Real-time trajectory scaling algorithm for hydraulic manipulators subject to limited on-board power

Lionel Hulttinen, Jouni Mattila

Tampere University, Faculty of Engineering and Natural Sciences, lionel.hulttinen@tuni.fi

Abstract.

Online trajectory generation is a practical necessity when implementing semi-autonomous control functionality for hydraulic working machines, such as earthmoving excavators or forestry loader cranes. In these human-in-the-loop systems, smooth and consistent trajectories need to be generated from hand-lever commands provided by the user, while considering the various local and global constraints of the actuation and power delivery system. For hydraulic manipulators, many velocity-level scaling algorithms have been presented in the last years, which take into account the inherent limitations of the hydraulic power unit by considering the sum of demanded volumetric flows compared to the pump capacity. However, there is a need to generate consistent trajectories that include continuous and bounded acceleration references, and additionally also respect the limitation set by the maximum rated power of the prime mover, which the previous velocity-level approaches are not able to satisfy. In this paper, a power-limited on-line trajectory generation method for load-sensing hydraulic systems is formulated, and simulation results with a 2-DOF manipulator are provided to demonstrate the proposed algorithm.

Keywords. Motion planning, hydraulic manipulator, on-line trajectory scaling.

1. Introduction

Online trajectory generation is an important consideration when developing high-performance motion control systems for hydraulic working machines, such as excavators or various types of loader cranes. When retrofitting these machines into semi-autonomous robotic systems, where the human operator is controlling the tip of the crane instead of commanding each hydraulic cylinder separately, a remarkable effort is required to take into account the different constraints imposed by the actuators and their power transmission.

General methods to generate consistent and smooth motion trajectories for hydraulic cranes from hand-lever/joystick commands have been presented in [1, 2]. However, these schemes consider the actuation bounds of the system by defining velocity and acceleration limits for every degree of freedom separately, which can become cumbersome to tune so that global constraints, such as limits on the sum of actuator velocities or power consumption, are not violated. Performance limits for coordinated motion control of these manipulators are affected by the net gross power provided by the prime mover carried on-board, which drives a common pump supplying multiple functions at the same time.

The most obvious implication of the power constraint is the limited net velocity of the actuators [3]. Unless equipped with a flow-sharing type of hydraulic system (also known as LUDV or 'social flow distribution') preventing fluid from taking the path of least resistance, pump flow saturation leads to lagging of the most loaded actuator. This can often be witnessed as a sudden deceleration of the lift cylinder mechanism and a consequent dipping of the tool center point, typically when fast motions are demanded from multiple actuators simultanenously. This is detrimental to controller performance, since the feedback controllers are not capable of handling control input saturations appropriately unless the reference generation knows the system's inherent limitations.

In the context of hydraulic manipulators, trajectory scaling algorithms that take account the volumetric flow limitations imposed by the hydraulic power unit have been presented in the last years [4, 5]. The firstly mentioned work solved this problem by formulating it as a quadratic program subject to an ℓ_1-norm constraint, while the latter work relied on path-velocity decomposition to uniformly scale down velocity references when the sum of demanded volumetric flows exceeded the pump capacity. While trajectory generation under pump saturation has been succesfully treated in these papers, for advanced motion control of hydraulic machinery, there is a need generate dynamically consistent trajectories that also include continuous acceleration profiles, which the previous velocity-level approaches are not able to provide.

Many works consider constrained motion planning for automated earthmoving excavators, in order to achieve minimum time solutions that respect, e.g., actuator force constraints [6, 7]. In this paper, we develop an online trajectory generation method that generates consistent, continuous trajectories while considering the limited on-board power. The main contribution is to combine an existing real-time trajectory scaling filter [8] and the pump saturation handlers [4, 5], and extend them to varying loading/velocity conditions by additionally considering the power-limiting (anti-stall) valve of the load-sensing hydraulic system. The functionality of the proposed power-limited on-line trajectory generator is verified by simulation experiments.

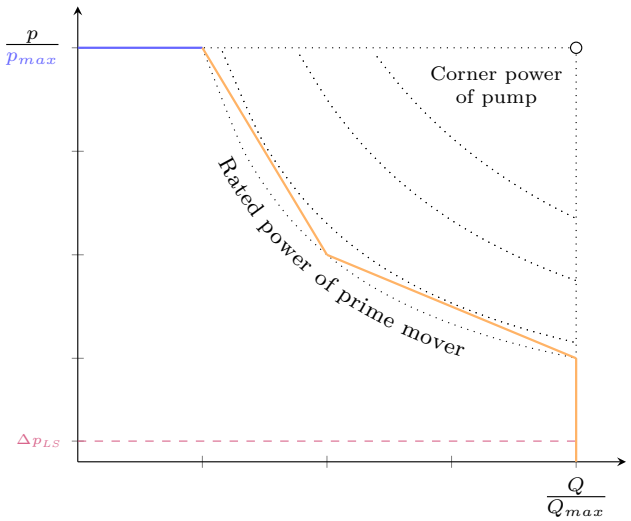

Figure 1. Ideal constant power hyperbola, and its discrete approximation by the power-limiting valve of the load-sensing pump controller.

