Global energy-optimised redundancy resolution in hydraulic manipulators using dynamic programming

Jarmo Nurmi*, Jouni Mattila

Tampere University of Technology, Department of Intelligent Hydraulics and Automation, P.O. Box 589 (Korkeakoulunkatu 6), FIN-33101 Tampere, Finland

Abstract

This paper addresses the problem of redundancy resolution in closed-loop controlled hydraulic manipulators. The problem is treated at the hydraulic level using proposed cost functions formulated into a dynamic programming approach of minimum-state representation. Bounds on joint range, actuator velocity and acceleration were enforced. This approach minimises the hydraulic energy consumption of the widely popular load-sensing and constant-supply pressure systems. The presented approach can resolve the redundancy more effectively from the hydraulic side than do actuator velocity or energy optimisation approaches, point-wise optimal approaches or some standard direct optimisation tools that may lead to inferior solutions, as shown in simulation results where up to 15–30% greater energy use is seen with some competing approaches. The results obtained motivate joint trajectory optimisation at the hydraulic level in prospective applications at construction sites where frequently driven work cycles of hydraulic construction cranes are automated.

Keywords:
redundancy resolution, hydraulic manipulator, construction crane, energy optimisation, global optimisation, dynamic programming, joint limits, load-sensing system, constant-pressure system

1. Introduction

Hydraulic manipulators are widely used for excavation and lifting applications at construction sites and for heavy-duty material handling in the forest industry due to their superior power-density and rugged nature. Although the hydraulic construction cranes are mainly open-loop controlled by human operators, manufacturers in these industries are interested in broadening their offerings through the automation of typical work cycles to improve the productivity and safety of their machines. On a technical level, this automation requires solving the inverse kinematics problem and realizing closed-loop control (see [1] for the proposed closed-loop control algorithm). Because construction cranes are typically equipped with redundant joints, the inverse kinematics problem transforms into a more difficult redundancy resolution problem, thus lending itself to sophisticated machine operation optimisation. Here, we resolve the redundancy of the construction crane from the standpoint of hydraulic energy minimisation. This approach of redundancy resolution entails moving the crane cylinder actuators in an energy-efficient fashion that is also subject to task-space reference.

Only a handful of articles discuss the redundancy resolution of hydraulic manipulators, including [2], in which point-wise optimal joint trajectories are given that minimise the energy consumed by hydraulic actuators. This point-wise solution is sub-optimal over the entire trajectory, and the problem is not fully considered at the hydraulic system level. In [3], some productivity problems in hydraulic knuckle booms are solved locally using redundancy to maximise the lifting capacity or velocity. Dynamic programming is also used to globally minimise the time required to move between two points in the workspace. However, no energy-related objectives were discussed. In [4], the work-

*Corresponding author
Email addresses: jarmo.nurmi@tut.fi (Jarmo Nurmi), jouni.mattila@tut.fi (Jouni Mattila)

ing cycle duration of non-redundant excavators is reduced locally by maximising its joint velocities. Although the article was written from a hydraulics standpoint, energy optimisation was disregarded.

In contrast, much work has been dedicated to resolving the redundancy of general manipulators (e.g. [5], [6]). Many of these papers discuss resolved redundancy pertaining to the minimisation of actuator energy consumption, which can lead to significant energy savings in manipulators in general. However, this solution is inevitably sub-optimal when dealing with many hydraulic system types. This generally arises from the pressure losses encountered in most hydraulic systems when the actuators are subject to unequal loads. Therefore, the energy optimisation of hydraulic manipulators that are mainly powered by load-sensing or constant-pressure systems calls for effective control approaches specifically tailored for these hydraulic systems and cost functions formulated at the hydraulic level, instead of the actuator level. Furthermore, contemporary articles on redundancy resolution mostly exemplify highly redundant manipulators that do not represent typical hydraulic manipulators, which have less kinematic redundancy. Therefore, the energy savings presented do not equal the savings typically achieved with hydraulic manipulators. Although the problem is simpler than most in terms of redundancy, the nonlinearities and non-convexity make the problem difficult to solve at the hydraulic level. For example, conventional direct optimisation methods yield local optimums, and the search for a global optimum among the local optimums is seen as time consuming.

In this paper, we effectively explore the redundancy resolution problem using popular hydraulic systems powered by constant-supply pressure or load-sensing variable displacement pumps as opposed to ineffective sub-optimal approaches. We focus on a common 3-degree-of-freedom (DOF) hydraulic manipulator design, which is redundant in one DOF in the typical manner, and propose cost functions at the hydraulic level to globally to minimise the manipulator's hydraulic energy consumption over prescribed workspace movements. To effectively resolve the redundancy, the proposed cost functions are formulated into a minimum-state dynamic programming approach, which ensures accurate tracking of a Cartesian path while minimizing the said cost functions. Bounds on joint ranges based on cylinder stroke, cylinder velocities and cylinder acceleration are enforced. We investigate popular load-sensing systems and analyse pump flow rate minimisation, which equally minimises the energy consumption of a constant-supply pressure hydraulic system. We compare our results to well-known sub-optimal control strategies. To the authors' knowledge, this is the first time joint trajectories have been globally optimised at the hydraulic level in prescribed Cartesian motions in relation to typical redundant hydraulic manipulators.

This paper is organised as follows. In Section 2, we introduce a typical hydraulic manipulator with a redundant degree-of-freedom and define its end-effector position and velocity. We also discuss the use of variable displacement pumps in the conventional constant-supply pressure and load-sensing systems. In Section 3, we define the optimal control problems in the continuous and discrete form, and we introduce the dynamic programming approach in Section 4. In Section 5, we propose the cost functions at the hydraulic level. In Section 6, we provide numerical simulations to compare and estimate the energy conservation attainable with a typical manipulator. In Section 7, we discuss important aspects of the optimal control problem, and we provide conclusions in Section 8.

2. Hydraulic manipulator with kinematic redundancy

Let us consider the planar 3-DOF hydraulic manipulator shown in Fig. 1, which represents the typical hydraulic manipulator configuration used in a number of applications for tasks involving heavy lifting at construction sites. The manipulator has a prismatic reach actuator that provides an additional DOF. Because of this redundancy property, the manipulator's end-effector tip can be controlled in an infinite number of joint trajectories, from an initial Cartesian point to the desired end point. This desirable redundancy opens up the possibility of finding joint trajectories that globally optimise the energy consumption of the manipulator at the hydraulic level while the end-effector satisfies Cartesian reference path constraints. To this end, the end-effector position and velocity are defined, and we discuss the variable displacement pumps heavily utilised in the manipulator's hydraulic systems.

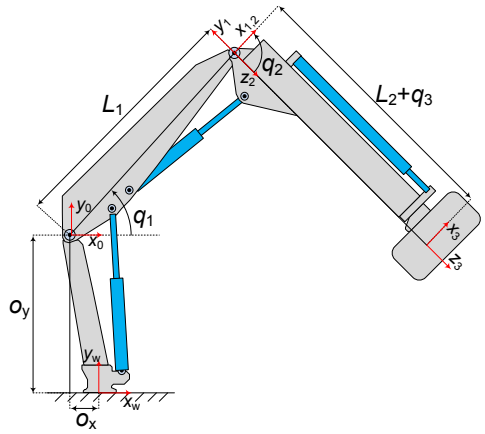


Figure 1: Typical 3-DOF kinematically redundant hydraulic manipulator

2.1. End-effector position and velocity

The joint space vector of the manipulator is written as

$$\mathbf{q} = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix}^{\mathrm{T}} \tag{1}$$

where joint coordinate q_1 denotes the lift angle, joint coordinate q_2 denotes the tilt angle (transfer angle) and the redundant joint coordinate q_3 denotes the extension length of the cylinder (reach). The joint coordinates q_1 (real positive), q_2 (real negative) and q_3 (real positive) are chosen based on the classical Denavit-Hartenberg (DH) convention [7]. Coordinate frames are attached to the links and numbered based on this DH convention. The world coordinate frame in the base is denoted with w (see Fig. 1).

Table 1: Denavit-Hartenberg parameters of the manipulator

Joint i	a_i	α_i	d_i	θ_i
1	L_1	0	0	q_1
2	0	$\pi/2$	0	$\pi/2 + q_2$
3	0	0	$L_2 + q_3$	0

The DH homogeneous transformation matrix $\mathbf{A}_i^{i-1} \in \mathbb{R}^{4x4}$, which determines the coordinate transformation from the link attached frame i to frame $i-1$, is

$$\mathbf{A}_i^{i-1} = \begin{bmatrix} \mathrm{c}_{\theta_i} & -\mathrm{s}_{\theta_i}\mathrm{c}_{\alpha_i} & \mathrm{s}_{\theta_i}\mathrm{s}_{\alpha_i} & a_i\mathrm{c}_{\theta_i} \\ \mathrm{s}_{\theta_i} & \mathrm{c}_{\theta_i}\mathrm{c}_{\alpha_i} & -\mathrm{c}_{\theta_i}\mathrm{s}_{\alpha_i} & a_i\mathrm{s}_{\theta_i} \\ 0 & \mathrm{s}_{\alpha_i} & \mathrm{c}_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

where e.g. s_{θ_i} denotes $\sin(\theta_i)$, c_{θ_i} denotes $\cos(\theta_i)$ and the matrix elements are obtained using the DH parameters (see Table 1). Using the DH transformation matrix in succession, we get

$$\mathbf{A}_3^0 = \mathbf{A}_1^0\mathbf{A}_2^1\mathbf{A}_3^2 \tag{3}$$

where $\mathbf{A}_3^0$ is the total coordinate transformation from the end-effector frame 3 to frame 0. To transform to world frame w, which is located in the manipulator base, we use a homogeneous transformation matrix on $\mathbf{A}_3^0$

$$\mathbf{T}_3^{\mathrm{w}} = \begin{bmatrix} \mathbf{R}_0^{\mathrm{w}} & \mathbf{o}_0^{\mathrm{w}} \\ \mathbf{0} & 1 \end{bmatrix} \mathbf{A}_3^0 \tag{4}$$

where the rotation matrix $\mathbf{R}_0^{\mathrm{w}}$ is the identity matrix $\mathbf{I}_{3x3}$ because the orientations of coordinate frames 0 and w are aligned, the translation vector $\mathbf{o}_0^{\mathrm{w}}$ is $\begin{bmatrix} o_{\mathrm{x}} & o_{\mathrm{y}} & 0 \end{bmatrix}^{\mathrm{T}}$, with its components denoting the offsets between the frames w and 0, and $\mathbf{T}_3^{\mathrm{w}}$ yields the transformation from frame 3 to w. Finally, we may determine the end-effector position in the world coordinate frame from $\mathbf{T}_3^{\mathrm{w}}$ to be

$$x_{\mathrm{w}} = L_1\cos(q_1) + (L_2 + q_3)\cos(q_1 + q_2) + o_{\mathrm{x}} \tag{5}$$

$$y_{\mathrm{w}} = L_1\sin(q_1) + (L_2 + q_3)\sin(q_1 + q_2) + o_{\mathrm{y}} \tag{6}$$

where x_{w} denotes the position of the end-effector in the direction of the x-axis in the world coordinate frame, y_{w} denotes the position of the end-effector in the direction of the y-axis in the world coordinate frame and because the manipulator operates in the xy-plane, the position of the end-effector in the z_{w} coordinate is zero. The manipulator link lengths are L_1 and L_2, and o_{x} and o_{y} are the offset of the world coordinate system in the direction of the x- and y-axes, respectively, from the first joint. The joints are actuated by hydraulic cylinders. The joints q_1 and q_2 are rotational joints, which rotate around the z-axis. The extension joint

q_3 is a prismatic joint whose length is determined by the length of the hydraulic cylinder. Because of the physical limits of the hydraulic cylinders that actuate the manipulator joints, the joint coordinates are lower and upper bounded. The maximum joint velocities in both directions of motion are also bounded as dictated by the maximum displacement of the hydraulic pump and cylinder size. System restrictions also limit maximum joint accelerations.

