moved along the $-z$ -direction until the pressure exceeds the threshold value, which means that there is contact between the surface and the tool. Using the third unit motion, the tool is moved along the z -direction. Iteratively, all the test

points are examined with the same procedure. Using a spiral toolpath on the xy -plane, the searching procedure could be minimized, since motions along the $-z$ - and z -direction are not required. However, as discussed in Experiment 1, moving multiple joints at the same time increases the error due to the nonlinearities in the hydraulic system. Moreover, spiral motion along the surface assumes that the tool is constantly in contact with the surface. A slight discrepancy in axes motion leads to a situation where the tool is unexpectedly moved into the $-z$ -direction. Thus, the pressure value exceeds the threshold value and the system assumes that the hole is found. To overcome this problem, simultaneous axes motion is minimized by splitting the task in different unit motions. Figure 11 presents the TCP position conducted experiment, wherein the different phases are illustrated:

- (1) *Phase 1* In the first phase, the TCP moves towards a test point using the first unit motion.
- (2) *Phase 2* If the TCP is directly above the test point, it moves along the $-z$ -direction with the second unit motion. It moves down until the pressure value exceeds a threshold value. Otherwise, if the z value of the TCP is below the pre-defined -0.1 m, we assume that the hole is detected.
- (3) *Phase 3* If the TCP cannot be moved further and the pressure value is above the threshold value, we assume that there is contact between the TCP and the surface.
- (4) *Phase 4* The TCP moves along the $-z$ -direction with the third motion.

The trajectory of the TCP shown in Fig. 11 and also the snapshots from Fig. 12 validate the capability of our approach for enhancing automation. The time required for the task was 143 s which is clearly slower than an experienced operator. However, key advantages of this enhanced automation are that the operator no longer needs to control

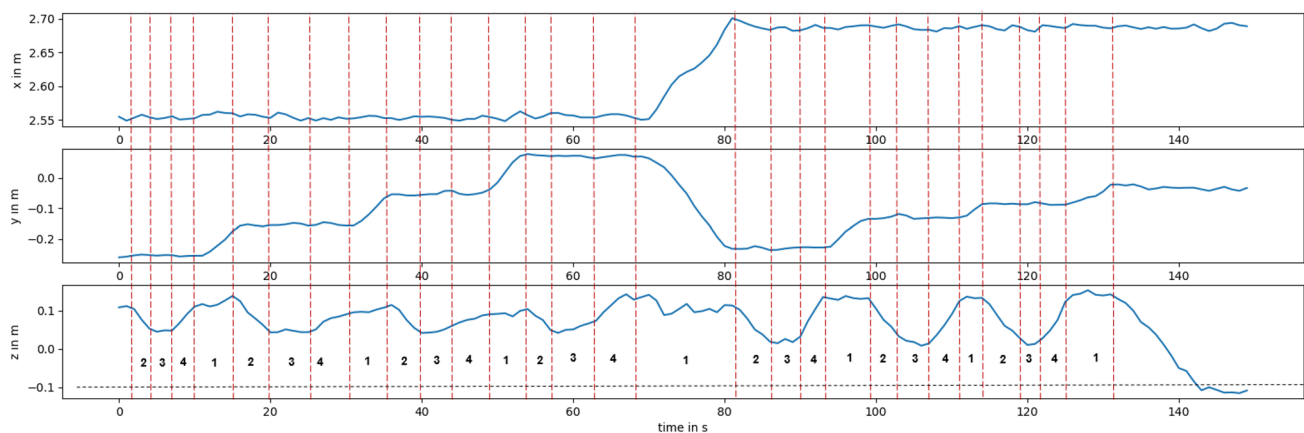


Fig. 11 Translational position of the TCP in the Cartesian space plotted against time

each joint independently and can concentrate only on the tool motion. Thus, the burden on human operators is reduced and the human operator can meanwhile perform other tasks. Moreover, since the approach solely relies on the pressure value of the main pump, the camera installation to provide detailed situational awareness for teleoperation is not required.

6 Current limitations

The presented method allows to deploy algorithmic control setups for efficient control of a teleoperated machine without serious hardware modification. However, since this method relies on the existing hardware and software architecture, inherent limitations exist. Some of the issues are summarised in the following section.

Teleoperated construction machinery is often controlled with levers. The 5 DoF actuated Brokk 170 can be fully controlled with two levers. However, as depicted in Fig. 3, the motion of the second joint is controlled by pressing the B2.RR button, whereas the other button B2.RL has to be used for other axes. In the CAN bus network specified by the manufacture, the third byte number of a specific Communication object identifier (COB-ID) describes the activation of the both buttons B2.RR and B2.RL. Since the same byte is responsible for two different groups of joints, motion (i.e., the second joint and the other four joints) cannot be controlled with other axes at the same time using the original control device. By establishing a communication link as described in Sect. 2.1, we can emulate every operation that can be done with the original control device. However, in other words, only those operations with the original input device can be emulated.

