IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 14, 2020, accepted July 21, 2020, date of publication July 31, 2020, date of current version August 14, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3013382

A Collaborative Robotic Approach to Autonomous Pallet Jack Transportation and Positioning

PIETRO BALATTI[1,2], (Member, IEEE), FABIO FUSARO[1,3], NICOLA VILLA[1], EDOARDO LAMON[1,2], (Member, IEEE), AND ARASH AJOUDANI[1], (Member, IEEE)

[1]HRI[2] Laboratory, Istituto Italiano di Tecnologia, 16163 Genoa, Italy
[2]Department of Information Engineering, University of Pisa, 56126 Pisa, Italy
[3]Department of Electronics, Information, and Bioengineering, Politecnico di Milano, 20133 Milan, Italy

Corresponding author: Pietro Balatti (pietro.balatti@iit.it)

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Programme Socio-Physical Interaction Skills for Cooperative Human-Robot Systems in Agile Production (SOPHIA) under Grant 871237, and in part by the Amazon Research Awards 2018.

ABSTRACT This paper proposes a novel loco-manipulation control framework for the execution of complex tasks with kinodynamic constraints using mobile manipulators. As a representative example, we consider the handling and re-positioning of pallet jacks (or lifts/carriers with similar characteristics) in unstructured environments. This task is associated with significant challenges in terms of locomotion, due to the mobility constraints that are imposed by their limited kinematics while moving, and manipulation, due to the existence of dynamic uncertainties while grasping and handling of pallet jacks. To tackle these challenges, our solution enables the robotic platform to autonomously reach a pallet jack location while avoiding the obstacles, and to detect and manipulate its handle by fusing the perception and the contact force data. Subsequently, the transportation of the pallet jack is achieved through a whole-body impedance controller and a trajectory planner which takes into account the mobility constraints of the robot-pallet jack chain. We demonstrate the effectiveness of the proposed solution in reaching and displacing the pallets to desired locations through simulation studies and experimental results. While these results reveal with a proof-of-concept the effectiveness of the proposed framework, they also demonstrate the high potential of mobile manipulators for relieving human workers from such repetitive and labor intensive tasks. We believe that this extended functionality can contribute to increasing the usability of mobile manipulators in different application scenarios.

INDEX TERMS Autonomous agents, field robots, manipulation planning, path planning for manipulators.

I. INTRODUCTION

The role of automation in logistics has been growing more and more in the last decades. This is due to the increasing demands imposed by the ''e-commerce revolution'' [1], whose constant progress cannot be responded by the decreasing available workforce in aging societies. The target activities with automation potential are mainly those that do not require a significant level of human expertise, such as handling, sorting, storage, and transportation, typical of warehouse environments.

Recent developments in robotized and automated warehouse systems have presented advantages such as full-time availability, and savings on space, labor costs, and on other operational costs, such as heating and lighting [2]. Nevertheless, the existing solutions require large initial investments, since they often need to be applied to customized production processes. Additionally, these solutions are not flexible enough, so they can be hardly employed in dynamically changing environments such as small and medium sized warehouses. For this reason, only 5% of the existing warehouses worldwide have been semi-automated [3].

One fundamental roadblock towards extending the automation capacity of the warehouses has been posed by the automatic yet flexible management of the pallets, which demand for their recognition, manipulation, and relocation (transportation and positioning through pallet jacks). Current solutions have aimed to tackle this through bespoke machines, designed for specific purposes. For instance, in [4], a voice-commandable prototype has been introduced, to enable human supervisors to use speech and pen-based gestures to assign tasks to the forklift. Other similar works

The associate editor coordinating the review of this manuscript and approving it for publication was Pengcheng Liu.

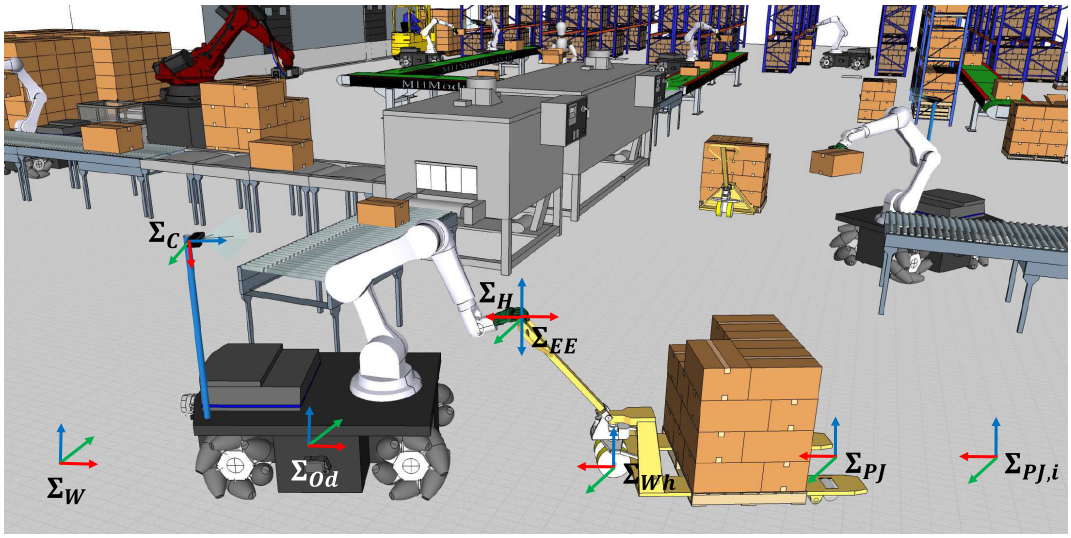


FIGURE 1. Concept illustration of a mobile collaborative robot while transporting and positioning of a manual pallet jack inside a warehouse. The depicted reference frames represent world (W), platform odometry (Od) and end-effector (EE), camera (C), pallet jack handle (H), wheels (Wh), current pose (PJ) and initial pose (PJ, i).

focused on forklifts control [5], [6], pallet manipulation [7], [8], and pallet recognition [9], [10], to separately tackle the issues related to the pallet repositioning tasks. Although such systems can be beneficial in large and fully autonomous warehouses, however, they impose additional costs and equipment waiting times to the others. In addition, since most of the small and medium sized warehouses have been designed and created for humans, the introduction of such automation systems would require a major re-design of the spaces, affecting their economic sustainability.

Collaborative robots (cobots), on the other hand, have demonstrated the potential to execute a large range of manufacturing and logistics tasks alongside their human partners [11], and add certain flexibility levels to the processes [12]. In logistics scenarios in particular, they can reconfigure from autonomous picking and palletizing [13] to navigating in warehouse environments [14], [15].

Towards understanding the potential of cobots in addressing the pallet management problem, in this work, we propose a novel autonomous pallet jack repositioning framework by means of a MObile Collaborative robotic Assistant (MOCA) [16]. MOCA is a mobile manipulator which is composed of an omnidirectional base, a robotic arm, and an underactuated soft robotic hand.

To the best of the authors' knowledge, all the existing frameworks tackling the pallet transportation and positioning problem are related to autonomous forklifts development [5], [17]. Instead, in this work, we propose a solution designed for a *multi-purpose platform*, MOCA, that has a high potential to perform several logistics tasks. It is important to notice that, this work does not claim to compare the performances that can be achieved with unmanned forklifts, but aims to provide a solution to this problem with a general purpose robot that can be employed for many applications. In small and medium-sized enterprises this is a crucial point, since they may not be able to afford multiple single-purpose, costly machines. On the other hand, they can exploit the flexibility of a multi-purpose machine, relieving the cost burden. In fact, the reconfigurability of this platform has already demonstrated to be effective also in other industrial scenarios, i.e., in human-collaborative manufacturing [18] and palletizing [19] tasks, and in teleoperation applications [16].

The latter work presented the first MOCA control framework, composed by a whole-body Cartesian impedance controller, exploited in manipulation tasks, and a decoupled velocity controller for the robotic base, for locomotion tasks. Nevertheless, some tasks such as pulling a pallet jack, would require coupled loco-manipulation skills to be executed. To achieve this, a uniform control strategy that allows us to exploit various types of arm and mobile base coordinated movements, becomes a crucial requirement.

To overcome these limitations and to improve the state of the art for the control of MOCA-like platforms, we present a prioritized weighted whole-body inverse dynamics control algorithm, obtained by solving an optimization problem. The presented controller allows to regulate the interaction forces with the environment, i.e. to shape the way the operational wrench is projected into joint torques, ensuring at the same time that the dynamic equations of the motion are satisfied.

Another challenge consists in dealing with the kinematic constraints imposed by the limited motion of the pallet jack when physically coupled and navigated by MOCA. In fact, standard pallet jacks provide one degree of freedom (DoF) for rotation and one DoF for translation, while the MOCA platform can freely move in space. Hence, to generate feasible paths for the pallet jack, we developed a kinematically-constrained motion planning algorithm for MOCA, in which

we can specify the pallet jack initial and desired position on the *x-y plane* and its orientation around the *z-axis*.

The proposed pallet repositioning framework for MOCA features also a navigation algorithm with obstacle avoidance capability that makes the robot reach the location of the pallet jack that needs to be relocated, and a perception algorithm to detect the pallet jack handle pose, supported by a contact force unit to ensure the handle's correct localization and grasping.

The main contributions of this paper are:

- a novel method for solving the repositioning of standard pallet jacks through a multi-purpose platform, that can be also employed for collaborative tasks, due to its high flexibility.
- a novel prioritized weighted whole-body inverse dynamics control algorithm, to enrich the coupled loco-manipulation skills required by the task.
- the integration of multiple state of the art components as the trajectory planning algorithms for both kinematically-constrained trajectories and collision-free paths, and a perception algorithm to detect the pallet jack handle.

Note that, this paper only focuses on the complex problem of pallet jack recognition and re-positioning, while the loading and unloading of the pallets on pallet jacks are out of the scope of this work.

II. METHODS

The presented framework can be subdivided into four operational modules, that include a *Pallet jack reaching* unit to navigate towards a desired pallet jack location, a *Handle detection* unit to localize more accurately the pallet jack handle, a *Handle preparation* unit to grasp and pull down the pallet jack handle, and a *Pallet jack repositioning* unit to move the pallet jack to a new location in the workspace. In this Section we deal with the theoretical components that are required for the implementation of the above-mentioned modules, whose details are illustrated in Sec. III.

To achieve an autonomous execution of the depicted tasks in a logistic scenario, we bestowed several components on MOCA, namely control, perception, and planning. The control framework's goal consists in enabling MOCA to perform loco-manipulation tasks autonomously, or in collaboration/coexistence with humans. Hence, a whole-body impedance controller [20], [21] is developed to respond to the flexibility needs of a dynamic warehouse environment, with the potential to ensure human safety.

A. MOCA WHOLE-BODY IMPEDANCE CONTROL

The whole-body impedance controller computes the high level torque references for the torque-controlled manipulator and for the velocity-controlled mobile base. The torque references are transformed in feasible velocity commands by means of an admittance interface.

