Computers and Electronics in Agriculture 183 (2021) 106037

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

A study case of Dynamic Motion Primitives as a motion planning method to automate the work of forestry cranes

Pedro La Hera[a,*], Daniel Ortiz Morales[b], Omar Mendoza-Trejo[a]

[a] SLU, Forest Biomaterials and Technology Department, Skogsmarksgränd 17, 901 83 Umeå, Sweden
[b] Cranab AB Karlsgårdsvägen 56, 922 32 Vindeln, Sweden

ARTICLE INFO

Keywords:
Forestry cranes
Dynamic motion primitives
Motion planning
Motion control systems
Robotics

ABSTRACT

Dynamic motion primitives (DMPs) is a motion planning method based on the concept of teaching a robot how to move based on human demonstration. To this end, DMPs use a machine learning framework that tunes stable non-linear differential equations according to data sets from demonstrated motions. Consequently, the numerical solution of these differential equations represent the desired motions. The purpose of this article is to present the steps to apply the DMPs framework and analyse its application for automating motions of forestry cranes. Our study considers an example of a forwarder crane that has been equipped with sensors to record motion data while performing standard work in the forest with expert operators. The objective of our motion planner is to automatically retract the logs back into the machine once the operator has grabbed them manually using joysticks. The results show that the final motion planner has the ability of reproducing the demonstrated action with above 95% accuracy. In addition, it has also the versatility to plan motions and perform similar action from other positions around the workspace, different than the ones used during the training stage. Thus, this initial study concludes that DMPs gives the means to develop a new generation of dynamic motion planners for forestry cranes that readily allow merging the operator's experience in the development process.

1. Introduction

1.1. Overview

Sweden is well known for its cut-to-length (CTL) harvesting practices featuring a combination of harvester and forwarder machines. In CTL, the harvester cuts trees to small lengths within the harvest area and then they are loaded onto a forwarder, which then transports them to the yard for unloading. A yearly average of 90 million cubic meters of wood are extracted out of Swedish forests using this system, making Sweden the third largest exporter of pulp and sawn timber (Eriksson and Lindroos, 2014).

With the passing of years, manufacturers of forestry machines have been offering larger and heavier machines as response to the demand for higher productivity. However, as it is impractical to keep making machines bigger, wood harvesting productivity in Sweden has stagnated for the past two decades. The most productive machines are heavy and have a negative effect on the forest environment and the operator. These effects come in the form of soil damages, chemical pollution, high fuel consumption, unergonomic levels of vibrations, and slow learning curve for human operators (Labelle and Lemmer, 2019). Therefore, further increases in size is no longer a viable solution that forestry can resort to increase productivity.

Increasing productivity of harvesting operations is a complex subject that can be approached from a number of different alternatives. Some of these include a combination of better forest operation planning, better reforestation planning, or better machine technology, just to mention some (Häggström and Lindroos, 2016). This article focuses on technological improvements that can be done to forestry machines in the short term. Our main topic involves the hardware and software used in forestry cranes, because cranes play a significant role in these processes, giving operators the ability to quickly, safely and accurately manoeuvre logs. However, despite seemingly simple construction, forestry cranes are difficult to operate using joysticks (Morales et al., 2014; Nurminen et al., 2006). Therefore, forestry companies employ skilled operators, rigorously trained to work efficiently and to minimize risk of accidents.

As witnessed by recent commercial products, as well as it is described in a variety of research articles, the next wave of technology development to boost the productivity of forestry machines features automation technology (Lindroos et al., 2019; Gingras and Charette, 2017; Reitz

[*] Corresponding author.
E-mail addresses: xavier.lahera@slu.se (P.L. Hera), daniel.morales@cranab.se (D.O. Morales), omar.mendoza.trejo@slu.se (O. Mendoza-Trejo).

https://doi.org/10.1016/j.compag.2021.106037
Received 23 October 2020; Received in revised form 29 January 2021; Accepted 31 January 2021
Available online 9 March 2021

et al., 2019). For forestry cranes, the most attractive concepts, in the short term future, involve a human-robot partnership, in which operators take advantage of advanced capabilities of ”intelligent computer-assisted support” (ICAS) to facilitate their work. Using an ICAS system implies that the machine is equipped with hardware and smart software capable to perform tasks semi-autonomously, easing operator’s job while increasing machine’s working performance (Morales et al., 2014). In the shorter term, these tasks imply to simply perform specific motions, while in the longer term more complete actions. The operator stays under command, either in the machine’s cabin (in the short term) or through remote operation (in the longer term (Westerberg and Shiriaev, 2013)).

If we consider that the machine operator is still the eyes of the operation, methods in robotics to accomplish this goal in forestry cranes involve adaptations and/or combinations of two main approaches:

- The first is known as Cartesian tip-control (Spong et al., 2006), an essential method used in robotics to ease the motion control of robotic arms. Applied to a forestry crane would imply that the ICAS decides how to move the crane’s cylinders, while the operator only focuses on moving the crane’s end-effector tool, i.e. the grapple or harvester head (see Fig. 2). In this case, the operator still controls the crane’s motions at all times, similar to conventional solutions. However, this form of operation is highly intuitive to people, easier to learn for beginners, and avoids unnecessary motions to optimize fuel consumption. Examples of research results showing this solution can be found in the work of (Münzer, 2004; Westerberg, 2014). Industrial examples applying this technology today include the Intelligent Boom Control from John Deere (IBC), the Crane Tip Control from HIAB, and there are many other examples sold by consultancy firms around Scandinavia (Manner et al., 2019; Technion, 2017).
- The second approach is known as semi-autonomous motion. Applied to a forestry crane would imply that the ICAS is given the ability to perform task-specific motions without the necessity of full human supervision. An example is when a forwarder crane grabs a log, and then it automatically retraces it back to the log-bunk. To this end, the operator may only need to click a button to perform certain actions. Therefore, this control method has more autonomy and can reduce operator’s task more dramatically, reducing mental fatigue, learning time, and enables the possibility to optimize the work in other ways. Examples of research results showing practical applications of this approach can be found in the work of (Morales et al., 2014; Westerberg, 2014; Fodor, 2017; Hansson and Servin, 2010). Currently, there are no commercially available industrial examples. However, industrial prototype machines involving this method are currently undergoing development in Scandinavia (CINTOC, 2020).

As observed by these two cases, boom-tip control is an ICAS method that has already reached the market in the recent years, showing the interest of industry behind these technologies. The latest reports suggest that boom-tip control can result in productivity increases within an averaged range of 5% (Manner et al., 2019). However, forestry is at the early stages adopting automation technologies. As research shows, higher gains can be expected from using semi-autonomous motions, as explained by the second case (Morales et al., 2014).

1.2. Semi-Autonomous motion control

Giving an ICAS the ability to control an entire motion opens a number of new possibilities to increase productivity of forestry crane work. This is due to the ability given to the machine to work autonomously performing portions of some tasks, as well as to start developing solutions that can lead to nearly fully automated work. However, planning and controlling motions of a heavy-duty hydraulic manipulator has shown to be a challenging topic, mainly because traditional methods used in robot motion control cannot guarantee robustness in highly dynamic environments (La Hera and Morales, 2015; Ding et al., 2018; Perdersen et al., 2018). This is the case for the complex nonlinear hydraulics dynamics used in these heavy-duty machines (Manring and Fales, 2019), as well as the complex scenery of the forest work environment.

Controlling the motion of a mechanical system often involves a combination of two essential technologies: the motion planning and the motion control (Spong et al., 2006). In a hydraulic system, a motion control algorithm will provide a signal representing the current/voltage or Pulse Width Modulation (PWM) sequence given to the hydraulic valve for moving the cylinders in a predefined way (Manring and Fales, 2019). On the other hand, the motion planning will deliver the specific positions, velocities and accelerations expected from the system to perform a motion. A representation of these two is sketched in Fig. 1.

In forestry, the problem of motion control has had substantial development over the past years. Although many challenges still remain in this topic, today, the most advanced machines come equipped with a new automation-based valve technology known as ”intelligent hydraulic valves” (EATON, 2019), helping to implement motion control more reliably. On the other hand, the topic of motion planning for forestry cranes is more scarce in literature and industry. The most relevant examples of motion planning for semi-autonomous crane motions can be seen in the seminal work of (Morales et al., 2014; Westerberg, 2014; Fodor, 2017; Hansson and Servin, 2010). In all these cases, the authors suggest to use variations of polynomial functions that are useful to embed motion patterns, almost similar to how industrial robot manipulators are programmed in factories. The polynomials help to specify a motion from an initial to a final position, respecting the cylinders’ range limits, as well as their constraints in velocities and accelerations. Nevertheless, research studies show that these methods are suited to well planned environments, such as factory floors, but they are slow and difficult to apply outdoors in highly dynamic environments (LaValle, 2006). The main reason is the high computational cost allocated to dynamically compute optimal polynomial parameters for every single motion, and the necessity to link multiple polynomials when the motions have complex shapes (Spong et al., 2006). This form of optimization problems are mathematically defined as infinite-dimensional constrained problems, which are not always easy and fast to compute (Fattorini et al., 1999). Therefore, these methods can lead to motion planning with high mathematical and computational complexity, which can limit the possibility to quickly and dynamically adapt motions from task to task. Thus, they are not directly appropriate for planning motions of forestry cranes, because time is an important performance criteria working with forestry machines, as discussed by the authors of (Morales, 2015).

In the context of alternatives to traditional motion planning methods, learning from demonstration (LfD) is the paradigm in which robots acquire new skills by learning to imitate an expert. The choice of LfD over other motion planning methods is compelling when ideal

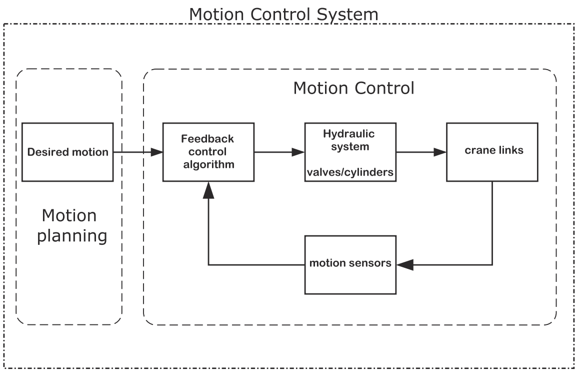

Fig. 1. A block diagram representation of the two components commonly found on a motion control system.

behaviour can be neither easily scripted (as it is done in traditional robot programming), nor easily defined as an optimization problem, but can be demonstrated. To this end, LfD is a subject involving a variety of machine learning alternatives (Ravichandar et al., 2020). Today, LfD is an approach that is both highly successful in industry, as well as in outdoor applications, whenever motions or actions need to be constantly modified. Thus, this makes LfD a more suitable solution for planning motions of heavy duty systems, such as forestry cranes.

