# Manipulation at optimum locations for maximum force transmission with mobile robots under environmental disturbances

Harun Tugal<sup>1,2</sup> · Kamil Cetin<sup>1,2</sup> · Yvan Petillot<sup>1,2</sup> · Matthew Dunnigan<sup>1,2</sup> · Mustafa Suphi Erden<sup>1,2</sup>

Received: 20 June 2021 / Accepted: 26 May 2022 / Published online: 5 July 2022 © The Author(s) 2022

#### Abstract

Remote manipulation plays a key role for applications in hazardous conditions, yet designing a robust controller enabling safe interaction with unknown environment and under the influence of disturbances is a challenge. In this study, we propose effective control and optimization methods for mobile robotic manipulator systems that can increase effort transmission to a task in desired directions. The vehicle position is optimized by utilizing constrained particle swarm optimization where the objective is to enhance directional manipulability of the robotic arm within the system. A forward dynamic controller is implemented to eliminate undesired excessive motions near singular joint configurations. A reset control algorithm along with an admittance type controller are developed for stable interaction with an unknown object under environmental disturbances. The experimentally validated results show that the proposed method phase out undesired position disturbances and increase the directional manipulability for the required task enabling augmented effort transmission for the task execution.

Keywords Directional manipulability · Forward dynamic control · Particle swarm optimization · Mobile robots

# **1** Introduction

Autonomous vehicles (AVs), that are capable of carrying task without supervision or teleoperation, and remotely operated vehicles (ROVs) are mainly used in applications where human access would imply high risk of danger or there exists limited access. They are mainly used for remote inspection and monitoring, with suitable electro–mechanical structures

This research was fully funded by EPSRC through the ORCA Hub project with the Grant Reference EP/R026173/1.

$\bowtie$	Harun Tugal haruntugal@gmail.com
	Kamil Cetin kamil.cetin@ikc.edu.tr
	Yvan Petillot y.r.petillot@hw.ac.uk
	Matthew Dunnigan m.w.dunnigan@hw.ac.uk
	Mustafa Suphi Erden m.s.erden@hw.ac.uk
1	Institute of Sensors, Signals and Systems, School of Engineering and Physical Sciences, Heriot Watt Universit

Engineering and Physical Sciences, Heriot-Watt University, EH14 4AL Edinburgh, UK

<sup>2</sup> Edinburgh Centre for Robotics, Edinburgh, UK

for remote repair, maintenance or material handling (Lozano-Perez, 2012). Interest in autonomous manipulation is constantly increasing, as performing manned tasks in unsafe conditions is not only dangerous, but also costly, as professional experts with extensive training are needed to perform such operations (see, e.g., Ruggiero, Lippiello, & Ollero, 2018).

Designing a safe and robust robotic vehicle manipulator system (VMS) that can accomplish manipulation in unknown conditions, however, is a challenging task from control point of view as the overall system has highly coupled non-linear dynamics along with the parametric uncertainties. Navigation within unstructured environment is difficult (Lozano-Perez, 2012). Further, grasping and manipulation of an object contains its own challenges since the system can be imposed to unstable dynamics (Colgate & Hogan, 1988). In addition, these robots, particularly for underwater and aerial applications, need to manipulate the environment without a stationary base. Motion controller for such systems requires the capability of simultaneously stable localization, trajectory tracking, and stable physical interaction with the environment (Andaluz et al., 2012).

To overcome these challenges, some studies consider a single kinematic/dynamic model for the overall system such that the robotic arm mounted on a mobile platform is defined



and controlled as a coupled system, resulting in a complex overall model, but enhancing the redundancy at the task space (see, e.g., De Luca et al., 2006; Lippiello & Ruggiero, 2012). Similar to this approach, the overall movement of mobile manipulators can be planned to carry out sub-tasks such as singularity avoidance, maximization of applicable force in desired directions, and manipulability enhancement. Manipulability is defined as the measure of a robotic system's ability to adjust its end-effector's position/orientation, (see Yoshikawa, 1985). In Yamamoto and Yun (1994), for instance, dynamic-coupling is presented such that mobile vehicle localizes itself to enhance manipulability of the robotic arm mounted on it; (see Yamamoto & Yun, 1996) for its extension encapsulating dynamic interaction between the vehicle and manipulator. A robust adaptive controller was proposed in Andaluz et al. (2012) based on a combined system dynamics for the mobile manipulators to tackle the point stabilization and trajectory tracking problems while increasing the manipulability of the robotic arm.

To reduce modelling and control complexities one can alternatively design independent controllers for each robotic system, the vehicle and the manipulator, and define different control objectives for each sub-system. In this approach any vehicle motion is considered to cause a perturbation in the task space and needs to be attenuated by the manipulator (see, e.g., Barbalata, Dunnigan, & Petillot, 2018; Ram, Pathak, & Junco, 2019). In Rigotti-Thompson et al. (2020), for a skid-steer mobile manipulator, the end-effector motion is decoupled from the base motion to compensate disturbed motions on uneven terrains via implementing an  $\mathcal{H}_{\infty}$  controller with/without feed-forward control actions.

The position of the vehicle relative to the task, on the other hand, determines the effectiveness of the robotic arm during the task execution. Energy efficiency, singularity avoidance, and dexterity have been used as the main criteria desired to be improved. Due to the non-availability of the objective function's gradient, it is a common approach to use metaheuristic algorithms for the base position optimization such as swarm intelligence and genetic algorithm (Dang & Nguyen, 2016; Mitsi et al., 2008; Ren et al., 2017). In Carriker et al. (1991), motion planning of a mobile manipulator is considered as an optimization problem where the vehicle position and manipulator joint angles were considered separately and optimum points, based on the end-effector position and force/torque, were estimated via simulated annealing. A decoupled approach has been established for base position optimization in Yu et al. (2018) where an internal penalty function and generalized Lagrange multiplier methods were utilized for obtaining near-optimal positions while nominating the positioning, orientation, and singularity avoidance as the constraints. Sparse uniform grid decomposition and sequential quadratic programming methods were implemented for base position optimization in Fan et al.

(2021) while considering joint range, joint speed, singularity avoidance and collision avoidance as search constraints. Prior studies on optimum positioning, however, have not demonstrated stable physical interaction with unknown objects under external disturbances. In our paper, we realize optimum positioning for maximum force transmission for physical interaction with environment.

The effect of the manipulator's motion on the vehicle is negligible in the ground based vehicles as the base is stationary. In the underwater and aerial applications, when the manipulator has relatively small mass with respect to the vehicle, the effects of the manipulator motion on base vehicle positioning are not significant and there fore a decoupled control approach can be adapted for a less complex and effective control (Barbalata et al., 2018). Decoupled control approach has also the advantage of easily accommodating different control cycle rates for the base vehicle and the robot arm manipulator. Thus, in this study, we adapt the decoupling approach and define different tasks for the vehicle and the robotic arm mounted on it; the vehicle moves to the optimum position to provide maximum manoeuvrability, (avoiding joint singularity and enhancing effort transmission from the joints to a task space) to the robot arm and the arm carries out the manipulation while attenuating any disturbances at the task space. We apply particle swarm optimization (PSO) to find an optimum vehicle position such that the force/torque transmission and velocity control capacities of the robotic arm are increased in pre-defined task directions, see Fig. 1.

In this paper, the primary aim is to enhance the force transmission of a manipulator at task space via optimal base localization while maintaining velocity control for task execution. For our method, we inspire from the observation that humans adopt their body posture and position while performing a force interaction task in order to apply effective force with their arms. Imitating this behaviour, the strength capabilities of a mobile robotic arm could be increased to accomplish



**Fig. 1** The vehicle can be relocated at optimum position based on the manipulation direction such that effort transmission and fine velocity control properties of the robotic arm is enhanced for a given task

manipulations that require excessive effort, such as for the initial drive to turn a valve. While doing this, task space position-force control and joint singularity problems should also be addressed for a stable manipulation under environmental disturbances. The presented control and optimization methods that we used in our overall system are as follows:

- Reset controller for attenuating any vehicle motion at the task space; i.e., contact point stabilization,
- Admittance controller for establishing a safe interaction with the environment,
- Forward dynamic controller for eliminating any excessive joint motions while the robotic arm is close to the singular joint configurations,
- PSO for optimum localization of the vehicle to enhance the force transmission and fine velocity control properties of the mobile robotic arm.

