Flow-limited path-following control of a double Ackermann steered hydraulic mobile manipulator

Lionel Hulttinen and Jouni Mattila
Innovative Hydraulics and Automation (IHA)
Tampere University, Tampere, Finland
firstname.lastname@tuni.fi

***Abstract*—This paper presents an approach to realize tool path-following control of double Ackermann steered wheeled mobile manipulators. Path-following is achieved by time scaling of motion trajectories when the vehicle-manipulator becomes subject to actuator flow bounds and/or tracking errors arising from limitations of the hydraulic actuation system. Our strategy combines 1) a platform path-following controller with explicit velocity bounds calculated from the a priori known pump flow limits, and 2) a manipulator trajectory scaling method based solely on the monitored position tracking error. The proposed method effectively allows the manipulator to readjust its position when facing unexpected disturbances, by propagating the scaling factor based instantaneous tool center point position errors to the platform path-following controller, to yield coordinated advancement on a predefined path. Simulation results are provided to verify the effectiveness of the method.**

I. INTRODUCTION

Wheeled nonholonomic mobile manipulators (WNMMs), consisting of an articulated manipulator mounted on top of a steerable mobile platform, can comprise of close to ten or more actuators. In order to orchestrate all these actuators in a synchronized manner to yield desired end-effector motion, sophisticated control algorithms are needed in order to avoid numerous actuator nonlinearities from ruining control performance. In the context of hydraulic mobile manipulators suitable for rough terrain operations, both the driveline and the manipulator arm hydraulic circuits are subject to pump flow bounds, limiting the realizable end-effector velocities. For example, steep turns of the platform combined with too high drive velocities could lead to excessive volumetric flow demands for the outer-wheel hydraulic motors. Consequently, valve or pump saturation could cause the vehicle to deviate from its intended path, jeopardizing the path-following task. Due to the before-mentioned difficulties arising from the hydraulic system limitations, accurate coordinated control of these multi-actuator systems is a challenging task, and control actions need to be programatically adjusted based on prior knowledge about available flow capacity of the pump.

A closely related classic problem in robotics is generation of dynamically feasible trajectories while considering limited actuation torques. A typical solution is to utilize the rigid-body dynamic model of the manipulator to solve for maximum admissible acceleration at every time instant, based on a priori known actuator torque- or power generating capabilities [1]. The general principle of these algorithms is to downscale velocity and acceleration commands at the expense of increased path traversal time, to prevent position tracking errors from growing excessively. However, these model-based approaches suffer from parameter uncertainty, and what is more, they make assumptions about the underlying controller structure, relying on the common computed-torque or similar control laws. To ensure general applicability, the scaling method should preferably be independent of the underlying low-level motion controllers, allowing the use of simple, e.g., individual joint-wise PD controllers. This is especially important in the context of heavy-duty hydraulic working machines, which are known to suffer from uncertain dynamics and various hard nonlinearities, including pump saturation and valve deadzone. As an alternative to the full-model-based time scaling approaches, some sources have proposed tracking error based schemes to deal with the actuation constraints of both manipulator arms [3] as well as mobile robotic platforms [4].

The subproblem related to the driveline actuation limitations of hydraulic WNMMs has been previously addressed using individual-wheel velocity-bounded path-following control [5], in which the required angular velocities of each individual hydraulic motor are calculated and, if necessary, the required platform velocity commands are scaled down to an admissible level. Meanwhile, bounded velocity control of hydraulically actuated manipulators has been treated more widely. In [6], path-constrained trajectory generation with individual link-wise velocity constraints for a forestry boom was presented. In [7], a simulation study was conducted about flow-bounded coordinated tool-center-point (TCP) control for a flexible concrete pump manipulator, by imposing a global constraint acting on the sum-flow of all actuators. However, there seems to be a lack of studies what comes to coordinated control of hydraulically actuated vehicle-manipulators, considering both the locomotion of the mobile base as well as motion control of the manipulator arm simultaneously. Most studies related to coordinated control of hydraulic mobile manipulators consider front-end loaders or similar machines, which have simple nonredundant planar mechanisms, instead of redundant long-reach manipulator arms.

In this work, we employ a TCP path-following control scheme for a double Ackermann steered wheeled hydraulic mobile manipulator. Instead individual wheel and/or link velocity limits, we consider pump saturation by a global sum-

flow constraint acting on the required wheel velocities. We combine the explicit bounded-velocity solutions for platform control from [5, 8] to a tracking error based manipulator time scaling scheme from [3], to yield a path-following control strategy for mobile manipulators. The proposed mobile manipulator path-following scheme is tested in simulation to validate its effectiveness.

