SCIENCE CHINA
Information Sciences

July 2020, Vol. 63 170204:1–170204:15
https://doi.org/10.1007/s11432-019-2741-6

• RESEARCH PAPER •
Special Focus on Bio-Robotic Systems: Modeling, Design, Control and Perception

Design and analysis of a whole-body controller for a velocity controlled robot mobile manipulator

Mantian LI[1], Zeguo YANG[1], Fusheng ZHA[1,2*], Xin WANG[2], Pengfei WANG[1], Ping LI[3], Qinyuan REN[3] & Fei CHEN[4*]

[1]*State Key Laboratory of Robotics and Systems, Harbin Institute of Technology, Harbin* 150001*, China;*
[2]*Shenzhen Academy of Aerospace Technology, Shenzhen* 518057*, China;*
[3]*Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou* 310027*, China;*
[4]*Active Perception and Robot Interactive Learning Laboratory, Department of Advanced Robotics, Istituto Italiano di Tecnologia, Genova* 30 16160*, Italy*

Received 17 August 2019/Revised 14 October 2019/Accepted 29 November 2019/Published online 25 May 2020

Abstract Mobile manipulators, which are intrinsically redundant when the manipulator and mobile base are moving together, are known for their capabilities to carry out multiple tasks at the same time. This paper presents a whole-body control framework, inspired by legged bio-robots, for a velocity controlled non-holonomic mobile manipulator based on task priority. Control primitives, such as manipulability optimization, trajectory tracking of the end-effector and mobile base, and collision avoidance, are considered in the framework and arranged at different priorities. Lower priority tasks are projected into the null space of control tasks with higher priorities. As a result, lower level tasks are completed without affecting the performance of higher priority tasks. Several experiments are implemented to verify the effectiveness of the proposed controller. The proposed method is proved to be an effective way to solve the whole-body control problem of velocity controlled mobile manipulators.

Keywords mobile manipulator, whole-body control, velocity controlled, task priority, null-space projection

Citation Li M T, Yang Z G, Zha F S, et al. Design and analysis of a whole-body controller for a velocity controlled robot mobile manipulator. Sci China Inf Sci, 2020, 63(7): 170204, https://doi.org/10.1007/s11432-019-2741-6

1 Introduction

With the rapid development of artificial intelligence (AI) technologies in recent years, more and more AI technologies have been applied to different kinds of bio-robotic systems [1–6]. Legged bio-robotic systems mounted with manipulators have been drawing much attention of both industrial and academic community for their advantages of combining manipulation and locomotion in uneven terrains. When robots are applied in flat areas, legged robotic systems are replaced with wheeled systems for stability, which are also called wheeled mobile manipulators [7–15]. When manipulators and mobile base are cooperatively moving together, the mobile manipulators can also be considered as robots with redundant degrees of freedom. As a result, both legged and wheeled mobile manipulators have the advantage of carrying out multiple tasks, but the problem lies in the arrangement of these tasks in the controller.

Inspired by whole-body motion of humans and animals, control of bio-robots often is composed of control of multiple control primitives, such as position control of the center of mass (CoM), position and force control of hands and feet, and collision avoidance. Whole-body control is a widely used methodology

* Corresponding author (email: zhafusheng@hit.edu.cn, fei.chen@iit.it)

to simultaneously achieve multiple control tasks and regulate the priorities of these tasks. Null-space projection technique is an effective method to regulate task priorities, which represents tasks with Jacobian matrices and projects the Jacobian matrices of lower priority tasks into the null space of those of higher priority tasks. In addition, null-space projection of Jacobian matrices is also a commonly used methodology to solve the control problem of redundant robots [16]. Siciliano and Slotine [17] proposed a framework to solve the multiple tasks control problem of redundant manipulators based on task priority [18] and achieve task priorities via null-space projection technique. Mobile manipulators are redundant robots with mobile base, so task priority strategy based on null space is applied to deal with the similar problem of mobile manipulators. Khatib et al. [19] presented a method to decompose operational and postural control tasks of human-like robots and discussed the dynamic behavior of the secondary control objectives. Furthermore, they extended their control structure to a multi-level hierarchical control strategy [20] and put forward a method to classify all control objectives as three categories (i.e., postural tasks, operational tasks and physical constraints) [21]. To handle the discontinuity caused by the insertion of unilateral tasks, Mansard et al. [22] proposed a continuous controller based on pseudo-inverse of Jacobian matrix.