The task-space vector denoting the end-effector Cartesian position may be written as

$$\mathbf{x}_\mathrm{t} = \begin{bmatrix} x_\mathrm{w} & y_\mathrm{w} \end{bmatrix}^\mathrm{T} \tag{7}$$

Differentiating $\mathbf{x}_\mathrm{t}$ in view of Eqs. (5)–(7) with respect to time yields

$$\dot{\mathbf{x}}_\mathrm{t} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \tag{8}$$

where the Jacobian matrix $\mathbf{J}(\mathbf{q}) \in \mathbb{R}^{2x3}$ is the partial derivative of $\mathbf{x}_\mathrm{t}$ with respect to $\mathbf{q}$ and $\dot{\mathbf{q}}$ is the time derivative of $\mathbf{q}$.

To solve the joint velocities $\dot{\mathbf{q}}$ from Eq. (8) we need to invert the non-square matrix $\mathbf{J}(\mathbf{q})$. A particular inversion is obtained using the weighted right-side pseudo-inverse [6]

$$\dot{\mathbf{q}}^\dagger = \mathbf{J}(\mathbf{q})^\dagger \dot{\mathbf{x}}_\mathrm{t} = \mathbf{W}^{-1}\mathbf{J}(\mathbf{q})^\mathrm{T}\left(\mathbf{J}(\mathbf{q})\mathbf{W}^{-1}\mathbf{J}(\mathbf{q})^\mathrm{T}\right)^{-1}\dot{\mathbf{x}}_\mathrm{t} \tag{9}$$

where $\mathbf{W}$ is a weighting matrix and $\mathbf{J}(\mathbf{q})^\dagger$ denotes the pseudo-inverse of the Jacobian matrix. This pseudo-inverse minimises the instantaneous Euclidean norm of joint velocities $\dot{\mathbf{q}}^\mathrm{T}\dot{\mathbf{q}}$ when the weighting matrix $\mathbf{W}$ equals the identity matrix $\mathbf{I}$, thus yielding a least-square solution. Variations of this weighting approach that are more sophisticated have been presented (e.g. weighting with the manipulator inertia matrix minimises the instantaneous kinetic energy). The pseudo-inverses have also been derived in actuator coordinates [8]. Because these pseudo-inverse approaches lead to the instantaneous minimisation of the performance criteria, the minimum over the whole task-space trajectory must be found using an optimal control methodology.

2.2. Control of hydraulic systems by variable displacement pumps

Variable displacement pumps are hydraulic components capable of outputting a variable flow rate through the hydraulic or electric alteration of the pump displacement. The pumps predominantly used in hydraulic manipulators are axial-piston types in which the output flow is controlled by adjusting the angle of the swashplate, which is a tilted disc usually actuated by pressure-regulated hydraulic control pistons acting against a spring load. The dynamic characteristics of common hydraulic piston pumps and their control principles are widely known and have been thoroughly studied in the literature (see [9–11]).

These variable displacement pumps are used in constant-supply pressure (CP) and load-sensing systems (LS) in hydraulic manipulators. The pressure and flow are usually regulated through hydromechanical feedback realised with a regulator valve subject to hydraulic pressure and spring forces, which control the valve's position. In the CP system, the regulator valve controls the pump displacement in such a way as to maintain a constant supply pressure, regardless of changes in the flow demand or load pressure. The pump flow is simultaneously matched to the required flow of the actuators. This desired control action is accomplished by fixing the spring tension force acting on the spool of the regulator valve to a force corresponding to the desired supply pressure. The hydraulic circuit of the CP system is shown in Fig. 2. CP systems have been widely adopted in hydraulic manipulators used in feedback control applications because of their enhanced stability and simplicity over other systems; however, these benefits come at the cost of energy efficiency.

The more complex LS system shown in Fig. 3 controls the pump displacement through the regulator valve so the supply pressure at the pump outlet tracks a time-varying reference, which is continuously adjusted to a fixed-pressure delta above the highest load pressure sensed from the hydraulic pilot line. Like in the CP system, the pump flow is matched to the actuator requirement. Abiding by this principle, the pressure losses over the valves controlling the actuators may be notably reduced compared to the CP system, but the potential disadvantages are poor energy efficiency with unequal loading and the increased chance of stability problems because of the decreased damping [12]. Variations of the LS systems include the electrical LS systems [13], which eliminate the long LS pilot line; and flow control systems [14], which remove the pressure feedback on the load at the pump. LS systems are immensely popular in open-loop-controlled manipulators in which energy must be distributed to multiple actuators with a single

pumping unit; however, they can also be encouragingly stable in feedback control applications [15].

The vast popularity of these systems in applications and their different operating principles imply the need for a tailored solution to effectively resolve kinematic redundancy at the hydraulic system level. The controllability of the hydraulic pump's flow rate and supply pressure enables this energy saving redundancy resolution.

3. Problem formulation

Let us formulate a dual-objective problem in which the secondary objective is to minimise a performance cost function L_p related to the energy consumption of the hydraulic system (cost functions presented later) while the manipulator end-effector is primarily required to track a time-dependent planar path $\mathbf{r}(t)$ denoted with

$$\mathbf{r}(t) = \begin{bmatrix} r_\mathrm{x}(t) & r_\mathrm{y}(t) \end{bmatrix}^\mathrm{T} \tag{10}$$

where t denotes time and the differentiable Cartesian x and y-coordinate references are $r_\mathrm{x}(t)$ and $r_\mathrm{y}(t)$, respectively.

3.1. Continuous-time formulation

The complex optimal control problem with fixed terminal time is defined as follows:

$$\min_{\mathbf{u}\in\boldsymbol{\rho}} \int_0^{t_\mathrm{f}} L_\mathrm{p}\big(\mathbf{x}(t),\mathbf{u}(t)\big)\,\mathrm{d}t \tag{11}$$

subject to:

$$\begin{aligned} &\mathbf{x}(0) = \mathbf{x}_0, \\ &\dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t),\mathbf{u}(t)\big), \\ &\mathbf{g}_\mathrm{e}\big(\mathbf{x}(t),\mathbf{u}(t)\big) = 0, \\ &\mathbf{g}_\mathrm{i}\big(\mathbf{x}(t),\mathbf{u}(t)\big) \le 0 \end{aligned} \tag{12}$$

where L_p is the performance cost to be minimised (real), t_f is the terminal time, $\mathbf{x}_0 \in \mathbb{R}^6$ is the initial system state vector, $\mathbf{x} \in \mathbb{R}^6$ denotes the system state vector, $\mathbf{u} \in \mathbb{R}^3$ denotes the control vector, $\mathbf{f} \in \mathbb{R}^6$ denotes the system dynamics, $\mathbf{g}_\mathrm{i} \in \mathbb{R}^{18}$ denotes the inequality constraints, $\mathbf{g}_e \in \mathbb{R}^2$ denotes the equality constraints and $\boldsymbol{\rho}$ is the control policy search space of feasible $\mathbf{u}(t)$, defined at time indices from 0 to t_f and constrained by joint acceleration limits at each time index (see Eq. (15)).

The system dynamics $\mathbf{f}\big(\mathbf{x}(t),\mathbf{u}(t)\big)$ are defined using

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= u_1 \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= u_2 \\ \dot{x}_5 &= x_6 \\ \dot{x}_6 &= u_3 \end{aligned} \tag{13}$$

where state vector $\mathbf{x} = [q_1\ \dot{q}_1\ q_2\ \dot{q}_2\ q_3\ \dot{q}_3]^\mathrm{T}$ and control vector $\mathbf{u} = [\ddot{q}_1\ \ddot{q}_2\ \ddot{q}_3]^\mathrm{T}$. The time indices were omitted for brevity.

To ensure satisfactory path tracking in the task space, we require that $\mathbf{g}_\mathrm{e}$ contains the time-varying equality constraint

$$\mathbf{x}_\mathrm{t}(t) - \mathbf{r}(t) = 0 \tag{14}$$

However, tracking the path could be equivalently forced with a velocity equality constraint $\dot{\mathbf{x}}_\mathrm{t}(t) - \dot{\mathbf{r}}(t) = 0$, positional inequality constraint $|\mathbf{x}_\mathrm{t}(t) - \mathbf{r}(t)| \le \delta$ with a small constant $\delta \in \mathbb{R}_+$, or, similarly, a velocity inequality constraint. Notice that when using any one of these constraints, we do not need to find weights for the performance and tracking objectives.

The following state and control constrains are contained in $\mathbf{g}_\mathrm{i}$

$$\begin{aligned} x_{i_\mathrm{min}} \le x_i \le x_{i_\mathrm{max}} \\ u_{i_\mathrm{min}} \le u_i \le u_{i_\mathrm{max}} \end{aligned} \tag{15}$$

where x_{i_min} denotes the minimum feasible value of state $i \in \{1,2,3,4,5,6\}$ (joint position or joint velocity limit), x_{i_max} denotes the maximum feasible value of state $i \in \{1,2,3,4,5,6\}$ (joint position or joint velocity limit), u_{i_min} is the minimum joint acceleration of joint $i \in \{1,2,3\}$ and u_{i_max} is the maximum joint acceleration of joint $i \in \{1,2,3\}$. These inequality constraints are based on physical bounds on joint ranges, joint velocity and joint acceleration. The constraints can be readily reduced into the general form denoted with $\mathbf{g}_\mathrm{i}$ in Eq. (12).

3.2. Discrete-time formulation

The discrete-time problem may be formulated as follows. Firstly, let the continuous time from 0 to t_f

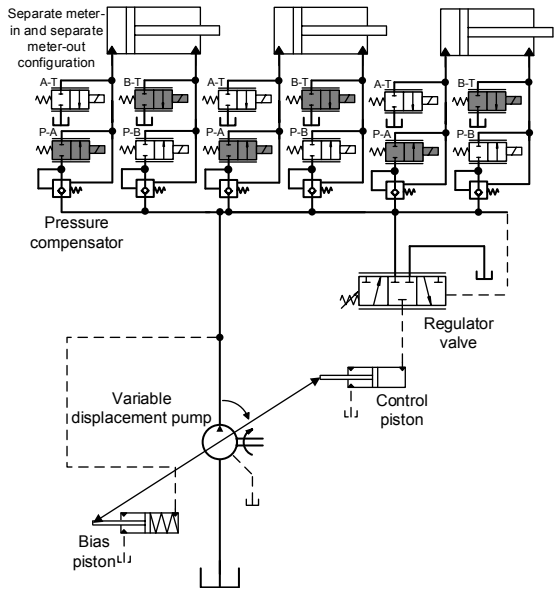


Figure 2: Hydraulic circuit of a constant pressure system

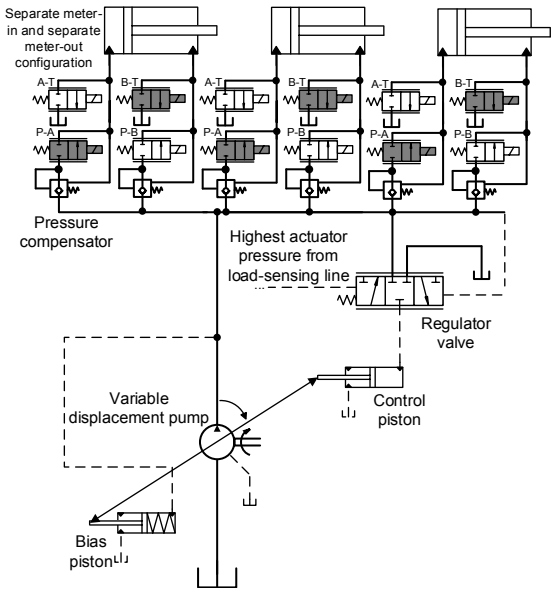


Figure 3: Hydraulic circuit of a load-sensing system

be discretised into N intervals of equivalent length t_f/N. Then the discrete version of the cost functional from Eq. (11) may be written as

$$\min_{\mathbf{u}\in\boldsymbol{\rho}} \left\{ T_\mathrm{s} \sum_{k=0}^{N-1} \Big[L_{\mathrm{p},k}\big(\mathbf{x}_k, \mathbf{u}_k\big) + \ell_k\big(\mathbf{x}_k, \mathbf{u}_k\big) \Big] \right\} \quad (16)$$

where T_s is the integration step, k denotes the discrete time index (stage), $\mathbf{x}_k$ denotes the discrete state vector, $\mathbf{u}_k$ denotes the discrete control vector, $L_{\mathrm{p},k}$ is the discrete performance cost at time stage k, $\ell_k\big(\mathbf{x}_k, \mathbf{u}_k\big)$ is the additive term at time stage k that penalises the violation of joint limits (joint ranges, joint velocity and joint acceleration) and $\boldsymbol{\rho}$ is the control policy search space of feasible $\mathbf{u}_k$, defined for time indices from 0 to $N-1$. When the joint limits are violated $\big($based on the limits defined in Eq. (15)$\big)$, a substantial constant much higher than the normally highest cost function value is added to term ℓ_k to ensure that controls that lead to exceeding joint limits are avoided. The final cost at N, which is independent of the control, is omitted.