II. PLATFORM KINEMATICS AND CONTROL

The system considered in this work is a nonholonomic mobile platform with four-wheel steering (4WS), the steering mechanism being a *double Ackermann steering*. Despite its resemblance to car-like steering, in which only front wheels are steered, 4WS has differing kinematic characteristics due to its ability to steer both pairs of wheels separately. This allows sharper turns of the chassis, and consequently, improved maneuverability owing to a smaller admissible turning radius. Another notable difference compared to other types of hydrostatically driven mobile platforms, such as skid-steered tracked vehicles (e.g., excavators and bulldozers) with dual path HSTs [9], or articulated-frame-steered vehicles (e.g., forestry forwarders and front-end loaders), is that velocity of each individual wheel is separately controllable.

Each wheel is actuated by a hydraulic hub motor with nominal displacement of $100\ \mathrm{cm}^3$ per revolution, connected to the wheel centroid via a gear with a transmission ratio of 17.7, while steering axles are actuated by symmetric cylinders (see Fig. 1). The limited maneuverability of the chassis due to the steering angle limits can be described using the concept of instantaneous center of rotation (ICR), illustrated in Fig. 2. For this particular platform, the minimum turning radius r_{min} equals to 3.75 m for the outer wheels, or about 1 m for the inner wheels, since the maximum steering angle of each wheel is approximately $\pm 45°$. More detailed specifications about the platform can be found in [5].

A. Steering kinematics of the mobile platform

While it would be possible to use the double Ackermann steering in various locomotion modes, such as crab steering, in this study the double Ackermann is used in a symmetric, *negative 4WS configuration*. This means front and rear wheel steer angles are mirrored (i.e., front and rear wheels are oriented in equal angles, but in opposite directions with respect to the chassis-fixed frame).

Neglecting the steering dynamics and assuming ideal Ackermann steering, the platform steering kinematics can be approximated by a simplistic bicycle model [10], reducing both pairs of wheels to a single pair of imaginary middle wheels located between the actual front and rear wheels. The relation between the time derivative of chassis configuration $\dot{q}_b = (\dot{\phi}, \dot{x}, \dot{y})$ and velocities measured in the chassis-fixed frame $\{\mathrm{b}\}$ is expressed as follows

$$\begin{bmatrix} \omega_{bz} \\ v_{bx} \\ v_{by} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{x} \\ \dot{y} \end{bmatrix} \tag{1}$$

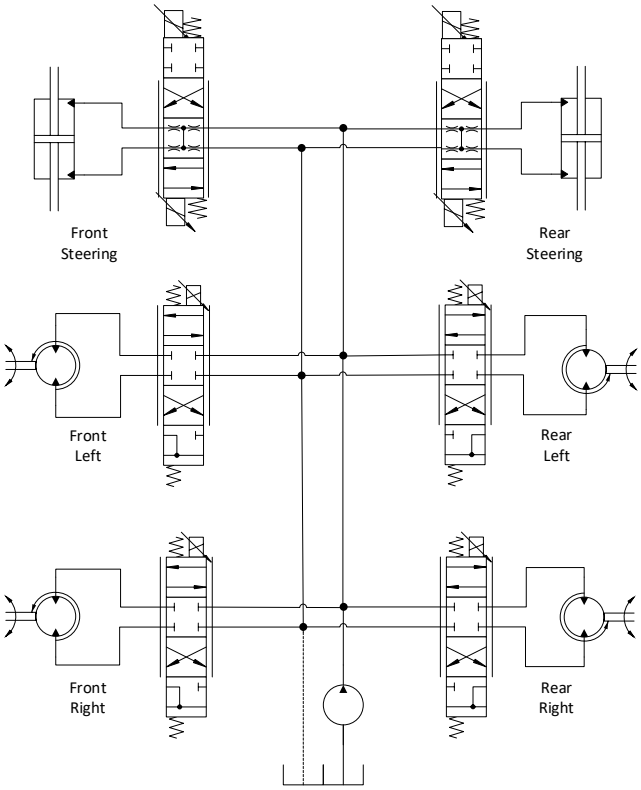

Fig. 1. Simplified hydraulic diagram of the 4WS drivetrain.