1.3. Problem formulation

To put this into perspective, Fig. 3 shows a sketch of a forwarder crane lifting logs. The motion of the boom-tip results from individually moving each boom cylinder (see Fig. 2). These cylinders are controlled by the operator via joysticks placed in the machine's cabin. As discussed in (La Hera and Morales, 2019), the redundancy in the number of degrees of freedom provoke the scenario that there exists infinitesimal ways to control these cylinders for performing the same crane motion. Thus, the difficulty for the operator is to coordinate a set of motions that are both comfortable and profitable for performing the work efficiently.

According to concepts of LfD, if the crane is equipped with motion sensors at each joint, then it is possible to record all crane motions performed by a professional operator. If the data of these motions is available to a learning algorithm, then this algorithm can be used consequently as a motion planner that is able to reproduce similar motions to what an expert human operator performs.

Among LfD methods, Dynamic Movement Primitives (DMPs) is one of the most applied methods in robotics, due to its mathematical ability of representing complex actions using well-known stable differential equations (Ijspeert et al., 2013). The advantage of DMPs over other machine learning methods is that it provides a formal mathematical framework to learn motions from demonstrations. After the learning process takes place, the resulting DMPs' models have the versatility of planning trajectories at different speeds, amplitudes, and include perturbations, without the necessity of retraining these models (Ravichandar et al., 2020).

Therefore, our hypothesis is that DMPs is one of the most appropriate methods to plan motion of forestry cranes given the following considerations:

- the machine has the necessary sensor hardware to measure motion
- the machine's computer has the computational ability and memory to record motion data
- the machine has the necessary motion control system to consequently use the motion planner after the learning phase

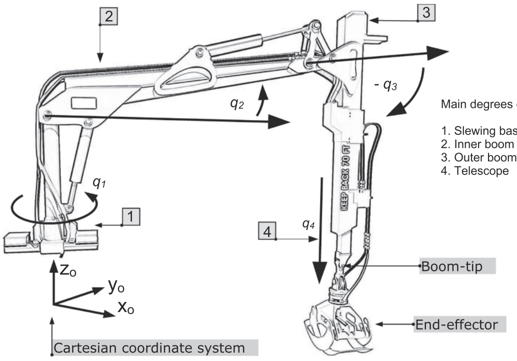

Fig. 2. Forwarder crane. It is a hydraulic manipulator with four degrees of freedom, specified in this graph as the slewing q_1, inner boom q_2, outer boom q_3, telescope q_4. It holds an end-effector attached at the boom-tip, serving as a tool to grab logs, also called grapple. The sensors for these angles measure positive in counter-clockwise direction.

Although these considerations may be trivial in the area of robotics, it is important to recall that machines equipped with motion sensors are rare in the forestry industry. Similarly, most forestry machines are not equipped with sophisticated computing power. Nevertheless, at least in Scandinavia, most forestry machines since the year 2018 fulfil these requirements to some extent.

1.4. Main proposition

The purpose of this article is to investigate the application of the DMPs method originally introduced by the authors of (Ijspeert et al., 2013). To this end, the goal is to develop a motion planning system resulting from learning motions performed by professional operators. To the best of our knowledge, this is the first time that this method is being introduced for forestry applications.

The requirements of using DMPs in our case is to develop a motion planning strategy that is 1) simple to implement, 2) easy to adapt to different working conditions, 3) low computational cost, and 4) allows to incorporate motion optimization (more on this will be explained in the discussion section). As research shows, DMPs provide the following advantages:

- motions can be quickly planned without the necessity to constantly calculate optimal polynomial coefficients.
- mathematically, it is a machine learning method involving sets of stable differential equations that can be trained with relatively small data sets. Consequently, the solutions of these equations represent the desired trajectory: position, velocity and acceleration. In addition, these differential equations have shown to be flexible and robust enough handling obstacle avoidance, changes in velocities, and changes in amplitudes.
- Multiple machines can be programmed similarly with motions that have been taught by a single expert. However, re-training the system is simple and can be given as an option in every machine.
- It is additionally possible to improve the learning process of the algorithm by modifying the learning data sets exploiting the kinematic redundancy. For instance, this modification can be done by using additional motion optimization over the original data set, thus resulting in a data set of optimal motions (Dong et al., 2020). More on this will be explained in the discussion section.

This article considers the case of a forwarder crane as a first study case showing the application of DMPs for developing a motion planning strategy to automatically collect logs. The objective is to develop a motion planning capable of automatically retracing the crane back to the carrier (log-bunk) once a machine operator has grabbed logs. This would resemble a task specific motion that can be accessed with the click of a button. Thus, simplifying the work for a machine operator by almost half. To this end, this article presents the essential steps to develop this motion planning system based on the DMPs framework. The study uses data that has been recorded from professional operators in Sweden. This data has been originally presented in (La Hera and Morales, 2019) to analyse and compare the work of machine operators. Here, the data will be strictly used in the steps of machine learning, but no analysis of the expertise of machine operators over the control of the crane will be presented. For interested readers, analysis of crane motions using similar data has been presented by the authors of (La Hera and Morales, 2019; Morales et al., 2014).

2. Materials and methods

2.1. Materials

2.1.1. Experimental setup and recorded data

The machine in this study is a Komatsu Forwarder 830 (Komatsu Forest, 2011). According to specifications, this machine uses a crane

CRF 5.1 from CRANAB, with a maximum length of 9.3 m. The crane itself is equipped with a G28 grapple, which rotates via a G121 hydraulic motor from Indexator.

To gather data, we equipped the crane with motion sensors at the main degrees of freedom, i.e. four joints, as it is specified in Fig. 2. These sensors were installed externally, by adding static metallic holders in the frames of the joints [1]. The sensors are high resolution quadrature encoders from the brand Heidenhain, item number ROD 426–5000. They provide a measurement resolution of 5000 pulses per revolution, meaning that they can measure as low as 0.072 degrees (0.0012 rad) for the angular joints and 0.0007 m (0.7 mm) for the telescope. We also installed a real-time data acquisition unit (DAQ) able to record signals at a frequency of 1 kHz (1000 recordings every second). The sensors and the DAQ work in parallel to the Komatsu system, meaning that they have no interference with the normal method of operating the machine.

The crane motions controlled by the machine operators involved those used to collect logs inside the forest. Referring to Fig. f3, in the remaining of this article these motions will be named as the retracting and expanding motions. They refer to the actions of expanding the crane to grab logs from the ground, and retracting the crane to bring logs back into the log-bunk. For interested readers, a full analysis of these motions are presented in (La Hera and Morales, 2019). Here, this recorded data is strictly used as a set for training DMPs.

2.2. *Methods*

Referring to Fig. 2, forwarder cranes are a RRRP[2] type of manipulators, having one degree of redundancy (Spong et al., 2006). The redundant degree of freedom is the telescopic link, implying that motions of this link allow to optimize overall crane motion-performance (Morales et al., 2014).

The following subsections present the mathematical principles required to analyse the crane kinematics. The kinematics analysis is a necessary step to calculate the exact coordinates for each link, as well as the Cartesian coordinates of the boom-tip given the measurements from sensors. This is done through the calculation of the forward kinematics (Spong et al., 2006).

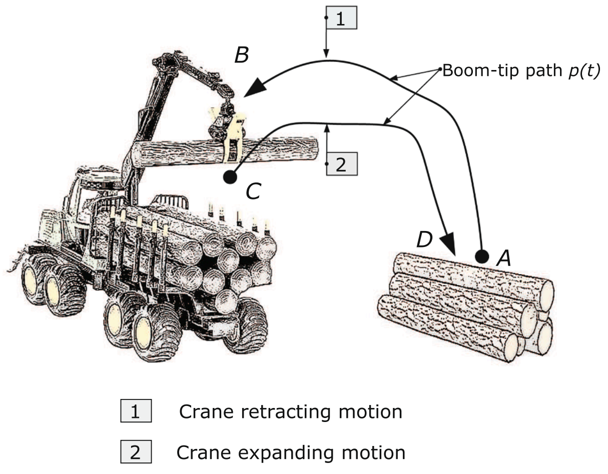

Fig. 3. Crane boom-tip motion patterns for grabbing and releasing trees. 1) Retracting path to the load bunk. 2) Expanding path from the load bunk. The paths are drawn as examples to show the direction of the motion. Usually the retracting path is higher than the expanding path, because the crane needs to avoid hitting the log-bunk when it is holding logs. The expanding path, however, can be much lower, because the crane can go back to the side crossing through the empty spaces between the log-bunk poles.

In addition, the inverse kinematics is formulated, because it is a necessary analytical tool for motion planning. As it is convenient to define a desired motion as a path of the boom-tip (see Fig. 3), the inverse kinematics analysis is useful to relate this desired boom-tip path to the required motions of the individual joints.

Consequently, this section provides the procedure to analyse data and train a motion planning model using the DMPs framework. Lastly, we indicate how the performance of the resulting motion planner is tested.

2.2.1. *Forward kinematics analysis*

A forwarder crane is an open kinematic chain formed by four links from the base to the joint where the end effector is attached, see Fig. 2. The joints are structured as follows:

1. Revolute joint for *slewing*, associated with q_1.
2. Revolute joint for the *inner boom*, associated with q_2.
3. Revolute joint for the *outer boom*, associated with q_3.
4. Prismatic joint for *telescopic extension of the outer boom*, associated with q_4.