The state dynamics may be discretised by using the well-known explicit, forward Euler method

$$\mathbf{x}_{k+1} = \mathbf{F}_k\big(\mathbf{x}_k, \mathbf{u}_k\big) = \mathbf{x}_k + T_\mathrm{s}\mathbf{f}\big(\mathbf{x}_k, \mathbf{u}_k\big) \quad (17)$$

where $\mathbf{F}_k\big(\mathbf{x}_k, \mathbf{u}_k\big)$ denotes the discretised system dynamics and T_s is the integration step.

The main problems with the continuous and derived discrete formulation are their high-dimensionality and complexity. The high-dimensionality implies that dynamic programming as such is impractical, whereas, for example, the complexity and non-convexity of some of the constraints and cost functions we shall introduce signify that conventional direct optimisation methods would yield locally optimal solutions, depending on the initial guess of the control. The global solution could, however, be searched by iterating through the vast number of solutions generated from different initial guesses; however, this process is ad hoc and time consuming. Nevertheless, because we are searching for a global solution, dynamic programming is a viable candidate, but only with the modified modelling approach suggested in the following section.

4. Dynamic programming solution

Dynamic programming (DP) is a powerful discrete-time method that has one remarkable property in that it can provide a global solution to non-convex optimal control problems. The main disadvantage of DP is its computational complexity: as the number of system states or controls increase,

the computational complexity increases exponentially [16]. Hence, dynamic programming can be considered a practical approach only when dealing with low-dimensional problems. DP is based on the well-known principle of optimality defined by Richard Bellman in 1957 [17]:

> An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.

In practice, the discretisation of states and controls is required, which yields a grid where the variables involved can take only a finite number of discrete values. The cost-to-go is evaluated only at these discrete points. The finer the discretisation, the closer to the global optimum the solution obtained will be. With the discretisation in place, the solution to our optimal control problem may be obtained using the recursive Bellman equation (principle of optimality):

$$J_k\big(\mathbf{x}_k\big) = \min_{\mathbf{u}_k \in \mathbb{U}} \Big\{ L_{\mathrm{p},k}\big(\mathbf{x}_k, \mathbf{u}_k\big) + \ell_k\big(\mathbf{x}_k, \mathbf{u}_k\big) + J_{k+1}\big(\mathbf{F}_k(\mathbf{x}_k, \mathbf{u}_k)\big) \Big\} \tag{18}$$

where $J_k\big(\mathbf{x}_k\big)$ is the optimal cost-to-go from stage $k \in \big\{0, 1, \ldots, N-1\big\}$ to the final stage $N-1$ defined for each state vector combination at stage k such that $\mathbf{x}_k \in \mathbb{X} = \big\{\mathbb{X}_1 \times \mathbb{X}_2 \times \cdots \times \mathbb{X}_{N_\mathrm{x}}\big\}$ (N_x is the number of states) and computed over all control vector combinations at stage k such that $\mathbf{u}_k \in \mathbb{U} = \big\{\mathbb{U}_1 \times \mathbb{U}_2 \times \cdots \times \mathbb{U}_{N_\mathrm{u}}\big\}$ (N_u is the number of controls). Here, $\mathbb{X}_i = \big\{x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(N_{\mathrm{x}_i})}\big\}$ and $\mathbb{U}_i = \big\{u_i^{(1)}, u_i^{(2)}, \ldots, u_i^{(N_{\mathrm{u}_i})}\big\}$ denote the discrete sets of the state and control i, respectively. The joint limits defined in Eq. (15) determine the maximum and minimum values in these discrete sets of feasible state and control values. The variables N_{x_i} and N_{u_i}, denote the number of discretised states and controls, respectively. The sum $L_{\mathrm{p},k} + \ell_k$ represents the running cost and $\mathbf{F}_k\big(\mathbf{x}_k, \mathbf{u}_k\big)$ is the discretised version of system dynamics $\mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big)$. It should be emphasised that we search for the optimal control policy (an optimal control at each discrete time stage k) producing the minimum cost-to-go over the entire trajectory from the initial to the final stage, and this minimum should be the smallest attainable one (i.e. the global minimum instead of the local minimum).

The main implication from the principle of optimality is that the controls we find optimal from stage k to N are also optimal at a later stage, e.g. from stage $k+1$ to N. It also indicates that the problem is naturally approached in a backwards fashion from the last to the initial stage. For demonstration purposes, consider a simple problem whose solution map is shown in Fig. 4. The system has a single state x, which can take on seven values; the control u can take on three values denoted with the marker coding (circle, star and triangle); and we deal with an N-stage problem of which we show the last three stages and the computed optimal controls corresponding to each state. In the last stage N, we have no optimal controls to compute.

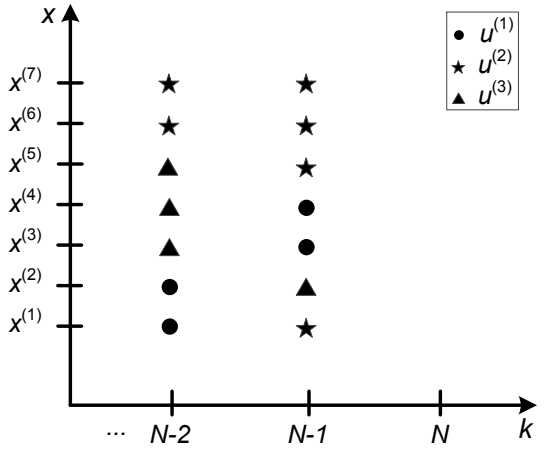


Figure 4: Example map of marker-coded optimal controls to illustrate the dynamic programming method

The procedure of computing the optimal controls at stages $N-2$ and $N-1$ of the example map is described in the following. First, let us define the running cost $c_k\big(x_k, u_k\big)$ as $L_{\mathrm{p},k}\big(x_k, u_k\big) + \ell_k\big(x_k, u_k\big)$. Because the final cost is omitted, we begin the procedure at stage $N-1$. In view of Eq. (18), we evaluate the optimal cost-to-go J_{N-1} at a particular state $x_{N-1}^{(1)}$ over the plausible controls $u_{N-1} \in$

$\{u^{(1)}, u^{(2)}, u^{(3)}\}$ as

$$\begin{aligned} J_{N-1}(x_{N-1}^{(1)}) = \min \Big\{ & c_{N-1}\big(x_{N-1}^{(1)}, u^{(1)}\big), \\ & c_{N-1}\big(x_{N-1}^{(1)}, u^{(2)}\big), \\ & c_{N-1}\big(x_{N-1}^{(1)}, u^{(3)}\big) \Big\} \\ = u^{(2)} & \end{aligned} \tag{19}$$

where $u^{(2)}$ is the example optimal control, which minimises the optimal cost-to-go J_{N-1} at the particular state. We store this control and the optimal cost-to-go in the map and repeat this procedure for all of the states.

As we step backwards into stage $N-2$, we utilize the optimal controls and optimal cost-to-go obtained at stage $N-1$ to evaluate the optimal cost-to-go J_{N-2} at each state; for example, for the particular state $x_{N-2}^{(3)}$ we may obtain

$$\begin{aligned} J_{N-2}(x_{N-2}^{(3)}) = \min \Big\{ & c_{N-1}\big(x_{N-2}^{(3)}, u^{(1)}\big) + \\ & J_{N-1}\big(F_k(x_{N-2}^{(3)}, u^{(1)})\big), \\ & c_{N-1}\big(x_{N-2}^{(3)}, u^{(2)}\big) + \\ & J_{N-1}\big(F_k(x_{N-2}^{(3)}, u^{(2)})\big), \dots \Big\} \\ = u^{(3)} & \end{aligned} \tag{20}$$

where $u^{(3)}$ is the optimal control, which minimises the optimal cost-to-go J_{N-2} at the particular state. In the computation, for example, the solution to J_{N-1} at state $F_k(x_{N-2}^{(3)}, u^{(1)})$ is known from computations performed at the previous time stage and we utilise this to evaluate the cost J_{N-2} at state $x_{N-2}^{(3)}$. As we continue with the procedure in this recursive manner to the initial stage, we may resolve the optimal policy of any length between 1 and N stages. The frequently occurring problem is that the next state derived from the state dynamics $F_k\big(x_k, u_k\big)$ with some particular control leads to a state unspecified in the state grid and hence the cost-to-go is not evaluated at this state; this problem is solved through the linear interpolation of the cost-to-go.

The DP method can easily deal with those discontinuities arising in hydraulic circuits and help solve optimisation problems arising in complicated hydraulic systems. DP was, for example, used to parametrise hydraulic excavator hybrids in [18]. For more details on the DP method, see [16], [17] and [19].

4.1. Proposed modelling approach

DP is a particularly complex approach when dealing with high-dimensional problems such as ours. Indeed, the discrete optimal control problem formulated in Eqs. (16) and (17) is too high-dimensional to solve using the DP algorithm discussed in the previous section. Here we present an improved modelling approach based upon the fact that we should only optimise the movement of redundant DOFs. This reduces our problem dimension from six to a maximum of two states and from three to one control, which enables effective use of the DP algorithm. Considering our emphasis on a typical hydraulic manipulator with three joints, from which one is redundant (Fig. 1), we optimise the movement of this redundant joint. The motion of the other joints can be effectively solved using inverse matrix computations since the remaining system is non-singular.

Let us optimise the motion of the extension joint q_3 of the manipulator. At the velocity level, the simplified system dynamics are hence

$$q_{3,k+1} = q_{3,k} + T_\mathrm{s} u_{3,k} \tag{21}$$

where $q_{3,k}$ is the extension joint position at stage k, the control input $u_{3,k}$ at stage k is $\dot{q}_{3,k}$ and T_s is the integration time step. At the acceleration level, a two-dimensional, double-integrator system could be written similarly with the acceleration of the redundant joint $\ddot{q}_{3,k}$ as input. By substituting the optimised position of the redundant joint $q_{3,k}$ into the algebraic solution of the position of the other joints $q_{2,k}$ and $q_{3,k}$, we obtain [7] (see the

atan2 version also therein)

$$
\begin{aligned}
q_{2,k} &= -\operatorname{acos}\left(\frac{n_1}{2L_1\left(L_2+q_{3,k}\right)}\right) \\
q_{1,k} &= \operatorname{asin}\left(\frac{n_2}{n_3}\right)
\end{aligned}
$$

where

$$
\begin{aligned}
n_1 &= \left(r_{\mathrm{x},k}-o_{\mathrm{x}}\right)^2+\left(r_{\mathrm{y},k}-o_{\mathrm{y}}\right)^2-L_1^2 \\
&\quad -\left(L_2+q_{3,k}\right)^2 \\
n_2 &= \big(L_1+\left(L_2+q_{3,k}\right)\cos(q_{2,k})\big)\big(r_{\mathrm{y},k}-o_{\mathrm{y}}\big) \\
&\quad -\big(L_2+q_{3,k}\big)\sin(q_{2,k})\big(r_{\mathrm{x},k}-o_{\mathrm{x}}\big) \\
n_3 &= L_1^2+\left(L_2+q_{3,k}\right)^2 \\
&\quad +2L_1\left(L_2+q_{3,k}\right)\cos\left(q_{2,k}\right)
\end{aligned}
\tag{22}
$$