1) Inverse kinematics: Required steering angles ψ_{id} of individual wheels are resolved in the steering frames $\{\mathrm{s}_i\}$ at the end of the drive axes, while required wheel velocities are resolved in the wheel frames $\{\mathrm{m}_i\}$. This is due to an additional velocity compensation term, which arises from the offcentering of the wheels from the steering pivot point (and is therefore dependent on the steering angle). The required linear velocities in the steering frames are calculated as

$$\begin{bmatrix} v_{s1x} \\ \vdots \\ v_{s4y} \end{bmatrix} = \begin{bmatrix} l_{1y} & 1 \\ \vdots & \vdots \\ -l_{4x} & 0 \end{bmatrix} \begin{bmatrix} \omega_{\mathrm{d}} \\ v_{\mathrm{d}} \end{bmatrix} \tag{2}$$

where l_{1y} is chassis half-width and l_{4y} is chassis half-length. Required steering angles for each wheel are extracted from the steering frame velocities as

$$\psi_{\mathrm{id}} = \tan^{-1}\left(\frac{v_{siy}}{v_{six}}\right) \tag{3}$$

Compensating for the wheel offset radius d, the required linear velocities in the wheel center frames are calculated as

$$\begin{bmatrix} v_{m1x} \\ \vdots \\ v_{m4y} \end{bmatrix} = \begin{bmatrix} v_{s1x} \\ \vdots \\ v_{s4y} \end{bmatrix} + \begin{bmatrix} d\sin(\psi_1) & 0 \\ \vdots & \vdots \\ -d\cos(\psi_4) & 0 \end{bmatrix} \begin{bmatrix} \omega_{\mathrm{d}} \\ v_{\mathrm{d}} \end{bmatrix} \tag{4}$$

from which the required hydraulic motor rotational speeds can be extracted as

$$n_{\mathrm{id}} = \mathrm{sign}(v_{\mathrm{d}}) \frac{\sqrt{v_{mix}^2 + v_{miy}^2}}{r_{wheel}} 2\pi k_{gear} \tag{5}$$

where r_{wheel} is the wheel radius and k_{gear} is the gear ratio.

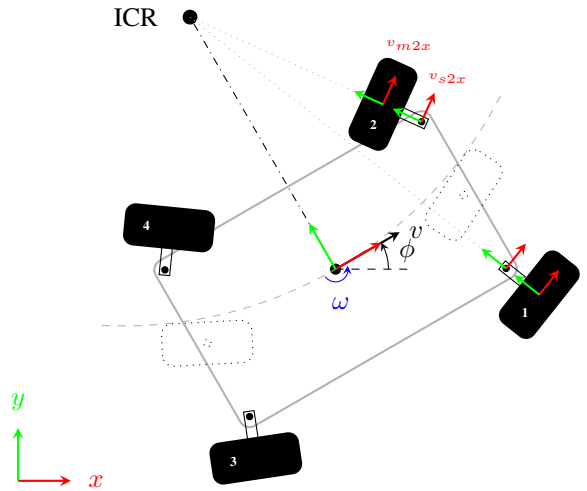

Fig. 2. Mobile platform with ideal symmetric four-wheel steering.

B. Flow-limited platform control

1) Path-velocity decomposition: The bounded velocity path-following controller is based on principle of path-velocity decomposition, which allows to decouple motion references from task geometry. Given a desired path $p(s) = (x_d, y_d)$ defined with respect a normalized path progress variable $s \in [0, 1]$, using the chain rule, the modified velocity reference can be expressed as a function of s and $\dot{s} > 0$ as

$$(\dot{x}_d, \dot{y}_d) = \frac{\partial s}{\partial t}\frac{\partial}{\partial s}p(s) = p'(s)\dot{s} \tag{6}$$

The heading reference can then be determined from $p'(s)$ as

$$\phi_d(s) = \text{atan2}(y'_d(s), x'_d(s)) \tag{7}$$

The motion references can be calculated from the expressions above as

$$\omega' = \frac{\partial \phi_d}{\partial s} = \frac{x'_d(s)y''_d(s) - y'_d(s)x''_d(s)}{x'_d(s)^2 + y'_d(s)^2} \tag{8}$$

$$v' = \left\| \frac{\partial p(s)}{\partial s} \right\| = \sqrt{x'_d(s)^2 + y'_d(s)^2} \tag{9}$$