The abovementioned controllers are mainly implemented on legged bio-robots. Whole-body control methodologies based on null-space projections are also utilized to solve the coordinating control of wheeled mobile manipulators. To achieve whole-body control of wheeled mobile robots, Dietrich et al. [23] put forward a task-based whole-body controller via impedance control and verified the efficacy of the controller on a wheeled mobile robot called Rollin' Justin [24]. Several commonly used control objectives are contained in the controller. Because the upper body of the robot is mainly torque controlled, while the mobile base is velocity controlled, an admittance control interface is introduced [25]. In addition, Adorno et al. [26, 27] utilized a dual-quaternion-based method to represent the configuration of wheeled mobile manipulators and proposed a whole-body controller based on task priority. Some other researchers have also proposed controllers of wheeled mobile manipulators without referring to the whole-body control concept. An online approach, which allows users to assign weighting factors to individual tasks, was presented in [28]. A unified approach for planning and control of mobile manipulators by combining the event-based planning and control method with the nonlinear feedback technique was proposed in [29]. Ref. [30] put forward the elastic strip technique to deal with task-consistent and non-task-consistent behaviors of mobile manipulators. Ref. [31] solved the trajectory tracking problem of mobile manipulators based on model predictive control. Ref. [32] came up with a feedback controller to achieve dragging and following tasks of mobile manipulators. Ref. [33] provided a flexible reactive control framework to satisfy different constraints with system redundancy.

Joint velocities are frequently-used control inputs for manipulators. Different from torque controlled robots, controllers for all control primitives should be managed in velocity level for such kind robots. In addition, interactions between the robot and environment, which are often measured by force sensors mounted on the end-effector, must be transformed into end-effector twists and handled by changing joint velocities. Contributions of this paper mainly consist of three aspects. Firstly, because manipulators mounted on mobile bases do not work in independent workspace as traditional industrial manipulators and sometimes have to cooperate with human workers, mobile manipulators are often collaborative manipulators for safety. Most academic research on whole-body control of mobile robots with collaborative manipulators is about torque controlled manipulators. However, famous collaborative manipulators, such as Universal Robots and TechmanRobot which are prevalent in industrial applications, are velocity or position controlled and have no access to torque control. Hence, dynamical model of the robotic systems cannot be utilized. In this paper, we present a whole-body controller for velocity controlled mobile manipulators with commonly used collaborative manipulators based on task priority. Both joint position and joint velocity control approaches are available for these collaborative manipulators, but only velocity inputs are available for most mobile base. For consistency, we choose joint velocity as the input of manipulator in our algorithm. Moreover, a method to deal with interactions between mobile manipulators and the environment by changing whole-body joint velocities is presented. The interaction forces between the end-effector of the manipulator and the environment are measured through a six dimensional force/torque sensor mounted on the end-effector of the manipulator and transformed into velocities of the end-effector

Figure 1 (Color online) System overview.

via admittance control laws. Furthermore, those interaction velocities are transformed into joint velocity inputs of the manipulator and the mobile base via inverse differential kinematics. Finally, a new concept called extended Cartesian coordinates is proposed in order to take non-holonomic constraints into consideration all the time. Several practical experiments are implemented to verify the effectiveness of the proposed controller and discuss its performance based on the results of the experiments. Then, we make a comparison with similar research carried out by previous researchers.

2 Kinematic modeling

The mobile robotic system used in this research is shown in Figure 1. The hardware system can be simply described by a differential driven base equipped with a UR10 robot, a six degrees of freedom (6-DOFs) the manipulator, which is velocity/position controlled. The effectiveness of velocity control and that of position control of manipulator are the same. As mentioned above, we choose joint velocities as inputs of the manipulator just for consistency of control inputs. Manipulators with such kind of control inputs are much safer and cheaper than those collaborative manipulators with torque control inputs, because the torque sensors mounted on each joint are a significant part of the costs of manipulators. The whole-body configuration can be represented by $\boldsymbol{q} = [x_{\rm b}, y_{\rm b}, \theta, q_1, \ldots, q_6]^{\rm T}$, where $\boldsymbol{q}_{\rm b} = [x_{\rm b}, y_{\rm b}, \theta]^{\rm T}$ represents the mobile base configuration, $\boldsymbol{q}_{\rm m} = [q_1, \ldots, q_6]^{\rm T}$ represents the upper-arm configuration.