To the best of our knowledge, this is the first research attempt to implement directional manipulability along with a metaheuristic optimization algorithm to extend interaction capabilities of a mobile manipulator. Valve turning, similar to a wheel rotation, in underwater environmental conditions is considered as a case study for the experimental evaluation (see, e.g., Di Lillo et al., 2020; Korpela, Orsag, & Oh, 2014). The contributions of this paper can be summarised as

- Metaheuristic optimisation method is combined with control architectures to maintain optimal vehicle position while interacting with unknown objects under environmental disturbances.
- Maximum force transmission is achieved by adopting directional manipulability with the optimisation algorithm, where the algorithm finds the robot configuration that will provide maximum force in selected task space directions.
- The optimisation and control frameworks are demonstrated to transmit maximum force with a physical system that can emulate any (land, underwater, or aerial) VMS, under environmental disturbances.

## 1.1 Notation

Bold lower-case letters, x, represent the column vectors, nonbold upper case-letters, X, denote matrices, non-bold lower case-letters, x, represent single variables, and  $\top$  denotes the transpose. The transformation from a vector to a skewsymmetric matrix is denoted by []<sub>×</sub> operator. The identify matrix is denoted as I.

### 2 Methods

## 2.1 Reference frames and transformations

When combining different sensors and motion frames, it is required to utilize a single frame of reference for each controlled motion. The force/torque sensor, for instance, provides measurements in sensor frame (s) which differs from the end-effector frame (e) and more importantly needs to be transferred to the contact/task space (t) as that is the main concern during an interaction, see Fig. 2 for the frame of references in this study.

The force/torque sensor measures the force applied by the environment on the end-effector  $(f_s = [f_s^x f_s^y f_s^z \tau_s^x \tau_s^y \tau_s^z]^\top)$ ; thus, the force applied by the manipulator on the environment  $f_e$  is obtained by taking the negative of the refined (eliminating sensor bias and gravity component due to the sensor's own and tool masses) sensor reading (Jamisola et al., 2005). Generally, due to the displacement  $(d_t)$  between the contact point (t) and the sensor point (s) the torque reading is the sum of the force sensor frame. Thus, the sensor frame force/torques measurements need be transformed to the task frame as

$$\boldsymbol{f}_{m} = \begin{bmatrix} {}^{t}\boldsymbol{R}_{e} & \boldsymbol{0} \\ \left[ \boldsymbol{p}_{s}^{t} \right]_{\times} {}^{t}\boldsymbol{R}_{e} {}^{t}\boldsymbol{R}_{e} \end{bmatrix} \boldsymbol{f}_{e},$$

where  $p_s^t \in \mathbb{R}^3$  is the position vector of the origin of point *s* with respect to the coordinate frame of *t* and  ${}^tR_e$  is the rotation matrix between the end-effector frame and the contact point reference frame.

Based on the forward kinematics of the manipulator, which returns end-effector's position  $p_e^b \in \mathbb{R}^3$  and rotation matrix  ${}^eR_b \in \mathbb{R}^{3\times 3}$ , the position of the contact point with respect to the base is obtained as

$$\boldsymbol{p}_t^b = \boldsymbol{p}_e^b + {}^e\boldsymbol{R}_b^\top \boldsymbol{p}_t^e,$$



Fig. 2 Motion's and sensors' frames of references

where  $p_t^e$  denotes the position of the contact point *t* relative to the end-effector. Here, the orientation of the contact point coincidences with the end-effector's; thus,  ${}^tR_b = {}^eR_b$ . Hereupon, the rotation matrix of the manipulator ( ${}^eR_b$ ) is to be used for the transformations from the base to the contact frames.

Once the desired task space velocities are obtained, the robot's Jacobian matrix can be utilized to transform the desired task space velocities,  $\dot{\boldsymbol{x}}_r$ , into the desired joint space velocities,  $\dot{\boldsymbol{q}}_r = [\dot{q}_1, \dots, \dot{q}_n]^{\top}$  as

$$\dot{\boldsymbol{q}}_r = \boldsymbol{J}^{\dagger}(\boldsymbol{q})\dot{\boldsymbol{x}}_r,\tag{1}$$

where  $J^{\dagger} \in \mathbb{R}^{n \times 6}$  is the pseudo inverse of the manipulator Jacobian matrix.

The position of the contact point  $(p_t^g)$  in the earth-fixed frame (global), on the other hand, based on the vehicle's position and orientation along the manipulator's joint configurations is given as

$$\boldsymbol{p}_t^g = \boldsymbol{p}_v^g + {}^g\boldsymbol{R}_v(\boldsymbol{r}_b^v + {}^v\boldsymbol{R}_b\,\boldsymbol{p}_t^b),$$

where  $p_t^g$  is the position of the vehicle with respect to the global frame,  ${}^vR_g$  is the rotation matrix between the body-fixed vehicle frame and the earth fixed global frame, the manipulator's base position with respect to the vehicle's body fixed frame is denoted with the position vector  $r_b^v$ , and the matrix  ${}^vR_b$  is the rotation matrix between the manipulator-base frame and the body-fixed vehicle frame.

#### 2.2 Position-force controllers

The robotic manipulator is required to perform two tasks simultaneously; position control for object manipulation with position disturbance attenuation and force control for establishing a safe and steady contact with the object. The former can be accomplished with a predetermined trajectory,  $p_r$ , and a controller minimizing the perturbations at the end-effector introduced by the vehicle's motion. This controller needs to have fast position tracking performance without excessive overshoots and steady state errors via using noisy sensor measurements that might contain time delays due to the discrepancy between the control and measurement cycles. Thus, we applied a reset controller method with a variable reset band which can be described in the state-space form as

$$R:\begin{cases} \dot{\boldsymbol{x}}(t) = A_r \boldsymbol{x}(t) + B_r \boldsymbol{e}_r(t), & \text{if } (\boldsymbol{e}_r(t), \dot{\boldsymbol{e}}_r(t)) \notin \mathcal{B}_h \\ \boldsymbol{x}(t^+) = A_p \boldsymbol{x}(t), & \text{if } (\boldsymbol{e}_r(t), \dot{\boldsymbol{e}}_r(t)) \in \mathcal{B}_h \\ \boldsymbol{u}(t) = C_r \boldsymbol{x}(t) + D_r \boldsymbol{e}_r(t), \end{cases}$$



Fig. 3 Reset action can reduce the overshot without introducing additional control efforts on the system

where  $e_r$ , x, and u are error, state, and output vectors, respectively (Villaverde et al., 2011). The middle equation represents the reset action applied when the error crosses a threshold, generally zero value. The diagonal matrix  $A_p$ performs reset algorithm on the desired states as  $A_p =$ diag  $(1 - d_r, ..., 1 - d_r, 0, ..., 0)$  with a reset ratio  $d_r$ which is a constant value with values  $d_r \in [0, 1]$  and allowing us to apply a partial reset  $(d_r \neq 1)$  strategy. The variable reset band can be defined as

$$\mathcal{B}_{h} = \left\{ (\boldsymbol{e}(t), \dot{\boldsymbol{e}}(t)) \in \mathbb{R}^{2} \mid |h\dot{\boldsymbol{e}}(t) + \boldsymbol{e}(t)| \le \gamma \right\}, \qquad (3)$$

where *h* and  $\gamma$  are positive constants and associated with the system's time delay and band range, respectively (Banos & Barreiro, 2012).