2. BACKGROUND

Load-sensing (LS) hydraulic systems with variable displacement pumps and pressure-compensated valves are an industry standard for mobile hydraulic working machines, such as excavators. In these systems, the highest load signal determines the pressure level at the pump, and the delivered flow is adjusted according to demand. However, prime movers cannot always fully match the power that could demanded by the pump in some conditions. Often, the power required during standard operation is used as a dimensioning criteria for the rated power of the prime mover, which might only be a third of the corner power of the pump (see Fig. 1). This leads to a system where full actuator speed can only be maintained at a low loading condition, but in high loading condition, a power-limiting valve in the pump's LS controller starts to limit system performance in order to prevent the prime mover from stalling [10].

An LS controlled variable-displacement pump with power limiting is illustrated in Fig. 2. The pump displacement is controlled by a control piston, whose stroke is determined by the load-sensing (FR) and pressure-limiting (DR) spools, so that pump displacement is adjusted according to the required pressure and volumetric flow. If the combination of required pressure and flow exceeds the maximum power setting of the power-limiting (LR) valve, this relief valve reduces the load-sensing pressure signal acting on the FR spool, which in turn decreases the pump displacement in order to meet the power constraint.

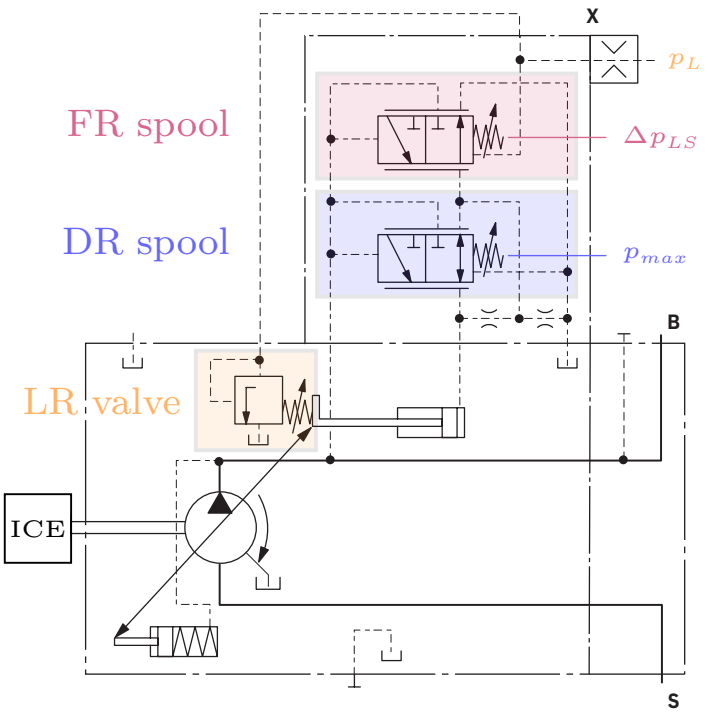

Figure 2. Hydraulic diagram of variable displacement axial piston pump Bosch Rexroth A10VSO with load-sensing controller [9]; the power-limiting (LR) valve is marked in yellow.

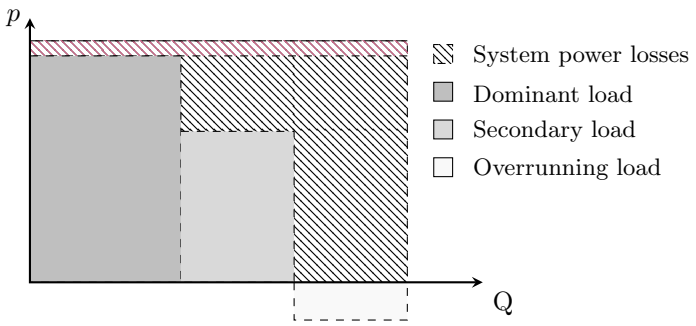

Figure 3. Example of power draw in a load-sensing hydraulic system.