The mobile base is driven by four Omnidirectional wheels, that allows the platform to avoid nonholonomic constraints, and has $m = 3$ DoFs, 2 for the translation on the plane, referred as x and y and 1 for the plane normal axis, i.e. the rotation around the *z-axis*. The resulting dynamics of the mobile base, with virtual joints $\boldsymbol{q}_v \in \mathbb{R}^m$, can be described by:

$$\boldsymbol{M}_{adm}\ddot{\boldsymbol{q}}_v^{des} + \boldsymbol{D}_{adm}\dot{\boldsymbol{q}}_v^{des} = \boldsymbol{\tau}_v^{vir} + \boldsymbol{\tau}_v^{ext}, \tag{1}$$

where $\boldsymbol{M}_{adm} \in \mathbb{R}^{m\times m}$ and $\boldsymbol{D}_{adm} \in \mathbb{R}^{m\times m}$ are the virtual inertial and virtual damping, $\dot{\boldsymbol{q}}_v^{des} \in \mathbb{R}^m$ is the input velocity sent to the mobile platform and $\ddot{\boldsymbol{q}}_v^{des} \in \mathbb{R}^m$ its desired acceleration, $\boldsymbol{\tau}_v^{ext} \in \mathbb{R}^m$ and $\boldsymbol{\tau}_v^{vir} \in \mathbb{R}^m$ are the external and the virtual torque, respectively.

The dynamics of a n-DoFs torque-controlled arm can be formulated as:

$$\boldsymbol{M}_r(\boldsymbol{q}_r)\ddot{\boldsymbol{q}}_r + \boldsymbol{C}_r(\boldsymbol{q}_r, \dot{\boldsymbol{q}}_r) + \boldsymbol{g}_r(\boldsymbol{q}_r) = \boldsymbol{\tau}_r + \boldsymbol{\tau}_r^{ext}, \tag{2}$$

where $\boldsymbol{q}_r \in \mathbb{R}^n$ is the joint angles vector, $\dot{\boldsymbol{q}}_r \in \mathbb{R}^n$ and $\ddot{\boldsymbol{q}}_r \in \mathbb{R}^n$ its velocity and acceleration respectively, $\boldsymbol{M}_r \in \mathbb{R}^{n\times n}$ is the symmetric and positive definite inertial matrix of the arm, $\boldsymbol{C}_r \in \mathbb{R}^n$ is the Coriolis and centrifugal force, $\boldsymbol{g}_r \in \mathbb{R}^n$ is the gravity vector, $\boldsymbol{\tau}_r \in \mathbb{R}^n$, and $\boldsymbol{\tau}_r^{ext} \in \mathbb{R}^n$ are the commanded torque vector and external torque vector, respectively.

Before retrieving the formulation of the whole-body dynamics of the robot, we make a few assumptions that will simplify the model. First, thanks to the high gains of the low level velocity controller, it is possible to neglect the effect of the arm motion onto the base motion, i.e. $\ddot{\boldsymbol{q}}_v \approx \ddot{\boldsymbol{q}}_v^{des}$. Second, the motion planner and controller ensures smooth trajectory generation and tracking for the mobile platform, minimizing the effects of the base motion onto the arm motion. Thus, we can write the following whole-body decoupled dynamics:

$$\begin{pmatrix} \boldsymbol{M}_{adm} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{M}_r \end{pmatrix}\begin{pmatrix} \ddot{\boldsymbol{q}}_v \\ \ddot{\boldsymbol{q}}_r \end{pmatrix} + \begin{pmatrix} \boldsymbol{D}_{adm} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C}_r \end{pmatrix}\begin{pmatrix} \dot{\boldsymbol{q}}_v \\ \dot{\boldsymbol{q}}_r \end{pmatrix} + \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{g}_r \end{pmatrix} = \begin{pmatrix} \boldsymbol{\tau}_v^{vir} \\ \boldsymbol{\tau}_r \end{pmatrix} + \begin{pmatrix} \boldsymbol{\tau}_v^{ext} \\ \boldsymbol{\tau}_r^{ext} \end{pmatrix}, \tag{3}$$

that can be summarized by

$$\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{C}\dot{\boldsymbol{q}} + \boldsymbol{g} = \boldsymbol{\tau} + \boldsymbol{\tau}^{ext}. \tag{4}$$

The operational space equations of motion [22] can be obtained by substituting the differential kinematics relationship in (4):

$$\dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{5}$$

and its derivative:

$$\ddot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{6}$$

where $\boldsymbol{x} \in \mathbb{R}^6$ is the operational coordinates vector, $\boldsymbol{q} = \begin{bmatrix}\boldsymbol{q}_v & \boldsymbol{q}_r\end{bmatrix}^T \in \mathbb{R}^{(n+m)}$, and $\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{6\times(n+m)}$ represents the analytical Jacobian matrix.

The result is:

$$\boldsymbol{\Lambda}(\boldsymbol{x})\ddot{\boldsymbol{x}} + \boldsymbol{\mu}(\boldsymbol{x}, \dot{\boldsymbol{x}})\dot{\boldsymbol{x}} + \boldsymbol{F}_{\boldsymbol{g}} = \boldsymbol{F} + \boldsymbol{F}^{ext}, \tag{7}$$

where $\boldsymbol{F}_{\boldsymbol{g}}$, $\boldsymbol{F}$, and $\boldsymbol{F}^{ext} \in \mathbb{R}^6$ are the gravity, the input, and the external force, respectively. $\boldsymbol{\Lambda}(\boldsymbol{x}) \in \mathbb{R}^{6\times6}$ represents Cartesian inertia and $\boldsymbol{\mu}(\boldsymbol{x}, \dot{\boldsymbol{x}}) \in \mathbb{R}^{6\times6}$ represents Cartesian

Coriolis and centrifugal matrix, obtained with the following equations:

$$\boldsymbol{\Lambda}(\boldsymbol{x}) = \boldsymbol{J}(\boldsymbol{q})^{-T}\boldsymbol{M}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})^{-1}, \tag{8}$$

$$\boldsymbol{\mu}(\boldsymbol{x},\dot{\boldsymbol{x}}) = \boldsymbol{J}(\boldsymbol{q})^{-T}(\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}) - \boldsymbol{M}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})^{-1}\dot{\boldsymbol{J}}(\boldsymbol{q}))\boldsymbol{J}(\boldsymbol{q})^{-1}, \tag{9}$$

where $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{(n+m)\times(n+m)}$ is the joint-space inertia matrix, $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \mathbb{R}^{(n+m)\times(n+m)}$ is the joint-space Coriolis and centrifugal matrix. The static relationship between joint torques in (4) and operational forces in (7) is

$$\boldsymbol{\tau}^{ext} = \boldsymbol{J}^T\boldsymbol{F}^{ext}. \tag{10}$$

The whole-body impedance controller models the operational wrench $\boldsymbol{F}^{ext} \in \mathbb{R}^6$ as a mass-spring-damper system, in the following way:

$$\boldsymbol{F}_{ext} = \boldsymbol{\Lambda}_d\ddot{\tilde{\boldsymbol{x}}} + \boldsymbol{D}_d\dot{\tilde{\boldsymbol{x}}} + \boldsymbol{K}_d\tilde{\boldsymbol{x}} \tag{11}$$

where $\boldsymbol{\Lambda}_d \in \mathbb{R}^{6\times6}$, $\boldsymbol{D}_d \in \mathbb{R}^{6\times6}$ and $\boldsymbol{K}_d \in \mathbb{R}^{6\times6}$ are the desired Cartesian inertia, damping and stiffness, respectively. The resulting torque input, proposed by *Wu et al.* [16], is the linear combination of the torque due to the impedance controller

$$\boldsymbol{\tau}_{imp} = \boldsymbol{g}(\boldsymbol{q}) + \boldsymbol{J}^T(\boldsymbol{q})(\boldsymbol{\Lambda}(\boldsymbol{x})\ddot{\boldsymbol{x}} + \boldsymbol{\mu}(\boldsymbol{x},\dot{\boldsymbol{x}})\dot{\boldsymbol{x}} - \boldsymbol{D}_d\dot{\tilde{\boldsymbol{x}}} - \boldsymbol{K}_d\tilde{\boldsymbol{x}}) \tag{12}$$

and the null-space torque,

$$\boldsymbol{\tau}_{null} = \boldsymbol{N}(\boldsymbol{q})\boldsymbol{\tau}_0, \tag{13}$$

where

$$\boldsymbol{\tau}_0 = -\boldsymbol{D}_0\dot{\boldsymbol{q}} - \boldsymbol{K}_0(\boldsymbol{q} - \boldsymbol{q}_0), \tag{14}$$

where $\boldsymbol{q}_0$ is a desired joint configuration, $\boldsymbol{K}_0 \in \mathbb{R}^{(n+m)\times(n+m)}$ and $\boldsymbol{D}_0 \in \mathbb{R}^{(n+m)\times(n+m)}$ are the desired joint-space stiffness and damping, and

$$\boldsymbol{N}(\boldsymbol{q}) = \boldsymbol{I} - \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{\Lambda}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})^{-1} \tag{15}$$

is the *dynamically consistent null-space projector* proposed by Khatib [22]. In the prioritized controller, the first task $\boldsymbol{\tau}_{imp}$ consists in maintaining a desired Cartesian impedance behavior, while the second task $\boldsymbol{\tau}_{null}$, with lower priority with respect to the first task, consists in keeping a desired, low-gain, joint-space impedance $\boldsymbol{\tau_0}$. This task, in the case it is not in conflict with the first one, allows the robot to avoid singular configurations and joint limits. This is achieved by choosing a suitable value for the desired joint configuration $\boldsymbol{q}_0$, i.e. both setting the vector far enough from each relative joint limit and ensuring its manipulability far enough from 0. The main advantage of this controller consists in the fact that we can plan Cartesian forces (and hence trajectories, thanks to the impedance controller) in the end-effector frame, but it is not possible to regulate how the Cartesian task is projected at the joint level. In other words, we would like to exploit the redundancy provided by MOCA DoFs, to determine which joints have to be mainly used, in some defined operational modes, such as manipulation and locomotion. In manipulation mode, for instance, we would like to favor the arm motion in close-proximity manipulation tasks, by projecting the Cartesian force mainly on the arm joints, and similarly for the locomotion mode, favoring the mobile base mobility when navigating in free spaces.