The roll-pitch-yaw (α, β, γ) Euler angles are applied to represent the orientation of the end-effector, so its six-dimensional position can be expressed by

$$\boldsymbol{x} = [x, y, z, \alpha, \beta, \gamma]^{\rm T}. \tag{1}$$

Then, whole-body forward kinematics can be expressed by

$$\boldsymbol{x} = f(\boldsymbol{q}). \tag{2}$$

Whole-body differential kinematics is obtained by differentiating (2):

$$\dot{\boldsymbol{x}} = J_1 \dot{\boldsymbol{q}}. \tag{3}$$

Because the mobile base is differential driven, the following kinematic constraint must be satisfied:

$$\boldsymbol{J}_{\rm bkc} \dot{\boldsymbol{q}} = 0, \tag{4}$$

where

$$\boldsymbol{J}_{\mathrm{bkc}} = [\cos\theta, \sin\theta, 0, \ldots, 0] \in \mathbb{R}^9$$

is the kinematic constraint expressed in whole-body Jacobian form, which must be satisfied by any control objectives. Inspired by the extended Jacobian concept in [34], we combine kinematic constraint Jacobian with the whole-body Jacobian and put forward the extended Cartesian coordinates as follows:

$$\dot{\boldsymbol{x}}_{\mathrm{e}} = \boldsymbol{J}_{\mathrm{e}}\dot{\boldsymbol{q}}, \tag{5}$$

where

$$\dot{\boldsymbol{x}}_{\mathrm{e}} = [0, x, y, z, \alpha, \beta, \gamma]^{\mathrm{T}}, \tag{6}$$

$$\boldsymbol{J}_{\mathrm{e}} = \begin{bmatrix} \boldsymbol{J}_{\mathrm{bkc}} \\ \boldsymbol{J}_1 \end{bmatrix}. \tag{7}$$

The extended Cartesian coordinates tracking is treated as the main task in this paper so that the kinematic constraint can be satisfied all the time. Only collisions that could be avoided within the null space of the main task will be discussed. When the distance between the robot link and obstacles is smaller than a user-defined distance $d_{\min}$, we deem that the collision cannot be avoided. The robot will be stopped for safety when encountering inevitable collisions.

3 Whole-body controller

Control primitives represent the intentions of bio-robots, while whole-body controller is responsible for the arrangement of multiple intentions. This section provides a detailed introduction to the structure of whole-body hierarchical controller, end-effector Cartesian controller, base cartesian controller and control objectives discussed in this paper. The ith priority control objective is defined by Jacobian $\boldsymbol{J}_i$ and related to joint velocity inputs by

$$\dot{\boldsymbol{x}}_i = \boldsymbol{J}_i\dot{\boldsymbol{q}}_i. \tag{8}$$

Joint velocity inputs of lower priority tasks will be projected into the null space of high priority tasks layer by layer and finally into the null space of the main task. As a result, the final whole-body joint velocity inputs can be obtained by

$$\dot{\boldsymbol{q}} = \dot{\boldsymbol{q}}_1 + \boldsymbol{N}_1\dot{\boldsymbol{q}}_2 + \boldsymbol{N}_2\dot{\boldsymbol{q}}_3 + \cdots, \tag{9}$$

where $\boldsymbol{N}_i$ $(i = 1, 2, 3, \ldots)$ are null-space projection matrices. Experiments shown in [35] indicate that null-space projections of augmented Jacobian have better performance than that of successive Jacobian, so augmented Jacobian is utilized in this paper. The null-space projection matrices are computed by

$$\boldsymbol{N}_i = \boldsymbol{I} - \boldsymbol{J}^{\#}_{\mathrm{AU}i}\boldsymbol{J}_{\mathrm{AU}i}, \tag{10}$$

where

$$\boldsymbol{J}^{\#}_{\mathrm{AU}i} = \boldsymbol{w}^{-1}\boldsymbol{J}^{\mathrm{T}}_{\mathrm{AU}i}(\boldsymbol{J}_{\mathrm{AU}i}\boldsymbol{w}^{-1}\boldsymbol{J}^{\mathrm{T}}_{\mathrm{AU}i})^{-1} \tag{11}$$

denotes the pseudo-inverse Jacobian matrix. In this research, the weighting matrix $\boldsymbol{w} = \boldsymbol{I}$ is an identity matrix.

The whole-body joint velocity inputs corresponding to the ith task can be obtained by

$$\dot{\boldsymbol{q}}_i = \boldsymbol{J}^{\#}_i\dot{\boldsymbol{x}}_i. \tag{12}$$

The final whole-body joint velocity inputs would be obtained by substituting (10)–(12) into (9).

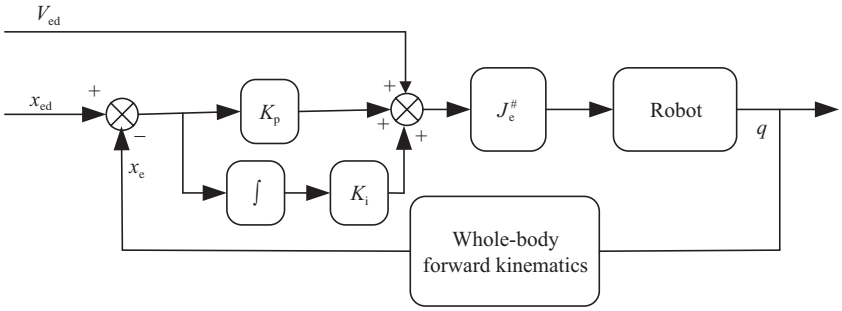

Figure 2 End-effector Cartesian controller.