and $r_{\mathrm{x},k}$ and $r_{\mathrm{y},k}$, respectively, are the desired Cartesian task-space position samples in the x and y coordinates at time stage k. That is, we also have available discrete samples from the desired Cartesian velocity $\dot{\mathbf{x}}_{\mathrm{t}}$ defined in Eq. (8) at each time stage k. Then we may use Jacobian column vectors $\mathbf{J}_i \in \mathbb{R}^2$ to rewrite Eq. (8) in discretised form as

$$
\begin{aligned}
\dot{\mathbf{x}}_{\mathrm{t},k} &= \mathbf{J}(\mathbf{q}_k)\dot{\mathbf{q}}_k \\
&= \mathbf{J}_1(\mathbf{q}_k)\dot{q}_{1,k}+\mathbf{J}_2(\mathbf{q}_k)\dot{q}_{2,k}+\mathbf{J}_3(\mathbf{q}_k)u_{3,k}
\end{aligned}
\tag{23}
$$

where the control $u_{3,k}$ is the joint velocity of the extension joint coordinate $\dot{q}_{3,k}$. From Eq. (23), we solve the desired Cartesian velocity with the contribution of the optimised extension joint included so

$$
\begin{aligned}
\dot{\mathbf{x}}_{\mathrm{t_r},k} &= \dot{\mathbf{x}}_{\mathrm{t},k}-\mathbf{J}_3(\mathbf{q}_k)u_{3,k} \\
&= \mathbf{J}_1(\mathbf{q}_k)\dot{q}_{1,k}+\mathbf{J}_2(\mathbf{q}_k)\dot{q}_{2,k}
\end{aligned}
\tag{24}
$$

where $\dot{\mathbf{x}}_{\mathrm{t_r},k}$ contains the Cartesian velocity required from joints q_1 and q_2 to maintain the desired Cartesian trajectory $\dot{\mathbf{x}}_{\mathrm{t},k}$. Because the remaining system is non-singular, we solve the angular joint velocities $\dot{q}_{1,k}$ and $\dot{q}_{2,k}$ in a straight forward manner using matrix inversion:

$$
\begin{bmatrix}\dot{q}_{1,k}\\ \dot{q}_{2,k}\end{bmatrix} = \begin{bmatrix}\mathbf{J}_1(\mathbf{q}_k) & \mathbf{J}_2(\mathbf{q}_k)\end{bmatrix}^{-1}\dot{\mathbf{x}}_{\mathrm{t_r},k}
\tag{25}
$$

where the Jacobian inverse matrix $\begin{bmatrix}\mathbf{J}_1(\mathbf{q}_k) & \mathbf{J}_2(\mathbf{q}_k)\end{bmatrix}^{-1}$ is symbolically computed in advance.

Extending the formulation to the acceleration level, we replace Eq. (21) with a standard two-dimensional double-integrator system with $\ddot{q}_{3,k}$ as the control $u_{3,k}$. Then, as before, we solve the unknown joint positions and velocities with Eqs. (22)–(25) and solve the non-singular system to obtain the joint accelerations

$$
\begin{aligned}
\begin{bmatrix}\ddot{q}_{1,k}\\ \ddot{q}_{2,k}\end{bmatrix} &= \begin{bmatrix}\mathbf{J}_1(\mathbf{q}_k) & \mathbf{J}_2(\mathbf{q}_k)\end{bmatrix}^{-1} \\
&\quad \times\left(\ddot{\mathbf{x}}_{\mathrm{t},k}-\dot{\mathbf{J}}(\mathbf{q}_k,\dot{\mathbf{q}}_k)\dot{\mathbf{q}}_k-\mathbf{J}_3(\mathbf{q}_k)u_{3,k}\right)
\end{aligned}
\tag{26}
$$

where the subtraction on the right-hand side in parenthesis contains the Cartesian acceleration required from joints q_1 and q_2 to maintain the desired Cartesian trajectory $\ddot{\mathbf{x}}_{\mathrm{t},k}$ and the Jacobian time derivative $\dot{\mathbf{J}}(\mathbf{q},\dot{\mathbf{q}})$ is symbolically computed in advance using the well-known chain rule of differentiation. In the above, we assumed that the desired Cartesian trajectory is twice-differentiable and sampled.

A similar approach formulated with torque input can be found in [20], but the approach here presents a simpler and more general system. In [21], the solution obtained in [20] was criticised for being a complex formulation of the acceleration, whereas our velocity level one-dimensional approach is merely a function of the joint positions and velocities. This considerably simplifies the redundancy resolution and yields the global solution for the CP system with relative ease. Furthermore, our two-dimensional acceleration level approach, which is not formulated from the perspective of manipulator dynamics, preserves simplicity over [20] and facilitates the introduction of general cost functions dependent on the manipulator motion state. The major benefit of the acceleration level solution over the velocity level solution is that the acceleration level solution satisfies physical joint acceleration limits. For simplicity, the highly complex pump equations and actuator pressure dynamics are omitted from the system dynamics, which means that we assume the pump can respond to the flow required by the actuators and the hydraulic fluid is incompressible. The problem formulation presented here means that a standard DP solution is feasible with the proposed approach.

5. Proposed cost functions

In hydraulic manipulators, the energy consumption of the actuators does not equal the energy consumed by the hydraulic components because of the pressure losses over the control valves when the actuators are subject to unequal loads. Thus, to effectively solve the redundancy resolution problem, we propose cost functions formulated from the standpoint of the hydraulic system instead of the manipulator dynamics or actuators.

5.1. Pump flow rate

Minimizing the pump flow rate over the Cartesian trajectory decreases pumping effort and minimises the hydraulic energy consumption of the CP system. The minimisation is plausible because of the controllability of the variable displacement pump. This minimisation has particular relevance when the hydraulic cylinders have different sizes, thus leading to varying cylinder flow requirements. Due to the high variability of cylinder sizes in practice, the problem of finding joint trajectories of the least pump flow over the Cartesian path is tractable. The different piston and piston rod areas in single-rod cylinders extend the optimisation potential.

The discontinuity and nonlinearity of the pump flow cost function, which occurs because of the variation of displaced area as a function of the direction of motion, is not a problem in standard DP. In view of Eq. (16), the pump flow rate cost function such that $Q_{\mathrm{p},k} \geq 0$ may be written at time stage k by

$$\begin{aligned} L_{\mathrm{p},k} &= Q_{\mathrm{p},k} \\ &= \sum_{i=1}^{3} \Big\{ Q_{\mathrm{A}i,k} - Q_{\mathrm{B}i,k} \Big\} \\ &= \sum_{i=1}^{3} \Big\{ v_{i,k} \big[A_{\mathrm{A}i} H(v_{i,k}) - A_{\mathrm{B}i} H(-v_{i,k}) \big] \Big\} \\ &= \sum_{i=1}^{3} \Big\{ \dot{q}_{i,k} r_{\mathrm{n}_i,k} \big[A_{\mathrm{A}i} H(\dot{q}_{i,k}) - A_{\mathrm{B}i} H(-\dot{q}_{i,k}) \big] \Big\} \end{aligned} \tag{27}$$

where $Q_{\mathrm{A}i,k}$ is the flow rate to the piston side of the cylinder i, $Q_{\mathrm{B}i,k}$ is the flow rate to the piston rod-side of the cylinder i, $H(\dot{q}_{i,k})$ is the piecewise Heaviside step function, $v_{i,k}$ denotes the cylinder velocity of actuator i, the discontinuous cylinder area is denoted with $A_{\mathrm{A}i} H(\dot{q}_{i,k}) - A_{\mathrm{B}i} H(-\dot{q}_{i,k})$, in which $A_{\mathrm{A}i}$ is the piston area and $A_{\mathrm{B}i}$ is the piston rod-side area of cylinder i, and $r_{\mathrm{n}_i,k}$ is the torque arm of cylinder i. For simplicity, the hydraulic fluid is assumed to be incompressible. Using a cylinder differential connection would change the piston side area $A_{\mathrm{A}i}$ to $A_{\mathrm{A}i} - A_{\mathrm{B}i}$ in the case of cylinder extension. This change would mean that the pump flow rate requirement for this particular movement reduces to $v_{i,k}(A_{\mathrm{A}i} - A_{\mathrm{B}i})$ or $v_{i,k} A_{ri}$, where A_{ri} is the circular area of the piston rod of cylinder i.

The Heaviside step function is defined by

$$H\big(\dot{q}_{i,k}\big) = \begin{cases} 0 & \text{if } \dot{q}_{i,k} < 0 \\ \frac{1}{2} & \text{if } \dot{q}_{i,k} = 0 \\ 1 & \text{if } \dot{q}_{i,k} > 0 \end{cases} \tag{28}$$

where the intermediate value at zero joint velocity is defined as half for convenience. This discontinuous Heaviside step function may be rewritten in differentiable form after some mathematical manipulations:

$$H\big(\dot{q}_{i,k}\big) \approx \hat{H}\big(\dot{q}_{i,k}\big) = \frac{1}{1 + e^{-2s\dot{q}_{i,k}}} \tag{29}$$

where s is a real positive number. By using this approximation, the flow rate objective is differentiable for all real $\dot{q}_{i,k}$ and may be convenient in practice.

In Eq. (27), the $A_{\mathrm{B}i}$ area is negatively signed to enforce positive $Q_{\mathrm{B},k}$ when the joint velocities are negative. We used the properties of tangential velocity, which states that $v_{i,k}$ can be written as $\dot{q}_{i,k} r_{\mathrm{n}_i,k}$, and also the property, which states that $H(v_{i,k})$ can be rewritten as $H(\dot{q}_{i,k})$ since $r_{\mathrm{n}_i,k}$ is positive. The torque arm $r_{\mathrm{n}_i,k}$ at time stage k is defined by

$$r_{\mathrm{n}_i,k} = \frac{L_{i_1} L_{i_2} \sin\big(q_{i,k} + q_i^{(0)}\big)}{\sqrt{L_{i_1} + L_{i_2} - 2L_{i_1} L_{i_2} \cos\big(q_{i,k} + q_i^{(0)}\big)}} \tag{30}$$

if $q_{i,k}$ is rotational, otherwise $r_{\mathrm{n}_i,k}$ is one. The distances L_{i_1} and L_{i_2} are the constant distances between the center of rotation and lower and upper cylinder joints, respectively. The initial value on the torque arm corresponds to a fully retracted piston when the joint coordinate $q_{i,k}$ is at its minimum value $q_{i_{\min}}$. Hence, $q_i^{(0)}$ is denoted by $q_{i_{\mathrm{c}}} - q_{i_{\min}}$. The quantity $q_{i_{\mathrm{c}}}$ is the angle of the triangle opposite a fully retracted cylinder. The triangle is formed by the upper cylinder joint, lower cylinder joint and rotational joint $q_{i,k}$.

5.2. Constant supply pressure system: Energy consumption

By optimizing the movement of the redundant joint to minimise the flow delivered to the actuators, the energy consumption of the CP system is minimised. We see this from the hydraulically produced power of the constant pressure system, written in view of Eq. (16)

$$L_{\mathrm{p},k} = \frac{p_{\mathrm{s}} Q_{\mathrm{p},k}}{\eta_{\mathrm{t},k}} \tag{31}$$

where p_{s} is the constant supply pressure, $Q_{\mathrm{p},k}$ is the pump generated flow rate defined in Eq. (27) at time stage k and $\eta_{\mathrm{t},k}$ is the total energy efficiency of the hydraulic pump and driving motor at time stage k. Because weighting $L_{\mathrm{p},k}$ with $1/p_{\mathrm{s}}$ yields the pump flow rate cost, the objectives are ideally equal. This holds the assumption that the constant pressure has been fixed pre-optimisation. Moreover, based on our assumption that the hydraulic energy cannot be reused, the energy consumed maintains positivity. The simplified one-dimensional system model previously presented can be used with this cost function, but the joint acceleration limits may not be obeyed. The total energy efficiency depends on the prevalent operating point, i.e. the supply pressure, displacement and rotation speed of the pump, but to showcase the basic capability of the redundancy resolution and due to a lack of realistic efficiency data, we assume a constant energy efficiency in our investigations.