These joint variables help define the vector of generalized coordinates as $q = [q_1, q_2, q_3, q_4]^T \in \Re$ for this four degree-of-freedom system. The forward kinematics, i.e. calculation of the boom-tip coordinates given the joint angles, can be conveniently expressed using the Denavit-Hartenberg (DH) convention (Spong et al., 2006), where each link configuration is represented by the homogeneous transformation

$$A_i(q_i) = \mathrm{Rot}_{z,\theta_i}\,\mathrm{Trans}_{z,d_i}\,\mathrm{Trans}_{x,a_i}\,\mathrm{Rot}_{x,\alpha_i}, \tag{1}$$

parametrized by joint angle θ_i, link offset d_i, link length a_i, and link twist α_i (see Fig. 4). Table 1 provides the values required to perform these calculations. The Cartesian position of the boom tip with respect to the base frame of the manipulator is defined by

$$p = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = [\, I_{3\times3} \quad 0_{3\times1} \,]\, T_0^4(q) \begin{bmatrix} 0_{3\times1} \\ 1 \end{bmatrix}, \tag{2}$$

where $T_0^4(q) = A_1(q_1)A_2(q_2)A_3(q_3)A_4(q_4)$, and p is the vector of the boom-tip's Cartesian coordinates for a given value of the joint coordinates q. Notice that the values for q are measured through sensors.

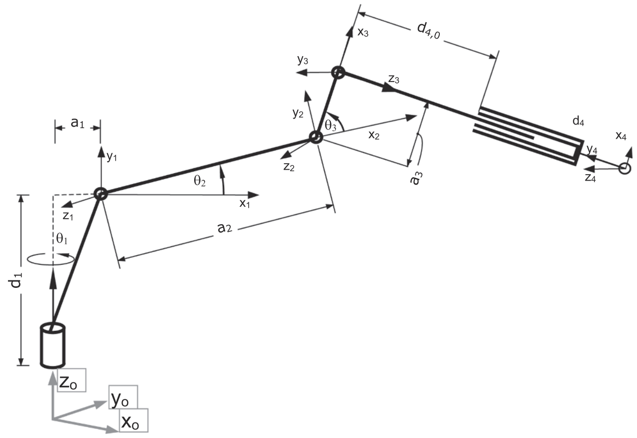

Fig. 4. Sketch of the forwarder crane with its specified coordinates to define the Denavit-Hartenberg convention. The values corresponding to this figure are in Table 1.

[1] On keeping up with our development, today, the company CRANAB offers cranes with fully embedded motion sensors (Cranab, 2011).
[2] R = revolute, P = prismatic.

Table 1
DH parameters of the 4-link manipulator.

Link i	θ_i [rad]	d_i [m]	a_i [m]	α_i [rad]
1	$q_1(t)$	3.24	0.02	$\pi/2$
2	$q_2(t) + \theta_{2,0}$	0	3.40	0
3	$\pi/2 + q_3(t) - \theta_{2,0}$	0	0.21	$\pi/2$
4	0	$d_{4,0} + q_4(t)$	0	$-\pi/2$
Constants: $\theta_{2,0} = 0.0496\,\mathrm{rad}$, $d_{4,0} = 2.42\,\mathrm{m}$				

2.2.2. *Inverse kinematics analysis*

The inverse kinematics, i.e. the calculation of joint angles given the boom-tip Cartesian configuration, is a set of nonlinear equations resulting from analysing the crane geometrically. As this is a redundant four degrees of freedom system, a closed form solution for the inverse kinematics does not exists (Spong et al., 2006). However, in these cranes, the redundant degree of freedom is the telescopic link q_4. Therefore, it is possible to find a closed form solution for the remaining links, provided that both the Cartesian coordinates p of the boom-tip are given, as well as the value of q_4, i.e.

$$q_{1,2,3} = F(p^0, q_4), \tag{3}$$

where p^0 is the Cartesian coordinate of the boom-tip. The functions of $F(\cdot)$ to perform this calculation are the following

$$\begin{aligned} q_1 &= \operatorname{atan2}(y, x), \\ q_2 &= \operatorname{atan2}\left(z - d_1, \sqrt{x^2 + y^2}\right) - \theta_{20} \\ &\quad + \cos^{-1}\left(\frac{a_2^2 + x^2 + y^2 + (z - d_1)^2 - (d_{40} + q_4)^2 + a_3^2}{2a_2\sqrt{x^2 + y^2 + (z - d_1)^2}}\right) \\ q_3 &= \theta_{20} - \tan^{-1}\left(\frac{a_3}{d_{40} + q_4}\right) \\ &\quad + \cos^{-1}\left(\frac{x^2 + y^2 + (z - d_1)^2 - a_2^2 - (d_{40} + q_4)^2 + a_3^2}{2a_2\sqrt{(d_{40} + q_4)^2 + a_3^2}}\right). \end{aligned} \tag{4}$$

where atan2 is the programming version of the arctangent function. Therefore, having an specified path $p^*(t)$ and a trajectory for $q_4^*(t)$, the Eqs. (4) are useful to explicitly calculate the remaining degrees of freedom.

2.2.3. *Dynamic motion primitive framework*

Unlike traditional polynomial approximations used in robotics, DMPs main basic idea is to use an analytical dynamical system with well understood response to generate desired trajectories. This dynamical system should inherently guarantee stability properties and should provide a response that resembles a desired trajectory by using a ”forcing term”. The essence of the DMPs framework consists in training a model of the forcing term using data sets of demonstrated motions (Ijspeert et al., 2013), such that the response of the dynamical system resembles the learned motion. To guarantee that the dynamical response goes from an initial to a desired goal position, the behaviour of the response is in the form of an attractor towards the goal. One of the simplest possible systems presenting these properties is the second order damped-spring model

$$\tau\ddot{m} = \alpha_n(\beta_n(g - m) - \dot{m}) + f, \tag{5}$$

which can be converted into a first order model of the form:

$$\begin{aligned} \tau\dot{n} &= \alpha_n(\beta_n(g - m) - n) + f, \\ \tau\dot{m} &= n, \end{aligned} \tag{6}$$

where $[m, \dot{m}, \ddot{m}] \in \Re$ are interpreted as the desired trajectory’s position, velocity, and acceleration respectively. τ is a time scaling constant, α_n and β_n are positive constants that determine the spring-damper properties, and g is the final/goal position. The forcing term determining the shape of the response is f and acts in the time interval of the motion $t \in [0, T]$. When $f = 0$, the remaining system is a globally stable second-order linear system with $(n(T), m(T)) = (0, g)$ as unique point attractor from the initial state $(n(0), m(0)) = (n_0, m_0)$. According to (Ijspeert et al., 2013), the choice $\beta_n = \alpha_n/4$ makes (6) a critically stable damped system that converges monotonically towards the point attractor g.

In order to remove the time dependency of (6), a re-parametrization of time $t \in [0, T]$ can be done by the first-order canonical system

$$\tau\dot{s} = -\alpha_s s, \tag{7}$$

where α_s is a constant, such that if $s_0 = 1$, the state vector converges monotonically to zero at the rate given by this value. Therefore, $s = 1$ indicates the start of the motion and s near zero will indicate that the goal g has been reached. The complete system (6) and (7) has a unique equilibrium point at $(n, m, s) = (0, g, 0)$.

The forcing term f can be defined in many different ways by using a variety of nonlinear functions. The authors of (Ijspeert et al., 2013) suggest using a nonlinear term based on Gaussian basis functions defined as

$$f(s) = \frac{\sum_{i=1}^{N} \psi_i(s)\omega_i}{\sum_{i=1}^{N} \psi_i(s)} s(g - m_0), \tag{8}$$

where N is the number of basis functions and

$$\psi_i(s) = \exp\left(-\frac{1}{2\sigma_i^2}(s - c_i)^2\right) \tag{9}$$

with the Gaussian basis functions centres located at c_i and defined as:

$$c_i = e^{-\alpha_s i \frac{T}{N}}, \quad i = 0, 1, \ldots, N \tag{10}$$

In (9), σ_i and c_i are constants that determine the width and centres of the basis functions. The choice of (8) is done due to its well studied properties as a nonlinear regression technique using a sum of Gaussian kernels. This approximation process can be interpreted as a simple kind of neural network, which interestingly does not require any iterative learning. This was the context in which this was originally applied to machine learning (Broomhead and Lowe, 1988).

When the nonlinear function f is trained using data, it renders the model (6) to behave non-linearly, displaying a behaviour that resembles the desired motion. The term $s(g - m_0)$ is useful as a scaling factor that changes the amplitude of the motion, such that different motions can be done with a single model.

To finalize, the aim of the learning process is to find the weights ω_i, such that (8) is able to approximate the forcing term f. To this end, Eq. (6) can be rearranged as

$$f = \tau\dot{n} - \alpha_n(\beta_n(g - m) - n) \tag{11}$$

such that the demonstrated motion $[m_{demo}, \dot{m}_{demo}, \ddot{m}_{demo}]$ can be replaced in (11), resulting in

$$f_{target} = \tau^2\ddot{m}_{demo} - \alpha_n(\beta_n(g - m_{demo}) - \tau\dot{m}_{demo}). \tag{12}$$

which is the target forcing function that (8) needs to approximate. The approximation can be done in software using a locally weighted regression set up to minimize

$$J_i = \sum_{t=1}^{T} \psi_i(t)(f_{target(t)} - \omega_i(s(t)(g - m_0)))^2, \tag{13}$$

with the solution being (Broomhead and Lowe, 1988)

$$\omega_i = \frac{h^T \psi_i f_{target}}{h^T \psi_i h}, \tag{14}$$

where

$$h = \begin{pmatrix} s_{t_0}(g - m_0) \\ \vdots \\ s_{t_P}(g - m_0) \end{pmatrix}, \tag{15}$$

$$\psi_i = \begin{pmatrix} \psi_i(t_0) & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \psi_i(t_P) \end{pmatrix}, \tag{16}$$

$$f_{target} = \begin{pmatrix} f_{target}(t_0) \\ \vdots \\ f_{target}(t_P) \end{pmatrix} \tag{17}$$

For interested readers, the statistical properties of the error associated to this regression approach can be found in (Broomhead and Lowe, 1988). A variety of alternative methods to solve (13) using optimization are discussed in literature (Stulp et al., 2012).