The advantages of the reset control over a linear compensator can be illustrated with a system consisting of pure integrator and time delay. A numerical example is adapted from Davó et al. (2018), such that a transfer function,  $P(s) = \frac{e^{-0.15s}}{s}$ , is considered to be in a feedback loop with a base compensator given by  $\frac{1.4s+0.3}{s}$  plus a reset band. The step response of the closed loop system with ( $d_r = 0.8$ ) and without ( $d_r = 0$ ) reset actions while h = 0.15 and  $\gamma = 0.001$ are illustrated in Fig. 3 where the overshoot is eliminated with the reset action. We refer the reader to Banos and Barreiro (2012), Zhao et al. (2019) for more detailed information about how the reset control overcomes limitations of the linear controllers.

The objective of force tracking, on the other hand, requires a force/torque sensor near the contact point, i.e., the endeffector of the robot, so that the interaction force can be measured to generate the desired velocities via an admittance type control algorithm whose motion dynamics in the task space are

$$\boldsymbol{f}_r - \boldsymbol{f}_m = M_a \boldsymbol{\dot{v}}_f + D_a \boldsymbol{v}_f + K_a (-\boldsymbol{p}_o), \tag{4}$$

where  $f_r$ ,  $f_m$ , and  $v_f \in \mathbb{R}^6$  denote the reference, measured force/torque, the desired end effector velocity vectors,



Fig. 4 Block diagram representation of the implemented position-force control algorithm for the manipulator interacting with the environment

respectively. The object's position with respect to the endeffector is denoted by  $\mathbf{p}_o = \begin{bmatrix} p_o^x & p_o^y & p_o^z & 0 & 0 \end{bmatrix}^\top$ . The diagonal matrices,  $M_a$ ,  $D_a$ , and  $K_a \in \mathbb{R}^{6\times 6}$  are the admittance controller's virtual mass  $(m_i)$ , damping  $(d_i)$ , and stiffness  $(k_i)$ parameters for position (i = p) and orientation (i = o). At the contact point, we want the end effector to be perpendicular to the inspected surface at task space. Thereby, the reference force/torque can be denoted as  $f_r = \begin{bmatrix} 0 & 0 & f_r^z & 0 & 0 \end{bmatrix}^\top$ , which implies that the force controller makes an effort to apply constant force,  $f_r^z$ , in z direction and zero torque in x and y directions to prevent misalignment at the tool frame. To eliminate force components affecting in other directions, calculated  $\mathbf{v}_f$  will be multiplied with a task matrix  $\Omega = \text{diag}(0, 0, 1, 1, 1, 0)$  before using in the manipulator control algorithm as in Moura et al. (2018), Cetin et al. (2021).

The vehicle motion affects the contact point position. To phase out this, similar to the chicken head stabilization, the measured base motion distortion ( $p_v$ ) is embedded into the reset controller's reference trajectory. In this way, the environmental disturbances affecting the main body of the overall system would be eliminated at the contact point.

One can utilize Jacobian matrix as in (1) for transforming task space velocities into the joint space velocities but, the motion might be unstable near singular joint configurations. Thereby singularity handling algorithms need to be implemented for safe operations (Jamisola et al., 2005). Here, following Scherzinger et al. (2017) and Lee et al. (2020), to solve the inverse kinematics problem we implement a forward dynamic control (FDC) method. The designed control architecture is illustrated in Fig. 4 where the Jacobian transpose is utilized, rather than Jacobian inverse, and nominal robot model,  $P_m$ , generates the desired joint velocities (Scherzinger, Roennau, & Dillmann, 2019). The control input,  $\tau_c$ , within the FDC method is generated via an impedance controller which is denoted as a proportionalderivative (PD) controller for non-redundant robots as



Fig. 5 Determining the injected damping via a direct-delta function  $(b_{max} = 1.25 \text{ N s m}^{-1})$ 

$$M_m(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{c}_m(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{g}_m(\boldsymbol{q}) = \boldsymbol{\tau}_c,$$
  
$$\boldsymbol{\tau}_c = \boldsymbol{J}^\top \boldsymbol{f}_c + \boldsymbol{G}_m(\boldsymbol{q}) - \boldsymbol{B}\dot{\boldsymbol{q}} \text{ and } \boldsymbol{f}_c = \boldsymbol{K}\boldsymbol{e} + \boldsymbol{D}\dot{\boldsymbol{e}}, \qquad (5)$$

where  $M_m \in \mathbb{R}^{6\times 6}$  denotes the manipulator's positive definite inertia matrix,  $c_m$  and  $g_m \in \mathbb{R}^6$  represent Coriolis an centrifugal terms and gravitational components, respectively. The task space pose error is denoted as  $e \in \mathbb{R}^6$ . The diagonal matrices,  $K = k_i \mathbf{I}$ ,  $D = d_i \mathbf{I} \in \mathbb{R}^{6\times 6}$  are the impedance controller's gains for position (i = p) and orientation (i = o). Additional joint space damping,  $B = b\mathbf{I} \in \mathbb{R}^{6\times 6}$ , is injected to dissipate the energy generated near singular configurations as the task space inertia and Coriolis increase near singular configurations (Lee et al., 2020). On the other hand, to eliminate the effect of the injected damping at a nonsingular motions, *b* is modified with a direct-delta function as

$$b = b_{max} \exp\left(-\frac{\sigma_{min}^2}{a^2}\right),\tag{6}$$

where *a* is a positive constant determining the width of the direct-delta function, see Fig. 5,  $b_{max}$  is the maximum damping constant, and  $\sigma_{min}$  is the minimum singular value of the Jacobian matrix measuring the distance from the singular joint configuration.

#### 2.3 Directional manipulability and constrained PSO

It is a common approach to use a measure of dexterity in studies that aim at optimizing the base location of serial manipulators. The manipulability index,  $\mu$ , is employed as a proxy to measure the dexterity associated with a feasible configuration of the manipulator. It can be derived for non-redundant manipulators as

$$\mu = \sqrt{\det(J(\boldsymbol{q})J(\boldsymbol{q})^{\top})} = \prod_{i} \sigma_{i}, \tag{7}$$

where  $\sigma_i$  denotes the singular values for the Jacobian matrix (Yoshikawa, 1985). One can use the dexterity measure

to define an objective function for the optimum base localisation. In a near-singular configuration, the determinant of the Jacobian of a serial manipulator almost diminishes and hence  $\mu$  in (7) approaches to zero. Maximizing this index as an optimization criterion maximizes the transformation from joint velocities to task-space velocities.

At task space, to enhance not only the effecting velocities but also the force transmission, the duality between the velocity and force control directions needs to be addressed (Chiu, 1988). Using the dexterity measure, considering the duality criterion, and counting the task space control directions we define a directional optimization index as

$$o_{i} = {}^{f}w \left[ {}^{f}\boldsymbol{u}^{\top} (J_{i}(\boldsymbol{q})J_{i}(\boldsymbol{q})^{\top})^{f}\boldsymbol{u} \right]$$
  
+  ${}^{v}w \left[ {}^{v}\boldsymbol{u}^{\top} (J_{i}(\boldsymbol{q})J_{i}(\boldsymbol{q})^{\top})^{-1}{}^{v}\boldsymbol{u} \right],$ (8)

where  ${}^{f}w$  and  ${}^{v}w$  are positive constants representing the relative importance of the force and velocity controllers, respectively, and the directional vectors can be defined as  ${}^{f}u = {}^{c}\mathcal{R}_{b}^{\top} \begin{bmatrix} 0 \ 0 \ 1 \ 1 \ 1 \ 0 \end{bmatrix}^{\top}$  and  ${}^{v}u = {}^{c}\mathcal{R}_{b}^{\top} \begin{bmatrix} 1 \ 1 \ 0 \ 0 \ 0 \end{bmatrix}^{\top}$  with  ${}^{c}\mathcal{R}_{b} = \text{diag}({}^{c}R_{b}, {}^{c}R_{b})$  matrix which is the diagonal combination of the rotation matrix,  ${}^{c}R_{b}$ , between the manipulator-base frame and the contact point. One can define a cost function which is minimised as a function of the total optimization index along a predetermined task trajectory as

$$f_o(o) = \frac{1}{\sum_{i=0}^n o_i},$$
(9)

where  $i = [0 \dots n]$  stands for the points on the task trajectory. Minimizing this cost function corresponds to maximizing the total optimization index and in turn maximizing the effective force transmission in the selected directions and enhancing the velocity control in others. The two weighting factors,  ${}^{f}w$  and  ${}^{v}w$ , represent the relative importance of the force and velocity controllers in orthogonal directions within the optimization index and allow for managing the trade-off between maximizing the force in the selected directions. These are user defined parameters according to whether maneuverability in the direction of force transmission or maneuverability in the direction of velocity control are prioritized and to what degree.