Here, a generic IK solver rapidly reaches its limits, since it typically assumes with its output that the machine can move all the five axes at the same time. However, in this case, there is a strict condition that the second joint is not controllable with other joints together. This problem of

disjoint sets of controllable joints will be further investigated in our future work.

Moreover, as shown in Experiment 1, the nonlinearities in the hydraulic system greatly affect high-precision position control. Although reasonable results are obtained with the PTP movement control, it is desired to control the complete path from the start posture to the goal posture to maximize the controllability and thus the safety on-site during machine operation. Consequently, we will consider these nonlinearities of the system as uncertainties in our next work to design a more robust controller.

7 Conclusion

This paper presented an approach enabling teleoperated construction machinery to be programmed, so that construction robots can be obtained at a significant reduction in effort. To avoid major adaptations to existing hardware and software architecture, we proposed a control method which leverages the existing hardware components and CAN bus protocol.

By analyzing the communication between the original input device and the machine, a communication link between a host PC and the machine was established. Based on this result, a generic CLIK algorithm was implemented to convert the desired motion descriptors expressed in task space to corresponding joint motions. Then, the joint motions are converted to corresponding CAN bus messages, which generate the corresponding PWM signals for the valve system. The proposed method successfully enables controlling the teleoperated machine in task space instead of controlling each axis separately with joystick. The first results are showcased in this paper.

Although the experiments presented in this work are still within a simple framework, the feasibility of our method that proven construction machinery can be adapted and advances from robotic can be utilized for the control of this machinery is validated. While industrial robotic technology has been greatly advanced during the last decades, construction sites



Fig. 12 Snapshots of Brokk 170 recorded during experiment 3

have always been a great challenge for robotic automation due to the unstructured nature and dynamically changing outdoor environment. Thus, the current degree of robotic automation within construction is still very low despite of the considerable importance of construction in the society. We expect that this work is going to be another option for automating construction sites, since it can reduce the effort to obtain a programmable machine suited for construction sites and other hazardous environments.

We concluded this work by emphasizing the existing limitations of our approach. While programming movements was already realized in this work (see Experiment 1 and 2), the lack of consideration of dynamic properties of a hydraulic machine resulted in limitations to keep up precision position control. The identified problems will be further investigated in our next work. Finally, we plan to integrate a motion planning module to produce the optimal trajectory within the workspace of the construction machine.

Acknowledgements This work is partially supported by the BROKK DA GmbH.