2.2.4. Motion data set

The original raw data of q contains a large quantity of sampled sensor measurements. This data, in its original form, is usually difficult to interpret, because machine operators work non-stop for extended hours performing a variety of crane motions. Technically, the original data is in files with large vectors containing all measurements of q and time. In order to make sense out of this data and apply the DMPs framework, the following preliminary data processing steps are required:

- **Velocity estimation**. It is important to recall that quadrature encoders measure position, but not velocity, nor acceleration. Data for velocity and acceleration needs to be estimated from sampled data of position measurements. Therefore, the first step in our data processing is to estimate $\dot{q}$ and $\ddot{q}$. To this end, we use an estimation method based on Algebraic derivatives, which is fairly new and has proven to be more robust than other approaches (Mboup et al., 2009). The estimators are formulated as

$$\widehat{\dot{q}}(t) = -\frac{6}{T_{win}^3} \int_0^{T_{win}} \varepsilon_1 q(\tau) d\tau \tag{18}$$

$$\widehat{\ddot{q}}(t) = -\frac{3}{T_{win}^3} \int_0^{T_{win}} \varepsilon_2 q(\tau) d\tau \tag{19}$$

where

$$\varepsilon_1 = T_{win} - 2\tau \tag{20}$$

$$\varepsilon_2 = 3T_{win}^2 - 12T_{win}\tau + 10\tau^2 \tag{21}$$

in which T_{win} is the time window for the integral operation, and $\widehat{(\cdot)}$ is the notation for an estimated variable. These derivatives are readily implemented through finite impulse response (FIR) filters (Mboup et al., 2009).

- **Trimming the data.** Despite all the motions that can be done with a forwarder crane, the essential logging work involves four main motions. Referring to Fig. 3, these are the motions to expand the crane for approaching logs and retract the crane for bringing logs back into the log-bunk. These motions happen either to the left or right of the vehicle. Trimming the data refers to the task of extracting these motions from the original data set. To this end, we use a technique that points at the places where the slewing motion reversed direction, which is indicated through values of $\dot{q}_1$ crossing zero. The function $\text{sign}(\dot{q}_1)$ tells whether the motion is for expanding or retracting the crane. The function $\text{sign}(q_1)$ tells whether the motion happened to the left or right of the vehicle, usually perpendicular to the vehicle's frame. In addition, the amplitude of q_1 indicates whether this is a true or false motion. A true motion is a complete path that takes the logs from the ground to the bunk, as seen in Fig. 3, while a false motion happens when an operator performs small adjusting motions.
- **Calculation of the boom-tip path**. This step calculates the crane path in terms of Cartesian coordinates. To this end, the calculations involve introducing the values of q into the forward kinematics (2) to obtain p. A visual example of these calculations is shown in Fig. 5. In addition, the estimation of the boom-tip velocity $\dot{p}$ can also be done using (18).
- **Removing the time off-set**. After trimming data, the resulting time vectors contain the time at which these motions happen. However, in order to treat each motion individually, the off-set in time needs to be removed, so that each individual motion is in the time interval $t \in [0, T_n]$, where n is the index relating to each individual trajectory that was found during the trimming step, and T_n is the duration for each of these trajectories.
- **Re-parametrization**. Trimming the data to separate motions leads to vectors that have different lengths, i.e. the number of elements differ, because each motion has different duration. Thus, performing further algebraic operations over vectors with different lengths is fundamentally impractical. To tackle this problem, our data processing parametrizes each motion according to a new normalized variable. To this end, we use the arc length calculated as:

$$L(t) = \int_{p_0}^{p_t} \sqrt{x'(t)^2 + y'(t)^2 + z'(t)^2} dt \tag{22}$$

which provides a monotonic variable for each trajectory. The resulting arc length can be further normalized to give a unity variable $\theta \in [0, 1]$ as follows

$$\theta(t) = \frac{L(t)}{L(T_n)}. \tag{23}$$

Consequently, performing an interpolation of q and p according to θ allows defining vectors of similar length.
- **Averaging**. Once all the data processing has taken place, the final data sets have matrix form, where each matrix is for one kind of motion, and each column in these matrices corresponds to an individual trajectory. This implies matrices of the form

$$\xi_k = \begin{pmatrix} \xi(\theta(0))_1 & \dots & \xi(\theta(0))_N \\ \vdots & & \\ \xi(\theta(T_n))_1 & \dots & \xi(\theta(T_n))_N \end{pmatrix}, \tag{24}$$

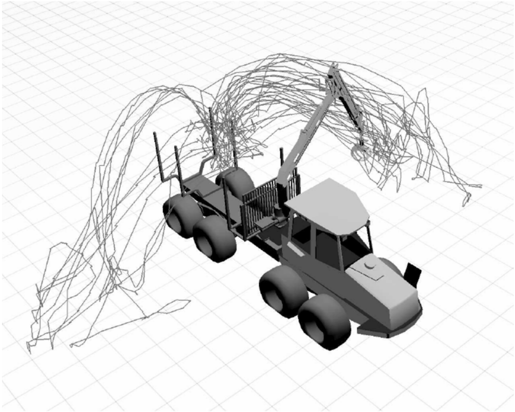

Fig. 5. Forwarder crane boom-tip path. This is an example of using the trimmed data $q(t)$ to reconstruct boom-tip paths $p(t)$.

where k refers to one of the four motions: expand left, expand right, retract left, retract right, and ξ refers individually to the variables q_1, q_2,q_3,q_4,x,y,z,θ, as well as all the estimated derivatives $\dot{q}_1$,$\dot{q}_2$,$\dot{q}_3$,$\dot{q}_4$,$\dot{x}$, $\dot{y}$, $\dot{z}$, $\dot{\theta}$. For instance, a matrix $\xi_k = q_{el}^1$ represent the expanding left trajectories for the joint q_1. All of these matrices contain N columns, each one being an individual trajectory found during the trimming process.

From this point there are two ways to apply the DMPs framework. The first consists of batching all trajectories into (17), such that the operation (14) averages all data. The second method consists in finding the average of all trajectories, and using an averaged set of trajectories to apply DMPs. Considering that expert machine operators have the ability of controlling crane motions using repetitive motions, as demonstrated by (La Hera and Morales, 2019), the latter approach will be considered here, as it has the ability to capture more characteristics of a motion than using a single trajectory. To this end, the step in data processing consists on finding a vector for each matrix corresponding to averaged trajectories, i.e.

$$\overline{\xi}_k = \begin{pmatrix} \overline{\xi}(\theta(0)) \\ \vdots \\ \overline{\xi}(\theta(T_n)) \end{pmatrix} = \begin{pmatrix} \frac{1}{N}\sum_{i=1}^{N}\xi(\theta(0))_i \\ \vdots \\ \frac{1}{N}\sum_{i=1}^{N}\xi(\theta(T_n))_i \end{pmatrix}. \tag{25}$$

where $\overline{(\cdot)}$ is used as notation for the mean of the trajectories.

- **Modifying initial conditions**. One important consideration when using data sets is initial conditions. As explained in Section 2.2.3, system (6) is a globally stable second-order linear system with $(n(T)$ $,m(T)) = (0,g)$ as unique point attractor from the initial state $(n(0)$, $m(0)) = (n_0, m_0)$. In order to guarantee a smooth behaviour, the motion needs to resemble a point-to-point motion starting and ending at rest (Spong et al., 2006). The problem at this stage is that during the trimming of data and final averaging, this requirement might not necessarily be fulfilled, because of the high amount of trajectories being used. Therefore, the final step is to modify the initial conditions of the final averaged trajectories to guarantee that we have a point-to-point motion behaviour starting and ending from rest. To this end, we use a method consisting on filtering velocities with a trapezoidal-like signal, in a way that the final trajectory does not alter the amplitude of motion. Thus, the final trajectories are modified versions of the originals, but they fulfil the requirements to apply the DMPs framework.

2.2.5. Training procedure

There exists two principles to use data sets for training the model of the forcing term (11):

- **Joint Coordinates**. The first method consists in using data of joint positions $\overline{q}_i(t)$, such that the model (6) consequently generates the desired joint positions $q^*(t)$, velocities $\dot{q}^*(t)$ and accelerations $\ddot{q}^*(t)$. A motion controller (see Fig. 1) is often able to directly use this information to control motions (Spong et al., 2006). However, obstacles are often detected in the Cartesian coordinate system. Therefore, in order to adaptively modify trajectories online for obstacle avoidance, then an inverse kinematics calculation is needed to transform the obstacle's world coordinate system into joint coordinates. This method can lead to a mathematical framework that complicates implementation, because these two are in different coordinate systems.
- **Cartesian coordinates**. The second approach consists on using data of the Cartesian paths $\overline{p}(t)$, such that the model (6) consequently generates the desired boom-tip path $[p^*(t),\dot{p}^*(t),\ddot{p}^*(t)]$. However, to produce a motion, the motion controller commonly will control the crane's individual joints q. Therefore, given a desired path $p^*(t)$, it is often necessary to find the desired individual joints positions $q^*(t)$. To this end, an inverse kinematics calculation is required. In the case of obstacle avoidance, the detection in the Cartesian coordinates can be directly placed in the motion planning. Thus, leading to straightforward implementation.

As the case of Cartesian coordinates leads to an straightforward implementation of obstacle avoidance, this is the method we continue exploring further. However, our training procedure uses a data set including $\overline{p}(t)$, as well as $\overline{q}_4(t)$, because this leads to an explicit calculation of the inverse kinematics through (4).

Another important observation is that the training procedure is done using only the final retracting motion to one side of the vehicle. Due to the properties of the model (6), all other variations of similar motions can be done consequently, irrespective of which side of the vehicle the motions are directed to. This will be validated through simulation tests. We recall that the motion planning resulting from the learning phase will be used to automatically bring logs back to the log-bunk, once they have been grabbed by the machine operator.