PSO is a bio-inspired population based heuristic optimization technique (Kennedy & Eberhart, 1995). The idea is intuitive; a group of swarm/particles search for an optimum location (i.e. cornfield) altogether and each member has the knowledge of their current local optimum (particle best position  $p_{pbest}$ ) and global optimum (group best position  $p_{gbest}$ ) locations during the search. Each particle, k, is associated with two vectors, the velocity vector  $V_k = [v_k^1, v_k^2, \dots, v_k^d]$ and position vector  $P_k = [p_k^1, p_k^2, \dots, p_k^d]$ , where d stands for the dimension of the search space. The velocity and position of each particle are initialized with random vectors within the feasible ranges for each parameter. The optimizer (evolutionary process) adjusts the velocity and position of the particle k on dimension d as

$$v_k^d = \omega v_{k-1}^d + \epsilon_1 r_1^d (p_{pbest}^d - p_k^d) + \epsilon_2 r_2^d (p_{gbest}^d - p_k^d),$$
  

$$p_k^d = p_{k-1}^d + v_k^d,$$
(10)

where  $p_k$  and  $v_k$  denote the particle's position and velocity, respectively,  $r_1^d$  and  $r_2^d$  are two uniformly distributed random numbers independently generated in [0, 1]. The particles are evaluated based on their new position scores;  $p_{nbest}$  and  $p_{gbest}$  are updated at every iteration. The search process is repeated until a pre-determined stopping criterion is met; e.g., the search is repeated for a fixed number of times  $(l_{max})$  or a defined cost function value is achieved. The effect of the particle's inertia, cognitive and social behaviours are denoted as  $\omega$ ,  $\epsilon_1$ , and  $\epsilon_2$ , respectively, see for instance (Eberhart & Kennedy, 1995; Shi & Eberhart, 1998; Zhan et al., 2009) for more information about the effect of each variable and how to tune them. One can implement this algorithm to find an optimum position for the vehicle such that the cost function in (9) is reduced to maximize the manipulability of the manipulator mounted on it.

Algorithm 1 Pseudo code for constrained PSO algorithm. Create and initialize  $P_k$  and  $V_k$  in the feasible search space with a population size  $(n_{pop})$ repeat for For each particle  $k \in [1, ..., n_{pop}]$  do Calculate  $F_{\mu}$  via (9) and (12), if  $F_{\mu}(p_k) < F_{\mu}(p_{pbest})$  and  $g(p_k) \leq 0$  then  $p_{pbest} = p_k$ ,  $F_{\mu}(p_{pbest}) = F_{\mu}(p_k).$ end if Choose the particle with best  $F_{\mu}$  as  $p_{gbest}$ , Update positions and velocities via (10). if  $|v_k| > v_{max}$  then  $v_k = \operatorname{sign}(v_k) v_{max}$ . end if end for until stopping condition is true

When the mobile manipulator is in the vicinity of an object (such a a valve) which the robotic manipulator can and is desired to make a contact with (within its workspace), safety becomes the main concern as collision may occur between the vehicle and the environment where the object is located (such as a pipe on which the valve is located). In order to prevent collision, a safety workspace restricting the potential movement of the vehicle can be defined. Particles are then initialized in that workspace and the search space is constrained within the PSO algorithm. In this study, we illustrate this concept by using a 3D sphere to restrict the motion of the vehicle once the contact is established as

$$g(p) = (p^{x} - p_{0}^{x})^{2} + (p^{y} - p_{0}^{y})^{2} + (p^{z} - p_{0}^{z})^{2} - r^{2} \le 0,$$
(11)

where *r* denotes the vehicle's operational workspace radius allowed during the interaction,  $p_0^i$  is the position of the vehicle when the first contact is established, and  $p^i$  is the relative position of the vehicle with respect to  $p_0^i \in \mathbb{R}^3$ . Also,  $v_k$ is restricted to the range  $[-v_{max}, v_{max}]$  to prevent particles from converging outside of the search space (Corne et al., 1999). Accordingly, the constrained optimization problem to find a location for the vehicle can be described as

Find 
$$p^x$$
,  $p^y$ ,  $p^z$  to minimize  $f_o$ ,  
Subject to  $g(p) \le 0$ .

The original PSO design lacks a mechanism to handle the constraints in an effective way. Most of the constrained PSO designs adopt the popular constraint handling techniques that are built for evolutionary algorithms (Runarsson & Yao, 2005; Takahama & Sakai, 2006). Different methodologies have been implemented regarding the constraint handling with PSO including penalty methods (Parsopoulos & Vrahatis, 2002), comparison criteria or feasibility tournament (Coello, Pulido, & Lechuga, 2004; He & Wang, 2007), and Lagrange-based method (Krohling & Dos Santos Coelho, 2006). Here, a feasible region constraint handling algorithm and a penalty method are implemented; initialization is made within the feasible search space such that constraint (11) is satisfied and the particles only memorize their and the group's best feasible solutions located in the search space (Isiet & Gadala, 2019; Xiaohui & Eberhar, 2002). The utilized constrained PSO algorithm solves the cost function as

$$F_o = \begin{cases} f_o, & \text{if feasible,} \\ f_o + \beta_1 \sum_{i=1}^d s + \beta_2 \sum_{i=1}^d y, & \text{if infeasible,} \end{cases}$$
(12)

where  $F_o$  and  $f_o$  are the penalty and original functions,  $\sum_{i=1}^{d} s$  is the sum of *d* violated constraints,  $\sum_{i=1}^{d} y$  is sum of the amount of *d* violated constraints, and  $\beta_1$  and  $\beta_2$  are the penalty factors. See Algorithm 1 for the depicted constrained search method.

#### 2.4 Case study: valve turning

Valve manipulation (e.g., Di Lillo et al., 2020) similar to a wheel rotating under environmental disturbances is adopted as a case study in the validations. Namely, a gripper attached to a robotic arm mounted on a moving platform grasps a valve from its outer circle (or from its extreme if it is a T-shape valve) such that the end-effector's *z*-coordinate frame

is perpendicular to the valve's rotation plane. In this configuration, our control direction in the task frame would be defined as; x, y, and yaw for velocity control and z, roll, and pitch for force/torque control. In this way, valve manipulation can be carried out with the velocity control and the orientation between the gripper and the valve itself will be maintained with the force/torque control.

An imitation of underwater manipulation was chosen as a case study where the manipulator motion and environmental interaction force affect the motion of the base vehicle. These effects are generated via modelling for the evaluation of the performance of the designed controller. The general equation of motion of an underwater vehicle can be written as

$$M_r \dot{\boldsymbol{\nu}} + C_r(\boldsymbol{\nu})\boldsymbol{\nu} + D_r(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{g}_r(\boldsymbol{\eta}) = \boldsymbol{\tau}_r + \boldsymbol{\tau}_d, \quad (13)$$

where  $\mathbf{v}$  is the body fixed linear (surge, sway, and heave) and angular (roll, pitch, and yaw) velocity vector,  $\mathbf{\eta}$  is the earth fixed position and attitude vector. The vehicle's positive definite inertia matrix including the added inertia, Coriolis and centripetal terms, and hydrodynamic lift and damping matrices are denoted by  $M_r$ ,  $C_r$ , and  $D_r \in \mathbb{R}^{6\times 6}$ , respectively. The vectors  $\mathbf{g}_r$ ,  $\mathbf{\tau}_r$ , and  $\mathbf{\tau}_d \in \mathbb{R}^6$  denote gravitational components and buoyancy forces, propulsion forces, and induced disturbance due to the manipulator motions (Fossen, 2011). The induced disturbances due to the manipulator motions,  $\mathbf{\tau}_d = [\mathbf{\tau}_f^{\top}, \mathbf{\tau}_m^{\top}]^{\top}$ , can be calculated as

$$\boldsymbol{\tau}_{d} = \begin{bmatrix} {}^{\boldsymbol{v}}\boldsymbol{R}_{b} & \boldsymbol{0} \\ [\boldsymbol{r}_{b}^{\boldsymbol{v}}]_{\times} {}^{\boldsymbol{v}}\boldsymbol{R}_{b} {}^{\boldsymbol{v}}\boldsymbol{R}_{b} \end{bmatrix} \boldsymbol{f}_{b}, \tag{14}$$

where  $f_b$  represents the forces and moments acting on the base frame of the manipulator.