To achieve these behaviors, the robot joint torques vector $\boldsymbol{\tau} \in \mathbb{R}^n$ is computed according to a prioritized weighted inverse dynamics algorithm. The controller is obtained by solving the problem[1] of finding the torque vector $\boldsymbol{\tau}$ closest to some desired $\boldsymbol{\tau_0}$ that realizes the operational forces $\boldsymbol{F}$, according to the norm induced by the positive definite weighting matrix $\boldsymbol{W} \in \mathbb{R}^{(n+m)\times(n+m)}$,

$$\begin{aligned} &\min_{\boldsymbol{\tau}} \frac{1}{2}\|\boldsymbol{\tau} - \boldsymbol{\tau_0}\|^2_{\boldsymbol{W}} \\ &\text{s.t. } \boldsymbol{J}\boldsymbol{\tau} = \boldsymbol{F} \end{aligned} \tag{16}$$

where $\|\boldsymbol{\tau} - \boldsymbol{\tau_0}\|^2_{\boldsymbol{W}}$ indicates the weighted norm $(\boldsymbol{\tau} - \boldsymbol{\tau_0})^2$ $\boldsymbol{W}(\boldsymbol{\tau} - \boldsymbol{\tau_0})$ and

$$\bar{\boldsymbol{J}} = (\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{J}^T)^{-1}\boldsymbol{J}\boldsymbol{M}^{-1} \tag{17}$$

is the *dynamically consistent pseudo-inverse of* $\boldsymbol{J}(\boldsymbol{q})$, and the constraint

$$\bar{\boldsymbol{J}}\boldsymbol{\tau} = \boldsymbol{F} \tag{18}$$

is the general relationship between the generalized joint torques and the operational forces [23]. We point out that torque in (10) is just one solution of (18). The solution of problem (16) is retrieved by means of the Lagrangian multipliers method. Consider the Lagrangian cost function

$$\mathcal{L} = \frac{1}{2}(\boldsymbol{\tau} - \boldsymbol{\tau_0})^T\boldsymbol{W}(\boldsymbol{\tau} - \boldsymbol{\tau_0}) + \boldsymbol{\lambda}^T(\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{J}^T\boldsymbol{F} - \boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{\tau}), \tag{19}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^6$ is the vector of unknown multipliers of the constraint. The solution requires to satisfy the following:

$$\begin{aligned} \left(\frac{\partial\mathcal{L}}{\partial\boldsymbol{\tau}}\right)^T &= \boldsymbol{W}(\boldsymbol{\tau} - \boldsymbol{\tau_0}) - \boldsymbol{M}^{-1}\boldsymbol{J}^T\boldsymbol{\lambda} = \boldsymbol{0}, \\ \left(\frac{\partial\mathcal{L}}{\partial\boldsymbol{\lambda}}\right)^T &= \boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{J}^T\boldsymbol{F} - \boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{\tau} = \boldsymbol{0}. \end{aligned} \tag{20}$$

The desired optimal solution is obtained by combining the mentioned conditions:

$$\begin{aligned} \boldsymbol{\tau} = \boldsymbol{W}^{-1}\boldsymbol{M}^{-1}\boldsymbol{J}^T\boldsymbol{\Lambda}_{\boldsymbol{W}}\boldsymbol{\Lambda}^{-1}\boldsymbol{F} + \\ + (\boldsymbol{I} - \boldsymbol{W}^{-1}\boldsymbol{M}^{-1}\boldsymbol{J}^T\boldsymbol{\Lambda}_{\boldsymbol{W}}\boldsymbol{J}\boldsymbol{M}^{-1})\boldsymbol{\tau_0}, \end{aligned} \tag{21}$$

where

$$\boldsymbol{\Lambda}_{\boldsymbol{W}} = \boldsymbol{J}^{-T}\boldsymbol{M}\boldsymbol{W}\boldsymbol{M}\boldsymbol{J}^{-1} \tag{22}$$

can be regarded as the *weighted Cartesian inertia*, analogous to the Cartesian inertia in (8). In (21), it is possible to recognize the two tasks in the controller, the Cartesian force task $\boldsymbol{F}$ with higher priority and the joint-space torque $\boldsymbol{\tau_0}$ projected in the null space of the first task through a *dynamically consistent null-space projector*, analogously to

[1] For the sake of readability, the dependencies on $\boldsymbol{q}$ and $\boldsymbol{x}$ are dropped from now on.

(12), (13), and (15). The formulation in (21) contains the prioritized tasks, but the input torques of the controller needs also to compensate for the other terms that are not present in the formulation of the problem in (16), like gravity and Coriolis terms, such as in (12). Moreover, $\boldsymbol{F}$ and $\boldsymbol{\tau_0}$ can be computed according to (11) and (14) respectively.

Only the choice of $\boldsymbol{W}$ is left. The structure of the matrix can be defined as

$$\boldsymbol{W}(\boldsymbol{q}) = \boldsymbol{H}^T \boldsymbol{M}^{-1}(\boldsymbol{q}) \boldsymbol{H} \tag{23}$$

where $\boldsymbol{H} \in \mathbb{R}^{(n+m)\times(n+m)}$ is the tunable positive definite weight matrix of the controller. In particular, in this paper, $\boldsymbol{H}$ is diagonal and dynamically selected depending on the task. A possible choice is represented by

$$\boldsymbol{H} = \begin{bmatrix} \eta_B \boldsymbol{I}_{m\times m} & \boldsymbol{0}_{m\times n} \\ \boldsymbol{0}_{n\times m} & \eta_A \boldsymbol{I}_{n\times n} \end{bmatrix} \tag{24}$$

where m and n are the dimensions of the joint space of the base and the arm, respectively, and $\eta_B, \eta_A > 0$ constant scalar values. For instance, to obtain higher mobility of the arm with respect to the base, we set $\eta_B > \eta_A$.

Noteworthy, if $\boldsymbol{H} = \boldsymbol{I}$, then $\boldsymbol{W} = \boldsymbol{M}^{-1}$ and $\boldsymbol{\Lambda_W} = \boldsymbol{\Lambda}$. These simplifications leads to the well-known [22]:

$$\boldsymbol{\tau} = \boldsymbol{J}^T \boldsymbol{F} + (\boldsymbol{I} - \boldsymbol{J}^T \boldsymbol{\Lambda} \boldsymbol{J} \boldsymbol{M}^{-1}) \boldsymbol{\tau_0}. \tag{25}$$

This means that the control algorithm proposed by Wu *et al.* in [16] is equal to the solution of problem (16), if weighting the norm with the inverse of the joint-space inertia matrix $\boldsymbol{M}$.

B. KINEMATICALLY-CONSTRAINED TRAJECTORY PLANNING FOR HOLONOMIC ROBOTS

In this paragraph we want to address the problem of generating trajectories for an holonomic robot that is pulling a wheeled object. The goal consists in designing trajectories for the end-effector of the robot, given the initial and goal pose of the pulled object, and the kinematics constraints that the object imposes on the range of possible motions of the holonomic robot. In this particular scenario, the pulled wheeled object is a pallet jack. To be able to embed its kinematics limitation onto the trajectory generator, we model the kinematic structure of the pallet jack and how it is coupled with the robot end-effector.

1) PALLET JACK KINEMATIC MODEL AND TRAJECTORY GENERATION

The pallet jack kinematic structure is composed by two rear fixed wheels and one front steered wheel. A quite common kinematic model for this structure is represented by the *bicycle model* (Fig. 2) [24]. In this model, the configuration of the vehicle is represented by the generalized coordinates $q = \begin{bmatrix} x & y & \theta \end{bmatrix}^T$, where the q is described by the vehicle frame $\boldsymbol{\Sigma}_{PJ}$ in the frame $\boldsymbol{\Sigma}_{PJ,i}$ shown in Fig. 2, with its *x-axis* in the vehicle's forward direction and its origin in the center of the rear fixed wheel axle. Its velocity is defined as v in the vehicle's x-direction, zero in the y-direction (the wheels cannot slip sideways). In the vehicle frame $\boldsymbol{\Sigma}_{PJ}$:

$$\dot{x}^{PJ} = v, \quad \dot{y}^{PJ} = 0. \tag{26}$$

The dashed lines in Fig. 2 are the no motion lines, so the direction along which the wheels cannot move. They intersect in a point called Instantaneous Centre of Rotation (ICR). The vehicle frame $\boldsymbol{\Sigma}_{PJ}$ follows a circular path with radius r around ICR with an angular velocity:

$$w = \frac{v}{r}, \quad r = \frac{L}{\tan\gamma} \tag{27}$$

Transforming the velocities in the frame $\boldsymbol{\Sigma}_{PJ,i}$ and applying the (27) we obtain the motion equations:

$$\begin{aligned} \dot{x} &= v\cos\theta \\ \dot{y} &= v\sin\theta \\ \dot{\theta} &= \frac{v}{L}\tan\gamma \end{aligned} \tag{28}$$

As we can notice, when $v = 0$, then $\dot{\theta} = 0$, so it is not possible to change orientation without applying a linear velocity. If $\gamma = \pi/2$, then the vehicle cannot move forward (the front wheel is orthogonal with respect to the back one) and the model enters in an undefined region.

Defined the kinematics, now the problem is to drive the pallet jack to a desired pose (x^*, y^*, θ^*). With the aim of generating feasible trajectories for the pallet jack model, a controller is introduced. We consider the coordinate transformation into polar coordinates using the notation shown in Fig. 2 [24]:

$$\begin{aligned} \rho &= \sqrt{{\Delta_x}^2 + {\Delta_y}^2} \\ \alpha &= atan2(\Delta_y, \Delta_x) - \theta \\ \beta &= -\theta - \alpha \end{aligned} \tag{29}$$

This results in a system description, in the new polar coordinates

1) if $\alpha \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right]$ (the goal is in front of the vehicle):

$$\begin{bmatrix} \dot{\rho} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} -\cos\alpha & 0 \\ \frac{\sin\alpha}{\rho} & -1 \\ -\frac{\sin\alpha}{\rho} & 0 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix}, \tag{30}$$

2) if $\alpha \in \left(-\pi, -\frac{\pi}{2}\right] \cup \left(\frac{\pi}{2}, \pi\right]$ (the goal is behind the vehicle):

$$\begin{bmatrix} \dot{\rho} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} \cos\alpha & 0 \\ -\frac{\sin\alpha}{\rho} & 1 \\ \frac{\sin\alpha}{\rho} & 0 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix}. \tag{31}$$

Now we need to design the control signals v and w to drive the vehicle to a unique equilibrium at $(\rho, \alpha, \beta) = (0, 0, 0)$.