Thus, the LR valve ensures that the hydraulic power at the pump does not exceed the rated power of the prime mover and cause stalling, while allowing the pump to be used at any operating point below the power hyperbola (e.g., to conduct fast movements under low loading conditions). This power-limiting mechanism used alongside an LS system is merely a relief valve and should be distinguished from actual closed-loop power controllers, which maintain the pump power constant and do not allow flow control below the power hyperbola.

This power limiting function in the LS system can be considered an additional global saturation nonlinearity of the hydraulic system, in addition to the flow/velocity and pressure/force limitations of the individual actuators. This power saturation in high loading conditions due to mismatch between the setting of the LR valve (rated power of the prime mover) and the corner power the pump should be considered in the trajectory generation of these machines, in order to prevent biasing and accumulation of controller tracking errors due to unrealizable motion references.

2.1. Power consumption in a load-sensing system

The load flow consumed by an actuator is calculated as

$$Q_L = |\dot{x}|A_c \tag{1}$$

where $\dot{x}$ is linear velocity of the piston and A_c is the effective pressurized area depending on the direction of motion. To take into account overrunning loads, the load pressure of each actuator is defined as

$$p_L = -\operatorname{sign}(f\dot{x})\frac{|f|}{A_c} \tag{2}$$

where f is the cylinder piston force. Here $f\dot{x} < 0$ signifies a resistive load (positive load pressure), while $f\dot{x} > 0$ signifies an overrunning load (negative load pressure) [11].

In a load-sensing hydraulic system, the power draw of the pump is determined by the highest load pressure p_L and the sum of load flows ΣQ_L (see Fig. 3) as

$$P = (\max(p_L) + \Delta p_{LS})\Sigma Q_L \tag{3}$$

where Δp_{LS} denotes the standby pressure at zero stroke, which is generally about 20 bar. The pump power draw is kept below a specified maximum power P_{max}, which is set by the pump manufacturer based on customer orders.

3. Proposed Scaling Algorithm

Using path-time decomposition, a desired trajectory can be expressed as a function of an n-times differentiable path $\mathbf{f}(\cdot)$, and a path coordinate s and its n time-derivatives as

$$\mathbf{q}_d = \mathbf{f}(s) \tag{4a}$$

$$\dot{\mathbf{q}}_d = \mathbf{f}'(s)\dot{s} \tag{4b}$$

$$\ddot{\mathbf{q}}_d = \mathbf{f}'(s)\ddot{s} + \mathbf{f}''(s)\dot{s}^2 \tag{4c}$$

The objective of a trajectory scaling system is to generate a value for the highest time derivative of s, the path coordinate, so that any physical constraints of the system are not violated. As a classic example, trajectory scaling systems use the dynamic model of the manipulator to determine in real-time the maximum feasible accelerations, given bounded actuation torques [12], or maximum feasible jerk by additionally considering the actuator dynamics [13].

For the purpose of trajectory scaling, we can simplify the dynamic equations by neglecting any closed-chain structures. In this case, we can express the inverse dynamic equation for a serial manipulator in joint-space as

$$\tau = \mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \tau_{ext} \tag{5}$$

where $\mathbf{H}(\mathbf{q})$ is the joint-space mass matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is the matrix of centrifugal and Coriolis forces, $\mathbf{g}(\mathbf{q})$ is a vector of gravitational torques, and τ_{ext} is a vector of external disturbance forces due to friction, payload and/or contact with the environment. Equation (5) can then be parametrized as

$$\tau(\ddot{s}, \dot{s}, s) = \mathbf{m}(s)\ddot{s} + \mathbf{b}(\dot{s}, s) \tag{6}$$

where

$$\mathbf{m}(s) = \mathbf{H}(\mathbf{q}_d)\mathbf{f}'(s) \tag{7}$$

$$\mathbf{b}(\dot{s}, s) = \mathbf{H}(\mathbf{q}_d)\mathbf{f}''(s) + \mathbf{C}(\mathbf{q}_d, \dot{\mathbf{q}}_d)\dot{\mathbf{q}}_d + \mathbf{g}(\mathbf{q}_d) + \tau_{ext} \tag{8}$$

From Eq. (6) upper and lower bounds for the path acceleration coordinate $\ddot{s}$ can be solved, which can be used as a part of a trajectory scaling algorithm to generate feasible motion references.