2.2.6. Evaluation of the final motion planner

To verify the performance of the final motion planner two simulation tests are considered:

- **Test 1**. It consists of simulating the final model (5) with different initial conditions, resembling the act of collecting logs from different locations. These locations represent the places where the machine operator grabs logs and would use the automatic motion planner to bring the logs into the machine. Ideally, a machine operator should mainly collect logs along the y-axis, where (s) he has the highest visibility. Referring to Fig. 6, six different variations around the y-axis are considered here, and to both sides of the machine. These locations are selected according to the characteristics of how machine operators control cranes (La Hera and Morales, 2019). The purpose is to observe the ability of the motion planner for dealing with variations in initial conditions, amplitude of motion, and velocities, despite being trained only with a single data set.
- **Test 2**. It consists of simulating the final model (5) with different initial conditions that are further from the machine than the original data set. To this end, the telescopic link needs to open to nearly its maximum extension. As it will be observed later, one of the main problems with machine operators is their inability to control four links simultaneously (La Hera and Morales, 2019). Therefore, they often refrain from using the telescopic link, as it is highly unintuitive to use it. However, as explained earlier, properly using the telescope

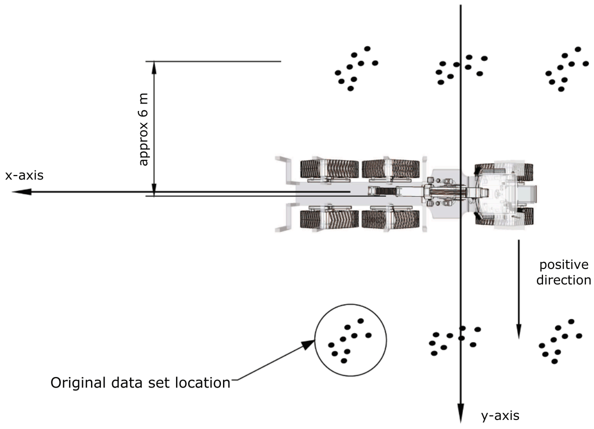

Fig. 6. Different initial conditions for testing. The dark dots represent the locations of the logs, from which the motion planning starts.

translates into optimizing motion performance, because this is the redundant degree of freedom in the system. The idea is to observe the behaviour of the motion planner to large variations in initial conditions, and the modification required in (5) to move this link as desired.

3. Results

3.1. Results of data processing

3.1.1. Trimming data and averaging

Fig. 7 shows the position trajectories that are found for retracting the crane from the left of the machine after the trimming process from data of one machine operator. As explained earlier, these are Cartesian motions representing the boom-tip trajectories p for retracting the crane from the left of the vehicle, as well as the telescopic link q_4. The dark bold signals are the averaged trajectories, as explained by (25). In addition, Fig. 8 shows the velocities that are found using (19), with the dark bold signals being averaged velocities. Both of these figures show data plotted according to the parametric variable $\theta(t)$, which is shown in Fig. 9. This monotonic variable is presented according to time to give an understanding of the duration of motion, as well as the duration of each trajectory. Thus, it is observable that the duration of motion can vary greatly and it mainly depends on how much load the crane is holding: the higher the load, the slower the motion.

3.1.2. Modification of initial conditions

The trapezoidal-like signal to modify initial conditions is presented in Fig. 10. This signal has a smooth profile at its edges, to minimize the effects over the behaviour of the original accelerations. Similarly, Fig. 11 shows the Cartesian position trajectory after filtering. As observed, the variation of the filtered trajectory compared with the original version is minor, laying in the range of millimetres. Differences in the range of millimetres can be considered negligible, because these cranes move in the range of several meters. The velocities are not presented, as it is difficult to observe any noticeable variation.

3.2. Results of applying the DMPs framework

3.2.1. Forcing term (11)

Having the demonstrated trajectories for the boom-tip path enables the possibility to find the forcing terms (11). Referring to Fig. 12, there are four forcing terms for training a motion planning model able to plan trajectories for $[x, y, z]$ and q_4, one for each of these variables. In Fig. 12,

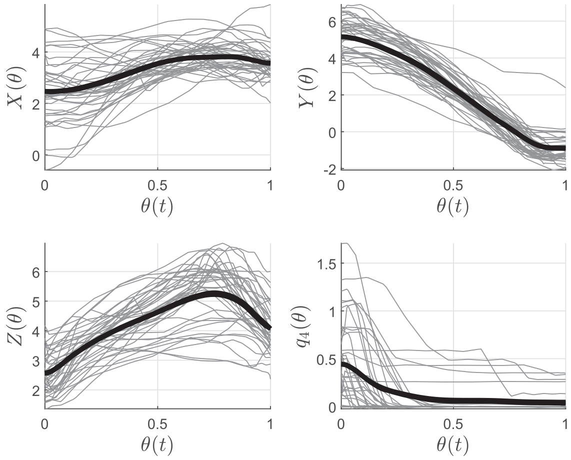

Fig. 7. Final data set. Cartesian coordinate trajectories p found after trimming the original data set, including the telescopic link q_4. The bold signal represents the average of all trajectories.

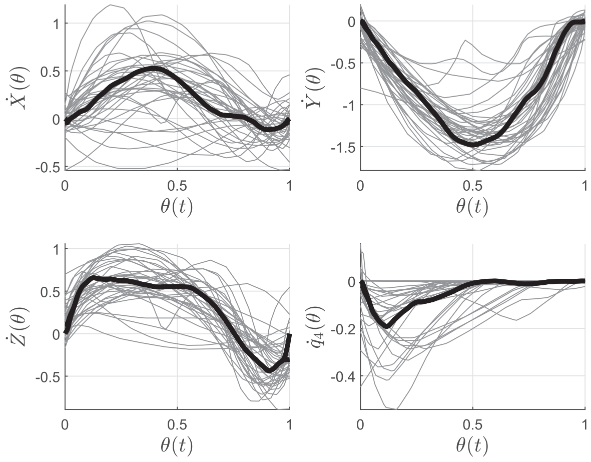

Fig. 8. Final data set. Cartesian coordinate velocities $\dot{p}$ found after trimming the original data set, including the telescopic link's velocity $\dot{q}_4$. The bold signal represents the average of all trajectories.

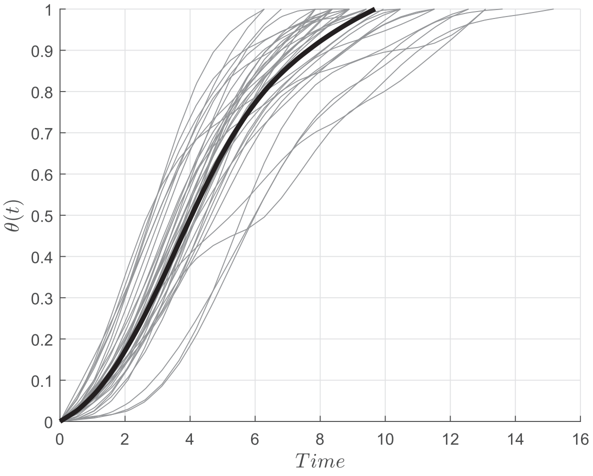

Fig. 9. Final data set. Curve length (23) normalized in respect to time. The bold signal is the averaged length used for interpolations.

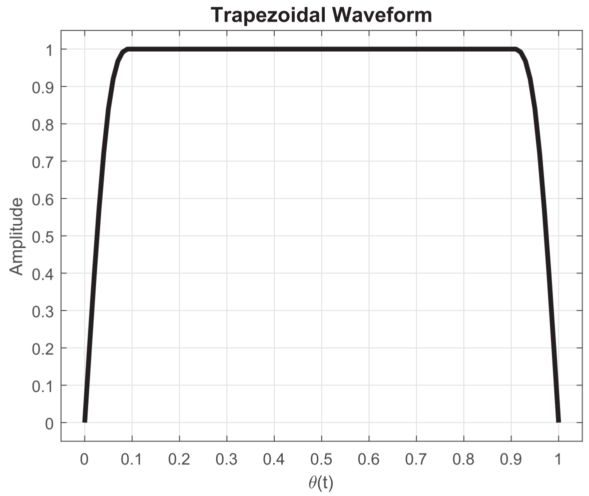

Fig. 10. Trapezoidal-like wave form with smooth initial and final profiles.

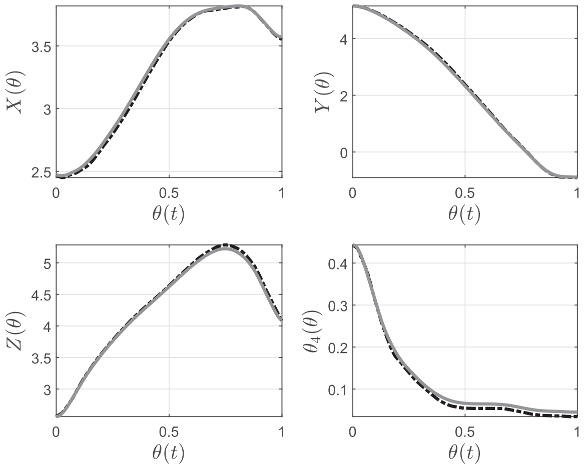

Fig. 11. The solid gray signal is the original one found through our signal processing approach. The dotted signal is the one reconstructed after the velocity has been filtered to correct initial conditions. The plot shows a slight variation in the range of millimetres. However, for a machine that has a reach of nearly 10 meters, this variation is minor.

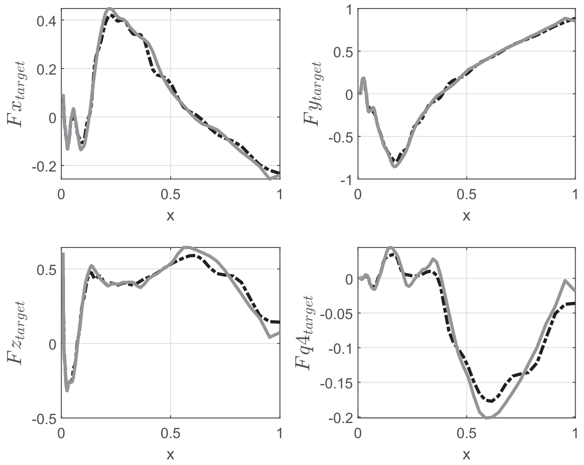

Fig. 12. Forcing term (11) denoted in this figures as $Fx_{target}, Fy_{target}, Fz_{target}, Fq4_{target}$ referring to the forcing terms for the Cartesian coordinates $[x, y, z]$ and q_4. The grey signals are the forcing term resulting from the computation of (11). The dashed black signals are the result from using the locally weighted regression method (13).

the grey solid signals represent those of using Eq. (11) with the data set used for demonstration (see Fig. 11). As observed, the forcing term is a nonlinear function with complex shapes at the beginning.