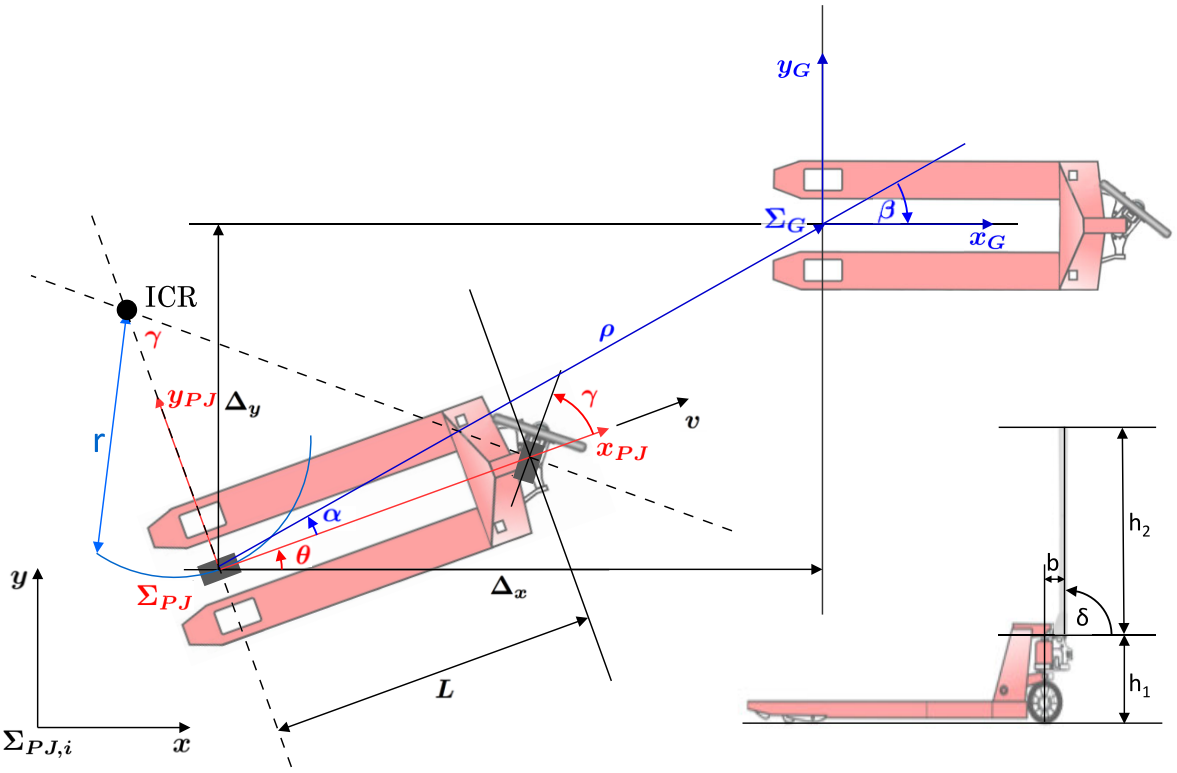


FIGURE 2. Bicycle model applied to the pallet jack. The two dark grey wheels represent the bicycle approximation. The steering wheel angle is γ. The two wheel axes are extended by the dashed lines and intersect at the Instantaneous Center of Rotation (ICR). Δ_x and Δ_y represent the error between Σ_{PJ} and Σ_G in Cartesian coordinates. The distance between Σ_{PJ} and the goal pose Σ_G is ρ, α is the angle between x_{PJ} and ρ and β is the angle of ρ with respect to $\Sigma_{PJ,i}$. In the right bottom part, the lateral view is shown with the corresponding notation.

Moreover, to avoid discontinuity at $\rho = 0$, we define the linear control law as:

$$\begin{aligned} v &= \sigma k_\rho \rho \\ w &= \sigma(k_\alpha \alpha + k_\beta \beta) \end{aligned} \tag{32}$$

where the terms $k_\rho \rho$ and $k_\alpha \alpha$ drive the vehicle along a line toward $\boldsymbol{\Sigma}_G$, $k_\beta \beta$ rotates the line such that $\beta \to 0$ and σ is the direction of the motion: $+1$ if the goal is in front, -1 otherwise. Combining (30) and (32), we obtain the closed-loop system:

$$\begin{bmatrix} \dot{\rho} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} -\sigma k_\rho \rho \cos\alpha \\ -\sigma(-k_\rho \sin\alpha + k_\alpha \alpha + k_\beta \beta) \\ -\sigma k_\rho \sin\alpha \end{bmatrix} \tag{33}$$

This controller drives the vehicle to the pose $(0, 0, 0)$. To drive to an arbitrary pose (x^*, y^*, θ^*), we apply a change of coordinates:

$$x' = x - x^*, \quad y' = y - y^*, \ \theta' = \theta, \ \beta = \beta' + \theta^* \tag{34}$$

The control parameters that makes the closed-loop system stable are:

$$k_\rho > 0; \quad k_\beta < 0; \ k_\alpha - k_\rho > 0. \tag{35}$$

However, since we aim for a robust position control, we want to ensure that the vehicle does not change direction during its approach to the goal applying the strong stability condition [25]:

$$k_\rho > 0; \quad k_\beta < 0; \ k_\alpha + \frac{5}{3}k_\beta - \frac{2}{\pi}k_\rho > 0. \tag{36}$$

2) END-EFFECTOR TRAJECTORY PLANNER

Since we want to compute the trajectory to be carried out by the pallet jack expressed w.r.t. its initial frame, $\boldsymbol{\Sigma}_{PJ,i}$, we first need to derive its transformation w.r.t. the world frame $\boldsymbol{\Sigma}_W$:

$$\boldsymbol{T}^W_{PJ,i} = \boldsymbol{T}^W_{Od} \boldsymbol{T}^{Od}_{EE} \boldsymbol{T}^{EE}_{H} \boldsymbol{T}^{H}_{Wh} \boldsymbol{T}^{Wh}_{PJ}, \tag{37}$$

where $\boldsymbol{T}^W_{Od}$ represents the odometry, i.e. the transformation from $\boldsymbol{\Sigma}_W$ to $\boldsymbol{\Sigma}_{Od}$, and $\boldsymbol{T}^{Od}_{EE}$ the transformation from $\boldsymbol{\Sigma}_{Od}$ to the end-effector frame $\boldsymbol{\Sigma}_{EE}$. Since we suppose that after the grasping the end-effector is rigidly coupled with the pallet jack handle, the translational component of their frames, $\boldsymbol{\Sigma}_{EE}$ and $\boldsymbol{\Sigma}_H$, results to be the same, while the orientation of the handle is rotated by π *rad* on the *y-axis*:

$$\boldsymbol{T}^{EE}_{H} = \begin{bmatrix} \boldsymbol{R}_y(\pi) & \boldsymbol{0} \\ \boldsymbol{0} & 1 \end{bmatrix}. \tag{38}$$

The transformation of the wheel frame $\mathbf{\Sigma}_{Wh}$ w.r.t. the handle frame $\mathbf{\Sigma}_H$ is calculated as:

$$\boldsymbol{T}_{Wh}^{H} = \begin{bmatrix} \boldsymbol{I}_{3\times3} & \begin{bmatrix} x_{Wh}^{H} \\ x_{Wh}^{H} tan\gamma \\ z_{Wh}^{H} \end{bmatrix} \\ \mathbf{0} & 1 \end{bmatrix}, \tag{39}$$

where x_{Wh}^{H} and z_{Wh}^{H} are computed as:

$$\begin{aligned} x_{Wh}^{H} &= -(b + h_2 cos\delta) \\ z_{Wh}^{H} &= -(h_1 + h_2 sin\delta). \end{aligned} \tag{40}$$

Lastly, the transformation from the pallet jacket to its wheels is constant and represented by:

$$\boldsymbol{T}_{PJ}^{Wh} = \begin{bmatrix} \boldsymbol{I}_{3\times3} & \begin{bmatrix} -L \\ 0 \\ 0 \end{bmatrix} \\ \mathbf{0} & 1 \end{bmatrix}, \tag{41}$$

After modeling and designing a controller for the pallet jack, we have to bring it to a desired pose taking into account it is not actuated. Giving as input a desired pose (x^*, y^*, θ^*), the trajectory is simulated through the controller. From this, we obtain the pose (x, y, θ) and hence $\boldsymbol{T}_{PJ}^{PJ,i}$, and the steering wheel angle γ, at each time-step. At this point, it is possible to retrieve the robot end-effector desired trajectory $\boldsymbol{T}_{EE}^{W}$ through (37):

$$\begin{aligned} \boldsymbol{T}_{PJ}^{W} &= \boldsymbol{T}_{PJ,i}^{W} \boldsymbol{T}_{PJ}^{PJ,i} \\ \boldsymbol{T}_{PJ}^{EE} &= \boldsymbol{T}_{H}^{EE} \boldsymbol{T}_{Wh}^{H} \boldsymbol{T}_{PJ}^{Wh} \\ \boldsymbol{T}_{EE}^{W} &= \boldsymbol{T}_{PJ}^{W} (\boldsymbol{T}_{PJ}^{EE})^{-1}. \end{aligned} \tag{42}$$

C. PLANNING COLLISION-FREE PATHS FOR HOLONOMIC ROBOTS

Typically, in cluttered environments such as warehouses, simple point-to-point direct motions are not suitable for autonomous mobile robots, due to the possible collision with dynamic obstacles. Human workers, machinery, conveyor belts, and other obstacles must be taken into account in planning the robot mobility. Therefore, we equipped MOCA with an obstacle avoidance algorithm capable of preventing collisions with fixed and moving obstacles. The algorithm exploits a global planner along with the TEB (Timed Elastic Band) local planner [26], without the need for a map initialization. Given a target location, the planner generates velocity commands for the mobile base, allowing it to navigate through the unknown area. This is obtained by updating a cost map which fuses the data sampled by the perception systems, such as front and rear lasers and the front camera, with the odometry estimation. Here we give an overview of the algorithm.

The trajectory optimization is described as the problem of finding the optimal control inputs $\boldsymbol{u}^* \in \mathbb{R}^m$:

$$\boldsymbol{x}^* = argmin\, J(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathcal{X} \tag{43}$$

$$\boldsymbol{u}^* = \boldsymbol{k}(\boldsymbol{x}^*) \tag{44}$$

where $\boldsymbol{x}$ is the optimization variable, $J(\boldsymbol{x})$ defines a nonlinear cost function, and $\boldsymbol{k} : \mathcal{X} \rightarrow \mathbb{R}^m$ denotes the mapping from $\boldsymbol{x}$ to the controls $\boldsymbol{u}$. Explicit definition of $J(\boldsymbol{x})$ can be found in [27].

In the case of mobile robot navigation, $J(\boldsymbol{x})$ is computed in order to minimize the trajectory in terms of time or distance between current and target pose. Additional terms are added to ensure a minimal distance from the obstacles, where the maximum is set at the center of the obstacle. The planning algorithm presented in [26] computes planar trajectories and it does not make assumptions on the obstacles shape or representation. A trajectory, in terms of subsequent robot positions, is defined as $s = \{z_k \in \mathbb{R}^2 \,|\, k = 1, 2, \ldots, N\}$.

An exploration graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is constructed for grouping an initial subset of admissible paths given the robot current z_s and goal z_g positions and a set of obstacle regions $\mathcal{O} = \{\mathcal{O}_l \,|\, l = 1, 2, \ldots, R\}$ with the set of vertices defined as $\mathcal{V} = \{z_s, \boldsymbol{\zeta}_i, z_g \in \mathbb{C} \,|\, \forall \boldsymbol{\zeta}_i \notin \mathcal{O}, i = 1, 2, \ldots, I\}$ and $\boldsymbol{\zeta}_i \in \mathbb{C}$ the waypoint samples. Due to the arbitrary shape of the obstacles, the waypoint sampling is computed with the probabilistic roadmap method [28].