3.1. *Power limit of the system*

Analogously to the pump flow limit or actuator torque bounds, there exists a sum-power limit restricting the curvilinear accelerations of the system. In [14], the power limit was defined as the sum power of the actuators

$$\left|\mathbf{f}'(s)\dot{s}^\top\left(\mathbf{m}(s)\ddot{s} + \mathbf{b}(\dot{s}, s)\right)\right| \leq P_{max} \tag{9}$$

from which the upper and lower bounds for the path acceleration can be solved as

$$\bar{\ddot{s}} = \begin{cases} \dfrac{P_{max} - \mathbf{f}'(s)\dot{s}^\top\mathbf{b}(\dot{s}, s)}{\mathbf{f}'(s)\dot{s}^\top\mathbf{m}(s)} & \text{if } \dot{s} \geq 0 \\ \dfrac{-P_{max} - \mathbf{f}'(s)\dot{s}^\top\mathbf{b}(\dot{s}, s)}{\mathbf{f}'(s)\dot{s}^\top\mathbf{m}(s)} & \text{if } \dot{s} < 0 \end{cases} \tag{10a}$$

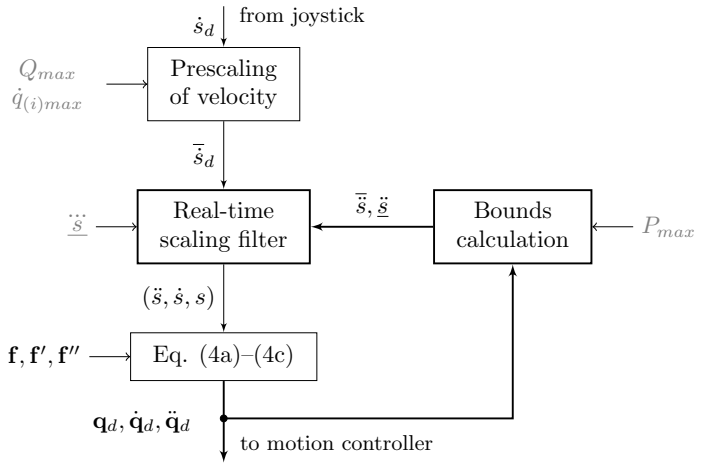

Figure 4. Block diagram of the proposed power-constrained trajectory scaling algorithm. $\dot{s}_d$ denotes user-requested velocity generated by joystick.

$$\underline{\ddot{s}} = \begin{cases} \dfrac{-P_{max} - \mathbf{f}'(s)\dot{s}^\top \mathbf{b}(\dot{s},s)}{\mathbf{f}'(s)\dot{s}^\top \mathbf{m}(s)} & \text{if } \dot{s} \geq 0 \\ \dfrac{P_{max} - \mathbf{f}'(s)\dot{s}^\top \mathbf{b}(\dot{s},s)}{\mathbf{f}'(s)\dot{s}^\top \mathbf{m}(s)} & \text{if } \dot{s} < 0 \end{cases} \tag{10b}$$

This formulation has the downside that it is not in line with the working principle of the load-sensing hydraulic system; for example, it models negative/assistive loads as generating power into the system, and does not consider the highest load pressure as the determining factor for pump power draw.

A correct definition of the power constraint for load-sensing systems would be defined as

$$\Sigma \left| \mathbf{f}'(s)\dot{s} \right| \max \left(\mathbf{m}(s)\ddot{s} + \mathbf{b}(\dot{s},s) \right) \leq P_{max} \tag{11}$$

which leads to constraints in the form

$$\bar{\ddot{s}} = \frac{P_{max} - \Sigma |\mathbf{f}'(s)\dot{s}| \mathbf{b}_i(\dot{s},s)}{\Sigma |\mathbf{f}'(s)\dot{s}| \mathbf{m}_i(s)} \tag{12a}$$

$$\underline{\ddot{s}} = \frac{-P_{max} - \Sigma |\mathbf{f}'(s)\dot{s}| \mathbf{b}_i(\dot{s},s)}{\Sigma |\mathbf{f}'(s)\dot{s}| \mathbf{m}_i(s)} \tag{12b}$$

where $\mathbf{m}_i(s)$ and $\mathbf{b}_i(\dot{s},s)$ denote the torque components corresponding to the highest load torque.