3.2.2. Basis functions (9)

Based on trial and error, we use twenty basis functions to approximate the forcing terms observed in Fig. 12. In our example, a higher amount does not bring any further improvements, and a lower amount reduces the matching accuracy. The information about the basis functions (9) is visually presented in Fig. 13, to give an understanding of how these functions look like. In Fig. 13, the first and second plots are for the basis functions in respect of the variable s from (7), as well as time. The last plot is the solution (9), including the places indicating the locations of the centres for the basis functions, according to (10).

3.2.3. Solution of (13) for the forcing term

Up to this point, the target functions (11) and the basis functions (9) have been calculated. The following step is to calculate the weights (14) that minimize the locally weighted regression (13). The forcing terms (8) resulting from this calculations can be observed in the dashed signals from Fig. 12.

As it can be observed in Fig. 12, the matching of the forcing term is not perfect. The standard error for this regression are 2.02%, 2.83%, 3.99%, 1.06% for the forcing term of the axis $[x, y, z]$ and q_4 respectively. Increasing the number of basis functions does not increase accuracy. However, the intention in this approach is not necessarily to have perfect matching of the forcing term, because the final motion results from integrating the model (5). As long as the error comparing the final trajectory with the demonstrated data set is within reasonable boundaries (a difference of a few millimetres), then it is not necessary to have perfect matching of the forcing term.

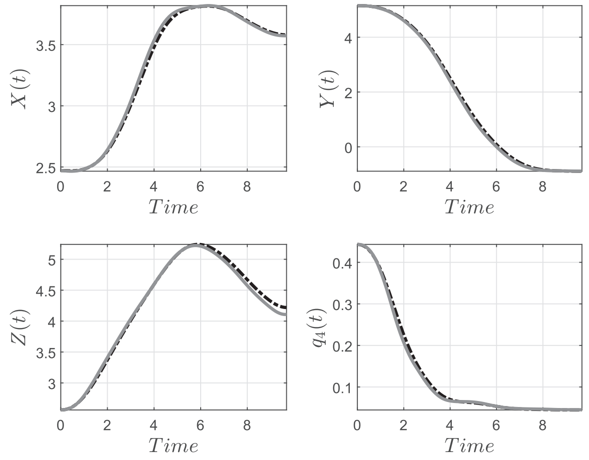

Fig. 14. Comparison of the demonstrated position trajectories against the results of the motion planning model (5). The grey signals are the data set used for training the model. The dashed black signals is the position trajectory using model (5).

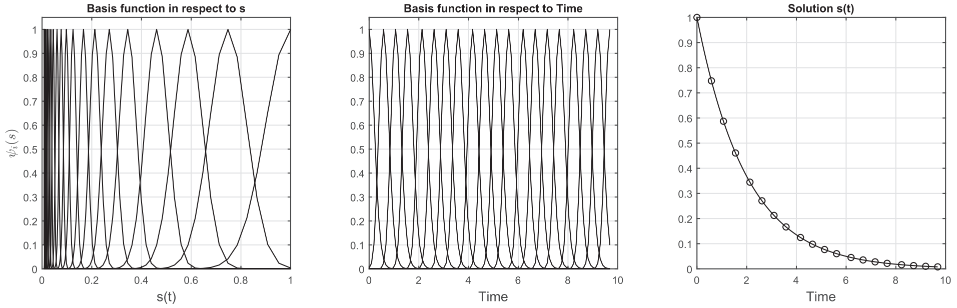

Fig. 13. Basis functions. The left plot refers to the basis function in respect to the variable $s(t)$. The middle plot refers to the basis function in respect of time. The right figure is the solution $s(t)$ in respect of time, and the points mark the centres for the basis functions.

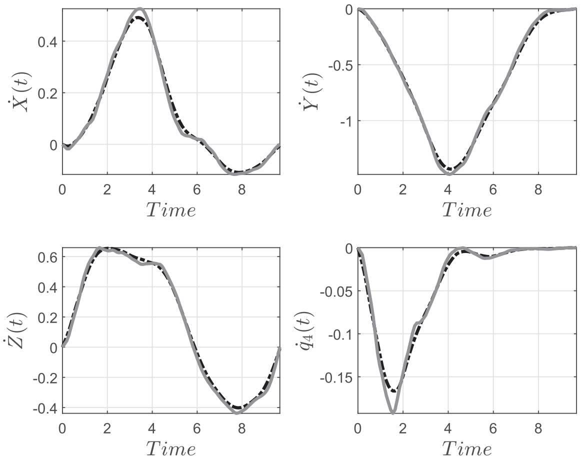

Fig. 15. Comparison of the demonstrated velocity trajectories against results of the motion planning model (5). The grey signals are the data set used for training the model. The dashed black signals is the position trajectory using model (5).

3.2.4. *Final motion planning resulting from model (5)*

Figs. 14 and 15 show a comparison of the original data set and the results of using model (5) for planning desired motions. Here, we compare the ability of the model to plan a motion that replicates the demonstration data. As observed, the matching is not perfect, but the difference is negligible for our purposes, considering that it is in the range of millimetres. The standard errors for this comparison are 2.18% , 6.4%, 5.6%, 0.67% for the position trajectories of the axis $[x, y, z]$ and q_4. This implies that in average we can replicate the desired path with an averaged error of 3.71%. Therefore, we consider that the resulting model is able to plan motions that resemble the demonstrated data with an accuracy of 96.3%. However, this accuracy is subjective, and can be changed by modifying the number of basis functions.

3.3. *Simulation tests for evaluating the final model (5)*

3.3.1. *Variation in initial conditions*

Referring to Fig. 6, the initial conditions for the first simulation test are planned according to the following characteristics: 1) around the vicinity of the original initial conditions, 2) at different offsets, and 3) in both sides of the machine. The results of this simulation are presented in Fig. 16. In addition, Fig. 17 presents the trajectories for each Cartesian coordinate, including the telescopic link q_4, demonstrating the ability of model (5) to plan trajectories despite these variations.

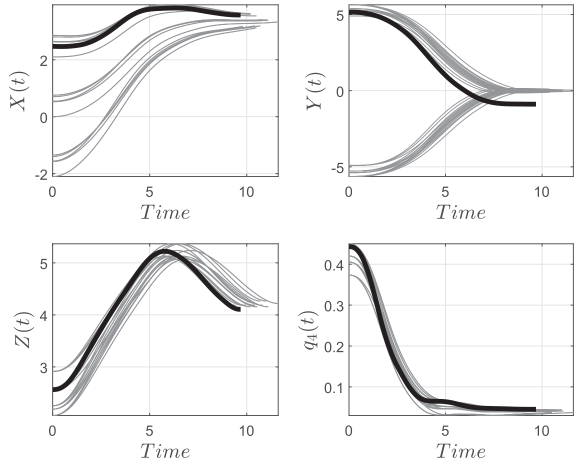

Fig. 17. The grey signals are the trajectories planned by the model (5). The solid bold lines are the trajectories used as data set for learning.

3.3.2. *Reaching logs that are located at further distances*

One of the main problems with reaching longer distances is that the telescopic link needs to open nearly to its maximum limit. This can be observed in Fig. 7 where some trajectories of the telescope start at an approximate opening of 1.5 meters. Nevertheless, it is also observed that the majority of motions happen at an approximate opening of 30% of the maximum, hence the averaged trajectory for q_4 observed in Fig. 7. In this simulation, the telescope will be used from 60% to 90% its maximum opening. The results of these simulations are presented in Fig. 18, where the original path is plotted with solid bold line. The individual Cartesian trajectories are presented in Fig. 19.

To understand the problem in this simulation, we refer to the degree of freedom q_4 in Fig. 19. As one can observe, the behaviour of q_4 changes in respect to what it is expected from Figs. 14 and 17, where we see that this link closes mainly at half the time of the whole motion, i.e. twice faster than the remaining links. Although it is not an indication of better performance, closing this link faster than the remaining degrees of freedom is something common from machine operators (La Hera and Morales, 2019). Therefore, the behaviour observed for q_4 in Fig. 19 can be perceived as odd for a trained operator. To modify this behaviour, an

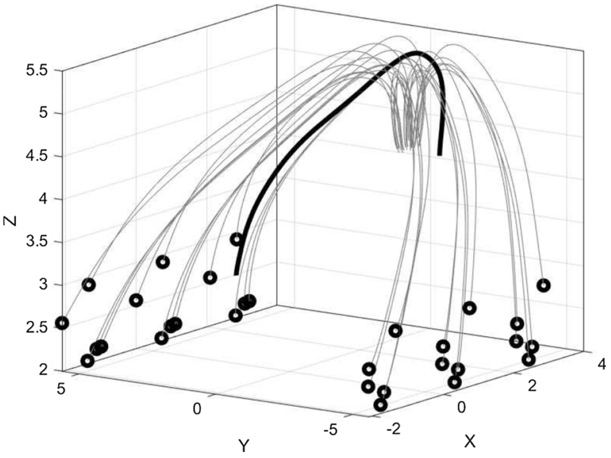

Fig. 16. This is a Cartesian coordinate plot for the simulated paths. The grey signals are the trajectories planned by the model (5). The dark dots represent all the new initial conditions used for the simulation. The solid bold line is the path used as data set for learning.

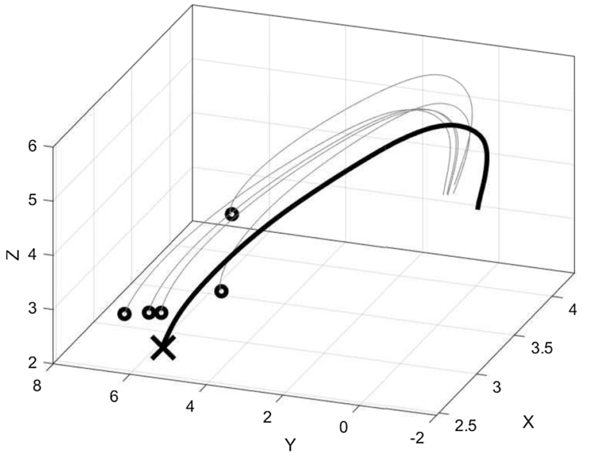

Fig. 18. This is a 3D Cartesian coordinate plot for the simulated paths. The grey signals are the trajectories planned by the model (5). The dark dots represent all the new initial conditions used for the simulation. The solid bold line is the path used as data set for learning.