5.3. Load-sensing system: Energy consumption

Systems based on LS architecture generate pressure losses over the control valves whenever the loading between actuators is unequal because of the system's nature, in which the highest actuator pressure is demanded at the pump level. Because the supply pressure varies depending on the load, instead of minimizing flow, we must specifically minimise the hydraulic power produced by the LS system

$$L_{\mathrm{p},k} = \frac{p_{\mathrm{s},k} Q_{\mathrm{p},k}}{\eta_{\mathrm{t},k}} \tag{32}$$

where $p_{\mathrm{s},k}$ is the supply pressure at time stage k, $Q_{\mathrm{p},k}$ is the pump flow rate at time stage k defined by Eq. (27) and $\eta_{\mathrm{t},k}$ is the total efficiency of the hydraulic pump and driving motor at time stage k.

Omitting the dynamics of the variable displacement pump, the supply pressure $p_{\mathrm{s},k}$ varies in unison with the highest actuator pressure

$$p_{\mathrm{s},k} = \max\Big\{p_{1,k}, p_{2,k}, p_{3,k}\Big\} + \Delta p_{\mathrm{LS}} \tag{33}$$

where $p_{1,k}$ denotes the chamber pressure of the lift cylinder, $p_{2,k}$ denotes the chamber pressure of the tilt cylinder, $p_{3,k}$ denotes the chamber pressure of the extension cylinder and Δp_{LS} is the LS pressure margin, which is conventionally set to approximately 2 MPa. The cylinder chamber pressure of actuator i is solved from

$$p_{i,k} = \left(\frac{|F_{i,k}|}{A_{i,k}} + p_{\mathrm{BP},i,k}\right) H\big(F_{i,k} v_{i,k}\big) \tag{34}$$

where the actuator force $F_{i,k}$ at time stage k can be computed using $T_{i,k}/r_{n_i,k}$, with $r_{n_i,k}$ defined by Eq. (30), $p_{\mathrm{BP},i,k}$ is the positive cylinder back-pressure at time stage k and piston area $A_{i,k}$ is denoted with $A_{\mathrm{A}i}H(\dot{q}_{i,k}) + A_{\mathrm{B}i}H(-\dot{q}_{i,k})$, which varies as a function of the direction of motion. We see that $H\big(F_{i,k}v_{i,k}\big)$ sets the required actuator pressure to zero when dealing with negative energy, and the absolute value on $F_{i,k}$ satisfies the requirement for non-negative actuator pressure.

The actuator forces $F_{i,k}$ and torques $T_{i,k}$ are solved from manipulator dynamics, which can be formulated based on well-known Lagrangian or robotic conventions. Centers of the mass positions of the links r_i and the link masses m_i in Fig. 5 are provided in Appendix A. Using these parameters, we obtain the inertia matrix and gravitational component using the procedure described in [7].

The back-pressure of cylinder i can be written as

$$p_{\mathrm{BP},i} = \frac{A_{\mathrm{B},i}}{A_{\mathrm{A},i}} p_{\mathrm{BP,B},i} H(v_i) + \frac{A_{\mathrm{A},i}}{A_{\mathrm{B},i}} p_{\mathrm{BP,A},i} H(-v_i) \tag{35}$$

where time indices are omitted for clarity. Back-pressure is taken from the rod-side when the cylinder extends by using the Heaviside function. Similarly, back-pressure is taken from the piston-side when the cylinder retracts. The cylinder area ratios scale the back-pressure from rod to piston-side and vice versa. In the case of a pressure-compensated hydraulic valve, the back-pressures of cylinder chambers A and B, i.e. $p_{\mathrm{BP,A},i}$ and $p_{\mathrm{BP,B},i}$, respectively, can be estimated in the steady-states

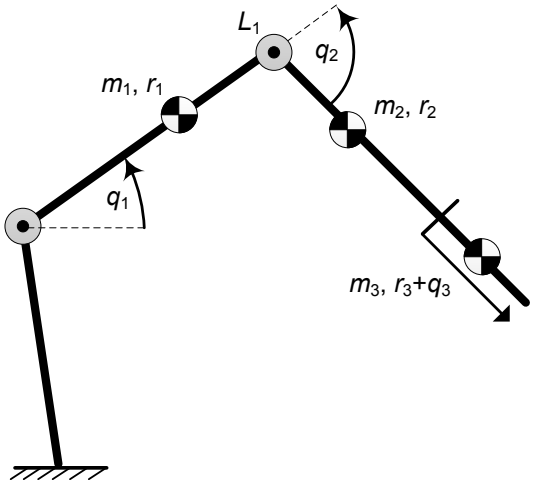


Figure 5: Positions of the center of masses of the links

by the multiplication of the valve's constant pressure difference with a certain coefficient [22]. When the back-pressures are insignificant, like in our closed-loop controlled separate meter-in and separate meter-out orifice configuration, these steady-state back-pressure functions can be omitted.

The maximum function in Eq. (33) is discontinuous but could be approximated with a differentiable expression for the sake of practical implementation after some mathematical manipulations:

$$\max\left\{p_{1,k}, p_{2,k}, p_{3,k}\right\} = n_{\mathrm{p}}/4 \tag{36}$$

where n_{p} is given by $p_{1,k} + p_{2,k} + |p_{1,k} - p_{2,k}| + 2p_{3,k} + \left|p_{1,k} + p_{2,k} + |p_{1,k} - p_{2,k}| - 2p_{3,k}\right|$ and the absolute values $|p_{i,k}|$ should be approximated with $\sqrt{{p_{i,k}}^2 + \epsilon}$. This approximation originates from the well-known definition of the maximum function of two variables via absolute values. The accuracy of this approximation improves when we decrease the value of the real positive ϵ (e.g. to class 10^{-6}).

6. Numerical examples

The purpose of these numerical examples is to compare the global and local solutions in relation to typical hydraulic manipulator applications and to showcase the superior performance of the global approach, which is due to the properties of the optimal control problem. We are particularly interested in showing how much energy can be saved in relation to the cost functions. We perform a comparison and discuss some parameters' effect on the solutions.

6.1. Setting up the numerical examples

Let us define our manipulator using the parameters supplied in Appendix A. By applying these, we obtain the workspace shown in Fig. 6. The reachable workspace without the extension joint ($q_3 = 0$) is shown with a circular marker. The load mass was fixed at 475 kg, which is a reasonable choice considering the heavy-duty lifting carried out at construction sites. This load mass was also close to the load capacity of the system. The same weight was employed to closed-loop control tests in [1]. We setup our comparison so a variety of optimisation methods complete a fictitious task cycle in which the end-effector is driven through triangular paths comprising diagonal, vertical and horizontal path, completed in this order. Throughout the experiments, the Cartesian paths between two points are generated with a quintic rest-to-rest polynomial trajectory, which provides smooth positional references for deriving the desired velocity and acceleration trajectories [23].

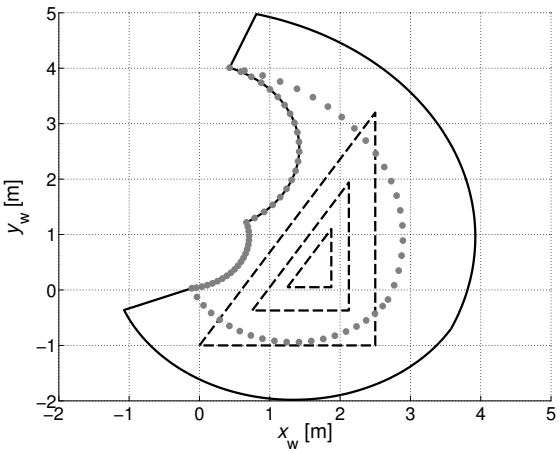


Figure 6: Test cycles for analyzing attainable energy-saving in the manipulator workspace

On the hydraulic side, the joint-actuating cylinders were sized as follows: ⌀80/45−0.545, ⌀80/45−0.545 and ⌀50/30 − 1.04, respectively, for the first (lift function), second (tilt) and third (extension) joints. These parameters were derived from a commercial construction crane. The cylinder velocities and accelerations were conservatively limited to the values shown in Appendix A. In addition, we briefly

investigate the parameter sensitivity's effect on the redundancy resolution.

The DP approaches with various cost functions are compared with non-conventional pseudo-inverse approaches formulated in actuator coordinates, and the fmincon function from the Optimization toolbox in Matlab and the DIDO application package [24, 25] are compared with the application to the described manipulator. The pseudo-inverse applied was the joint-limited (joint ranges and joint velocity limits satisfied) null-space projection method introduced by Flacco et al. [26], but we extended it to the actuator space so norm $\mathbf{v}^{\mathrm{T}}\mathbf{A}\mathbf{v}$ as the weighted or norm $\mathbf{v}^{\mathrm{T}}\mathbf{v}$ as the unweighted version are minimised instead of the standard norm $\dot{\mathbf{q}}^{\mathrm{T}}\dot{\mathbf{q}}$. The joint ranges (cylinder strokes) and velocity limits as well as the reference trajectory were satisfied. The weighting matrix is a diagonal matrix $\mathbf{A} \in \mathbb{R}^{3x3}$ in which the diagonal terms A_{ii} are given by $A_{\mathrm{A}i}H(v_i)+A_{\mathrm{B}i}H(-v_i)$. The Matlab function fmincon (ver. 2013b) was set up with the default interior-point method; the system used was Eqs. (11)–(15) integrated with a fixed-step Runge-Kutta solver. For the fmincon algorithm, we used the high-dimensional system because the solution resulting from the proposed low-dimensional system more frequently failed to converge while satisfying the constraints; whereas for the DIDO software, we successfully used the low-dimensional system. It is a well-known fact that pseudospectral methods (that is implemented for example in DIDO software) can yield satisfactory trajectories with relatively few discretisation nodes, albeit the value of the cost may then be imprecise. To improve the cost of the DIDO solution and smoothness of DIDO trajectories, the DIDO solver was run with higher nodes via a bootstrapping approach. In this approach, regarding the CP case, the solver was first run with a 20 discretisation node solution that was inputted as a guess for a 30 node solution that, in turn, was inputted as a guess for a 60 node solution that was finally inputted as a guess for a 90 node solution. The LS case was treated similarly, but using a higher number of nodes. We noticed that a DIDO solution is obtained significantly faster when using a continuous form of the flow rate cost function. Thus, we employed the continuous Heaviside approximation. The position of the extension joint was initialised with the smallest feasible value satisfying the other joint ranges, and the remaining joint positions were solved using Eq. (22) in all of the methods. The initial and final joint velocities were set to zero in the optimal control methods.

Energy savings were firstly computed using a simplified model which did not possess actuator pressure dynamics, friction effects or LS pump pressure dynamics. Energy savings were secondly computed using a full-scale, closed-loop simulation of the hydraulic system to demonstrate that our simplified problem formulation is pragmatic and the neglected hydraulic aspects of the problem could be omitted. Actuator pressure dynamics, friction effects and pump pressure dynamics were included in the full simulation model. Energy saving results are presented for both the simplified and full simulation case in the following sections. In the closed-loop simulation case, hydraulic cylinder chamber pressures were controlled independently using pressure-compensated valves, which were set-up in a separate meter-in and separate meter-out orifice configuration. The separate meter-in and meter-out valves were controlled so that valve notch connections P-A & B-T and P-B & A-T were simultaneously opened (see Figs. 2 and 3). The notches' magnitude of opening was of course otherwise controlled independently. Hydraulic system parameters were set to the same values in each closed-loop simulation to yield comparable results. The adaptive robust control approach [27, 28] was used as the motion controller; however, the cross-port valve was excluded. The adaptive robust controller was implemented for each manipulator function, allowing the controller's robustness to modelling uncertainties to dominate neglected joint coupling terms. To avoid cavitation, the so-called offside cylinder chamber pressures during the motion were regulated to a constant 1 MPa.