3.2. Real-time trajectory scaling filter

The block diagram of presented trajectory scaling method is illustrated in Fig. 4. As an input, the operator provides a desired path velocity command $\dot{s}_d$ by the hand-lever/joystick. This reference velocity is prescaled within admissible values $\bar{\dot{s}}_d$ determined by the capacity of the

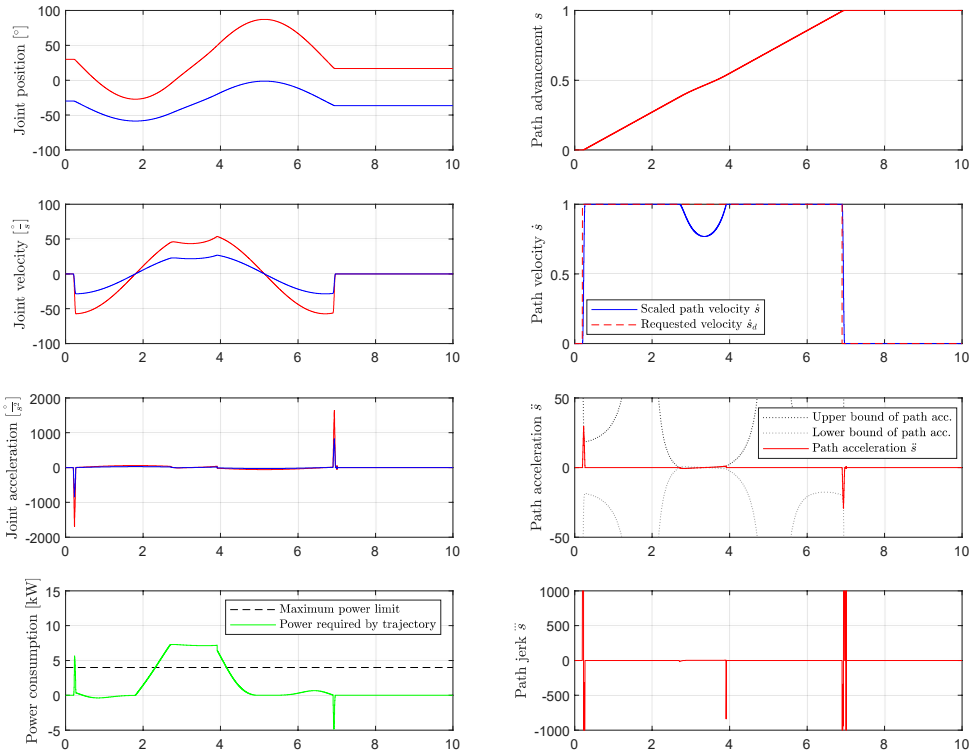

Figure 5. Results with constraints from [14].

pump Q_{max} and/or the maximum velocities $\dot{q}_{(i)max}$ of the individual actuators, determined by the flow saturation of the pressure-compensated valves.

Next, after prescaling of the user-provided velocity command, the desired velocity command $\bar{\dot{s}}_d$ is fed to a real-time suitable trajectory scaling filter. The previously calculated bounds $(\bar{\ddot{s}}, \underline{\ddot{s}})$ on the path acceleration, derived from the power limit, are fed as inputs to a trajectory scaling filter adopted from [8] (presented in detail in Appendix). In [8], this scaling filter was used to enforce bounds on robot control torques and their time derivatives. The same real-time scaling filter has also been recently employed in teleoperation systems to prevent inconsistencies between the master device and a hydraulic slave manipulator [15].

The working principle of the trajectory scaling filter is to generate $\dddot{s}$ so that $(\ddot{s}, \dot{s}, s)$ do not cause constraint violations. The $\dddot{s}$ generated by the filter passes through a chain of three integrators to yield $\ddot{s}$, $\dot{s}$ and s, from which the reference trajectory is computed according to (4a)–(4c). If a constraint violation is about to occur (e.g., if the power constraint cannot be maintained), the trajectory scaling filter is to temporarily gives up tracking the desired velocity path coordinate $\dot{s}_d$, leading into a deviation between $\dot{s}$ and $\dot{s}_d$.

4. Simulation Experiments

In order to demonstrate the effectiveness of the proposed method, a simulation study considering a planar 2R manipulator is carried out. Sinusoidal trajectories were applied for both joints, the path geometry being defined by