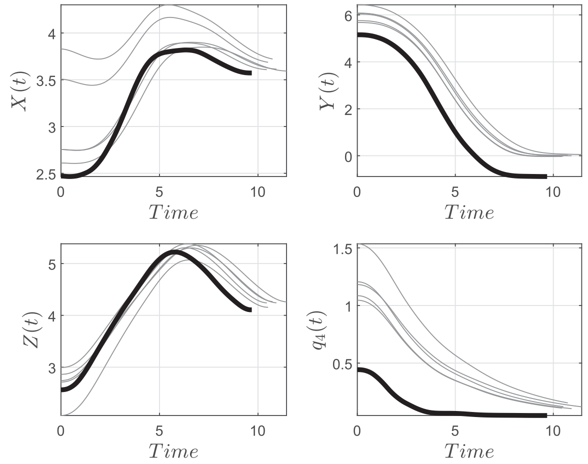

Fig. 19. The grey signals are the trajectories planned by the model (5). The solid bold lines are the trajectories used as data set for learning.

additional modification of the model (5) can be done by adding an additional attracting force. To this end, an attracting force towards $q_4 = 0.4$ meters is sufficient to force the system to speed up the motion. This can be done by adding the term

$$f_a = k(m - m_0)\left(\frac{1 - \text{sign}(m - m_0)}{2}\right) \tag{26}$$

to (5), resembling the behaviour of a spring, where f_a is the additional term, k represents the spring constant, and m_0 the attraction state. The operator involving sign is a mathematical form of an if-else command used to tell the system when to activate or deactivate this function. Results of simulating the model (5) with this new term are presented in Figs. 20 and 21. It can be observed from Fig. 21 that the new behaviour of q_4 goes in agreement to what we would be expecting from the training data set. This simulation test demonstrates the ability of the model (5) of accepting additional terms to modify the motion behaviour.

4. Discussion

Dynamic motion primitives is a machine learning method that has received a lot of attention the past years as a new form of dynamic motion planning approach. The main concept behind this method is to tune stable differential equations according to motion data sets recorded from demonstrated actions performed by people. The numerical solution to these differential equations represent the desired motions after the learning phase has taken place. Afterwards, modifying the quantities of the differential equations parameters has the ability to make these trajectories perform at different velocities, towards different desired goal positions, and from any desired initial position. Additional functions having spring-damper force characteristics can be added to account for external factors, such as obstacle avoidance. This versatility has made DMPs to be successfully integrated in different robotic systems throughout the world to program robot motions by demonstrations, allowing to readily change robot tasks without the necessity of complex re-programming. However, no studies have previously examined its application for heavy-duty hydraulic manipulators, as those used in forestry. Thus, this article stands as the first study case to exemplify its application within forestry for performing one particular action, i.e. bringing the logs into the machine automatically once an operator has grabbed them using joysticks. However, similar analysis to what is presented here can be followed as one possibility to perform the whole crane-cycle work with nearly full automation.

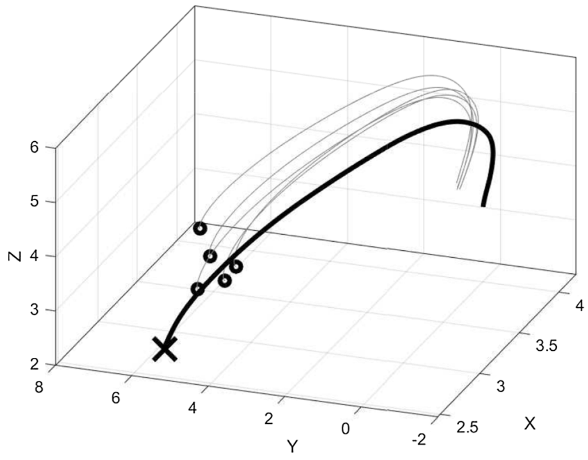

Fig. 20. This is a 3D Cartesian coordinate plot for the simulated paths. The grey signals are the trajectories planned by the model (5). The dark dots represent all the new initial conditions used for the simulation. The solid bold line is the path used as data set for learning. In this simulation, $k = 0.5$, and $m_0 = 0.4$, which are needed for the additional term (26).

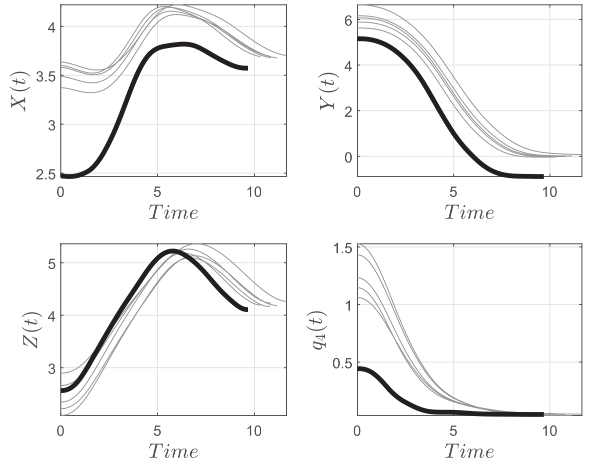

Fig. 21. The grey signals are the trajectories planned by the model (5). The solid bold lines are the trajectories used as data set for learning.

The philosophy behind the intended action of automatically retracing the crane back into the machine raises from the fact that the forest industry is recently introducing automation technology into forestry cranes. A common feature to forestry crane control is today the Cartesian control of the boom-tip, introduced as an automation method that is more intuitive to use and learn. Therefore, the action of grabbing logs can be done by the operator, not only because of the simplified joystick control method, but also because automating the action of grasping logs is more challenging from the engineering standpoint. Nevertheless, retracing the crane back to carry logs into the machine is more feasible as an incremental step into ICAS functions that can be provided to machine operators in the near future.

4.1. Discussion about results

To apply the DMPs framework, we have used a learning data set recorded from machine operators performing regular logging work with a forwarder machine. Our experience working with DMPs indicates that its application depends highly on how we treat the recorded data prior applying the DMPs framework. Therefore, we have paid special attention into specifying all the necessary data processing steps required to apply the DMPs framework when joint position measurements are available. These steps can be summarized as a sequence that estimates derivatives out of position measurements (i.e. velocity and acceleration), separates the data according to actions, remakes the initial and final conditions, and finally averages multiple trajectories. The performance verification of our final motion planning results demonstrates that this method can be instrumental in automating crane motions, because it can reproduce the demonstrated motion quite reliably, i.e. above 95% accuracy. Nevertheless, the performance highly depends on the ability of approximating the forcing term of the differential equations with reasonable accuracy, e.g. 97% in our case. As we are talking about cranes that have several meters of reaching capacity, a few millimetres difference between the planned and demonstrated motion is negligible. Therefore, these results are successful for our particular case.

This study has several strengths. First, we were able to postulate a generalized methodology to apply DMPs using joint position data recorded with expert machine operators. Second, our results demonstrate the ability of using the final set of differential equations to plan motions that are very different to the original data set, making its application more generalized, without the necessity of complex modifications. Third, we used the example of the telescopic link to demonstrate the influence of additional factors into the DMPs differential equations, such that we can influence the behaviour of the desired trajectory. Other complementary considerations can be similarly added through other mathematical functions to account for external factors.

Despite the benefits of our approach, some limitation to our study should be addressed. First, the objective of the article has been to cover the steps required to develop a motion planning system based on the DMPs framework. However, motion feedback control systems are necessary to successfully implement this in a real machine, which are not commonly available in forestry machines at large. Therefore, we emphasize that the implementation of our solution is directed towards the newest forestry machine models from Scandinavian manufacturers, which are coming out to the market with new automation features. In addition, this study assumes that the only obstacle in the workspace is the machine's log-bunk. Therefore, the motion planner can provide an obstacle-free path to bring logs into the log-bunk. However, the motion planner is blind, because computer vision systems are not available in forestry machines. Therefore, the use of our approach is directed towards similar logging methods to the cut-to-length system from Scandinavia, which leads to obstacle-free work for forwarder machines. Second, the only modification over to the training data set was to modify the initial conditions. However, understanding that machine operators control cranes by following very repetitive motions opens the possibility to apply optimization for finding a better set of joint trajectories able to optimize different performance criteria. Examples of this optimizations are to minimize energy during motion, to work at the fastest speed, to maximize the loading capacity of the grapple, etc. Therefore, having as basis the recorded data showing the paths machine operators perform, optimization can be instrumental to modify joint trajectories, such that the resulting motion planner is able to plan optimal trajectories, which include the experience of expert operators moving the crane over defined paths. Nevertheless, such an analysis falls outside the scope of the present study, but it can be important to recall that similar concepts have formerly taken place in the work of (Morales et al., 2014; Dong et al., 2020).

4.2. General discussion

Automation dedicated to forestry machines is slowly starting to take place as a new standard to increase both the efficiency of machines, as well as the work productivity. The newest forestry machine models from Scandinavian manufacturers, which also include the North American company John Deere, are slowly starting to provide products featuring new automation technology. Most of the features presented so far are dedicated towards improvements in motion control systems to facilitate the work with cranes. Nevertheless, research on advanced automation features for forestry cranes exists in literature as far as two decades ago. Therefore, it is possible to anticipate that the incremental step to appear in forestry machines is the semi-automatic control of some crane motions.