Funding Open Access funding enabled and organized by Projekt DEAL.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Brell-Cokcan S, Braumann J, Cokcan B, Kleindienst M (2012) Just-in-time design: developing parametric design tools for architectural design. In: Beyond codes and pixels: Proceedings of the 17th International Conference on Computer-Aided Architectural Design Research in Asia 2012, pp 455–464
- BROKK (2018) Troubleshooting guidelines. Technical report, BROKK AB. <http://www.brokkinc.com/media/troubleshooting.pdf>
- Buss SR (2004) Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods. *IEEE J Robot Autom* 17(1–19):16
- Chan TF, Dubey RV (1995) A weighted least-norm solution based scheme for avoiding joint limits for redundant joint manipulators. *IEEE Trans Robot Autom* 11(2):286–292
- Chen J, Haas E, Barnes M (2007) Human performance issues and user interface design for teleoperated robots. *IEEE Trans Syst Man Cybern Part C: Appl Rev* 37:1231–1245
- Dariush B, Gienger M, Jian B, Goerick C, Fujimura K (2008) Whole body humanoid control from human motion descriptors. In: *IEEE International Conference on Robotics and Automation*, Pasadena, CA, 2008, pp 2677–2684. <https://doi.org/10.1109/ROBOT.2008.4543616>
- Feng C, Xiao Y, Willette A, McGee W, Kamat V (2015) Vision guided autonomous robotic assembly and as-built scanning on unstructured construction sites. *Autom Constr* 59:128–138
- Gifftthaler M, Sandy T, Dörfler K, Brooks I, Buckingham M, Rey G, Kohler M, Gramazio F, Buchli J (2017) Mobile robotic fabrication at 1:1 scale: the In situ Fabricator. *Constr Robot* 1:3–14. <https://doi.org/10.1007/s41693-017-0003-5>
- Heimig T, Kerber E, Stumm S et al (2020) Towards robotic steel construction through adaptive incremental point welding. *Constr Robot* 4:49–60. <https://doi.org/10.1007/s41693-019-00026-4>
- Hutter M et al (2015) Towards optimal force distribution for walking excavators. In: *International Conference on Advanced Robotics (ICAR)*, Istanbul, 2015, pp 295–301. <https://doi.org/10.1109/ICAR.2015.7251471>
- Jamison Gerbo E, Saliklisa EP (2014) Optimizing a trussed frame subjected to wind using rhino, grasshopper, Karamba and Galapagos. In: *Proceedings of the IASS-SLTE 2014 Symposium Shells, Membranes and Spatial Structures: Footprints Brazil 2014*
- Jud D, Leemann P, Kerscher S, Hutter M (2019) Autonomous Free-Form Trenching Using a Walking Excavator. *IEEE Robot Autom Lett* 4:3208–3215
- Keating SJ, Leland JC, Cai L, Oxman N (2017) Toward site-specific and self-sufficient robotic fabrication on architectural scales. *Sci Robot* 2:eaam8986. <https://doi.org/10.1126/scirobotics.aam8986>
- Kim J, Jin M, Choi W, Lee J (2019) Discrete time delay control for hydraulic excavator motion control with terminal sliding mode control. *Mechatronics* 60:15–25. <https://doi.org/10.1016/j.mechatronics.2019.04.008>
- Lee HJ, Brell-Cokcan S, Schmitz K (2019) A general approach for automating teleoperated construction machines. In: Berns K, Görges D (eds) *Advances in Service and Industrial Robotics. RAAD 2019. Advances in Intelligent Systems and Computing*, vol 980. Springer, Cham. https://doi.org/10.1007/978-3-030-19648-6_25
- Lee HJ, Brell-Cokcan S (2020) A probabilistic motion control approach for teleoperated construction machinery. In: Hisashi O, Hiroshi F, Kazuyoshi T (eds) *Proceedings of the 37th International Symposium on Automation and Robotics in Construction (ISARC)*. Kitakyushu, Japan: International Association for Automation and Robotics in Construction (IAARC), S. 571–577
- Motzko C, Klungenberger J, Wöltjen J, Löw D (2016) *Bewertungsmatrix für die Kostenplanung beim Abbruch und Bauen im Bestand. Datenbanksystem zur Analyse und Bewertung in Bezug auf Kosten, Technologien und Dauern. Abschlussbericht 2015*. Fraunhofer IRB Verlag
- Naboni E (2014) Integration of Outdoor Thermal and Visual Comfort in Parametric Design. <https://doi.org/10.13140/2.1.1448.4808>
- Nakamura Y, Hansafusa H (1986) Inverse kinematic solutions with singularity robustness for robot manipulator control. *J Dyn Syst Meas Control* 108(3):163–171
- Osumi H (2014) Application of robot technologies to the disaster sites. Report of JSME Research Committee on the Great East Japan Earthquake Disaster, pp 58–74
- Park J, Lee B, Kang S, Kim PY, Kim HJ (2017) Online Learning control of hydraulic excavators based on echo-state networks. *IEEE Trans Autom Sci Eng* 14(1):249–259. <https://doi.org/10.1109/TASE.2016.2582213>

- Roudsari S, Pak M (2013) Ladybug: a parametric environmental plugin for grasshopper to help designers create an environmentally-conscious design. Proceedings of BS 2013: 13th Conference of the International Building Performance Simulation Association, pp 3128–3135
- Steinhagen G, Braumann J, Brüninghaus J, Neuhaus M, Brell-Cokcan S, Kuhlenkötter B (2016) Path planning for robotic artistic stone surface production. In: Reinhardt D, Saunders R, Burry J (eds) *Robotic Fabrication in Architecture, Art and Design 2016*. Springer, Cham. https://doi.org/10.1007/978-3-319-26378-6_9
- Stumm S, Brell-Cokcan S (2019) Haptic programming: foreword by sigrid brell-çokcan and johannes braumann, association for robots in architecture. https://doi.org/10.1007/978-3-319-92294-2_4
- Tomatsu T, Nonaka K, Sekiguchi K, Suzuki K (2015) Model predictive trajectory tracking control for hydraulic excavator on digging operation. In: *IEEE Conference on Control Applications (CCA)*, Sydney, NSW, 2015, pp 1136–1141. <https://doi.org/10.1109/CCA.2015.7320765>
- Volden JAB (2015) Engineering geological evaluation of the applicability of drill & split in tunnels at the Follo line project. Master's thesis, NTNU

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.