## **3 Experimental validation**

## 3.1 Experimental setup

The experimental setup consists of a parallel manipulator known as a Stewart platform, a Universal Robot (UR3), a Robotiq 2-finger gripper (2F-140), an Intel RealSense camera (D455), and an ATI Gamma force/torque sensor (with ATI Force/Torque 9105-NETB sensor box). The camera and UR3 robot are mounted on the Stewart platform and the force/torque sensor attached to the robot's end effector via a mechanical adaptor and the gripper is attached to the end-of the sensor. Thus, the setup enables the robot to interact with the environment (in this case a PVC valve with diameter equal to 0.1 m), see Fig. 6. The platform and UR3 robot each have six degrees of freedom and each controlled by their own control boxes providing 33 Hz and 125 Hz control cycles, respectively.



**Fig. 6** Experimental setup consisting of a Stewart platform, UR3 robotic arm, force/torque sensor, 2-finger gripper, and a camera. A rigid wheel type valve was used as an object that the robot arm is interacting with and manipulating

The explained swarm algorithm is used to find the optimum positions for the platform and waypoints to generate a smooth trajectory using a quartic spline. An April-tag marker is used to locate the valve and also to measure any perturbation effecting the platform (Wang & Olson, 2016). To create a stable interaction with the environment, the FDC method, in (5) with (6), along with the reset control, in (2) with (3), and admittance control, in (4), architectures were implemented when an object enters into the workspace of the manipulator, by using the force/torque sensor's measurements with a sampling frequency  $(f_s)$  equivalent to the robot's control cycle ( $f_s = 125 \text{ Hz}$ ). The Stewart platform, robotic manipulator, and gripper are all controlled via the ROS middleware for ease of integration by a computer with a 3.20 GHz CPU. This experimental setup imitates a mobile manipulator systems that could be used for subsea or aerial applications.

#### 3.2 Results

Initially, to evaluate the controller actions on the chickenhead problem, the base of the manipulator was moved in z-direction  $\pm 0.18$  m as a perturbation (Experiment 1), with the controller parameters given in Table 1. Thus, the performance of the reset action was observed with the default proportional-integral (PI) controller. The measured base motion by means of the fiducial marker was used within the controller to eliminate the undesired move at the contact point. The evaluation is based on the Euclidean norm of the recorded position errors with respect to the reference point computed as

$$||E_{cont.}|| = \sqrt{\sum_{i=x}^{l} \sum_{j=1}^{n} e_{ij}^{2}},$$

Tuble I controller und optimization parameter	Table 1	Controller and	optimization	parameters
---	---------	----------------	--------------	------------

PI+Reset	$k_p = 20$	$i_p = 2$	$d_r = 0.7$
Control	$k_{o} = 2$	$i_o = 1$	$\gamma = 10^{-4}$
Adt.	$m_p = 2 \mathrm{kg}$	$d_p = 350 \mathrm{Ns/m}$	$k_p = 250 \mathrm{N/m}$
Control	$m_o = 0.2 \mathrm{kg}$	$d_o = 7 \mathrm{Ns/m}$	$k_o = 0 \mathrm{N/m}$
PSO	$l_{max} = 100$	$\omega = 0.5$	$\epsilon_1 = \epsilon_2 = 2$
	$n_{pop} = 50$		$w^f = w^v = 1$
FDC	$k_p = 50$	$d_p = d_o = 0.02$	$b_{max} = 1.25$
	$k_{o} = 5$		



**Fig. 7** End-effector position, position error (*E*), and integral error term within the controller when base of the manipulator is moved  $\pm 0.18$  m in the z-direction. Reset action eliminates the overshoot and reduce overall tracking error via resetting the integral action (undermost figure);  $||E_{PI}|| = 0.212$  and  $||E_{PI+Reset}|| = 0.154$ 

where the controlled axes are denoted by l = x, y, z and n denotes the data size. For a  $t_{max}$  seconds experiment, with data registration at a rate  $f_s$ , the amount of samples recorded will be  $n = t_{max} f_s$ . Figure 7 shows that despite the fast base motion,  $|v_z| = 0.18 \text{ m s}^{-1}$ , the controller could relatively keep the end-effector position stationary with respect to the global frame with a maximum error equivalent to 1 mm and the reset action can reduce the overall position tracking error by approximately 27% when compared to the base controller.

Then, a simple experiment (Experiment 2) was executed via using a pendulum with a mass of 8 kg and l = 0.26 m to compare the torque measures generated by the manipulator to accomplish the same task in different base positions. The robot was required to hold a pendulum 0.05 m away from the equilibrium position which, creates restoring forces on the contact point due to the gravity.

A threshold force,  $f_{th}$ , was specified as 2 N which allowed us to determine whether or not a contact is established with the pendulum. Once the end-effector has contact with the



**Fig. 8** *Left* holding a pendulum with 8 kg mass at a position approximately 0.05 m from the equilibrium position. *Right* repeating the same task while the base is located an optimum posture such that force/torque transmission property of the manipulator, from joint to end-effector space, is enhanced



Fig. 9 Force measurements in z direction at task space during two experiments; non-optimum and optimum postures. Threshold force was determined as  $f_{th} = 2 \text{ N}$  to determine the contact

pendulum, the manipulator moves the mass to the desired position in 5 s and holds there around another 5 s. The task was repeated with a non-optimum<sup>1</sup> (practically the nominal position of the platform; 1.196 m above and parallel to the ground) and with an optimum postures where, in the later case, the base was located at the optimum position after establishing contact with the pendulum. The last 2 s of the joints' torques measurements were is used in a comparison index calculated as

$$\tilde{\tau} = \left\| \frac{\sum_{i}^{n} \tau_{mr}(j, i)^{\top}}{n} \right\|, \quad j \in [1, 6],$$

where  $\tau_{mr}$  denotes the stored joints' effort measurement as a matrix. A lower value of this index is desired for the same task.

Figure 8 shows different postures of the manipulator during the experiments. Figure 9 shows the end-effector interaction force in z direction at task space for both cases, which verifies that a very similar level of task effort (slightly more effort in case of optimal location) was performed by the robot with both the non-optimal and optimal base locations. Table 2

**Table 2** Joints' efforts in different postures and contact conditions. First Row: Less joint-torques are required in optimum posture compared to the non-optimum while holding a pendulum. Second Row: More effort is required for the gravity compensation in the optimum posture compared to the non-optimum, indicating that the less torque requirement in the First Row with the optimum posture, is obtained despite larger gravity compensation compared to the non-optimum posture

	$\tilde{\tau}_{nopt}$	$ ilde{ au}_{opt}$	Difference (%)
Pendulum	2.1974	1.9931	9.3
Free Space	2.1661	2.2715	-4.9

gives the joint torque effort computations for both robot configurations when the robot was pushing the pendulum and when the robot was in free space without pushing the pendulum. Moving the base location to a desired position reduces the required joints' efforts around 9.3% for the same task: in the non-optimum case  $\tilde{\tau}_{nopt} = 2.1974$  and in the optimum posture  $\tilde{\tau}_{opt} = 1.99$ . This is achieved despite the fact that slightly more force is applied to the end-effector while the robot was in the optimum posture, as seen in Fig. 9.