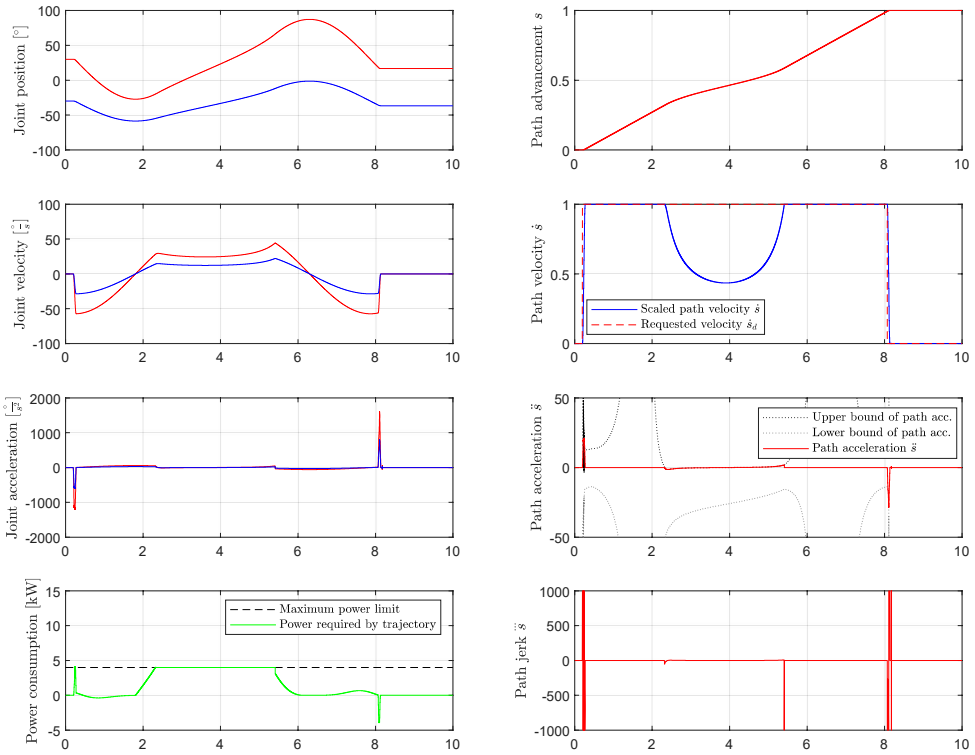

Figure 6. Results with proposed method.

$$\mathbf{f}(s) = \begin{bmatrix} \frac{\pi}{6} - \sin(s) \\ -\frac{\pi}{6} - 0.5\sin(s) \end{bmatrix} \tag{13a}$$

$$\mathbf{f}'(s) = \begin{bmatrix} -\cos(s) \\ -0.5\cos(s) \end{bmatrix} \tag{13b}$$

$$\mathbf{f}''(s) = \begin{bmatrix} \sin(s) \\ 0.5\sin(s) \end{bmatrix} \tag{13c}$$

In this study, a symmetric upper/lower bound for the path jerk coordinate $\dddot{s}$ is adopted and used as a tuning parameter representing the response dynamics of the load-sensing controller of the pump. The path jerk limit was set to $\underline{\dddot{s}} = -1000$, i.e., $|\dddot{s}| < 1000$.

4.1. Shortcomings of existing approach

We first employ the constraint (10a)–(10b) from [14] for the path geometry described by (13a)–(13c), using the trajectory scaling filter described in the previous section and illustrated in Fig. 4. The left side of Fig. 5 shows the generated motion trajectory $(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d)$ and the power consumption of this trajectory. The right side of Fig. 5 shows the path variable and its

time derivatives $(s, \dot{s}, \ddot{s}, \dddot{s})$, together with the upper/lower bounds for path acceleration $(\bar{\ddot{s}}, \underline{\ddot{s}})$ calculated from the power constraint.

When operating below the power hyperbola (i.e., at low loading conditions), only the prescaling according volumetric flow bound of the pump to takes place. In this case, the scaling filter is able to create continuous acceleration trajectories from the user-requested path velocity (compared to velocity-level trajectories in [5]).

On the other hand, if the power limit is about to be exceeded, the filter scales down path velocity until the power consumption of the trajectory is below the limitation set by the power hyperbola. As seen in Fig. 5, when violating the power limit, the trajectory filter gives up tracking the user-commanded path velocity $\dot{s}_d$. Thus, the net velocity of the manipulator, and consequently the sum of volumetric flows demanded from the pump, is always adjusted to the level determined by the LR valve setting.

However, when the trajectory is scaled using constraints (10a) and (10b), the definition of power consumption (shown in green in Fig. 5) does not correspond to the actual power draw on the pump in the load-sensing system. Thus, the definition of power consumption used in [14] could actually lead to violation of the power constraint in a load-sensing system, leading to biasing of the trajectory due to the mismatch between the filter constraint and the realizable control effort, affected by the constrained pump outflow determined by the LR valve setting.

4.2. Proposed approach

Next, for comparison, we employ the constraint (12a)–(12b) for the same path geometry. The resulting motion trajectory together with system power consumption, and the evolution of the path variable and its time derivatives, are illustrated in Fig. 6.

As can be seen from plot in the lower left corner, the bounds on system acceleration defined in the real-time scaling algorithm are capable to keep the power consumption of the trajectory within the specified upper limit P_{max}. Consequently, since meeting the power limit requires slowing down the net velocity of the trajectory in order to decrease volumetric flow demand from the pump, the trajectory duration is longer compared to the previous experiment.