Although being a relatively simple example, our study has targeted one possible scenario of automation that can take place quite soon, i.e. a click of a button to automatically retrace the crane to the machine. However, the dynamic motion primitives framework has many inherited properties that we have not addressed, to focus the reader in the design methodology. Some of the added benefits and also possibilities to improve our results are the following:

- **Re-training the model**. DMPs can be factory trained according to data from expert machine operators. Nevertheless, retraining the DMPs models is readily possible on-site, and can be provided as a feature to machine operators. This can be directly applied by machine operators to reprogram the motion planner and accomplish other tasks. For instance, the task of unloading the machine, which is the reversed process of our present study.
- **Partial automation**. The present study focuses on retracing the crane to the machine, in order to bring logs into the log-bunk automatically. This motion has the ability to reduce nearly 30% of the machine operator's tasks. Nevertheless, it is similarly feasible to provide an expanding motion for the crane to approach logs, reducing the work of the operator even further to nearly 60%. This leaves the operator with the simplified tasks of grasping and releasing logs.
- **Grabbing logs**. As presented in the work of (Li et al., 2017), the DMPs framework can also be used for teaching the crane how to grab logs. This is in concept a more challenging problem, because it requires the presence of computer vision software and hardware, as well as other sensors that are currently not present in forestry crane grapples. Nevertheless, research aiming towards these solutions have been taking place. In addition, industrial products to support such automation are already appearing on the market (Harr et al., 2020).
- **Full automation**. It is feasible to assume that a combination of all the possibilities described earlier can lead to automate the work with forwarder machines. As pointed out in (Lindroos et al., 2019) this is a goal for Scandinavian forestry, visioning the manufacturing of unmanned machines that are lighter and with reduced carbon footprint. Therefore, the efforts to automate the navigation of the vehicle is ongoing research (Ringdahl et al., 2011).

In conclusion, this article has provided an initial example of how DMPs could be applied in the forest industry to develop a new generation of motion planners for automating crane work. Results show that these motion planners can reliably reproduce the demonstrated motion after carefully preparing the training data set. Consequently, the motion planner is able to plan desired motions from different initial conditions, all resembling similar behaviour to the training data set. Additional functions can be introduced to alter the behaviour of the motion planner. As there exists different adjustments that can be done to the original data set by exploiting kinematic redundancy through optimization, our results should be considered as an initial study case. A combination of optimization and DMPs can be instrumental to plan motions that both reflect the experience of machine operators, while simultaneously considering the optimization of performance criteria factors such as, energy, time, etc.

Funding sources

This study was partly financed by the Swedish Foundation for Strategic Environmental Research MISTRA (program Mistra Digital Forest), the Kempe Foundations (project JCK-1713) and the Swedish Cluster of Forest Technology.

CRediT authorship contribution statement

Pedro La Hera: Conceptualization, Methodology, Formal analysis, Data curation, Software, Writing - original draft, Supervision, Visualization, Funding acquisition. **Daniel Ortiz Morales:** Conceptualization, Software, Investigation, Resources, Writing - review & editing. **Omar Mendoza-Trejo:** Resources, Visualization, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors would like to acknowledge the support of the company "The Swedish Cluster of Forest Technology" (In Swedish: skogstekniska klustret) for their help getting the machine to record the data used in this article. Similarly, we would like to acknowledge the technical help of the company Komatsu Forest for placing the sensors in the forwarder machine used in the study. Acknowledgements to Prof. Ola Lindroos for his support acquiring the financial support to carry on with this study. The authors gratefully acknowledge the financial support of the Kempe Foundation.

References

Broomhead, D.S., Lowe, D., 1988. Multivariable functional interpolation and adaptive networks, complex systems, 2.

CINTOC, 2020. Cintoc. http://cintoc.se/, Accessed on 2020.

Cranab, 2011. Cranab presents an entirely new generation of cranes. http://www.cranab.se/static/en/206/, Accessed on 2011.

Ding, Ruqi, Zhang, Junhui, Bing, Xu., Cheng, Min, 2018. Programmable hydraulic control technique in construction machinery: Status, challenges and countermeasures. Automat. Construct. 95, 172–192.

Dong, Xiaowei, Mendoza-Trejo, Omar, Morales, Daniel Ortiz, Lindroos, Ola, Hera, Pedro La, 2020. Simulation-based comparison between two crane-bunk systems for loading work when considering energy-optimal motion planning. Int. J. For. Eng. 31 (1), 70–77.

EATON, 2019. https://www.eaton.com/us/en-us/catalog/valves/cma-advanced-mobile-valves.resources.html, Online since 2019.

Eriksson, Mattias, Lindroos, Ola, 2014. Productivity of harvesters and forwarders in ctl operations in northern sweden based on large follow-up datasets. Int. J. For. Eng. 25 (3), 179–200.

Fattorini, Hector O., Fattorini, Hector O., et al., 1999. Infinite Dimensional Optimization and Control Theory, vol. 54. Cambridge University Press.

Fodor Szabolcs, 2017. Towards semi-automation of forestry cranes: automated trajectory planning and active vibration damping. PhD thesis, Umeå universitet.

Gingras Jean-Francois, Charette Francis, 2017. Fp innovations forestry 4.0 initiative. In: Bangor: 2017 Council on Forest Engineering Annual Meeting.

Häggström, Carola, Lindroos, Ola, 2016. Human, technology, organization and environment–a human factors perspective on performance in forest harvesting. Int. J. For. Eng. 27 (2), 67–78.

Hansson, Anders, Servin, Martin, 2010. Semi-autonomous shared control of large-scale manipulator arms. Control Eng. Practice 18 (9), 1069–1076.

Harr Joakim, Lindgren Magnus, Vännman Niklas, 2020. Rotator arrangement with an angle meter, January 2 2020. US Patent App. 16/486,199.

Ijspeert, Auke Jan, Nakanishi, Jun, Hoffmann, Heiko, Pastor, Peter, Schaal, Stefan, 2013. Dynamical movement primitives: learning attractor models for motor behaviors. Neural Comput. 25 (2), 328–373.

Komatsu Forest AB, 2011. Forwarder 830.3. http://www.komatsuforest.se/default.aspx?id=10876, Online since 2011.

Labelle, Eric R., Lemmer, Kevin J., 2019. Selected environmental impacts of forest harvesting operations with varying degree of mechanization. Croatian J. For. Eng.: J. Theory Appl. For. Eng. 40 (2), 239–257.

La Hera, Pedro, Morales, Daniel Ortíz, 2015. Model-based development of control systems for forestry cranes. J. Control Sci. Eng. 2015.

La Hera, Pedro, Morales, Daniel Ortiz, 2019. What do we observe when we equip a forestry crane with motion sensors? Croatian J. For. Eng.: J. Theory Appl. For. Eng. 40 (2), 259–280.

LaValle, Steven M., 2006. Planning Algorithms. Cambridge University Press.

Li, Zhijun, Zhao, Ting, Chen, Fei, Hu, Yingbai, Su, Chun-Yi, Fukuda, Toshio, 2017. Reinforcement learning of manipulation and grasping using dynamical movement primitives for a humanoidlike mobile manipulator. IEEE/ASME Trans. Mechatron. 23 (1), 121–131.

Lindroos Ola, Mendoza-Trejo Omar, La Hera Pedro, Ortiz Morales Daniel, 2019. Advances in using robots in forestry operations. In: University of Southern Queensland, Australia and John Billingsley, editors, Burleigh Dodds Series in Agricultural Science, Burleigh Dodds Science Publishing, pp. 233–260. ISBN 9781786762726. doi:10.19103/AS.2019.0056.18. https://shop.bdspublishing.com/store/bds/detail/product/3-190-9781838797782.

Manner, Jussi, Mörk, Anders, Englund, Martin, 2019. Comparing forwarder boom-control systems based on an automatically recorded follow-up dataset. Silva Fenn 53, 10161.

Manring, Noah D., Fales, Roger C., 2019. Hydraulic Control Systems. John Wiley & Sons.

Mboup, Mamadou, Join, Cédric, Fliess, Michel, 2009. Numerical differentiation with annihilators in noisy environment. Numerical Algorith. 50 (4), 439–467.

Morales Daniel Ortiz, 2015. Virtual Holonomic Constraints: from academic to industrial applications. PhD thesis, Umeå Universitet.

Morales, Daniel Ortíz, La Hera, Pedro, Westerberg, Simon, Freidovich, Leonid B., Shiriaev, Anton S, 2014. Path-constrained motion analysis: An algorithm to understand human performance on hydraulic manipulators. IEEE Trans. Human-Machine Syst. 45 (2), 187–199.

Morales, Daniel Ortiz, Westerberg, Simon, La Hera, Pedro X., Mettin, Uwe, Freidovich, Leonid, Shiriaev, Anton S., 2014. Increasing the level of automation in the forestry logging process with crane trajectory planning and control. J. Field Robot. 31 (3), 343–363.

Münzer Marc E., 2004. Resolved motion control of mobile hydraulic cranes. Aalborg University, Institute of Energy Technology.

Nurminen Tuomo, Korpunen Heikki, Uusitalo Jori, 2006. Time consumption analysis of the mechanized cut-to-length harvesting system.

Pedersen Niels Henrik, Johansen Per, Andersen Torben Ole, 2018. Challenges with respect to control of digital displacement hydraulic units.

Ravichandar Harish, Polydoros Athanasios S., Chernova Sonia, Billard Aude, 2020. Recent advances in robot learning from demonstration. Ann. Rev. Control Robot. Auton. Syst. 3.

Reitz Jan, Schluse Michael, Roßmann Jürgen, 2019. Industry 4.0 beyond the factory: An application to forestry. In: Tagungsband des 4. Kongresses Montage Handhabung Industrieroboter. Springer, pp. 107–116.

Ringdahl, Ola, Lindroos, Ola, Hellström, Thomas, Bergström, Dan, Athanassiadis, Dimitris, Nordfjell, Tomas, 2011. Path tracking in forest terrain by an autonomous forwarder. Scand. J. For. Res. 26 (4), 350–359.

Spong, M.W., Hutchinson, S., Vidyasagar, M., 2006. Robot Modeling and Control. John Wiley and Sons, New Jersey.

Stulp, Freek, Theodorou, Evangelos A., Schaal, Stefan, 2012. Reinforcement learning with sequences of motion primitives for robust manipulation. IEEE Tran. Robot. 28 (6), 1360–1370.

Technion, 2017. Crane control systems. https://technion.fi/crane-control-systems/, Online since 2017.

Westerberg Simon, 2014. Semi-automating forestry machines: motion planning, system integration, and human-machine interaction. PhD thesis, Umeå Universitet.

Westerberg, Simon, Shiriaev, Anton, 2013. Virtual environment-based teleoperation of forestry machines: Designing future interaction methods. J. Human-Robot Interact. 2 (3), 84–110.