2) Path-following control law [8]: For platform path-following, we employ a slightly modified version of the the control law from [5, 8]. The platform configuration errors are given by

$$\begin{bmatrix} e_\phi \\ e_x \\ e_y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi_d) & \sin(\phi_d) \\ 0 & -\sin(\phi_d) & \cos(\phi_d) \end{bmatrix} \begin{bmatrix} \phi - \phi_d \\ x - x_d \\ y - y_d \end{bmatrix} \tag{10}$$

Obviously, position tracking and orientation tracking are not independently achievable for nonholonomic vehicles, e.g., in order to compensate for chassis lateral errors, one must temporarily introduce heading errors, and vice cersa. The controller uses the following σ-function to generate a suitable approach angle so to compensate for lateral position error e_y

$$\sigma(e_y) = \sin^{-1}\frac{k_2 e_y}{|e_y| + \epsilon} \tag{11}$$

where $0 < k_2 \leq 1$ and $\epsilon > 0$ are tuning variables. Using the modified (approach angle compensated) orientation reference for heading feedback control yields the following control law

$$\begin{bmatrix} \omega_\text{d} \\ v_\text{d} \end{bmatrix} = \begin{bmatrix} w' + \dfrac{k_3(e_\phi - \sigma(e_y))}{\cos\sigma(e_y) + k_1 e_x} \\ \dfrac{v'}{\cos\sigma(e_y) + k_1 e_x} \end{bmatrix} \dot{s} \tag{12}$$

which is linear in the scaling factor $\dot{s}$. Choosing the scaling factor as

$$\dot{s} = \frac{v_{max}}{v'}(\cos\sigma(e_y) + k_1 e_x) \tag{13}$$

renders the control law explicit in terms of the magnitude of tangential velocity. This allows solving for the maximum admissible velocity $v_\text{d} = v_{max}$ separately at every time instant, based on analytical model-based equations [8].

3) Flow limit based velocity scaling: Our task is to define v_{max} so to prevent system actuation limits from becoming violated. To attain a set of feasible velocity commands under the pump-imposed flow constraint, required hydraulic motor rotational speeds $n_{i\text{d}}$ are mapped into required volumetric flows as $Q_{i\text{d}} = |n_{i\text{d}}|V_{mi}$, where V_{mi} is the displacement of the ith motor. Now, actuator flow demands have to satisfy a sum-flow constraint in the form $\sum Q_{i\text{d}} \leq Q_{max}$, where Q_{max} is the total available flow budget and the sum of all drive motor flow demands is calculated using equations (2–5) as

$$Q_{i\text{d}} = |n_{i\text{d}}(\tfrac{v'}{\cos\sigma(e_y)+k_1 e_x}, w' + \tfrac{k_3(e_\phi - \sigma(e_y))}{\cos\sigma(e_y)+k_1 e_x})|V_{mi} \tag{14}$$

To ensure the sum-flow requirement of all motors does not exceed the pump capacity Q_{max}, the velocity command in the control law (12) is determined at every time instant using (14), which yields

$$v_{max} = v'\frac{Q_{max}}{\sum Q_{i\text{d}}} \tag{15}$$

One could also limit individual motor speeds to not exceed a predefined maximum allowed speed n_{max}, by calculating maximum velocities in a fashion similar to (15) and choosing the minimum value out of all the maximum velocities, as done in [5]. Additionally, if the arm cylinders were pressure-compensated and connected on the same circuit, one could model them as predictable velocity sources and take into account their volumetric flow demands similar to above as $Q_{i\text{d}} = |\dot{x}_{i\text{d}}|A_{ci}$, where A_{ci} is the pressurized area of the ith cylinder (for asymmetric cylinders, the larger area should be used for conservativeness, as in [7]). Then, one could for example prioritize the arm movement instead of base movement, by allocating only a limited portion of the total flow budget for the drive motors.

In this study, we only consider the platform flow bounds analytically. For considering arm actuation limitations, we take a different approach, relying on purely tracking error based trajectory adaptation similar to [3].