In a given posture, the total joint torques are determined by the gravitational effect (the torques required to hold the robot at that joint configuration) and the force applied to the end effector (or any external disturbances effecting the system). To analyse the effect of the gravitational term in the effort calculations, we moved the arm to the previously obtained joint configuration in free space (without the pendulum, no force is applied at end-effector) and joints efforts are analysed in these configurations. It was observed that gravitational term in optimal posture is slight higher than the non-optimal posture, around 4.9%: in the free space non-optimum case  $\tilde{\tau}_{nopt} = 2.1661$  and in the optimum posture  $\tilde{\tau}_{opt} = 2.2715$ (Table 2). Despite this, the optimal posture overall requires less effort to apply the same level of force at the end effector. This verifies that the difference of 9.3% effort with the optimal base location is indeed due to optimally locating the base for maximum effort transmission.

Finally, a valve turning task at optimum base location was executed with the following experimental protocol (Experiment 3);

- 1. Manipulator grasps the valve under base disturbances,
- 2. Optimum base position is estimated,
- 3. Platform is moved to the optimum location,
- 4. Valve is manipulated with a pre-defined trajectory  $(0 \pi/2 \operatorname{rad})$ .

Different experimental scenarios were carried out for comparisons. First, a benchmark performance index was obtained via manipulating the valve with a fixed base location (T0) (platform is at its nominal position). Then, another experiment (T1) was carried out assuming that the robotic arm

<sup>&</sup>lt;sup>1</sup> The non-optimum posture for the experiment was set in a way that the manipulator can reach and execute the task within its workspace.

Table 3 Overall directional manipulability of the robotic arm mounted on the platform through the valve turning

Tests	Fixed veh. position: Benchmark	Optimal 2D pos. (T1): Ground VMS	Opt. 3D pos. (T2): UVMS	Optimal 3D pos. plus dist. (T3): UVMS under dist.
o <sub>total</sub>	161.85	283.47	428.61	391.62

Bold values indicate the main results

was mounted on a ground based vehicle which had omnidirectional motion capabilities which could translate the manipulator base location in two directions. Thus, an optimum base location was estimated in x- and y-directions (2D) and quartic splines were implemented to generate a smooth path for the vehicle in between the initial and optimal locations.

As a final effort, it was assumed that there existed an underwater vehicle manipulator system (UVMS) such that the vehicle could be controlled to move in x-, y-, and zdirections (3D). Optimal base locations were estimated for these directions and an AUV model along with its controller were implemented to generate the motions of the vehicle. The experiment repeated while assuming that there was no (T2) and there existed (T3) environmental disturbances affecting the vehicle motion in all three directions. The following position disturbance adopted from Dunnigan and Wronka (2011) was applied to evaluate the performance of the task execution controller

$$p_i(t) = Asin\left(\frac{2\pi t}{T_i} + \psi_i\right) \text{ for } i \in \{x, y, z\},$$

where A = 0.02 m denotes the amplitude,  $T_i$  is the period  $(T_x = 3 \text{ s}, T_y = 2.25 \text{ s}, T_z = 4.5 \text{ s})$ , and  $\psi_i$  is the phase  $(\psi_x = \pi/2 \text{ rad}, \psi_y = 0 \text{ rad}, \psi_z = \pi \text{ rad})$ . It must be stated that, the disturbance was unknown to the position controller and a camera in conjunction with the fiducial marker was utilized to measure the deviation from the desired path.

The performance index of the manipulator, as motivated, was enhanced at the optimum base locations, see the obtained total optimization index ( $o_{total} = \sum_{i=0}^{n} o_i$ ) within the aforementioned experiments in Table 3. In 2D optimization the overall directional manipulability was increased 75% and in 3D localization it is boosted approximately 165% compared to the fixed base position. Perturbations, naturally, reduce the overall performance as the base would be displaced from the desired position, see Fig. 10 for the base motion, optimum location, and deviation from the path due to the disturbances during T3. Figure 11 illustrates the base motion in the last experiment (T3) and the estimated optimum positions for all the experiments within the constrained search space.

For a safe operation, not exceeding the maximum force/ torque values applicable to the manipulator and the manipulated object is essential along with adequately attenuating any undesired base motion at the contact point. The interaction force and torque values during the last experiment (T3)



Fig. 10 The platform performs the motion of a hypothetical underwater vehicle under disturbances



**Fig. 11** Estimated optimum points (within T1, T2, and T3) and measured platform position in the last experiment within the constrained motion space

are illustrated in Fig. 12, where the force and torque values caused by the interaction are not exceeding 2N and 0.2 N m, respectively, with the help of the implemented force/torque controller. Also, the reset control architecture sufficiently attenuates any base motion effect at the contact point such that  $\max(|E|) = 0.005$  m, which can be compensated with the compliance mechanism of the gripper.



**Fig. 12** Force and torque measurement during the interaction where S and F denotes the start and finish points of the task execution. Bottom graph: position error at the contact point caused by the undesired base motions

To observe the effects of the interaction forces on the vehicle, the dynamics of an AUV, the Falcon underwater vehicle was emulated using the Stewart platform, via implementing (13) and (14) with the dynamic parameters adopted from Berg (2012) and Manhães et al. (2016). The simulated AUV position is transferred to the platform in addition to the artificially generated disturbances and the computed optimum location as a reference trajectory. One can state that the proposed approach is applicable to such underwater vehicles with strong controllers as the deviation of the base vehicle from the reference position is almost negligible and the implementation of the force/torque and the reset controllers leads to the limited interaction forces. Parameters used in the control architecture and the optimization algorithm (Bonyadi & Michalewicz, 2017) are given in Table 1. We refer the reader to Scherzinger et al. (2019) and Tugal et al. (2018) for more detailed information about the impact of the manipulator's model and impedance controller used in FDC approach and admittance controller parameters on the system's behaviour, respectively.

# **4** Conclusion

This study validates the feasibility of the directional manipulability to augment the force/torque transmission and velocity control from joint space to the task space for a mobile robotic manipulator system. Constrained particle swarm optimization was implemented to find an optimal position for the vehicle. To attenuate the undesired motion at the contact point along with avoiding excessive motions near the singular joint configurations and to establish a stable interaction with an unknown object; reset, forward dynamic, and impedance controllers were developed for the robotic arm. In this approach we combine a metaheuristic optimisation technique and the directional optimization index with the control architecture and in this way we both enhance manipulability and ensure stability while interacting. This approach can be used

The proposed optimization and control algorithms were evaluated with an experimental setup allowing to emulate mobile robotic manipulator dynamics within a limited motion space. The results show that a wheel type valve, typically used in underwater assets, can be manipulated at an optimum vehicle position where force transmission and fine velocity control properties of the robotic arm mounted on it are enhanced. Also, the arm can apply executable force/torque in the rotation direction of the valve in a safe manner despite moderate disturbances affecting the system.

with any vehicle-manipulator system or stationary robotic

**Funding** Funding was provided by Engineering and Physical Sciences Research Council (Grant No. EP/R026173/1).

## Declarations

systems.

**Conflict of interest** The authors declare that they have no conflict of interest.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecomm ons.org/licenses/by/4.0/.