Then, the set of edges $\mathcal{E}$ is constructed from the sampling of the waypoints. Each edge connects a pair of vertices $v_1, v_2 \in \mathcal{V}$ if:

- the direction with respect to the goal heading is forward oriented, i.e. $\dfrac{\Re[(v_2 - v_1)(z_g - z_s)]}{\mid v_2 - v_1 \mid\mid z_g - z_s \mid} > \theta, \ \theta \in [0, 1]$;
- the line segment $\mathcal{L} = \{v_1 + t(v_2 - v_1) \,|\, \forall t \in [0, 1]\}$ does not intersect any obstacle $\mathcal{O}_l$, i.e. $\mathcal{L} \cap \mathcal{O} = \emptyset$.

The paths between z_s and z_g are extracted from the generated graph $\mathcal{G}$ by a depth-first search augmented by a visited list.

Finally, the trajectory has to be optimized within the robot control feedback loop. At the beginning, the set of admissible trajectories T is empty. Hence, a new graph is created by seeding random samples in a region of interest. Thanks to the modified depth first search, the initialization of a single representative trajectory is computed for each class in T. The globally optimal trajectory $\boldsymbol{x}^*$ is selected according to (43) and then the sub-optimal control $\boldsymbol{u}^*$ is calculated. The current start and goal poses, z_0 and z_g respectively, are updated considering the actual robot state and perceptions.

III. OPERATIONAL MODULES

The proposed autonomous pallet jack recognition and handling framework can be divided into four main execution modules, as shown in Fig. 3: (A) a *Pallet jack reaching* unit in which the mobile robot navigates towards a desired pallet jack location while avoiding obstacles, (B) a *Handle detection* unit that combines the data coming from a perception module and the interaction forces at hand, (C) a *Handle preparation* unit responsible for grasping and pulling down the pallet jack handle, and (D) a *Pallet jack repositioning* unit to move the pallet jack to a new location in the workspace.

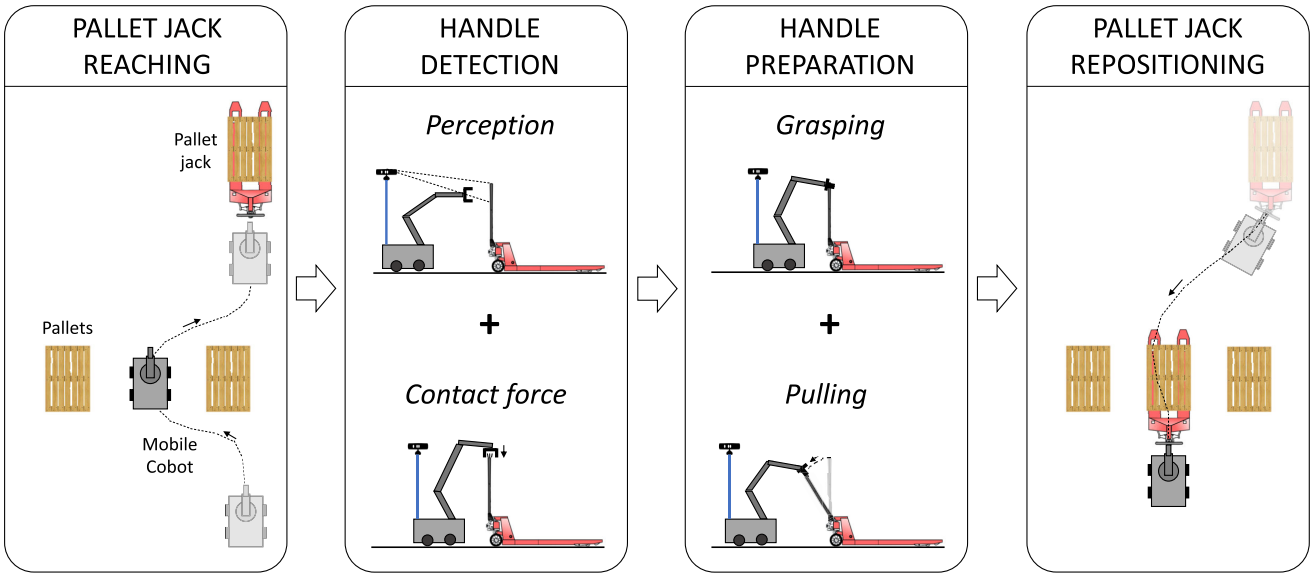


FIGURE 3. The framework consists of four main modules, that also represent the different phases of a pallet repositionning task. At first, the mobile cobot reaches the pallet jack location while avoiding the obstacles along its path. Fusing the data coming from perception and contact forces, MOCA is able to detect the pallet jack handle pose, that is grasped and pulled in the next phase. Lastly, the pallet jack is transported to the new desired location within the warehouse.

A. PALLET JACK REACHING

The presented system is designed to be operated in environments like warehouses, where typically a point-to-point direct motion is not feasible for autonomous robots. As mentioned earlier, human workers, pallets, and other obstacles must be taken into account in planning of the robot mobility. Therefore, the obstacle avoidance algorithm of Section II-C, capable of preventing collisions with both fixed and moving agents is integrated in *Pallet jack reaching* module. The Robot Operating System (ROS) package "move_base", which utilizes the ROS Global Planner along with the TEB (Timed-Elastic-Band) local planner [26], was used for this implementation, which eliminates the need for a map initialization. Given the target location, that can be retrieved in multiple ways, e.g. through a beacon system, exploiting cameras placed around the area, or using any other indoor positioning system (IPS), the algorithm is able to generate the mobile base velocity commands and navigate it through the unknown area. This is done by updating a cost map which fuses the data perceived by the perception sensors, such as lasers and cameras, and the odometry information. The data received from the sensors is employed to either insert or remove obstacle information into/from the costmap. Then, the costmap occupied cells are converted into a set of non-convex (concave) polygons in the following steps:

- Clusters are determined using the DBSCAN Algorithm [29].
- Clusters are converted to convex polygons and then to concave ones [30].

The obstacle poses are updated at each time-step. Thus, the TEB local planner is able to calculate a collision-free path related to the received information as explained in subsection II-C.

B. HANDLE DETECTION

Subsequent to locating the robot in front of the pallet jack, its handle pose must be detected with a high level of accuracy to ensure a successful prosecution of the next phases. To this end, we designed a handle detection algorithm that combines the vision and the interaction forces data.

1) PERCEPTION

The following section explains the vision algorithm implemented in the *handle detection* phase. To be able to detect and manipulate the handle, the mobile manipulator should identify the grasping 3D point along the pallet jack handle, together with the shaft orientation to allow the mobile robot to approach the pallet jack from a suitable angle. These two variables constitute the handle pose $\boldsymbol{T}_H^W$.

A simple approach to detect the handle grasping pose was to attach a colored band on the handle and proceed with the color segmentation of the colored point cloud. However, this approach is subject to the sensitivity of pixel colors to the environmental light. Moreover, the detection of a particular color in the scene depends on a set of color-defining parameters that have to be preset carefully. Therefore, this approach was considered unfeasible for the real warehouse environments, since, in addition to these issues, it requires also customization of the pallet jacks.

As reported in [31], learning-based and template matching techniques are among the most practical yet accurate for object recognition. Learning-based techniques are very strong in object recognition, even for a multiple of them,

but require expensive training and can be an over-fit to our relatively simple handle detection problem. On the other side, the template matching based algorithms can ensure a good approximation of the object position and orientation in the space. This technique also allows us to select a certain point on the template, for instance, a feasible grasping point on the handle, which leads to a simple handle grasping strategy. It also eliminates the need for any additional customization of the standard pallet jacks.

To apply this technique, first, a point cloud model of the handle was extracted from the scene. Following [32], a cluster extraction was performed from the observed depth point cloud, in order to select the candidates for the template alignment. Then, thanks to SHOT estimator algorithm, the SHOT descriptors are extracted both from the model and from the scene [33]. The correspondences between the model and the scene are computed by means of KdTree, a space-partitioning data structure that enables efficient nearest neighbor searches [34]. Finally, to evaluate the correspondences, we applied geometric consistency grouping, that is a clustering algorithm that enforces simple geometric constraints between pairs of correspondences [35]. The output of the the algorithm consists in a transformation matrix from the camera reference frame to a new reference, i.e., the rotated original frame[2] of the template, which aligns the model cloud with the real handle in the environment. Once the template fitting was performed, a rigid transformation from the new reference frame to the grasping point on the template was applied. The algorithm's output matrix $\boldsymbol{T}_H^C$, which corresponds to the pose of the handle in the camera frame $\boldsymbol{\Sigma}_C$, was transformed in the world frame $\boldsymbol{\Sigma}_W$ by:

$$\boldsymbol{T}_H^W = \boldsymbol{T}_{Od}^W \boldsymbol{T}_C^{Od} \boldsymbol{T}_H^C, \tag{45}$$

$$\boldsymbol{T}_C^{Od} = \begin{bmatrix} \boldsymbol{R}_z(-\frac{\pi}{2})\boldsymbol{R}_x(-\frac{\pi}{2}+\boldsymbol{\phi}) & \boldsymbol{p}_C^{Od} \\ \boldsymbol{0} & 1 \end{bmatrix}, \tag{46}$$

where $\boldsymbol{\phi}$ is the camera angle around the x axis of its frame.

2) CONTACT FORCE

Once the handle position $\boldsymbol{T}_H^W$ is localized, the robot end-effector must reach the handle from the top in order to grasp it firmly. The whole-body impedance controller is activated, with low impedance values to smoothly respond to unexpected collisions, while a higher mobility is assigned to the *arm* through (24). The robot is first sent to $\hat{\boldsymbol{T}}_H^W$, that represents $\boldsymbol{T}_H^W$ translated of 8 *cm* on the *z-axis*. From $\hat{\boldsymbol{T}}_H^W$, the robot end-effector starts to lower its position on the *z-axis* until the interaction forces on that axis, i.e., F_z, reach the threshold $F_{z,th}$, which identifies a physical contact.

C. HANDLE PREPARATION

Once the contact with the handle is established, the pallet jack needs to be prepared for the final phase. This state is subdivided into two actions.

[2] The reference frame of the model cloud when it was recorded.

1) GRASPING

This primitive simply makes the gripper/robotic hand close. In this way the robot and the pallet jack are physically coupled and they can be regarded as part of the same kinematic chain.

2) PULLING

The goal of this state is to make the pallet jack handle rotate δ_p *rad* to enable the pallet jack repositioning in the next phase. To do so, a circular trajectory with origin at the handle joint is designed exploiting (40) with δ going from $\pi/2$ *rad* to δ_p.