3.1 End-effector tracking controller

As mentioned above, kinematic constraints must be satisfied all the time, so the new concept called extended Cartesian coordinates will be applied in this study to make sure that non-holonomic constraints could be satisfied for all tasks. As a result, the Cartesian trajectory to be followed should be expressed in the extended form as shown in (6). Because our manipulator is velocity controlled, the operational space tracking controller for a velocity controlled manipulator introduced in [36] has been adopted in this paper for the end-effector to track a user-defined Cartesian trajectory. The controller is briefly shown in Figure 2 and could be formulated as

$$\dot{\boldsymbol{q}}_i = \boldsymbol{J}_{\mathrm{e}}^{\#}\left(\boldsymbol{V}_{\mathrm{ed}} + \boldsymbol{K}_{\mathrm{p}}\boldsymbol{x}_{\mathrm{ee}} + \boldsymbol{K}_{\mathrm{i}}\int \boldsymbol{x}_{\mathrm{ee}}\mathrm{d}t\right), \tag{13}$$

where

$$\boldsymbol{J}_{\mathrm{e}}^{\#} = \boldsymbol{J}_{\mathrm{e}}^{\mathrm{T}}(\boldsymbol{J}_{\mathrm{e}}\boldsymbol{J}_{\mathrm{e}}^{\mathrm{T}})^{-1} \tag{14}$$

denotes the pseudo-inverse of extended Jacobian ($\boldsymbol{J}_{\mathrm{e}}$) of extended Cartesian coordinates. $\boldsymbol{V}_{\mathrm{ed}}$ represents the desired extended Cartesian velocity, which is a twist with the extended term. $\boldsymbol{x}_{\mathrm{ee}}$ represents error of extended Cartesian position, which is a twist with the extended term that takes the end-effector from its current configuration to the desired configuration instead of a vector which is simply the difference between the desired position and the current position. $\boldsymbol{K}_{\mathrm{p}}$ and $\boldsymbol{K}_{\mathrm{i}}$ are proportional and integral positive-definite diagonal gain matrices, respectively. In our case, $\boldsymbol{K}_{\mathrm{i}}$ is set to be zero and $\boldsymbol{K}_{\mathrm{p}} = k_{\mathrm{p}}\boldsymbol{I}$ with $k_{\mathrm{p}} > 0$ and $\boldsymbol{I}$ represents an identity matrix. There is no gravity compensation term because it has been integrated in the built-in controller of the manipulator. As a result, the whole controller is actually a combination of a feedforward controller based on inverse differential kinematics and a proportional feedback controller, that is, the closed-loop reduces to a P controller, which is a linear controller. Without loss of generality, the system can be analyzed as a one-dimensional system. For each servo circle, the desired velocity is constant. Then, the stability becomes obvious. The feedforward controller provides motion even when there exists no position error. A further analysis of the controller reveals that the position error does not exactly converge to zero, but it can be reduced by choosing a larger gain under the physical constraints of robots.

3.2 Base tracking controller

Pose regulation of mobile base is of vital importance for the control of bio-robots. It can be used to guide the robotic system to a target pose along a predefined trajectory and keep the end-effector of the manipulator still. Moreover, it is also a useful tool to regulate the configuration of manipulator and avoid collisions of both manipulator and mobile base. The linear base controller introduced in [11] has been utilized to control the pose of mobile base. The controller is shown in Figure 3 and can be formulated as

$$v = v_{\mathrm{d}}\cos e_3 + k_1 e_1, \tag{15}$$

$$\omega = \omega_{\mathrm{d}} + k_2 e_2 + k_3 e_3, \tag{16}$$

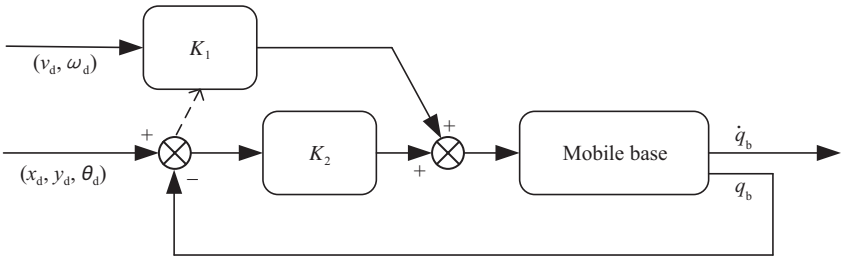

Figure 3 Mobile base Cartesian controller. K_1 and K_2 are matrices determined by tracking error and control objective.