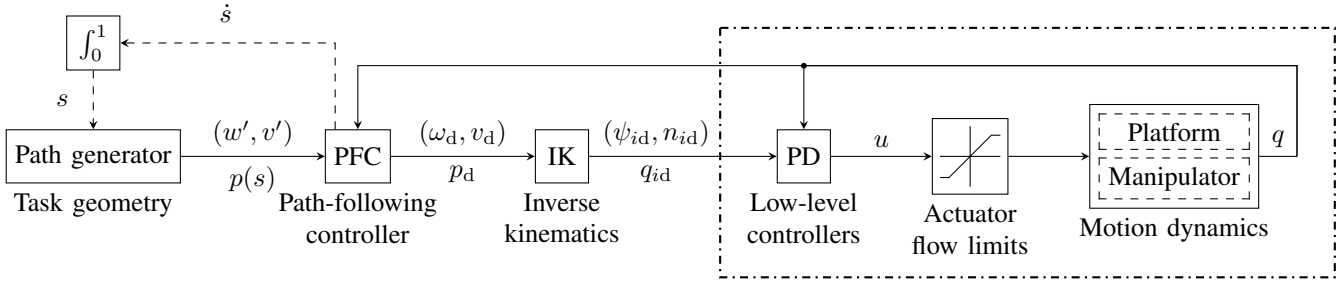

Fig. 3. Block diagram of the presented flow-bounded path-following control scheme.

III. PATH-FOLLOWING CONTROL STRATEGY

In this simulation study, the manipulator mounted on the 4WS platform is a RRRP crane (see Fig. 4), a topology commonly found in forestry and loader cranes; dimensions of the boom can be found in [11]. The pose of the end-effector in world coordinates is expressed as $p = f(q)$, where the configuration space of the mobile manipulator is $q = \begin{bmatrix} q_b & q_a \end{bmatrix} = \begin{bmatrix} \phi & x & y & q_1 & q_2 & q_3 & d_4 \end{bmatrix}$. To acquire differential kinematics, the base and manipulator Jacobians can be combined into a composite nonholonomic mobile manipulator Jacobian, expressed as

$$\dot{p} = J_b G u + J_a \dot{q}_a \tag{16}$$

where $J_a = \frac{\partial f(q)}{\partial q_a}$ is the analytical Jacobian of the arm, and $G(\phi)$ is a mapping to feasible motions that explicitly models the non-holonomity of the base [12].

For motion planning of the arm-plus-base system, two notable sources of redundancy exist; one is the intrisic redundancy of the planar arm (q_2, q_3, d_4), and another is between the orientation of the mobile base and the arm rotation joint (ϕ, q_1). To mitigate cross-coupling due to redundancies, some sources employ task decomposition to the mobile manipulator Jacobian, by using the redundant degrees-of-freedom to fulfill one or more user-specified tasks. Augmenting the Jacobian with an additional objective functions can render it invertible, but might at the same time introduce artificial (algorithmic) singularities [12], and hence the choice of auxiliary tasks while avoiding task singularities becomes cumbersome. What is more, input delays and sluggish actuator dynamics can be detrimental to the application of Jacobian-based motion generation schemes for relatively slow hydraulic machinery with heterogeneous actuator capabilities.

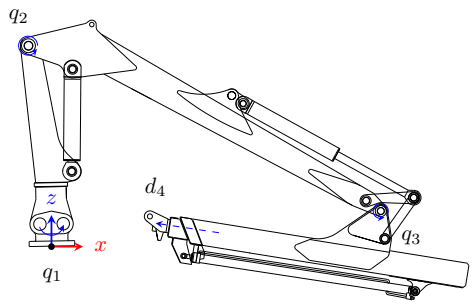

Fig. 4. Planar view of the RRRP manipulator mounted on the platform.

Alternatively, the mobile manipulation problem can be decomposed into two independent subtasks defined in different coordinates: the macro subtask, in which the platform must follow a path on a plane, and the micro subtask, in which the TCP of the manipulator must track a point which defined is relative to its base (i.e., the platform) [13].

A. Arm controller

It has been previously shown that path-following control for hydraulic platforms can be implemented accurately [5]. Building on top on this, we can resort to a decoupled approach to mobile manipulation, which consists of a primary controller that controls platform position in world coordinates, and a secondary controller that controls arm position in base (platform-fixed) coordinates, aiming to compensate for instantaneous platform tracking errors. Given a desired end-effector position ${}^b_w p_d$ in world coordinates, and measured platform configuration $({}^b_w R, {}^b_w p)$, the desired position in world coordinates can be transformed into platform coordinates by

$${}_b p_d = {}^b_w R^{-1} ({}^b_w p_d - {}^b_w p)$$

The desired position in platform coordinates is then fed to an analytic inverse kinematics solution to obtain desired joint angles as $q_{ad} = f^{-1}({}_b p_d)$. If we use a fixed extension length d_4, a standard closed-form solution for the planar RRP arm exists [6], and the only remaining redundancy is between the base locomotion and arm rotation. This approach does not take into account interaction between the base and manipulator, but is justifiable if the base locomotion is slow compared to the manipulator arm movement. The remaining practical challenge of this approach is how to define the hierachy between the two controllers so to slow down platform movement whenever the manipulator cannot follow the designed path.