### References

- Andaluz, V., Roberti, F., Toibero, J. M., & Carelli, R. (2012). Adaptive unified motion control of mobile manipulators. *Control Engineering Practice*, 20(12), 1337–1352. https://doi.org/10.1016/j. conengprac.2012.07.008
- Banos, A., & Barreiro, A. (2012). Reset control systems. Berlin: Springer.
- Barbalata, C., Dunnigan, M. W., & Petillot, Y. (2018). Coupled and decoupled force/motion controllers for an underwater vehiclemanipulator system. *Journal of Marine Science and Engineering*, 6(3), 1–23. https://doi.org/10.3390/JMSE6030096

- Berg, V. (2012). Development and Commissioning of a DP system for ROV SF 30k. PhD thesis, Norwegian University of Science and Technology.
- Bonyadi, M. R., & Michalewicz, Z. (2017). Particle swarm optimization for single objective continuous space problems: A review. *Evolutionary Computation*, 25(1), 1–54.
- Carriker, W., Khosla, P., & Krogh, B. (1991). Path planning for mobile manipulators for multiple task execution. *IEEE Transactions on Robotics and Automation*, 7(3), 403–408.
- Cetin, K., Zapico, C. S., Tugal, H., Petillot, Y., Dunnigan, M., & Erden, M. S. (2021). Application of adaptive and switching control for contact maintenance of a robotic vehicle-manipulator system for underwater asset inspection. *Frontiers in Robotics and AI*, 8, 215. https://doi.org/10.3389/frobt.2021.706558
- Chiu, S. L. (1988). Task compatibility of manipulator postures. *The International Journal of Robotics Research*, 7(5), 13–21.
- Coello, C. A. C., Pulido, G. T., & Lechuga, M. S. (2004). Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3), 256–279.
- Colgate, J. E., & Hogan, N. (1988). Robust control of dynamically interacting systems. *International Journal of Control*, 48(1), 65– 88. https://doi.org/10.1080/00207178808906161
- Corne, D., Dorigo, M., Glover, F., Dasgupta, D., Moscato, P., Poli, R., & Price, K. V. (1999). *New ideas in optimization*. Cambridge: McGraw-Hill.
- Dang, Q. V., & Nguyen, L. (2016). A heuristic approach to schedule mobile robots in flexible manufacturing environments. *Procedia CIRP*, 40, 390–395. https://doi.org/10.1016/j.procir.2016.01.073
- Davó, M. A., Gouaisbaut, F., Baños, A., Tarbouriech, S., & Seuret, A. (2018). Exponential stability of a PI plus reset integrator controller by a sampled-data system approach. *Nonlinear Analysis: Hybrid Systems*, 29, 133–146. https://doi.org/10.1016/j.nahs.2018.01.008 arXiv:1603.02458.
- De Luca, A., Oriolo, G., & Giordano, P. R. (2006). Kinematic modeling and redundancy resolution for nonholonomic mobile manipulators. In *IEEE international conference on robotics and automation* (pp. 1867–1873). https://doi.org/10.1109/ROBOT.2006.1641978
- Di Lillo, P., Simetti, E., Wanderlingh, F., Casalino, G., & Antonelli, G. (2020). Underwater intervention with remote supervision via satellite communication: Developed control architecture and experimental results within the dexrov project. *IEEE Transactions* on Control Systems Technology, 29(1), 108–123. https://doi.org/ 10.1109/tcst.2020.2971440
- Dunnigan, M. W., & Wronka, C. M. (2011). Comparison of control techniques for a robotic manipulator with base disturbances. *IET Control Theory and Applications*, 5(8), 999–1012. https://doi.org/ 10.1049/iet-cta.2010.0331
- Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. In *IEEE sixth international symposium on micro* machine and human science (pp. 39–43). https://doi.org/10.1109/ MHS.1995.494215
- Fan, Q., Gong, Z., Tao, B., Gao, Y., Yin, Z., & Ding, H. (2021). Base position optimization of mobile manipulators for machining large complex components. *Robotics and Computer-Integrated Manufacturing*, 70, 102–138. https://doi.org/10.1016/j.rcim.2021. 102138
- Fossen, T. I. (2011). Handbook of marine craft hydrodynamics and motion control. New York: Wiley. https://doi.org/10.1002/ 9781119994138
- He, Q., & Wang, L. (2007). A hybrid particle swarm optimization with a feasibility-based rule for constrained optimization. *Applied Mathematics and Computation*, 186(2), 1407–1422. https://doi.org/10. 1016/j.amc.2006.07.134
- Isiet, M., & Gadala, M. (2019). Self-adapting control parameters in particle swarm optimization. *Applied Soft Computing Journal*, 83, 105653. https://doi.org/10.1016/j.asoc.2019.105653

- Jamisola, R. S., Oetomo, D. N., Ang, M. H., Khatib, O., Lim, T. M., & Lim, S. Y. (2005). Compliant motion using a mobile manipulator: An operational space formulation approach to aircraft canopy polishing. *Advanced Robotics*, 19(5), 613–634. https://doi.org/10. 1163/156855305323383820
- Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In Proceedings of the IEEE international conference on neural networks (pp. 1942–1948).
- Korpela, C., Orsag, M., & Oh, P. (2014). Towards valve turning using a dual-arm aerial manipulator. In *IEEE international conference* on intelligent robots and systems (pp. 3411–3416). https://doi.org/ 10.1109/IROS.2014.6943037
- Krohling, R. A., & Dos Santos Coelho, L. (2006). Coevolutionary particle swarm optimization using gaussian distribution for solving constrained optimization problems. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 36*(6), 1407– 1416. https://doi.org/10.1109/TSMCB.2006.873185
- Lee, D., Lee, W., Park, J., & Chung, W. K. (2020). Task space control of articulated robot near kinematic singularity: Forward dynamics approach. *IEEE Robotics and Automation Letters*, 5(2), 752–759. https://doi.org/10.1109/LRA.2020.2965071
- Lippiello, V., & Ruggiero, F. (2012). Exploiting redundancy in Cartesian impedance control of UAVs equipped with a robotic arm. In *IEEE/RSJ international conference on intelligent robots and systems* (pp. 3768–3773). https://doi.org/10.3182/20120905-3-HR-2030.00158
- Lozano-Perez, T. (2012). Autonomous robot vehicles. Berlin: Springer.
- Manhães, M. M. M., Scherer, S. A., Voss, M., Douat, L. R., & Rauschenbach, T. (2016). UUV Simulator: A Gazebo-based package for underwater intervention and multi-robot simulation. In OCEANS 2016 MTS/IEEE Monterey (pp. 1–8). https://doi.org/10. 1109/OCEANS.2016.7761080
- Mitsi, S., Bouzakis, K. D., Sagris, D., & Mansour, G. (2008). Determination of optimum robot base location considering discrete end-effector positions by means of hybrid genetic algorithm. *Robotics and Computer-Integrated Manufacturing*, 24(1), 50–59. https://doi.org/10.1016/j.rcim.2006.08.003
- Moura, J., McColl, W., Taykaldiranian, G., Tomiyama, T., & Erden, M. S. (2018). Automation of train cab front cleaning with a robot manipulator. *IEEE Robotics and Automation Letters*, 3(4), 3058– 3065. https://doi.org/10.1109/LRA.2018.2849591
- Parsopoulos, K., & Vrahatis, M. (2002). Recent approaches to global optimization problems through particle swarm optimization. *Natural Computing*, 1(2/3), 235–306. https://doi.org/10.1023/A: 1016568309421
- Ram, R. V., Pathak, P. M., & Junco, S. J. (2019). Trajectory control of a mobile manipulator in the presence of base disturbance. *Simulation*, 95(6), 529–543. https://doi.org/10.1177/0037549718784186
- Ren, S., Xie, Y., Yang, X., Xu, J., Wang, G., & Chen, K. (2017). A method for optimizing the base position of mobile painting manipulators. *IEEE Transactions on Automation Science* and Engineering, 14(1), 370–375. https://doi.org/10.1109/TASE. 2016.2612694
- Rigotti-Thompson, M., Torres-Torriti, M., Auat Cheein, F. A., & Troni, G. (2020).  $\mathcal{H}_{\infty}$ -Based terrain disturbance rejection for hydraulically actuated mobile manipulators with a nonrigid link. *IEEE/ASME Transactions on Mechatronics*, 25(5), 2523–2533. https://doi.org/10.1109/TMECH.2020.2983072
- Ruggiero, F., Lippiello, V., & Ollero, A. (2018). Aerial manipulation: A literature review. *IEEE Robotics and Automation Letters*, 3(3), 1957–1964. https://doi.org/10.1109/LRA.2018.2808541
- Runarsson, T. P., & Yao, X. (2005). Search biases in constrained evolutionary optimization. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 35(2), 233–243. https://doi.org/10.1109/TSMCC.2004.841906