where coefficients $k_1, k_2, k_3 > 0$ are calculated according to [11]; v_d and ω_d are desired driving velocity and steering velocity, respectively; v and ω are final commanded steering velocity and steering velocity, respectively. The tracking error is defined as

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_\mathrm{d} - x \\ y_\mathrm{d} - y \\ \theta_\mathrm{d} - \theta \end{bmatrix}, \tag{17}$$

where $x_\mathrm{d}, y_\mathrm{d}, \theta_\mathrm{d}$ are desired Cartesian coordinates of the mobile base.

In addition, because the mobile velocity may be given as driving velocity v and steering velocity ω or Cartesian velocity $\dot{x}, \dot{y}, \dot{\theta}$, they are related by the following expression:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -\sin\theta & 0 \\ \cos\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}. \tag{18}$$

Differentiating (17) and setting $\sin e_3 = e_3$ leads to the following error dynamics:

$$\dot{\boldsymbol{e}} = \begin{bmatrix} -k_1 & \omega_\mathrm{d} & 0 \\ -\omega_\mathrm{d} & 0 & v_\mathrm{d} \\ 0 & -k_2 & -k_3 \end{bmatrix} \boldsymbol{e},$$

where $\boldsymbol{e} = [e_1, e_2, e_3]^\mathrm{T}$. In this paper, v_d and ω_d are chosen to be constant. Therefore, the system is actually time-invariant. Then, the characteristic polynomial can be easily obtained and its three roots are determined by the control parameters k_1, k_2, k_3 and the constant desired velocities. These parameters are properly chosen according to the stability requirements on the roots, and the system is asymptotically stable.

3.3 Control primitives

Control primitives based on task priority are represented by Jacobian matrices. This subsection discusses detailed formulations of some commonly used control objectives of velocity controlled mobile manipulators, including operational control primitives, postural control primitives, and physical constraints.

3.3.1 *End-effector Cartesian task*

End-effector tasks often represent the main intentions of bio-robots. Similarly, end-effector Cartesian tasks are treated as main tasks in this research. End-effector tasks contain user-defined trajectory tracking and human-robot interaction. Because the aforementioned extended Cartesian coordinates are utilized to describe operational tasks, human-robot interaction effects should also be transformed into extended Cartesian coordinates or their time derivatives. Then, the combined effects will be transformed into whole-body joint velocity inputs, that is

$$\dot{\boldsymbol{q}}_\mathrm{e} = \boldsymbol{J}_\mathrm{e}^{\#}(\boldsymbol{V}_\mathrm{e} + \boldsymbol{V}_\mathrm{ei}), \tag{19}$$

where $\boldsymbol{V}_{\rm ei}$ denotes the interaction effects between the end-effector of the robot and environment. The interaction effects are sensed by a six-dimensional force/torque sensor and transformed into an additional twist of the end-effector via a proportional admittance controller. That is $\boldsymbol{V}_{\rm ei} = \boldsymbol{K} \cdot \boldsymbol{F}_{\rm ei}$, where $\boldsymbol{K}$ is a user-defined diagonal matrix with all diagonal elements being positive, $\boldsymbol{F}_{\rm ei}$ is the forced measured by the force/torque sensor. The interaction effects are obvious and will not be displayed in the experiments.

3.3.2 *Collision avoidance*

Getting information of the environment via vision system is an important part of bio-robots. Because robot vision is beyond the scope this research, we assume that all the information about obstacles around the robot has been known in advance. The algorithm presented in [37] is applied to compute the distance between the robot and obstacles in each servo cycle. At the same time, potential collision points on both obstacles and the mobile manipulator can also be obtained. The distance between an obstacle and the robot is calculated by

$$d = \|\boldsymbol{x}_{\rm op} - \boldsymbol{x}_{\rm rp}\|, \tag{20}$$

where $\boldsymbol{x}_{\rm op}$ and $\boldsymbol{x}_{\rm rp}$ are position vectors of potential collision points on the obstacle and the robot, respectively.