In order to ensure better interaction performances while pulling the handle, the controller impedance parameters are tuned in such a way they are more stiff on the direction of the motion, and more compliant along the other axes. This is achieved by tuning the major axis of the Cartesian stiffness and damping ellipsoids in the direction of interaction:

$$\boldsymbol{K}_c = \boldsymbol{U}\boldsymbol{\Psi}_k\boldsymbol{U}^T, \quad \boldsymbol{D}_c = \boldsymbol{U}\boldsymbol{\Psi}_d\boldsymbol{U}^T, \tag{47}$$

where the diagonal matrix $\boldsymbol{\Psi}_k = diag(k_{high}, k_{low}, k_{low})$ and $\boldsymbol{\Psi}_d = diag(d_{high}, d_{low}, d_{low})$ coefficients are respectively the desired stiffness and damping coefficients along the direction of the vectors composing the $\boldsymbol{U}$ basis. The $\boldsymbol{U}$ columns are the basis vectors, the first one represents the desired motion vector, and the other two are shaped to form an orthonormal basis [36]. From now on, the controller weight matrix $\boldsymbol{H}$ (24) is designed in such a way all the rotations about the *z-axis* of the base with respect to the world frame are reduced, while, at the same time, the motion of the base is boosted with respect to the arm motion. In order to do that, we set different weights in correspondence of the virtual torques of the mobile platform:

$$\boldsymbol{H} = \begin{bmatrix} diag\{\eta_{B_x}, \eta_{B_y}, \eta_{B_{yaw}}\} & \boldsymbol{0}_{m\times n} \\ \boldsymbol{0}_{n\times m} & \eta_A \boldsymbol{I}_{n\times n} \end{bmatrix}, \tag{48}$$

where $\eta_{B_{yaw}} \gg \eta_A \geq \eta_{B_x} = \eta_{B_y}$. In this way, high virtual torques related to the yaw of the mobile base are discouraged, in favor of other DoFs. In the particular case of $\eta_A > \eta_{B_x} = \eta_{B_y}$, the motion of the base platform in the *x-y plane* is dominant, and if the end-effector frame has to perform a rotation about the *z-axis*, the controller will try to actuate more the arm joints than the wheels. Noteworthy, the algorithm does not constraint the motion of the robot: in the case that the rotation is not achievable just performing an arm motion, the wheels will be actuated.

D. PALLET JACK REPOSITIONING

The pallet jack is now ready to be relocated in the desired pose of the workspace. First, a trajectory is generated by taking into account the kinematic constraints of the pallet jack, as explained in Sec. II-B. Next, the whole-body impedance controller executes the trajectory with the aforementioned weights. Once the pallet jack desired goal is reached, the task is accomplished.

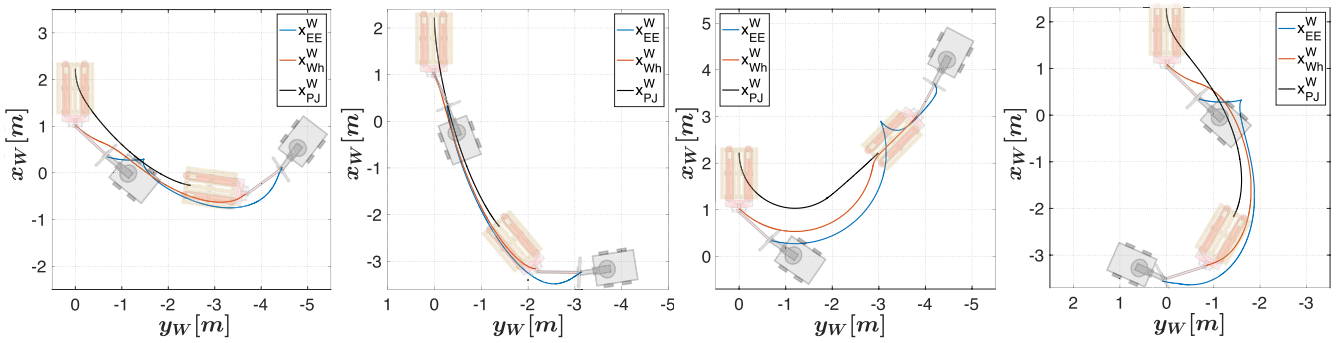


FIGURE 4. Simulation results: the four plots show the trajectories for different goals starting from the same configuration, i.e. $q^W_{PJ} = [2.2\ m\ 0.0\ m\ \pi\ rad]^T$. From left to right, the final desired configurations were set to: $q^W_{PJ} = [-0.2\ m\ -2.5\ m\ -\pi/2\ rad]^T$, $q^W_{PJ} = [-2.2\ m\ -1.4\ m\ -3\pi/4\ rad]^T$, $q^W_{PJ} = [2.3\ m\ -3.0\ m\ -\pi/4\ rad]^T$, $q^W_{PJ} = [-2.2\ m\ -1.4\ m\ 3\pi/4\ rad]^T$. The average errors of the final desired position and orientation across all trials were 0.0199 *m* and 0.1262 *rad*, respectively.

IV. SIMULATION RESULTS

Prior to the experiments, the repeatability and the accuracy of the pallet jack repositioning algorithm were extensively tested in simulations. Fig. 4 illustrates four typical scenarios with different ending desired positions and orientations. These simulation results provided a primary evidence in achieving the pallet jack desired pose with high accuracy in the repositioning algorithm. Notice that, the sketches of the pallet jack and the robot were added at a post-processing stage on the plots, to ensure a better understanding of the results.

V. EXPERIMENTAL SETUP

We conducted experiments inside a real warehouse, using a pallet jack with loaded pallets to demonstrate the validity of the proposed method in a typical industrial environment.

The pallet jack used in our experiments is a Tractel Pioneer 2500, weights 87 Kg, and has these physical characteristics, following the notation of the Fig. 2: $L = 1.2\ m$, $h_1 = 0.4\ m$, $h_2 = 0.85\ m$ and $b = 0.09\ m$.

The robotic platform is MOCA [16], which is the results of the integration of a Robotnik SUMMIT-XL STEEL mobile platform, a Franka Emika Panda robotic arm mounted on top, and an underactuated Pisa/IIT SoftHand. An additional ASUS Xtion Pro RGB-D camera is used by the vision module which was mounted on a pole attached to the backside of MOCA (see Fig. 3). In the presented experiments, the values in (45) were set to: $\boldsymbol{p}_C^{Od} = [-0.225\ m,\ -0.183\ m,\ 1.56\ m]^T$ and $\boldsymbol{\phi} = 0.24\ rad$.

During the *Pallet jack reaching* phase, described in Sec. III-A, the "move_base" algorithm takes advantage of the perception sensors mounted on the Robotnik SUMMIT-XL STEEL, namely a HOKUYO-10LX Laser and an ASUS Xtion Pro RGB-D camera located in front.

The software architecture relies upon ROS using C++ and Matlab as client libraries. The high modularity of this middleware enabled the data exchange among all the described modules through the well-known publisher/subscriber policy and the server/client paradigm.

VI. EXPERIMENTS

In this section, we first illustrate the performance of the proposed Weighted WB impedance controller for repositioning of a pallet jack. Next, an overview of the results of the four phases of the experiments (presented in previous sections), is provided.

For *Pallet jack repositioning* phase, the controller's parameters to calculate the trajectory of the pallet jack were set to

$$k_\rho = 0.2; \quad k_\beta = -\frac{6}{5};\ k_\alpha = -\frac{5}{3}k_\beta + \frac{2}{\pi}k_\rho + 1.$$

Moreover, we decided to saturate the pallet jack linear velocity v at 0.3 m/s and the steering wheel angle γ at $\pi/4\ rad$.

We initially carried out this phase with two different controllers: the whole-body impedance controller introduced in [16] (WB controller), and its improved version presented here that includes the prioritized weighted inverse dynamics algorithm (Weighted WB controller). In the latter, in order to limit the rotational movements of the robot base, so as to achieve better tracking performances, we set the weights of (48) as $\eta_A = \eta_{B_x} = \eta_{B_y} = 1$, $\eta_{B_{yaw}} = \sqrt{20}$. Fig. 5 shows the different behaviors of the two controllers with the same pallet jack desired goal, $\boldsymbol{q}^W_{PJ} = [0.1\ m\ \ 4.0\ m\ \ \pi/2\ rad]^T$, represented in the world frame $\boldsymbol{\Sigma}_W$. As can be noticed from Fig. 5a, the desired end-effector pose x^W_{dEE}, projected on the xy plane, is tracked more accurately with the Weighted WB controller. In fact, with the WB controller, the robot trajectory is subject to continuous variations as can be also noticed in Fig. 5b, where the position errors of the two controllers are compared. In the third subplot, after an initial transient of 15 s, the Weighted WB controller position error norm (blue curve) falls below the threshold of 3 cm, while the one related to the WB controller (red curve) is still subject to high fluctuations for a longer time producing a chattering phenomenon. The Cartesian controller generates a force according to the Cartesian tracking error. At first, when this error is small, due to the high compliance, such force is not enough to

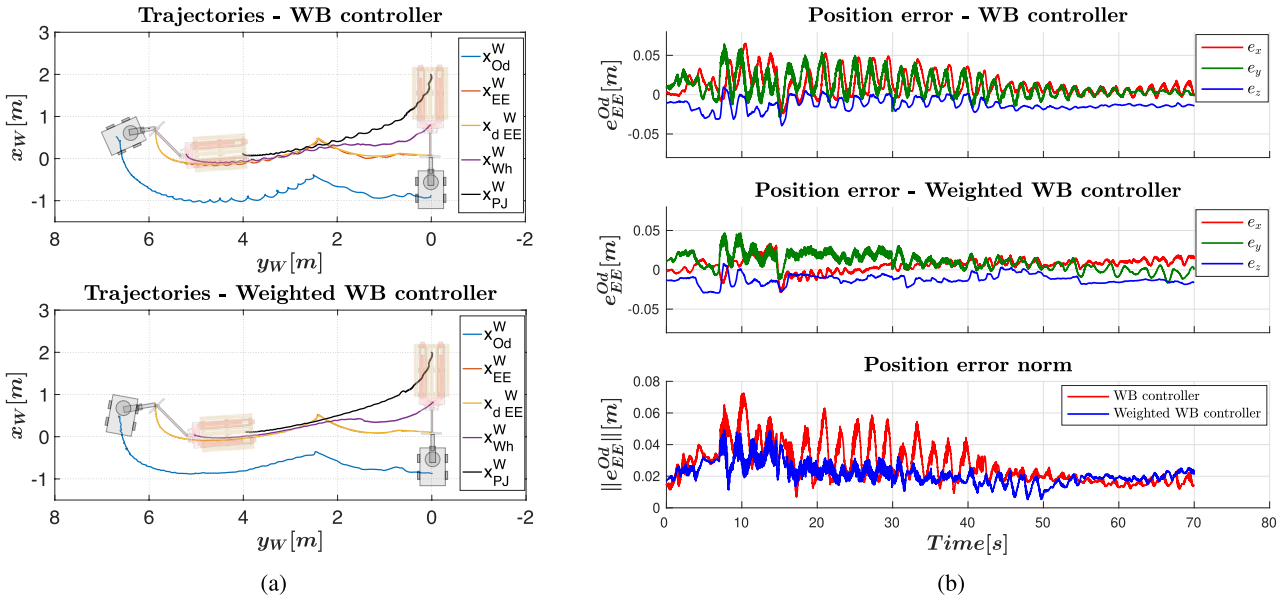


FIGURE 5. ***Pallet jack repositioning phase*: trajectory (a) and position error (b) comparison between the whole-body controller (WB) and the weighted inverse dynamics algorithm whole-body controller (Weighted WB).**

FIGURE 6. **Snapshots of the experiment which was performed in a real warehouse environment.**

pull the pallet jack, because of its load and friction. As a result, the tracking error increases until the generated force overcomes friction and gravity, moving the pallet jack, and hence, reducing again the tracking error. The WB controller generally tries to exploit all the DoFs of MOCA to achieve the desired motion, combining highly dynamical arm motion with slower base motion. For instance, the rotation of the end-effector around the *z-axis* are achievable by means of combined arm motion and the third joint of the base. The effect of the base motion results slower because of its higher inertia and partially delayed with respect to the arm motion, due to its lower control loop frequency. This delay might affect the end-effector pose limiting the tracking performance. To limit this effect, we opted to weight the rotational motions of the base around the *z-axis* (see Sec. 5), favoring the combined motion of the arm and just the linear motion of the base. The effects of this choice can be noticed in fig. 5b, where the norm of tracking error results lower.