B. Tracking error based velocity scaling

In addition to analytical maximum velocity solutions that prevent pump saturation, it is necessary to adapt path speed based on instantaneous position tracking errors of the TCP. We can augment (15) with an additional tracking error dependent term so to slow down the platform in case the TCP following errors grow excessively. The tracking error based scaling factor is propagated to the platform path-following controller, helping to synchronize platform and

arm movements in case of disturbances. We can define an additional scaling term as

$$\delta(||e||) = \begin{cases} 1 & \text{if } ||e|| < e_{ok} \\ \dfrac{e_{ok}}{||e||} & \text{otherwise} \end{cases}$$

where $||e|| = ||{}_b p_b - f(q_a)||$ is the norm of (filtered) TCP position errors in expressed in chassis-fixed coordinates, and e_{ok} is a threshold value of choice to be tuned based on the desired tradeoff between time expenditure and accuracy requirement of the task. The platform control law (12) then assumes a modified form

$$\begin{bmatrix} \omega_\mathrm{d} \\ v_\mathrm{d} \end{bmatrix} = \begin{bmatrix} (w' + \frac{k_3(e_\phi - \sigma(e_y))}{\cos\sigma(e_y) + k_1 e_x})\dot{s} \\ v_{max}\delta(||e||) \end{bmatrix} \tag{17}$$

and the modified time derivative of the path coordinate s is governed by

$$\dot{s} = \frac{v_{max}\delta(||e||)}{v'}(\cos\sigma(e_y) + k_1 e_x) \tag{18}$$

where v_{max} is calculated analytically based on the hydraulic flow limitations of the steering and drive circuit.

IV. SIMULATION CASES

A kinematic simulation model of the mobile vehicle-manipulator system was used to validate the proposed method. Actuator dynamics were modeled as first-order transfer functions at the velocity inputs. The test path was defined as a composite C2-continuous cubic Bézier curve in 2D plane (shown in Fig. 5), combined with a parametric sine wave $z_d(s) = \sin(8\pi s)$ to generate TCP height reference.

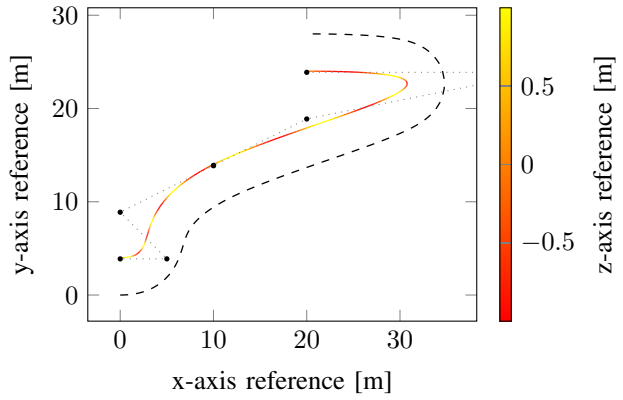

Fig. 5. Composite cubic Bézier curve and control points illustrated, showing TCP reference path (—), and its offset curve for platform reference (- -).

After defining the curve which the mobile manipulator TCP has to follow in world coordinates, we need to calculate an offset curve that the platform-fixed frame should follow. Calculating an offset curve might in some cases form self-intersecting loops creating "pockets", which should be trimmed/filtered out by the path planner. These loops are not are not considered in this study, instead the desired platform configuration is always fixed relative to the desired tool center point position.

A. Case 1: Flow limit based velocity solution

The purpose of this test is to demonstrate that the sum-flow constraint is not violated when using the flow-bounded maximum velocity solution, i.e, that pump saturation is avoided. In this test, the error-based time scaling was not activated. The pump has a maximum volumetric output of $Q_{max} = 63\frac{dm^3}{min}$, which will be used as a parameter for the flow-limited controller (12); in real world, some margin of safety should be used to account for inefficiencies.

Left side of Fig. 6 shows a plot of the flow demands and TCP tracking errors for the test. The initial platform position was intentionally set different from the path starting point. One can notice overshoot and oscillation in TCP position in the beginning, due to the platform advancing despite incorrect initialization. While the plaform trajectory eventually stabilizes, the TCP trajectory is unable to follow its height reference, leading to vast tracking errors. The resulting tracking errors can be used as a reference for the results of the next test, where platform velocity is adapted based on monitored TCP errors.