Furthermore, the artificial potential field concept introduced in [38] is applied to avoid potential collisions. The relationship between the repulsive force and the distance is described as follows:

$$F_{\rm rep} = \begin{cases} \dfrac{f_{\max}}{d_{\rm start}}(d - d_{\rm start})^2, & d \leqslant d_{\rm start}, \\ 0, & d > d_{\rm start}, \end{cases} \tag{21}$$

where $f_{\max}$ denotes the maximum repulsive force, and $d_{\rm start}$ denotes the distance to activate/deactivate the repulsive force. Notably, the proposed method is only used to achieve collision avoidance between links of the robot and obstacles. Active collision avoidance between the end-effector and obstacles, which is achieved through motion planning, is beyond the scope of this paper. The direction of repulsive force is represented by a unit vector computed by

$$\boldsymbol{n} = \frac{\boldsymbol{x}_{\rm rp} - \boldsymbol{x}_{\rm op}}{d}. \tag{22}$$

For a torque controlled robot, the repulsive force will be transformed into joint torque inputs through Jacobian transpose. Because our robot is velocity controlled, the repulsive force is replaced with a "repulsive velocity", a velocity term with same magnitude computed by (21). Then the "repulsive velocity" will be transformed into joint velocity inputs via the inverse of differential kinematics. In the case that the velocity of the potential collision point on the manipulator is just equal to the repulsive velocity, which is a small probability moment, the potential field meets a local minimum point. Even though the potential collision point may keep still at that moment, the potential point itself is changing all the time as the robot moves, that is, the potential collision point at that moment is not the potential collision point at the next moment. As a result, the robot automatically moves out from the local minimum point.

3.3.3 *Self-collision avoidance*

Humans and animals seldom hit themselves, so self-collision avoidance is also important for bio-robots. To carry out self-collision algorithm, the whole robotic system is divided into three parts. (1) Mobile base is described by a cuboid represented by the coordinates of its eight vertices. (2) For brevity, end-effector part is represented by a sphere which can fully contain the wrist and end-effector of the robot. (3) The upper arm and forearm are both represented by cylinders with proper radii and lengths. The self-collision algorithm proposed by Alexander et al. [39] is applied to solve the self-collision problem of our robot. Analogous to the collision avoidance algorithm, repulsive forces are replaced by "repulsive velocities" and transformed into joint velocity inputs via the inverse of differential kinematics.

3.3.4 *Joint limits*

Joint limits represent physical constraints bio-robots and should be carefully considered to avoid undesirable damage to robotic system. Obviously, only joints of upper arm have physical limits. All joint angles are limited to a scope ranging from $-360°$ to $360°$. The artificial potential field technique [38] is also applied to avoid joint limits. The distance between the ith joint and its limits is computed by

$$d_i = \min(\|\phi_i - \phi_{\rm ll}\|, \|\phi_i - \phi_{\rm ul}\|), \tag{23}$$

where ϕ_i denotes joint angle of the ith joint, $\phi_{\rm ll}$ and $\phi_{\rm ul}$ denote the lower and upper joint limits of the ith joint, respectively.

The "repulsive velocity" of the ith joint is formulated as follows:

$$F_{i,\rm rep} = \begin{cases} k_i d_i^2, & d_i \leqslant \phi_{\rm start}, \\ 0, & d_i > \phi_{\rm start}, \end{cases} \tag{24}$$

where k_i is a positive gain parameter determined by designers, and $\phi_{\rm start}$ is the minimum distance to be free of repulsive force.

3.3.5 *Singularity avoidance*

Singularities are configurations at which a robot loses control in one or more directions and should be carefully avoided in robot motion planning and control. Among several means to keep robot away from singular configurations, the manipulability measurement [40] has been widely used and proved to be an effective way to avoid singularities. Because singularities have nothing to do with the position of mobile base, only configurations of the upper arm will be taken into consideration in this control primitive. The manipulability measurement is defined as

$$\omega(\boldsymbol{q}_{\rm m}) = \sqrt{\det(\boldsymbol{J}(\boldsymbol{q}_{\rm m})\boldsymbol{J}^{\rm T}(\boldsymbol{q}_{\rm m}))}. \tag{25}$$

Increase of the manipulability measurement will cause the manipulator to move away from singularities. One way to achieve this is to choose the derivative of manipulability measurement with respect to joint angles as a secondary task [11]. The corresponding joint inputs can be calculated by the following equation:

$$\dot{\boldsymbol{q}}_{\rm m} = k_0 \left(\frac{\partial \omega(\boldsymbol{q}_{\rm m})}{\partial \boldsymbol{q}_{\rm m}} \right)^{\rm T}, \tag{26}$$

where $k_0 > 0$ is a user-defined gain parameter. By expressing the set of joint velocity inputs in whole-body form and projecting it into the null space of higher priority tasks, the upper arm will be kept away from singularities without affecting the performance of higher priority tasks.