As a summary, the proposed trajectory generator allows fast load lowering when needed, but prevents exceeding the system power limit during load lifting operations. Consequently, the trajectory scaling system prevents biasing of the reference trajectories and excessively increased controller tracking errors due to unfeasible motion commands due to limited power. If a sudden direction change of the load occurs (e.g., a shift from load lowering to lifting motion) and causes a drastic change in the highest load pressure, this might lead to temporary situtation where the commanded trajectory exceeds the power limit for short period of time before adapting to meet the power constraint.

5. CONCLUSION

An algorithm for online time scaling of motion trajectories for single-pump, multi-actuator load-sensing hydraulic systems subject to limited prime mover power was presented. The method takes advantage of an existing real-time suitable trajectory scaling filter, and extends it to handle the global saturation effects relevant for hydraulic mobile machinery, namely pump volumetric flow and prime mover power constraints. The method was shown to be able to generate consistent, continuous trajectories while satisfying the power limit.

If the power limit can not defined due to lack of an accurate dynamic model, the presented method can still be used to generate to acquire continuous acceleration references from flow-bounded velocity trajectories generated using the approach described in [5]. Integration of the proposed trajectory scaling algorithm to a momentum-based disturbance observer [16] or other means of real-time external force estimation to create a payload-aware/force-adaptive trajectory generator is left for future studies.

6. References

[1] K. L. Knierim and O. Sawodny, "Real-time trajectory generation for three-times continuous trajectories," in *2012 7th IEEE Conf. on Industrial Electronics and Applications (ICIEA)*, 2012, pp. 1462–1467.

[2] S. Fodor, C. Vazquez, and L. Freidovich, "Interactive on-line trajectories for semi-automation: Case study of a forwarder crane," in *2016 IEEE Int. Conf. on Automation Science and Engineering (CASE)*, pp. 928–933.

[3] A. R. Enes and W. J. Book, "Optimizing point to point motion of net velocity constrained manipulators," in *49th IEEE Conference on Decision and Control (CDC)*, 2010, pp. 6415–6420.

[4] J. Wanner and O. Sawodny, "Tool-center-point control of a concrete pump using constrained quadratic optimization," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 1, pp. 382–396, 2021.

[5] S. Lampinen, J. Niemi, and J. Mattila, "Flow-bounded trajectory-scaling algorithm for hydraulic robotic manipulators," in *2020 IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics (AIM)*, 2020, pp. 619–624.

[6] S. Yoo, C.-G. Park, S.-H. You, and B. Lim, "A dynamics-based optimal trajectory generation for controlling an automated excavator," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 224, no. 10, pp. 2109–2119, 2010.

[7] Y. Yang, J. Pan, P. Long, X. Song, and L. Zhang, "Time variable minimum torque trajectory optimization for autonomous excavator," *ArXiv*, vol. abs/2006.00811, 2020.

[8] C. Guarino Lo Bianco and O. Gerelli, "Trajectory scaling for a manipulator inverse dynamics control subject to generalized force derivative constraints," in *2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009, pp. 5749–5754.

[9] Bosch Rexroth AG, "Axial piston variable pump A10VSO," [Online] Accessed January 24, 2022.

[10] M. Vukovic, R. Leifeld, and H. Murrenhoff, "Reducing fuel consumption in hydraulic excavators – a comprehensive analysis," *Energies*, vol. 10, no. 5, 2017.

[11] A. Bonavolonta, C. Dolcin, P. Marani, E. Frosina, and A. Senatore, "Comparison of energy saving and recovery systems for hydraulic mobile machines," *AIP Conference Proceedings*, vol. 2191, 2019.

[12] O. Dahl and L. Nielsen, "Torque-limited path following by online trajectory time scaling," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 5, pp. 554–561, 1990.

[13] M. Tarkiainen and Z. Shiller, "Time optimal motions of manipulators with actuator dynamics," in *1993 Proceedings IEEE Int. Conf. on Robotics and Automation (ICRA)*, 1993.

[14] F. G. Flores and A. Kecskeméthy, "Time-optimal path planning along specified trajectories," in *Multibody System Dynamics, Robotics and Control*, H. Gattringer and J. Gerstmayr, Eds. Springer, 2013.

[15] F. Zhang, J. Zhang, M. Cheng, and B. Xu, "A flow-limited rate control scheme for the master-slave hydraulic manipulator," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 5, pp. 4988–4998, 2022.