In an additional set of experiments, we evaluated the performance of the proposed framework (with Weighted WB impedance controller), in four phases of the task, whose snapshots are illustrated in Fig. 6. A video of the experiment is available in the multimedia extension.

In the *Pallet jack reaching* phase, MOCA had to reach the pallet jack avoiding obstacles (notice the filled pallet in Fig. 6, which was placed in front of MOCA) after having received the target pose. In a more realistic scenario, the pallet jack could be localized within the warehouse in many ways, e.g. through a beacon system or exploiting cameras placed around the area. However, for the sake of simplicity in our proof-of-concept experiment, we assumed the pallet jack pose to be roughly known.

Fig. 7 left plot depicts the path of the robot during this phase. Since a pallet (as an obstacle) was autonomously detected on the way to the goal, the algorithm updated the trajectories to avoid it.

In the next phase, the *Handle detection* unit retrieved the pose of the handle in the Odometry frame $\mathbf{\Sigma}_{Od}$, i.e. $\boldsymbol{p}_H^{Od} = [0.75\ -0.42\ 1.27]\ m$, as shown on the point cloud in Fig. 8. The white point cloud corresponds to the real sight of the robot, the green one is the extracted handle cluster, and the red point cloud corresponds to the aligned template.

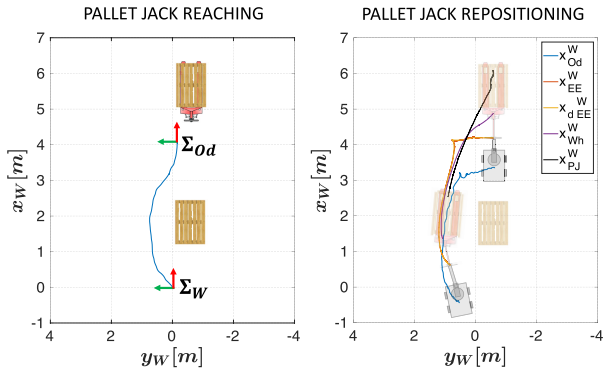


FIGURE 7. Experimental results of phases A and D.

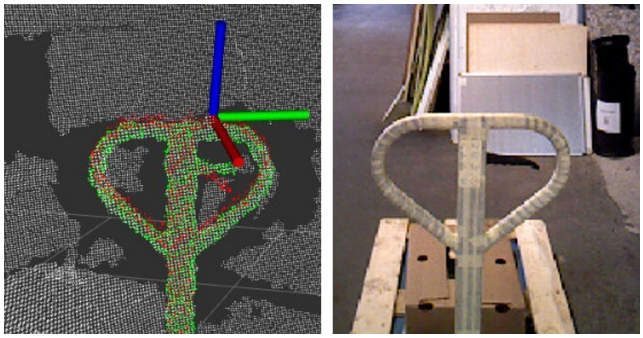


FIGURE 8. Handle detection through perception module.

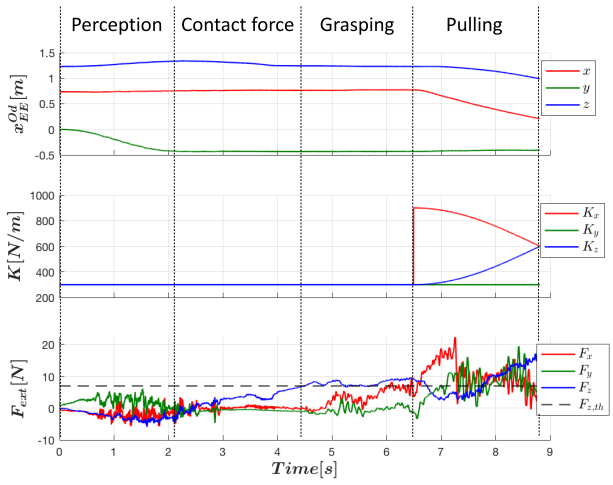


FIGURE 9. Experimental results of phases B-C: MOCA end-effector pose, Cartesian stiffness and external forces.

Subsequently, the robot end-effector reached $\hat{\boldsymbol{T}}^W_{ee}$ that can be noticed in the first column of the top plot in Fig. 9. The impedance values were set to be very compliant, 300 N/m in every Cartesian direction, to allow a safe and compliant interaction in case of unexpected collisions. Next, in the *Contact force* phase, the robot lowered its hand until a contact was detected, i.e., at time $t = 4.4$ s, when F_z goes beyond the threshold $F_{z,th}$ set to 7 N, as shown in the bottom plot.

This contact triggered the *Handle preparation* phase, where the Pisa/IIT SoftHand grasped the pallet jack handle at time t = 5-6 s (*Grasping* subphase). The hand only uses 1 actuator to activate its adaptive synergy, so its intrinsic compliance makes it adapt to the object that is being grasped, therefore we set the closure percentage to 100%, to ensure a stable grasping that guarantees the kinematic chain continuity between the robot and the pallet jack. Next, the circular trajectory in (40) is executed with δ_p set to $\pi/4$ (*Pulling* subphase). In (47), k_{high} and k_{low} were set to 900 N/m and 300 N/m respectively. In the second plot of Fig. 9, it is possible to notice the regulation of the impedance parameters based on the direction of the motion vector during *Pulling*. Here, the parameters of the weight matrix $\boldsymbol{H}$ were modified such that $\eta_A = \eta_{B_x} = \eta_{B_y} = 1$, and $\eta_{B_{yaw}} = \sqrt{20}$.

The right plot in Fig. 7 represents the results during the *Pallet jack repositioning* phase. Specifically, it shows the trajectories in the world frame $\boldsymbol{\Sigma}_W$ projected on the *x-y plane*. In this experiment, the goal position for the *Pallet jack repositioning* phase was set aside another filled pallet, which was positioned in front of a warehouse storage drawer, so it was set to $\boldsymbol{q}^W_{PJ} = [2.5\ m\ 0.9\ m\ \pi\ rad]^T$.

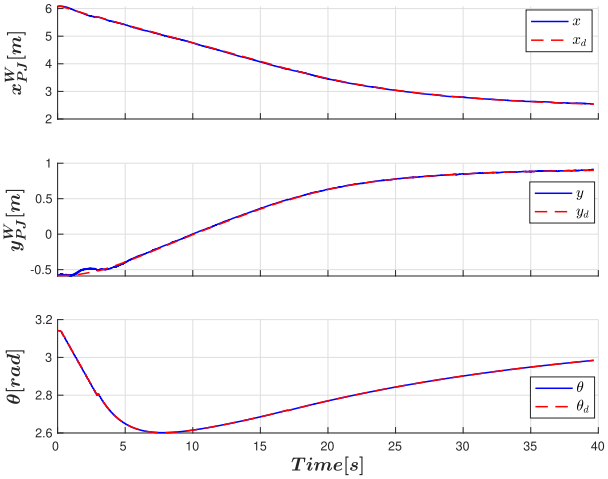


FIGURE 10. Experimental results of phase D. The desired trajectories (red dashed lines) are accurately tracked through MOCA.

Fig. 10 shows the desired pallet jack configuration $\boldsymbol{q}_d = [x_d \quad y_d \quad \theta_d]^T$ computed by the planner (Sec. II-B) at each time-step, projected in the world frame $\boldsymbol{\Sigma}_W$. Moreover, it shows the estimated pallet jack configuration $\boldsymbol{q} = [x \quad y \quad \theta]^T$ calculated reversing the (42) and using the measured $\boldsymbol{T}^W_{EE}$.

VII. CONCLUSION AND DISCUSSION

In this work we presented a novel collaborative robotic approach to the problem of autonomous pallet jack transportation and positioning. The proposed framework can navigate a mobile cobot to a target pose, detect the handle of the pallet jack and approach to grasp it, and transport it to a desired position and orientation in a warehouse environment. The experimental results with MOCA, a mobile collaborative

robotic platform recently developed at IIT, demonstrated the high potential of the proposed framework in seamless and accurate positioning of the pallets. This was achieved in spite of the dynamic uncertainties in perception, navigation, and manipulation, and the imposed kinematic constraints of the exploited standard pallet jack. Future works will aim to improve the framework reliability in response to uncertainties, such as checking the feasibility of the maneuver space in narrow environments, and ensuring collision-free paths during the repositioning phase, for instance by pausing the motion if a dynamic obstacle is moving along the planned path.

Although this work only focused on the problem of pallet jack transportation and repositioning, the proposed framework has the potential to automatize similar repetitive and effort intensive tasks, such as repositioning of large industrial carts, airport mobile stairways, etc. This is because, such systems exhibit similar mobility constraints that can be dealt with, using the proposed MOCA controller.

ACKNOWLEDGMENT

(Pietro Balatti and Fabio Fusaro contributed equally to this work.)

REFERENCES

[1] S. Lone, A. Khelladi, and S. Packiarajah, *European Ecommerce Report: 2018 Edition*. Brussels, Belgium: Ecommerce Eur., 2018.

[2] K. Azadeh, R. De Koster, and D. Roy, "Robotized and automated warehouse systems: Review and recent developments," *Transp. Sci.*, vol. 53, no. 4, pp. 917–945, Jul. 2019.

[3] G. Bäközkan and F. Göçer, "Digital supply chain: Literature review and a proposed framework for future research," *Comput. Ind.*, vol. 97, pp. 157–177, May 2018.

[4] S. Teller, M. R. Walter, M. Antone, A. Correa, R. Davis, L. Fletcher, E. Frazzoli, J. Glass, J. P. How, A. S. Huang, J. Hwan Jeon, S. Karaman, B. Luders, N. Roy, and T. Sainath, "A voice-commandable robotic forklift working alongside humans in minimally-prepared outdoor environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 526–533.