B. Case 2: Tracking error based adaptation

The error-based adaptation gain appearing in (17) was tuned as $\gamma = 20$. Compared to the previous experiment, the controller acts significantly more smoothly in the beginning, fixing the initial pose error before advancing the platform reference (see right side of Fig. 6). The task takes more time to be completed, but with significantly less errors (after having corrected the initialization error).

By slowing down the path velocity when tool errors grow, the controller is able to maintain the base joint orientation nearly constant during path execution. In comparison, when using the maximum velocity solution as-is, the manipulator lags behind in steep curves and has to compensate by quick movements, involving aggressive movements of the base rotation joint. Considering the low damping of hydraulic rack-and-pinion or similar type of actuators [14], it is beneficial to keep the base rotation joint as fixed as possible and only actuate the planar arm to realize path tracking.

In future studies, the tool reference point relative to the platform configuration should be changed dynamically during the task. This would allow partially compensating for platform immobility in steep curvatures (e.g., when the designed offset curve forms intersecting pockets that the non-holonomic platform is unable to follow), using the redundant degrees of freedom provided by the telescopic extension and the base rotation joints.

V. SUMMARY

In this paper an approach to realize path-following for hydraulic mobile manipulators was presented. In principle, in addition to calculating the maximum admissible platform velocity based on volumetric flow constraints imposed by the pump, the algorithm slows down the platform velocity based on the instantaneous manipulator tracking error, thus allowing the manipulator arm to readjust its position when facing unexpected disturbances. The presented solution has

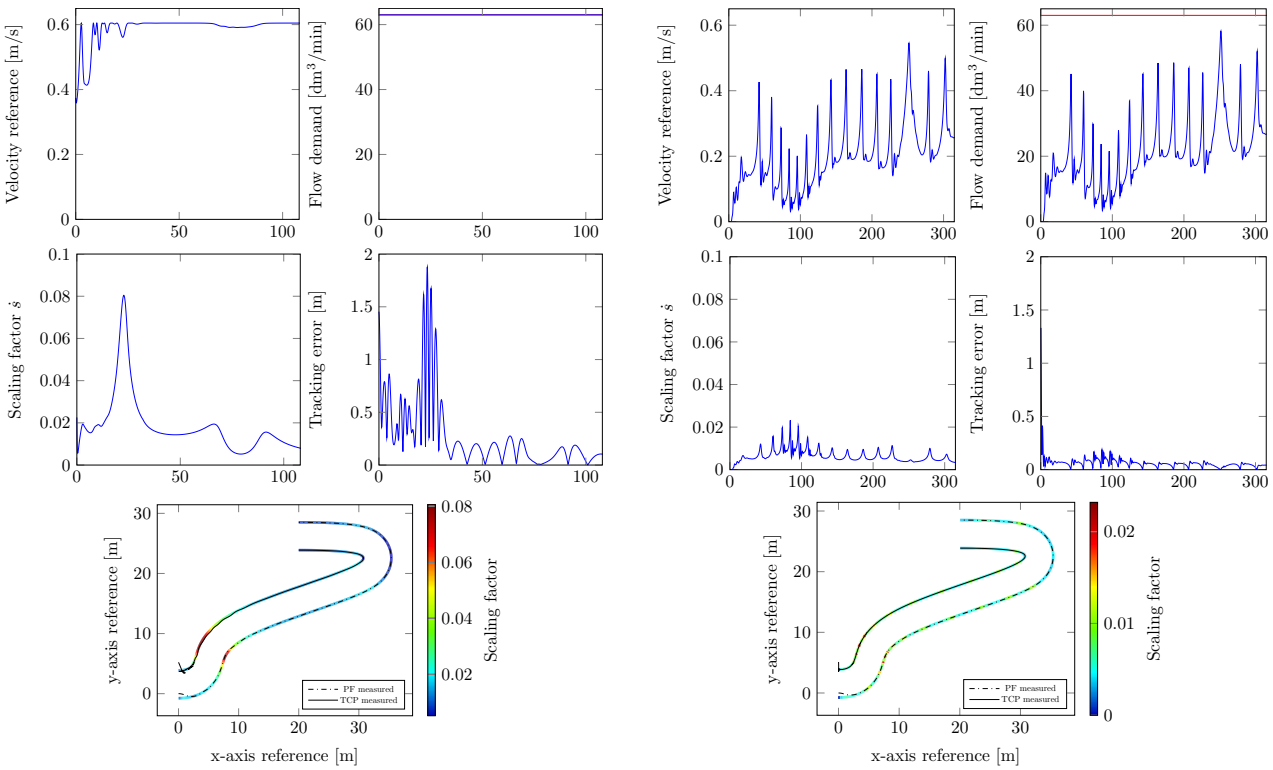

Fig. 6. Simulation experiments considering only the analytical flow limits without error-based adaptation (*left*), and with error-based adaptation in addition to the analytical flow limits (*right*).