6.2. Constant-pressure system

The cost function used for the comparison in Table 2 is Eq. (16), into which the pump flow cost from Eq. (27) is substituted. Therefore, the cost function is the sum of the pump flow rate over the Cartesian trajectory in which the largest triangular path is driven with 10 seconds spent on each edge. The cost function values are scaled with the minimum, i.e. best, result. We searched for the global solution from the infinite space of solutions without any restriction apart from the constraints on joint ranges, joint velocity limits and joint acceleration limits set in Appendix A. The maximum supply pressure level was not restricted in any way. We obtained closed-loop tracking performances comparable to [27, 28] when using the optimised joint

motion trajectories as the motion controller's references, i.e. the position tracking errors were generally less than 0.01 rad or 0.01 m.

The energy savings predicted by the simplified model, which neglected conventional hydraulic system dynamics, and the more accurate closed-loop simulation prediction, which included these effects, are mostly comparable, showing that the degree of model simplification in the problem formulation is justified. The DP approach with the CP cost function yields the lowest cost in both cases, although the DIDO approach set to minimise Eq. (27) has an almost identical cost to the DP approach. Still, keeping that DP approach as our baseline, a comparison shows that the DP methods written to minimise the LS cost in view of Eq. (32), the positive actuator energy cost or the actuator velocity cost $\mathbf{v}^{\mathrm{T}}\mathbf{v}$ yield sub-optimal joint trajectories that require a somewhat higher flow to perform the test cycle. The fmincon function yields a very poor local optimum that is significantly inferior to the global optimum. The weighted pseudo-inverse yields the worst solution, in which the pump flow over the test cycle was over 30% greater. Surprisingly, the unweighted pseudo-inverse written in the cylinder coordinates yields a decent result. All of the results concern the largest triangle path in Fig. 6 completed in 30 seconds, or 10 seconds per edge, which amounts to a Cartesian velocity of the end-effector of roughly 0.5 m/s on the diagonal path.

In reference to the smaller triangular paths, we saw the energy-saving potential clearly decreasing because of the smaller area covered, i.e. optimisation potential is lost with a decreasing range of motion. The DP solutions owing to the discretisation are prone to having slightly jagged edges, but proper filtering smooths these out. The cost function values in the tables throughout this paper were computed after this low-pass filtering. The low-pass filtered (cutoff frequency 5 Hz) cylinder velocity trajectories are in Figs. 7a–7c, which show the largest test cycle.

The proposed simplified first-order DP approach was parametrised by state and control grid size, with the state being divided into 200 discrete values and the control being divided into 101 discrete values. In addition, the algorithm was highly computationally efficient and the modelling accuracy (with its discretisation) was more than adequate. The time was discretised into 0.05 second steps. Surprisingly, this enabled a significantly faster solution and lower system memory consumption than the fmincon method, which uses the interior-point algorithm. As seen, the fmincon can very easily yield a local minimum to our problem. Therefore, global optimal control is seen as a highly attractive solution.

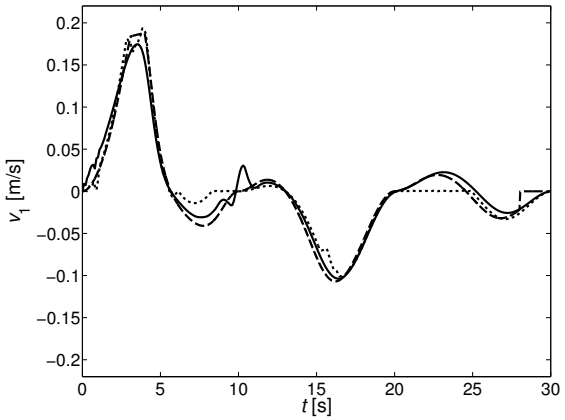
The second-order model, however, allows the inclusion of joint acceleration limits we applied to our comparison. DP with the joint acceleration constraints intuitively yields a cost that is always higher or the same as the cost obtained from DP without joint acceleration constraints. In terms of discretisation, the state space of the extension cylinder position was divided into 125 discrete values, the extension cylinder velocity into 101 values and the extension cylinder acceleration into 201 values. DP with the second-order model is still faster than the fmincon algorithm even though the increase in the number of states leads to an exponential increase in the computational burden. Decreasing the grid sizes in this case reduces the computational effort without any significant sacrifice in performance. Halving the state grid size halves the requirement on system memory and computation time, and the same applies for the control grid. The second-order approach was clearly not as computationally efficient as the first-order approach, but decreasing the grid sizes even more is plausible if faster execution is desired, albeit at the expense of inferior trajectories. The DIDO approach possessed a lower computational complexity than the DP approach and a somewhat lower memory requirement.

Figure 8 demonstrates the convergence properties of the dynamic programming algorithm in the CP case. It shows that even the coarser state and control grids are capable of producing solutions that are in reasonably close agreement with the global optimum. The anticipated difference between the coarser and denser grid solutions is the smoother motion profiles of the denser solutions. The discretisation used corresponds to the second to last marker in Fig. 8. Finally, as a major convenience, the problem concerning the CP system may be solved without any knowledge about the manipulator dynamics.

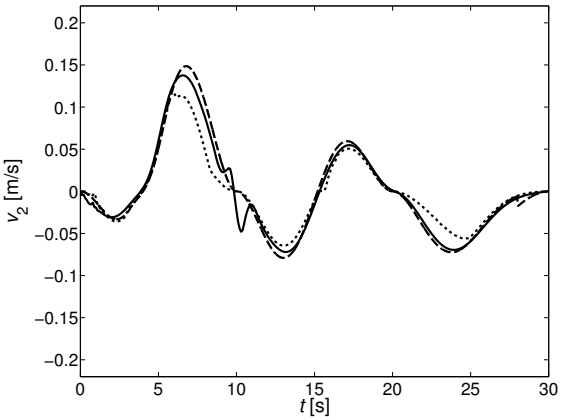
The energy saved can be increased by decreasing the cylinder areas since the actuator flows are reduced. In some cases, however, the decrease in cylinder area leads to an intolerable increase in the supply pressure level. Even minor realistic cylinder area changes can increase the energy savings and therefore extend the optimisation potential. Consider our numerical example in which

Table 2: Constant pressure system: Comparison of optimal and sub-optimal approaches (with joint limits) based on energy savings predicted by the simplified model and closed-loop simulated full-scale model

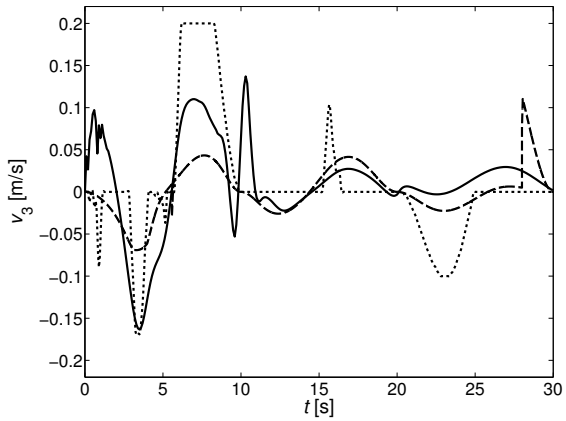
Method (minimised cost)	Simplified rel. cost function	Full rel. cost function
DP (CP)	1.000	1.000
DIDO (CP)	1.003	1.002
DP (LS)	1.062	1.060
DP (Act. velocity)	1.087	1.086
DP (Pos. actuator)	1.098	1.098
Act. p-inv. (Unweighted)	1.127	1.124
fmincon (CP)	1.166	1.163
Act. p-inv. (Weighted)	1.308	1.308



(a) Lift cylinder velocity v_1 (m/s) over time (s)



(b) Tilt cylinder velocity v_2 (m/s) over time (s)



(c) Ext. cylinder velocity v_3 (m/s) over time (s)

Figure 7: CP optimal dynamic programming trajectory (····), suboptimal fmincon trajectory (—) and pseudo-inverse trajectory (− −)

the sizes of the tilt and lift actuators are the same. If we changed the tilt cylinder size (joint 2) from $\oslash 80/45 - 0.545$ to $\oslash 80/56 - 0.545$, the flow requirement on the pump decreased by 5% over the

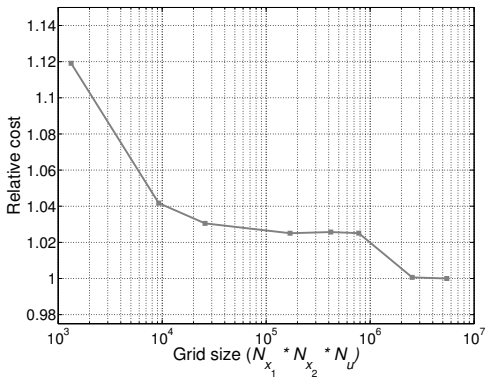


Figure 8: Convergence of the dynamic programming algorithm in the constant-pressure case; the scale for the x-axis is logarithmic to improve readability, and the y-axis cost function value is scaled with the cost obtained using the densest grid size

largest triangular path. The reduction was 4% in the medium and small triangular paths with this moderate change. But since the maximum supply pressure in the cycle increased by well over 5%, the change was clearly not advisable. If we changed the extension cylinder size from $\oslash 50/30 - 1.04$ to $\oslash 45/30 - 1.04$, the energy saved on the largest triangular path was 4%, with negligible effect on the supply pressure level. It may be that the extension cylinder of the typical hydraulic manipulator has the most potential for size optimisation. Component size optimisation is, however, not the main focus of this paper, but even this example demonstrates the energy savings obtained by sizing the hydraulic cylinders used in a work cycle.

6.3. Load-sensing system

As before, the cost function used for the comparison in Table 3 is Eq. (16), into which the LS cost from Eq. (32) is substituted. Therefore, the cost function is the sum of consumed LS energy over the Cartesian trajectory in which the largest triangular path is driven, with 10 seconds spent on each edge. The cost function values are scaled with the minimum result. The same state and control grids were applied as in the CP case, and the second-order model with joint ranges, cylinder velocity and acceleration limits were used. The closed-loop control performances in the simulations, in which the LS pump's time constant was set to 0.20 seconds [29], were similar to the CP case.