3.3.6 *Base Cartesian task*

Even though trajectory tracking of the end-effector is of vital importance for completing main tasks, trajectory tracking of mobile base is sometimes needed to avoid obstacles on the path without affecting the implementation of main tasks or to regulate the whole-body configuration of bio-robots. The base controller discussed in Subsection 3.2 is utilized to track a prescribed base Cartesian trajectory. In addition, a set of velocity commands can also be given to regulate or finely adjust the posture of upper arm in some cases. To guarantee the performance of high priority tasks, the resultant joint velocity inputs should be projected into the null space of higher priority tasks.

The total number of degrees of freedom (DOFs) of the mobile manipulator is nine. However, the number of DOFs that need to take all the aforementioned control primitives into account is much larger than nine. In fact, only two control primitives can be taken into account at the same time, so most of those control primitives are deactivated. The criterion to activate/deactivate control primitives is stated as follows. (1) Because the operational task of the end-effector is the main task in this paper, it

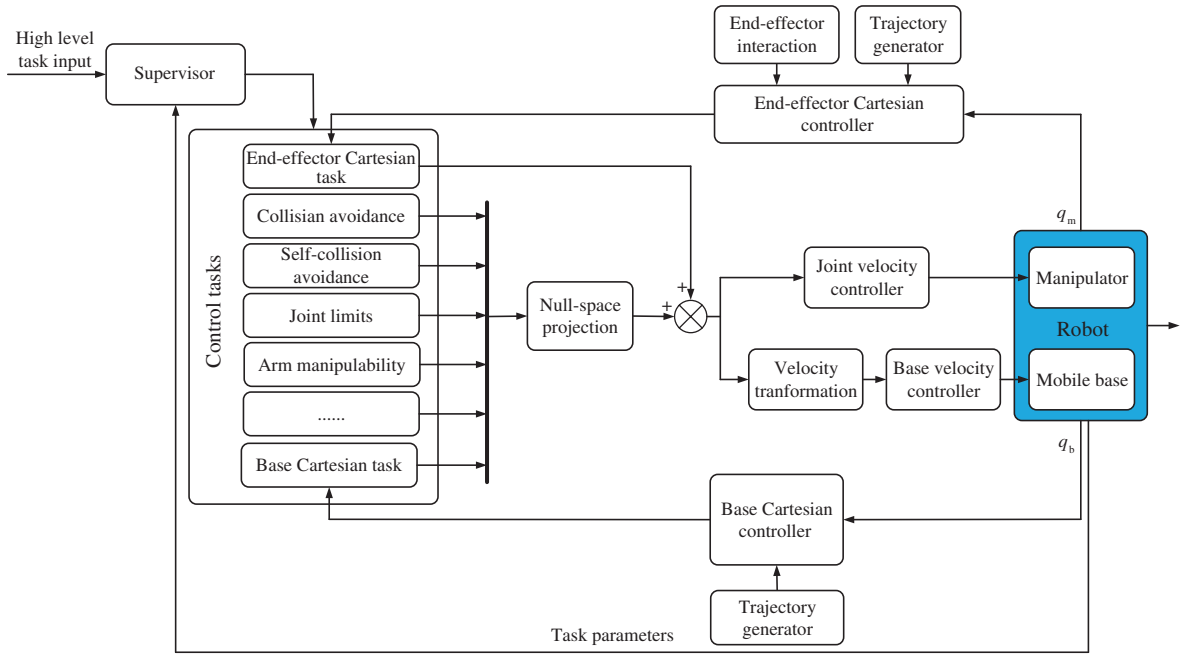

Figure 4 (Color online) Whole-body control scheme.

keeps active all the time as long as the robotic system is not stopped for safety. (2) Collision avoidance, self-collision avoidance and joint limits control primitives are activated or deactivated according to start distance, such as d_{start} and ϕ_{start}. (3) The singularity avoidance control primitive is activated if none of the three control primitives is activated or deactivated when any one of the three control primitives is activated. (4) The base controller task is activated only if a base Cartesian trajectory is planned by designers. Once this control primitive is activated, other control primitives besides the main task will be deactivated, because we think the collision avoidance and singularity problem have been both solved by trajectory planner.

The proposed whole-body control diagram can be explained by Figure 4. The left part shows the control objectives, which can be arranged at different priorities according to corresponding situations. The supervisor is applied to monitor information about tasks and dynamically change the priorities of tasks. The upper and lower parts represent trajectories planned by trajectory generators as designers want. The centric part shows the lower lever controllers of the mobile base and the upper manipulator. The Cartesian trajectories of the end-effector and the mobile base are planned by designers with some trajectory planning algorithm. The trajectory of the mobile base is planned only when necessary (such as the second experiment). Finally, lower priority control primitives are projected into the null space of the extended Jacobian.