the benefit of being solely based on tracking error, and thus being independent of the underlying low-level motion controllers. Moreover, it is agnostic of the type of disturbance that the TCP following error is caused by, whether wheel slippage of the platform due on an icy surface, or a sudden change of direction of the load force causing one of the boom actuators to lag behind.

Future work could include studies about energy-efficient coordinated control of heavy-duty mobile manipulators. As outlined in [9], hydro-mechanical system losses and geometrical path properties are interconnected, and larger turning radii are preferrable in terms of minimizing power consumption of the drivetrain. When combined with energy-optimal redundancy resolution for the planar arm [11], one could study the volumetric flow cost (and, consequently energy efficiency) of arm movement compared to that of platform movement given different task geometries. This information could be used to add intelligence at the path planning level, to generate an active path deformation strategy that would proactively exploit the redundancy between the manipulator and the platform in a way to minimize platform movement, with the aim of more energy-optimal task execution.

References

[1] O. Dahl, "Torque limited path following by on-line trajectory time scaling," 1989, Licentiate Thesis, Lund Institute of Technology.

[2] 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.

[3] E. Szádeczky-Kardoss and B. Kiss, "Tracking error based on-line trajectory time scaling," in *2006 International Conference on Intelligent Engineering Systems*, pp. 80–85.

[4] ——, "On-line trajectory time-scaling to reduce tracking error," in *Intelligent Engineering Systems and Computational Cybernetics*, J. A. T. Machado, B. Pátkai, and I. J. Rudas, Eds. Springer, 2009, pp. 3–14.

[5] H. Liikanen, M. Aref, R. Oftadeh, and J. Mattila, "Path-following controller for 4wds hydraulic heavy-duty field robots with nonlinear internal dynamics," in *2019 10th IFAC Symposium on Intelligent Autonomous Vehicles (IAV)*, vol. 52, no. 8, 2019, pp. 375–380.

[6] D. Ortiz Morales, S. Westerberg, P. X. La Hera, U. Mettin, L. Freidovich, and A. S. Shiriaev, "Increasing the level of automation in the forestry logging process with crane trajectory planning and control," *Journal of Field Robotics*, vol. 31, no. 3, pp. 343–363, 2014.

[7] J. Wanner and O. Sawodny, "Tool-center-point control of a flexible link concrete pump with hydraulic limitations using quadratic programming," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, 2019, pp. 561–566.

[8] R. Oftadeh, R. Ghabcheloo, and J. Mattila, "A time-optimal bounded velocity path-following controller for generic wheeled mobile robots," in *2015 IEEE/RSJ Int. Conf. on Robot. Autom. (ICRA)*, pp. 676–683.

[9] T. Mononen, M. Aref, and J. Mattila, "Analysis of the effects of path properties on autonomous motion control of a hydraulic tracked vehicle," in *2018 25th IEEE International Conference on Mechatronics and Machine Vision in Practice (M2VIP)*, vol. 52, no. 8, pp. 98–106.

[10] A. Kelly, *Mobile Robotics: Mathematics, Models, and Methods*. Cambridge University Press, 2013.

[11] J. Nurmi and J. Mattila, "Global energy-optimal redundancy resolution of hydraulic manipulators: Experimental results for a forestry manipulator," *Energies*, vol. 10, no. 5, 2017.

[12] A. De Luca, G. Oriolo, and P. R. Giordano, "Kinematic modeling and redundancy resolution for nonholonomic mobile manipulators," in *Proc. 2006 IEEE Int. Conf. on Robot. Autom. (ICRA)*, pp. 1867–1873.

[13] D. S. Alicja Mazur, "On path following control of nonholonomic mobile manipulators," *International Journal of Applied Mathematics and Computer Science*, vol. 19, no. 4, pp. 561–574, 2009.

[14] P. Mustalahti and J. Mattila, "Nonlinear model-based controller design for a hydraulic rack and pinion gear actuator," in *BATH/ASME 2018 Symposium on Fluid Power and Motion Control*.