[5] T. A. Tamba, B. Hong, and K.-S. Hong, "A path following control of an unmanned autonomous forklift," *Int. J. Control, Autom. Syst.*, vol. 7, no. 1, pp. 113–122, Feb. 2009.

[6] Z. Zhang and B. Xiao, "The influence of cargo moving and sliding mode control strategy for forklift," *IEEE Access*, vol. 8, pp. 16637–16646, 2020.

[7] M. Seelinger and J.-D. Yoder, "Automatic visual guidance of a forklift engaging a pallet," *Robot. Auto. Syst.*, vol. 54, no. 12, pp. 1026–1038, Dec. 2006.

[8] D. Lecking, O. Wulf, and B. Wagner, "Variable pallet pick-up for automatic guided vehicles in industrial environments," in *Proc. IEEE Conf. Emerg. Technol. Factory Autom.*, Sep. 2006, pp. 1169–1174.

[9] R. Cucchiara, M. Piccardi, and A. Prati, "Focus based feature extraction for pallets recognition," in *Proc. Brit. Mach. Vis. Conf.*, 2000, pp. 1–10.

[10] R. Bostelman, T. Hong, and T. Chang, "Visualization of pallets," *Proc. SPIE Intell. Robots Comput. Vis.*, vol. 6384, Oct. 2006, Art. no. 638408.

[11] A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, and O. Khatib, "Progress and prospects of the human–robot collaboration," *Auto. Robots*, vol. 42, no. 5, pp. 957–975, Jun. 2018.

[12] R. Bloss, "Collaborative robots are rapidly providing major improvements in productivity, safety, programing ease, portability and cost while addressing many new applications," *Ind. Robot, Int. J.*, vol. 43, no. 5, pp. 463–468, Aug. 2016.

[13] R. Krug, T. Stoyanov, V. Tincani, H. Andreasson, R. Mosberger, G. Fantoni, and A. J. Lilienthal, "The next step in robot commissioning: Autonomous picking and palletizing," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 546–553, Jan. 2016.

[14] M. Lutz, C. Verbeek, and C. Schlegel, "Towards a robot fleet for intralogistic tasks: Combining free robot navigation with multi-robot coordination at bottlenecks," in *Proc. IEEE 21st Int. Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2016, pp. 1–4.

[15] Z. Zhang, Q. Guo, J. Chen, and P. Yuan, "Collision-free route planning for multiple agvs in an automated warehouse based on collision classification," *IEEE Access*, vol. 6, pp. 26022–26035, 2018.

[16] Y. Wu, P. Balatti, M. Lorenzini, F. Zhao, W. Kim, and A. Ajoudani, "A teleoperation interface for loco-manipulation control of mobile collaborative robotic assistant," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3593–3600, Oct. 2019.

[17] A. Widyotriatmo and K.-S. Hong, "Configuration control of an autonomous vehicle under nonholonomic and field-of-view constraints," *Int. J. Imag. Robot.*, vol. 15, no. 3, pp. 126–139, 2015.

[18] W. Kim, M. Lorenzini, P. Balatti, Y. Wu, and A. Ajoudani, "Towards ergonomic control of collaborative effort in multi-human mobile-robot teams," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 1–4.

[19] E. Lamon, M. Leonori, W. Kim, and A. Ajoudani, "Towards an intelligent collaborative robotic system for mixed case palletizing," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, 2020, pp. 9128–9134.

[20] A. Dietrich, K. Bussmann, F. Petit, P. Kotyczka, C. Ott, B. Lohmann, and A. Albu-Schäffer, "Whole-body impedance control of wheeled mobile manipulators," *Auto. Robots*, vol. 40, no. 3, pp. 505–517, Mar. 2016.

[21] S. Kim, K. Jang, S. Park, Y. Lee, S. Y. Lee, and J. Park, "Whole-body control of non-holonomic mobile manipulator based on hierarchical quadratic programming and continuous task transition," in *Proc. IEEE 4th Int. Conf. Adv. Robot. Mechatronics (ICARM)*, Jul. 2019, pp. 414–419.

[22] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE J. Robot. Autom.*, vol. 3, no. 1, pp. 43–53, Feb. 1987.

[23] O. Khatib, "Inertial properties in robotic manipulation: An object-level framework," *Int. J. Robot. Res.*, vol. 14, no. 1, pp. 19–36, Feb. 1995.

[24] P. Corke, *Robotics, Vision and Control: Fundamental Algorithms in MATLAB Second, Completely Revised*, vol. 118. Springer, 2017. [Online]. Available: https://www.springer.com/gp/book/9783319544120

[25] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to Automes Mobile Robots*. Cambridge, MA, USA: MIT Press, 2011.

[26] C. Rosmann, F. Hoffmann, and T. Bertram, "Planning of multiple robot trajectories in distinctive topologies," in *Proc. Eur. Conf. Mobile Robots (ECMR)*, Sep. 2015, pp. 1–6.

[27] C. Rösmann, F. Hoffmann, and T. Bertram, "Integrated online trajectory planning and optimization in distinctive topologies," *Robot. Auto. Syst.*, vol. 88, pp. 142–153, Feb. 2017.

[28] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans. Robot. Autom.*, vol. 12, no. 4, pp. 566–580, 1996.

[29] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. KDD*, vol. 96, 1996, pp. 226–231.

[30] J.-S. Park and S.-J. Oh, "A new concave hull algorithm and concaveness measure for n-dimensional datasets," *J. Inf. Sci. Eng.*, vol. 28, pp. 587–600, May 2012.

[31] S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab, and V. Lepetit, "Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 858–859.

[32] K. Liu, W. Wang, and J. Wang, "Pedestrian detection with lidar point clouds based on single template matching," *Electronics*, vol. 8, no. 7, p. 780, Jul. 2019.

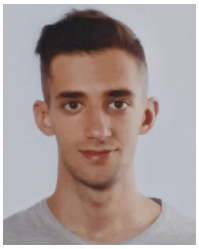
[33] F. Tombari, S. Salti, and L. Di Stefano, "A combined texture-shape descriptor for enhanced 3D feature matching," in *Proc. 18th IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 809–812.

[34] M. Muja and D. Lowe, "Flann-fast library for approximate nearest neighbors user manual," Univ. British Columbia, Vancouver, BC, Canada, Tech. Rep., 2009.

[35] H. Chen and B. Bhanu, "3D free-form object recognition in range images using local surface patches," *Pattern Recognit. Lett.*, vol. 28, no. 10, pp. 1252–1262, Jul. 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167865507000621

[36] P. Balatti, D. Kanoulas, N. G. Tsagarakis, and A. Ajoudani, "Towards robot interaction autonomy: Explore, identify, and interact," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 9523–9529.

PIETRO BALATTI (Member, IEEE) received the B.D. and M.D. degrees *(cum laude)* in computer science and engineering from the Politecnico di Milano, Italy, in 2014 and 2016, respectively. He is currently pursuing the Ph.D. degree with the Human-Robot Interfaces and physical Interaction Laboratory, Istituto Italiano di Tecnologia, and the Department of Information Engineering, University of Pisa, Italy. He was a member of the CoAware Team that won the KUKA Innovation Award, in April 2018, who was responsible of the robot control framework. His main research interests include the enhancement of robot interaction autonomy, within the fields of adaptive impedance control and physical human–robot interaction.



FABIO FUSARO received the B.D. degree in bioengineering and the M.D. degree (Hons.) in robotics engineering from the University of Genova, Italy, in 2017 and 2019, respectively. He is currently pursuing the Ph.D. degree with the Human-Robot Interfaces and physical Interaction Laboratory, Istituto Italiano di Tecnologia, in collaboration with the Politecnico di Milano. His main research interests include autonomous loco-manipulation of mobile manipulators in industrial environments and in human-aware task planning for collaborative robots.

NICOLA VILLA received the bachelor's and master's degrees in automation and control engineering from the Politecnico di Milano, Italy, in 2019. In the last six months of his university career, he has worked at his master's thesis on the navigation control for an AUV, developed for the European project UNEXMIN at ISEP, Porto, PT. He is currently working with the Human-Robot Interfaces and physical Interaction Laboratory, Istituto Italiano di Tecnologia, Genoa, Italy. His main research interests include interaction control of robotic arms and mobile manipulators, human–robot collaboration, service robotics, and computer vision for object recognition.

EDOARDO LAMON (Member, IEEE) received the B.Sc. degree in information engineering and the M.Sc. degree in control and automation engineering from the University of Padova, Italy. He is currently pursuing the Ph.D. degree with the Human-Robot Interfaces and physical Interaction (HRI^2) Group, Italian Institute of Technology (IIT), Genoa, and the Department of Information Engineering, University of Pisa, Italy. He is a part of the Program Committee of the International Workshop on Human Friendly Robotics 2020. His research, developed within the Horizon-2020 European project SOPHIA and the Amazon Research Awards 2019, focus on combining robotics knowledge with artificial intelligence, with the aim to boost collaborative robotics in smart factories. His research interests include human–robot collaboration, and robot control and learning.

ARASH AJOUDANI (Member, IEEE) received the Ph.D. degree in robotics and automation from the University of Pisa and IIT, in July 2014. He is currently a tenured Senior Scientist with the Italian Institute of Technology (IIT), where he leads the Human-Robot Interfaces and physical Interaction (HRI^2) Laboratory. He is the author of the book *Transferring Human Impedance Regulation Skills to Robots* [Springer Tracts in Advanced Robotics (STAR)], and several publications in journals, international conferences, and book chapters. His main research interests include physical human–robot interaction and cooperation, robotic manipulation, robust and adaptive control, assistive robotics, and tele-robotics. He has contributed to several successful European projects (H2020 and FP7), such as WALKMAN, WearHap, SOMA, and SoftPro. He was a recipient of the European Research Council (ERC) starting grant 2019. His Ph.D. Thesis was a Finalist for the Georges Giralt Ph.D. Award 2015–Best European Ph.D. Thesis in robotics. He was a Winner of the Amazon Research Awards, in 2019, the Solution Award, in 2019 (Premio Innovazione Robotica at MECSPE2019), the KUKA Innovation Award, in 2018, and the Werob Best Poster Award, in 2018; a Finalist for the Best Manipulation Paper Award at ICRA, in 2012, the Best Conference Paper Award at Humanoids, in 2018, the Best Interactive Paper Award at Humanoids, in 2016, and the Best Oral Presentation Award at Automatica (SIDRA), in 2014; and a Winner of the Best Student Paper Award and a Finalist for the Best Conference Paper Award at ROBIO, in 2013. He is the coordinator of the Horizon-2020 Project SOPHIA with a consortium of 12 partners from the leading European research and industrial organizations. He has been serving as the Executive Manager of the IEEE-RAS Young Reviewers' Program (YRP), and the Chair and a Representative of the IEEE-RAS Young Professionals Committee. He has been serving as a member for the Scientific Advisory Committee and an Associate Editor for several international journals and conferences, such as IEEE RAL, Biorob, and ICORR.