[16] B. Son, C. Kim, C. Kim, and D. Lee, "Expert-emulating excavation trajectory planning for autonomous robotic industrial excavator," in *2020 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 2656–2662.

Appendix: Trajectory scaling filter

The adopted trajectory scaling filter is a modified version of the original filter presented in [8]; the modified filter equations are provided below. The filter was modified in two ways; firstly, to allow the upper and lower bounds of $\ddot{s}$ both to be negative, and secondly, to enforce motion to stop when the user-provided velocity command is zero.

$$z^+ = \operatorname{sign}(\bar{\ddot{s}})\frac{\bar{\ddot{s}} - \ddot{s}_d}{T\,\underline{\dddot{s}}} \tag{14}$$

$$z^- = \frac{\underline{\ddot{s}} - \ddot{s}_d}{T\,\underline{\dddot{s}}} \tag{15}$$

$$\dot{z}^+ = -\operatorname{sign}(\bar{\ddot{s}})\operatorname{ceil}(z^+)\left(z^+ - \frac{\operatorname{ceil}(z^+)-1}{2}\right) \tag{16}$$

$$\dot{z}^- = \operatorname{ceil}(-z^-)\left(-z^- - \frac{\operatorname{ceil}(-z^-)-1}{2}\right) \tag{17}$$

In this paper $\dot{s}_d$ is generated by the user from the joystick, while $\ddot{s}_d$ is set to zero.

$$z = \frac{1}{T\alpha}\left|\frac{\dot{s}-\dot{s}_d}{T} + \frac{\ddot{s}-\ddot{s}_d}{2}\right| \tag{18}$$

$$\gamma = \begin{cases} z^+ & \text{if } z < z^+ \\ z^- & \text{if } z > z^- \\ z & \text{otherwise} \end{cases} \tag{19}$$

$$m = \operatorname{floor}\left(\frac{1+\sqrt{1+8|\gamma|}}{2}\right) \tag{20}$$

$$\dot{\tilde{z}} = -\frac{\gamma}{m} - \frac{m-1}{2}\operatorname{sign}(\gamma) \tag{21}$$

$$\dot{z} = \begin{cases} \frac{\ddot{s}-\ddot{s}_d}{T|\alpha|} & \text{if } z \geq 0 \text{ and } \frac{\ddot{s}-\ddot{s}_d}{T|\alpha|} \leq \dot{\tilde{z}} \\ \frac{\ddot{s}-\ddot{s}_d}{T|\alpha|} & \text{if } z < 0 \text{ and } \frac{\ddot{s}-\ddot{s}_d}{T|\alpha|} \geq \dot{\tilde{z}} \\ \frac{\ddot{s}-\ddot{s}_d}{T|\beta|} + \frac{\alpha+\beta}{|\beta|}\left(\frac{m-1}{2} + \frac{|\gamma|}{m}\right) & \text{otherwise} \end{cases} \tag{22}$$

We can define

$$\sigma = \begin{cases} 0 & \text{if } \dot{s}_d = 0 \text{ and } \dot{s} = 0 \\ \dot{z} - \dot{\tilde{z}} & \text{otherwise} \end{cases} \tag{23}$$

A switching control law for the path jerk is then defined as

$$\dddot{s} = \begin{cases} \bar{\dddot{s}}\min(1,\sigma) & \text{if } \sigma \geq 0 \\ \underline{\dddot{s}}\max(-1,\sigma) & \text{if } \sigma < 0 \end{cases} \tag{24}$$

where $\underline{\dddot{s}} < 0$ defines the lower bound for the jerk coordinate.

Acknowledgment

This work was supported in part by the STREAM project funded by Shift2Rail Joint Undertaking (JU), established under the European Union's Horizon 2020 Framework Programme for Research and Innovation, under Grant Agreement No. 101015418. The content of this work reflects only the author's view, and that the Shift2Rail JU is not responsible for any use that may be made of the information it contains.

Biographies

Lionel Hulttinen received B.Sc. (Tech.) and M.Sc. (Tech.) degrees in Automation Engineering from Tampere University of Technology (TUT) in 2017 and 2018, respectively. His research interests include system identification and model-based control of hydraulic robotic manipulators.

Jouni Mattila. received the M.Sc. (Eng.) and Dr. Tech. degrees from Tampere University of Technology (TUT), Tampere, Finland, in 1995 and 2000, respectively. He is currently a Professor of machine automation with the unit of Automation Technology and Mechanical Engineering, Tampere University. His current research interests include machine automation, nonlinear model-based control of robotic manipulators, and energy-efficient control of heavy-duty mobile manipulators.