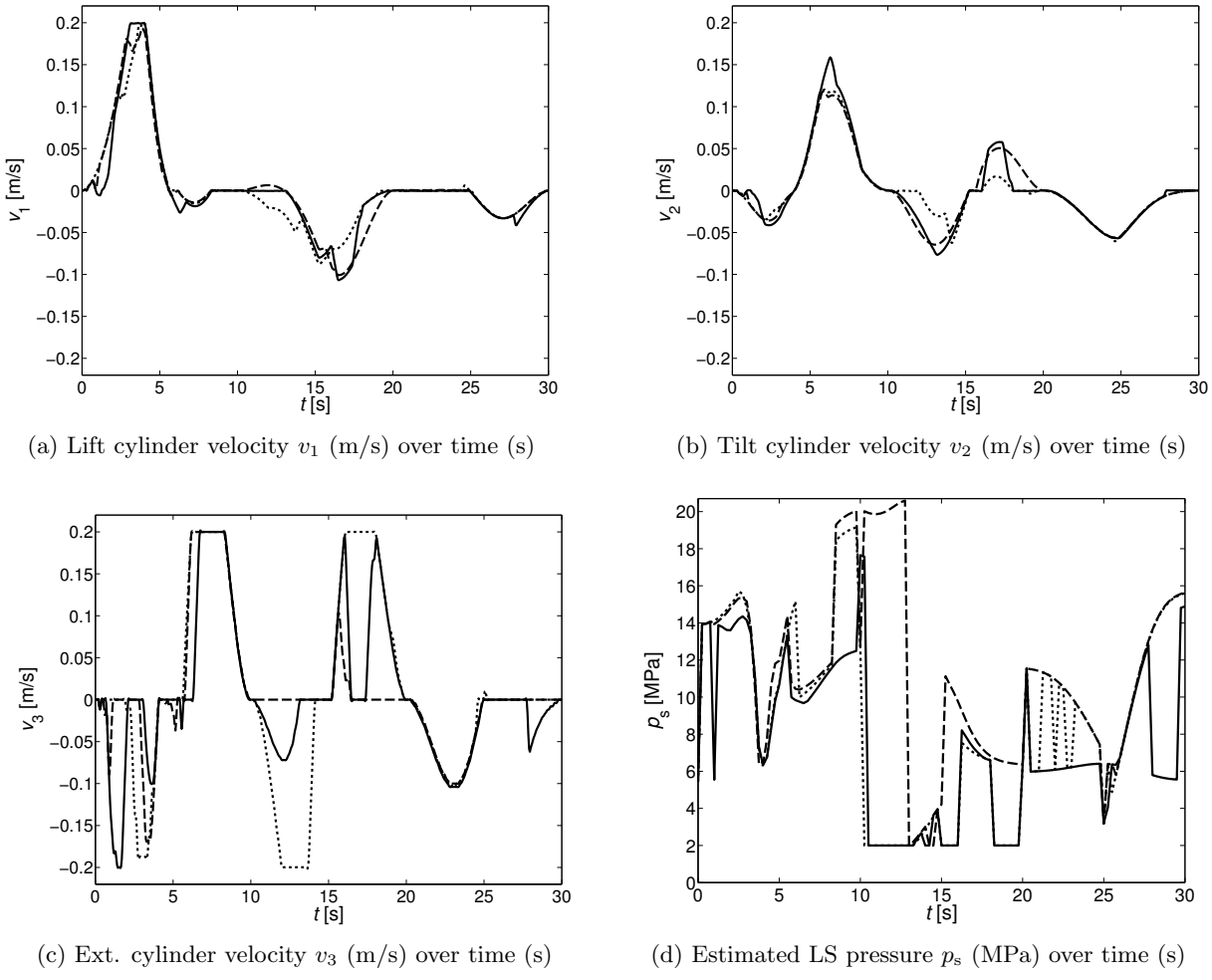
The energy savings predicted by the simplified model and closed-loop simulation are somewhat different compared to the CP case. The results imply that the energy savings obtainable in an actual LS system are lower than predicted. Still, the DP approach minimising the LS energy consumed yielded the best solution. We kept this as our baseline for comparison. The weighted pseudo-inverse formulated in the actuator coordinates produces a cost function value, which was almost 30% higher than that of the best DP solution in the simulations. The fmincon also produces a decent result, particularly in the simulated case. The computational demand of the fmincon algorithm is notably high. The pseudo-inverses produce inferior results, because they are only point-wise optimal and do not satisfy joint acceleration limits. The DP methods produce significantly better trajectories than the point-wise methods considering the moderate optimisation potential for the manipulator with one redundant joint. The DP approach minimising the CP energy consumed yields a cost which is reasonably close to the global optimum in the closed-loop simulation. As before, the DIDO solver produces a satisfactory trajectory, particularly for closed-loop simulation.

The DP method which penalises positive actuator work is nearly optimal even from the LS perspective. The numerical discrepancy in closed-loop simulation is minor, but the desired trajectories differ, as shown in the cylinder velocity trajectories below in Figs. 9a–9c. This discrepancy originates from the LS pressure losses which arise from the requirement of highest actuator pressure at the pump level. When these optimised joint trajectories are driven in closed-loop and compared, some of the optimisation potential of the LS system is lost because of the pump's non-ideal pressure dynamics. As before, the energy-saving potential decreased when the size of triangular path was reduced because of the smaller area covered.

The desired pump supply pressure (LS pressure) based on the simplified model is shown in Fig. 9d as the joints were driven through the optimal joint trajectories. This figure illustrates what the LS pressure would be with the other DP solutions for a comparison. The LS optimal solution clearly searches for a lower supply pressure than does the CP optimal solution. The actuator energy optimal solution also has a higher LS pressure demand in general than did the LS solution. The maximum actuator pressure at the pump in LS systems affects the consumption of all the actuators, which is why the LS solution must search for a path of lower

Table 3: Load-sensing system: Comparison of optimal and sub-optimal approaches (With inertial and gravitational effects and joint limits) based on energy savings predicted by the simplified model and closed-loop simulated full-scale model

Method (minimised cost)	Simplified rel. cost function	Full rel. cost function
DP (LS)	1.000	1.000
DIDO (LS)	1.087	1.012
DP (Pos. actuator)	1.109	1.027
fmincon (LS)	1.154	1.050
DP (CP)	1.201	1.083
DP (Act. velocity)	1.247	1.135
Act. p-inv. (Unweighted)	1.259	1.154
Act. p-inv. (Weighted)	1.413	1.293



(a) Lift cylinder velocity v_1 (m/s) over time (s)

(b) Tilt cylinder velocity v_2 (m/s) over time (s)

(c) Ext. cylinder velocity v_3 (m/s) over time (s)

(d) Estimated LS pressure p_s (MPa) over time (s)

Figure 9: LS optimal dynamic programming trajectory (—), CP optimal dynamic programming trajectory (− −) and positive actuator energy optimal dynamic programming trajectory (----)

pressure.

Figure 10 demonstrates the convergence properties of the dynamic programming algorithm in the LS case. Compared to the CP case, the coarser

state and control grids produce solutions that are further from the global optimum. True global optimality requires a reasonably dense grid size. The discretisation we used corresponds to the second to last marker in Fig. 10.

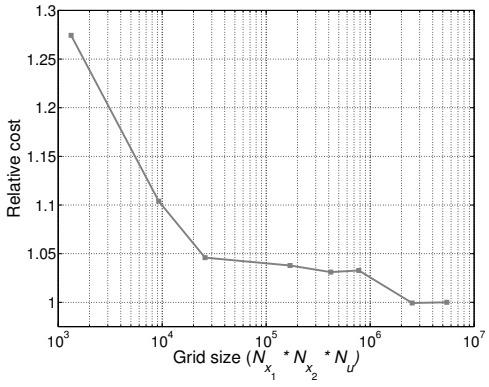


Figure 10: Convergence of the dynamic programming algorithm in the load-sensing case; the scale for the x-axis is logarithmic to improve readability, and the y-axis cost function value is scaled with the cost obtained using the densest grid size

Changing the tilt cylinder size (joint 2) from $\oslash 80/45 - 0.545$ to $\oslash 80/56 - 0.545$ increased the LS energy required over the largest triangle trajectory by almost 6%, even though the flow required by the actuators was reduced. A significant increase in LS pressure explains this. Changing the extension cylinder size from $\oslash 50/30 - 1.04$ to $\oslash 45/30 - 1.04$ was more favorable because the pump flow was reduced while the highest actuator pressure was mostly unaffected. The modification in area decreased the LS energy required by 4%. It is essential to note that while the extension cylinder pressure increased, the LS pressure for the most part did not.

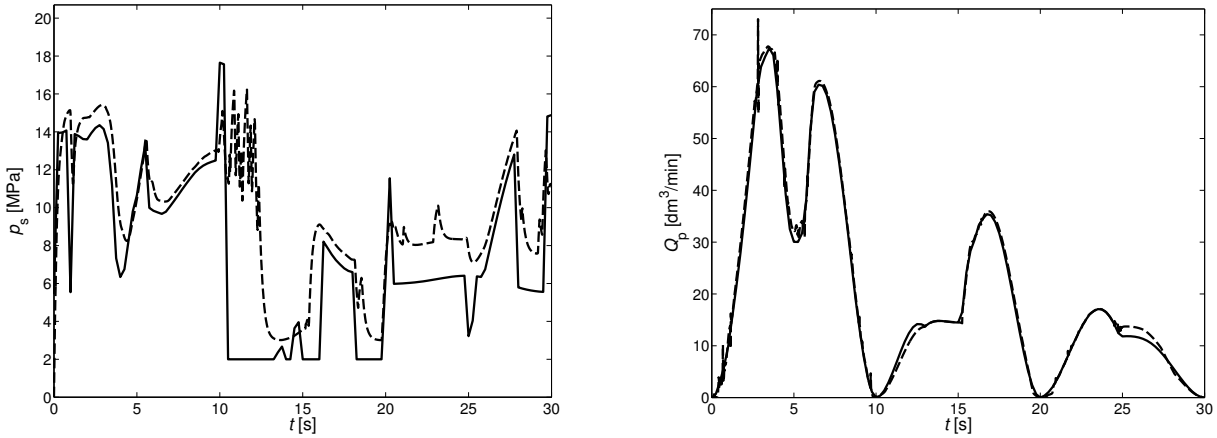
Finally, the simplified and the more complete closed-loop simulation model are compared in Figs. 11a–11b. The pump flow rate predictions of the models are consistent, although fluid compressibility was only included in the closed-loop simulation model. The supply pressure predictions are understandably less consistent because approximately 1 MPa of the supply pressure offset is due to the cylinder back-pressures omitted from the simplified model. Pump pressure dynamics and closed-loop control behaviour account for the remaining offset seen in the supply pressure comparison.

7. Discussion

In the literature and research papers thus far, the redundancy resolution problem has been resolved at the actuator level. Our results have shown this could be a suboptimal approach at the hydraulic system level. Moreover, the hydraulic redundancy resolution problem is a complex problem that cannot necessarily be resolved to global optimality using standard direct optimisation tools. Our results show that the local optimums obtained can be very poor and are close to point-wise optimal trajectories at times.

Moreover, the energy saved, as demonstrated using the DP and DIDO approaches, can be very significant (e.g. 30% compared to some point-wise optimal methods). These results demonstrate the reduction attainable in the energy consumption of a typical hydraulic manipulator which has one redundant joint. Therefore, the energy savings obtained cannot be exceptionally high. The numerical examples presented here are formulated using realistic parameters, so the results can be considered practically plausible. The extra reach by the extension cylinder in the numerical example was roughly one meter, where as in some other applications the reach may be over two meters. This could have some significance when determining the optimisation potential in other similar manipulators.

Some simplifications have been made in the problem formulation to ease the computational burden: (1) the ideal hydraulic fluid was assumed to be incompressible and (2) the pump dynamics were neglected. The first assumption seems reasonable because the amount of compressed fluid delivered by the pump should be relatively low compared to the incompressed fluid leaving the pump. This observation was verified because energy savings predicted by the simplified model had results comparable to the closed-loop simulations in the CP case. The second assumption means that we assumed that the pump responds to the commanded supply pressure within the discretised time interval of 0.05 seconds. This simplification, which was made for computational convenience, affects the validity of the results from a real manipulator because an actual pump cannot respond this fast. Particularly, we saw that including pump dynamics in a closed-loop simulation of the optimum joint trajectories, some energy saving potential in the LS case is lost. However, the simplification could still be justified because it decreases computational complexity. Moreover, the



(a) LS case: Supply pressure p_s (MPa) over time (s)

(b) CP case: Pump flow rate Q_p (dm^3/s) over time (s)

Figure 11: Model comparison: Simplified model used in optimisation (—) and full model used in closed-loop simulations (− −)

DP approach minimising the CP energy consumed, formulated without the knowledge of manipulator dynamics, was reasonably accurate in decreasing the LS case cost.

In this study we assumed a separate meter-in and separate meter-out valve configuration (SMISMO). This SMISMO configuration could allow for additional savings from differential connection utilization, but we decided to omit valve level optimization from this paper and focus on the redundancy resolution through cylinder motion optimisation that is the core of the paper. A differential connection to the hydraulic cylinder was thus not utilised in the simulations. At the moment this pump flow rate reducing connection has also not been used in the real hydraulic manipulator, which was the basis for the simulations. Cylinder back-pressures were also omitted from the optimization because the back-pressures could be regulated to a minor 1 MPa value in the closed-loop simulations using the SMISMO valve configuration. Furthermore, a fixed hydraulic pump efficiency was assumed in the simulations. Considering a variable pump efficiency could have influenced the simulation results and changed the outcome of this study.

Our approach is intended for offline optimisation, and the results are seen to motivate energy optimal redundancy resolution at the hydraulic level in future applications aiming to automate some construction crane work cycles. When the automation comes to fruition, the optimal joint trajectories could computed beforehand and retrieved from a database when needed. Online application of our solution is not possible when the workspace trajectory is unknown. To enable online application, one would need to predict the future trajectory over a short time horizon or design an improved point-wise optimal approach: the pseudo-inverse one used for comparison in this paper is clearly suboptimal. We assume that a point-wise optimal solution with appropriate adaptive weights could yield a satisfactory result, but finding the weights would be difficult.