- Scherzinger, S., Roennau, A., & Dillmann, R. (2017). Forward Dynamics Compliance Control (FDCC): A new approach to cartesian compliance for robotic manipulators. In *IEEE international conference on intelligent robots and systems (IROS)* (pp. 4568–4575). https://doi.org/10.1109/IROS.2017.8206325
- Scherzinger, S., Roennau, A., & Dillmann, R. (2019). Inverse kinematics with forward dynamics solvers for sampled motion tracking. In *IEEE 19th international conference on advanced robotics (ICAR)* (Vol. 1, pp. 681–687). https://doi.org/10.1109/ICAR46387.2019. 8981554, arXiv:1908.06252
- Shi, Y., & Eberhart, R. (1998). A modified particle swarm optimizer algorithm. In *IEEE international conference on evolutionary computation* (pp. 69–73). https://doi.org/10.1109/ICEMI.2007. 4350772
- Takahama, T., & Sakai, S. (2006). Constrained optimization by the ε constrained differential evolution with gradient-based mutation and feasible elites. In *IEEE congress on evolutionary computation*, *CEC*, 2006 (pp. 1–8). https://doi.org/10.1109/cec.2006.1688283
- Tugal, H., Gautier, B., Kircicek, M., & Erden, M. S. (2018). Handimpedance measurement during laparoscopic training coupled with robotic manipulators. In *IEEE/RSJ international conference* on intelligent robots and systems (IROS 2018) (pp. 4404–4410).
- Villaverde, A., Blas, A., Carrasco, J., & Torrico, A. (2011). Reset control for passive bilateral teleoperation. *IEEE Transactions on Industrial Electronics*, 58(7), 3037–3045. https://doi.org/10.1109/TIE.2010. 2077610
- Wang, J., & Olson, E. (2016). AprilTag 2: Efficient and robust fiducial detection. In *IEEE international conference on intelligent robots* and systems (pp. 4193–4198). https://doi.org/10.1109/IROS.2016. 7759617
- Xiaohui, H., & Eberhar, R. (2002). Solving constrained nonlinear optimization problems with particle swarm optimization. In *Proceed*ings of the sixth world multiconference on systemics, cybernetics and informatics (Vol. 5, pp. 203–206).
- Yamamoto, Y., & Yun, X. (1994). Coordinating locomotion and manipulation of a mobile manipulator. *IEEE Transactions on Automatic Control*, 39(6), 1326–1332. https://doi.org/10.1109/9.293207
- Yamamoto, Y., & Yun, X. (1996). Effect of the dynamic interaction on coordinated control of mobile manipulators. *IEEE Transactions* on Robotics and Automation, 12(5), 816–824. https://doi.org/10. 1109/70.538986
- Yoshikawa, T. (1985). Manipulability of robotic mechanisms. International Journal of Robotics Research, 4(2), 3–9.
- Yu, Q., Wang, G., Hua, X., Zhang, S., Song, L., Zhang, J., & Chen, K. (2018). Base position optimization for mobile painting robot manipulators with multiple constraints. *Robotics and Computer-Integrated Manufacturing*, 54, 56–64. https://doi.org/10.1016/j. rcim.2018.05.007
- Zhan, Z. H., Zhang, J., Li, Y., & Chung, H. S. H. (2009). Adaptive particle swarm optimization algorithm. *IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics*, 36(6), 1362– 1381. https://doi.org/10.1109/TSMCB.2009.2015956
- Zhao, G., Nešić, D., Tan, Y., & Hua, C. (2019). Overcoming overshoot performance limitations of linear systems with reset control. *Automatica*, 101, 27–35. https://doi.org/10.1016/j.automatica.2018.11. 038

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



781



Harun Tugal received the B.Sc. degree in electrical and electronics engineering from Karadeniz Technical University, Trabzon, Turkey, in 2011, the M.Sc. degree in advanced control and system engineering and the Ph.D. degree in electrical and electronics engineering from Manchester University, Manchester, U.K., in 2013 and 2017, respectively. He was a Research Associate with the Institute of Sensors, Signals and Systems, Heriot-Watt University, Edinburgh, U.K. His research

interests include human-robot interaction, bilateral teleoperation, underwater robotic systems, and machine learning.



Kamil Cetin received his B.Sc., M.Sc. and Ph.D. degrees in Electrical and Electronics Engineering from Dumlupinar University in 2002, Dokuz Eylul University in 2007 and Izmir Institute of Technology in 2016, respectively. During his Master studies, he also worked as an Electronics Engineer in a healthcare company in Izmir for four years. Then, he worked at two EU funded FP7 projects (ASPIRE and ISISEMD) and pursued his career as a Research Assistant in the Elec-

tronic Systems Department at Aalborg University in Denmark for three years. During his Ph.D. studies, he worked as a Researcher within a national project supported by The Scientific and Technological Research Council of Turkey. After his Ph.D. graduation, he worked at an EU funded H2020 project (WeDRAW) as a Post-Doctoral Researcher in the Robotics Brain Cognitive and Science Department at the Italian Institute of Technology in Genova for two years. From 2018 to 2021 he was as a Research Associate at Heriot-Watt University within ORCA Hub project. His research interests include robot manipulator control, telerobotic systems, nonlinear control methods, and underwater robotic systems.





Yvan Petillot is a full Professor of Robotics and Computer Vision at Heriot-Watt University. He is a leading member of the Oceans Systems Laboratory, the deputy director of the Institute for Sensor Signals and Systems and the deputy director of the joint research institute in Signal and Image Processing (ERP-SIP) with Edinburgh University. He is also currently a Royal Society Industry Fellow in collaboration with SeeByte Ltd, a company he cofounded in 2001 and in which he

was Chief Technical Officer until 2010. With over 15 years experience in Robotics, Image Processing and Autonomous Systems in the maritime domain, he has made very significant contributions to target detection and classification multiple vehicle collaboration and autonomous inspection and manipulation. He has a long history of successful collaborations with other leading researchers in Europe through a number of European Research Projects (ALIVE, Autotracker, Amason, 2005, FreeSubNet 2008, Trident, 2010, Eurathlon 2012) totaling over £1.5M of research funding. In parallel, Professor Petillot has a unique track record in technology transfer and engagement with industry.



Matthew Dunnigan background is in the area of modelling, control and parameter estimation of dynamic systems and he has over twenty years of relevant experience. He is a graduate of Glasgow University (B.Sc. in Electrical and Electronic Engineering) and Heriot-Watt University (M.Sc. in Digital Techniques and Ph.D. in Electrical Engineering). He is a Fellow of the Institute of Engineering and Technology. He is currently a Senior Lecturer and his research has concentrated on

the development and implementation of adaptive, state-space, neural/fuzzy and nonlinear control methods applied to different application domains (robotic manipulators/underwater vehicles, heave compensators, electric motors, laser cutting and electrodynamic actuators). Control techniques researched include self-tuning adaptive, nonlinear sliding-mode, fuzzy/neural, state-space estimators and adaptive inverse control.



Mustafa Suphi Erden received the B.S., M.S., and Ph.D. degrees in Electrical and Electronics Engineering from Middle East Technical University, Ankara, in 1999, 2001, and 2006. From 2007 to 2012 he was a postdoctoral researcher, successively in Delft University of Technology, the Netherlands; in Ecole Nationale Techniques Supérieure de Avancées-ParisTech, France: in Univ. Pierre & Marie Curie - Paris 6, France. He is an Assistant Professor with the School of Engi-

neering and Physical Sciences at Heriot-Watt University in Edinburgh, UK, since 2015. His research interests include physical human-robot interaction, assistive robotics, skill assistance, mechatronics design, medical robotics, walking robots, and machine learning.