Cylinder optimisation has some significance because it can further improve the redundancy resolution at the hydraulic level. In CP systems, this cylinder optimisation is as simple as decreasing the cylinder areas, which improves the energy savings if the pump supply pressure does not need to be boosted. In LS systems, we found the extension cylinder area to be most often reducible without increasing the LS pressure level, which leads to decreased LS energy requirement. However, on an LS system, cylinder area optimisation has intricate ties to the load mass and geometry of the manipulator. The effect of varying load mass was not evaluated, but it may be assumed that the load affects the energy savings obtained via redundancy resolution.

In our work, we did not consider energy recuperation because the energy recuperation and hybrid systems linked to these cases are still in their infancy on an application level. In addition, we focused on traditional systems because the energy optimisation at the hydraulic level had not yet been studied. In view of systems with energy recuperation, the pump flow rate cost function should be

written, omitting time indices, as

$$Q_{\mathrm{p}} = \dot{q}_1 r_{\mathrm{n}_1}(A_{\mathrm{A}1} - A_{\mathrm{B}1}) + \dot{q}_2 r_{\mathrm{n}_2}(A_{\mathrm{A}2} - A_{\mathrm{B}2}) + \dot{q}_3(A_{\mathrm{A}3} - A_{\mathrm{B}3}) \quad (37)$$

where r_{n_i} denotes the torque arms, $A_{\mathrm{A}i}$ denotes the piston-side area of the cylinder and $A_{\mathrm{B}i}$ is the rod-side area. Minimizing this cost gives the flexibility to maximally tap into the returning flow from the cylinder. For example, as cylinder one extends, the returning flow from the meter-out side can be used for simultaneously extending actuators. Making full use of this in real-time would require a hardware overhaul in conventional systems and possibly a type of model predictive control. However, with regard to the goal of automating some construction crane work cycles, the globally optimal joint trajectories could be computed beforehand with the offline approach presented using Eq. (37) as the cost function.

Although we focused on traditional hydraulic systems, the proposed hydraulic energy minimisation strategy has practical relevance in some other hydraulic systems. For example, the flow rate delivered by the dedicated pumps in valveless systems may be minimised using the analysed approach to decrease the pumping effort. In addition, the results naturally extend to some hydraulic transformer or multi-chamber cylinder systems operating on a single constant pressure source. For example, these multi-chamber systems' energy efficiency is better compared to traditional systems because the displaced cylinder area can be discretely controlled to reduce the pressure losses over the control valve. Using the efficient flow-reducing strategy, the inputted hydraulic energy of the multi-chamber system can be minimised effectively because manipulator dynamics are not required. However, the minimised flow objective is complicated in this case by the variety of choices for the cylinder areas in the multi-chamber cylinders. Overall, these cases imply that the flow-reducing optimal control presented should be practically relevant in future applications. Finally, the extension of the DP approach to non-planar manipulators with a redundant extension joint and a base-rotating actuator is possible in future work.

8. Conclusions

Because conventional methods can be insufficient when it comes to resolving kinematic redundancy at the hydraulic level, we proposed cost functions formulated into an effective, globally optimal approach to resolve the redundancy problem at the hydraulic level. The popular load-sensing and constant-pressure system architectures were treated. Due to the complex problem formulation required at the hydraulic level, the solutions obtained using particularly the point-wise optimal methods yielded significantly poorer results compared with the global approach, up to 15–30% greater energy use as seen in numerical examples. Furthermore, we found that a bootstrapped pseudospectral solution may produce practically comparable results to dynamic programming in terms of hydraulic energy minimisation. The results obtained are relevant to the typical planar feedback-controlled hydraulic construction crane, whose redundancy is founded on the prismatic reach function. However, extending the results to typical non-planar cranes is possible. Furthermore, we assumed a separate meter-in and separate meter-out valve configuration, which allowed negligible cylinder back-pressures in the closed-loop simulations presented. A differential connection to the hydraulic cylinder was not considered and we also assumed a fixed pump efficiency in the simulations. Considering a variable hydraulic pump efficiency could have influenced the simulation results and changed the outcome of this study. A variable pump efficiency, regular valve configuration and differential connection are to be considered in future work. Overall, the results obtained motivate a need and provide an efficient way to optimise the joint trajectories on prospective applications in which frequently driven work cycles for hydraulic construction cranes are automated.

Acknowledgment

This work was partially funded by the Graduate School of Concurrent Mechanical Engineering of Finland. This funding is highly appreciated. The authors also wish to sincerely thank the anonymous reviewers for critically reading the manuscript and suggesting significant improvements.

References

[1] J. Koivumäki, J. Mattila, High performance nonlinear motion/force controller design for redundant hydraulic construction crane automation, Automation in Construction 51 (2015) 59–77. doi:10.1016/j.autcon.2014.12.014.

[2] M. Vukobratovic, M. Kircanski, A dynamic approach to nominal trajectory synthesis for redundant manipulators, IEEE Transactions on Systems, Man and Cybernetics SMC-14 (4) (1984) 580–586. doi:10.1109/TSMC.1984.6313329.

[3] B. Löfgren, Kinematic control of redundant knuckle booms with automatic path-following functions, Ph.D. thesis, Royal Institute of Technology, KTH, Sweden (2009).
URL http://urn.kb.se/resolve?urn=urn%3Anbn%3Ase%3Akth%3Adiva-11495

[4] P. Hołobut, Time-optimal control of hydraulic manipulators with path constraints, Journal of Theoretical and Applied Mechanics 43 (3) (2005) 523–538.
URL http://www.ptmts.org.pl/jtam/index.php/jtam/article/view/v43n3p523

[5] Y. Halevi, E. Carpanzano, G. Montalbano, Minimum energy control of redundant linear manipulators, Journal of Dynamic Systems, Measurement, and Control 136 (2014) 051016–1. doi:10.1115/1.4027419.

[6] A. Deo, I. Walker, Minimum effort inverse kinematics for redundant manipulators, IEEE Transactions on Robotics and Automation 13 (5) (1997) 767–775. doi:10.1109/70.631238.

[7] L. Sciavicco, B. Siciliano, Modelling and control of robot manipulators, Springer, 2000. doi:10.1007/978-1-4471-0449-0.

[8] L. Beiner, J. Mattila, An improved pseudoinverse solution for redundant hydraulic manipulators, Robotica 17 (2) (1999) 173–179. doi:10.1017/S0263574799001216.

[9] G. Zeiger, A. Akers, Dynamic analysis of an axial piston pump swashplate control, Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 200 (1) (1986) 49–58. doi:10.1243/PIME_PROC_1986_200_093_02.

[10] N. D. Manring, F. A. Damtew, The control torque on the swash plate of an axial-piston pump utilizing piston-bore springs, Journal of Dynamic Systems, Measurement, and Control 123 (3) (2001) 471–478. doi:10.1115/1.1386654.

[11] P. Achten, Dynamic high-frequency behaviour of the swash plate in a variable displacement axial piston pump, Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 227 (6) (2013) 529–540. doi:10.1177/0959651813483419.

[12] P. Krus, On load sensing fluid power systems: With special reference to dynamic properties and control aspects, PhD Thesis, Linköping University, Sweden (1988).

[13] H. C. Pedersen, T. O. Andersen, M. R. Hansen, Controlling a conventional LS-pump based on electrically measured LS-pressure, in: Proceedings of the Bath/ASME Symposium on Fluid Power and Motion Control, 2008, pp. 531–546.
URL https://www.researchgate.net/publication/237344940_Controlling_a_Conventional_LS-pump_based_on_Electri-_cally_Measured_LS-pressure

[14] M. Axin, B. Eriksson, P. Krus, Flow versus pressure control of pumps in mobile hydraulic systems, Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 228 (4) (2014) 245–256. doi:10.1177/0959651813512820.

[15] J. Koivumäki, J. Mattila, Stable and high performance energy-efficient motion control of electric load sensing controlled hydraulic manipulators, in: Proceedings of the ASME/BATH Symposium on Fluid Power and Motion Control, American Society of Mechanical Engineers, 2013, pp. V001T01A024–V001T01A024. doi:10.1115/FPMC2013-4440.

[16] O. Sundstrom, L. Guzzella, A generic dynamic programming Matlab function, in: Proceedings of the IEEE Control Applications (CCA) & Intelligent Control (ISIC), 2009, pp. 1625–1630. doi:10.1109/CCA.2009.5281131.

[17] D. E. Kirk, Optimal control theory: An introduction, Courier Dover Publications, 2012.

[18] W. Shen, J. Jiang, X. Su, H. R. Karimi, Parameter matching analysis of hydraulic hybrid excavators based on dynamic programming algorithm, Journal of Applied Mathematics 2013. doi:10.1155/2013/615608.

[19] P. Elbert, S. Ebbesen, L. Guzzella, Implementation of dynamic programming for n-dimensional optimal control problems with final state constraints, IEEE Transactions on Control Systems Technology 21 (3) (2013) 924–931. doi:10.1109/TCST.2012.2190935.

[20] M. H. Choi, Redundancy resolution by minimization of joint disturbance torque for independent joint controlled manipulators, in: Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), 1999, pp. 392–397. doi:10.1109/AIM.1999.803199.

[21] C. Pholsiri, Task-based decision making and control of robotic manipulators, Ph.D. thesis, University of Texas, USA (2004).
URL http://hdl.handle.net/2152/1388

[22] S. Aranovskiy, A. Losenkov, C. Vazquez, Position control of an industrial hydraulic system with a pressure compensator, in: Proceedings of the 22nd Mediterranean Conference of Control and Automation (MED), 2014, pp. 1329–1334. doi:10.1109/MED.2014.6961560.

[23] R. N. Jazar, Theory of Applied Robotics: Kinematics, Dynamics, and Control, Springer, 2010. doi:10.1007/978-1-4419-1750-8.

[24] I. M. Ross, F. Fahroo, Pseudospectral knotting methods for solving nonsmooth optimal control problems, Journal of Guidance, Control, and Dynamics 27 (3) (2004) 397–405. doi:10.2514/1.3426.

[25] I. M. Ross, M. Karpenko, A review of pseudospectral optimal control: From theory to flight, Annual Reviews in Control 36 (2) (2012) 182–197. doi:10.1016/j.arcontrol.2012.09.002.

[26] F. Flacco, A. De Luca, O. Khatib, Motion control of redundant robots under joint constraints: Saturation in the null space, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2012, pp. 285–292. doi:10.1109/ICRA.2012.6225376.

[27] S. Liu, B. Yao, Programmable valves: A solution to bypass deadband problem of electro-hydraulic systems, in: Proceedings of the American Control Conference, Vol. 5, 2004, pp. 4438–4443.
URL http://ieeexplore.ieee.org/document/1384008/

[28] S. Liu, B. Yao, Coordinate control of energy saving programmable valves, IEEE Transactions on Control Systems Technology 16 (1) (2008) 34–45. doi:10.1109/TCST.2007.903073.

[29] Q. Yuan, J. Y. Lew, Modeling and control of two stage twin spool servo-valve for energy-saving, in: Proceed-

ings of the American Control Conference, IEEE, 2005, pp. 4363–4368. doi:10.1109/ACC.2005.1470666.

Appendix A. Parameters

Table A.1: Workspace and dynamics parameters of the manipulator

Parameter	Value
L_1	1.60 [m]
L_2	1.562 [m]
o_{x}	-0.225 [m]
o_{y}	0.957 [m]
r_1	0.771 [m]
r_2	0.663 [m]
r_3	1.856 [m]
m_1	80.11 [kg]
m_2	33.93 [kg]
m_3	570.19 [kg]
$q_{1_{\min}} : q_{1_{\max}}$	-0.212 : 1.517 [rad]
$q_{2_{\min}} : q_{2_{\max}}$	-2.545 : -0.321 [rad]
$q_{3_{\min}} : q_{3_{\max}}$	0 : 1.04 [m]
$v_{i_{\max}} : \forall i \in \{1,2,3\}$	0.2 [m/s]
$v_{i_{\min}} : \forall i \in \{1,2,3\}$	-0.2 [m/s]
$a_{i_{\max}} : \forall i \in \{1,2,3\}$	0.5 $[\mathrm{m/s^2}]$
$a_{i_{\min}} : \forall i \in \{1,2,3\}$	-0.5 $[\mathrm{m/s^2}]$