4 Experiments and discussion

In this section, we carry out experiments with a velocity controlled mobile manipulator to verify the effectiveness of the proposed bio-inspired whole-body controller. Furthermore, we discuss its performance according to the results of experiments and make comparisons with previous similar research achievements.

4.1 Multi-objective control experiment

This experiment refers to situations when several intentions should be considered by bio-robots. Three control tasks are taken into account in the first experiment, including an operational task (main task), arm collision avoidance (secondary priority task) and singularity avoidance (last priority task). Collision

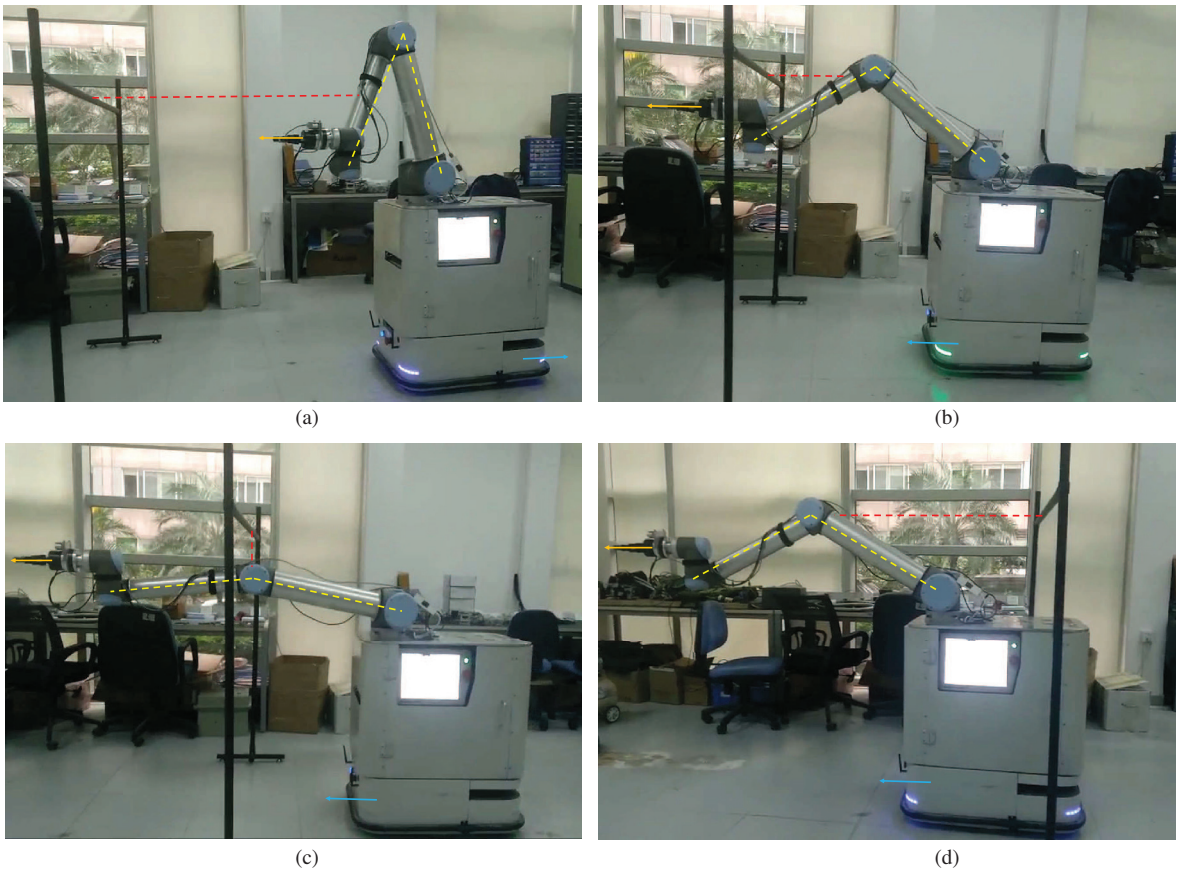

Figure 5 (Color online) Multiple tasks implementation experiments. (a) Configuration at the very beginning; (b) before collision; (c) avoiding collision; (d) after passing through the obstacle.

avoidance is treated as a secondary priority task, because only collisions between the robot links and the environment that can be avoided by artificial potential field method are considered in this paper. For collisions that can be avoided, the whole robotic system will stop for safety. That is, when the robot is approaching an obstacle and the distance is smaller than a user-defined $d_{\min}$, the robot will stop moving, which means the robot cannot complete the prescribed operational task because of obstacles. In this experiment, this safe distance is set to 10 cm. A cylindrical obstacle located 1.5 m away and 1.3 m above the ground has been known in advance. Repulsive force activation distance d_{start} and maximum repulsive force $f_{\max}$ are set to 0.3 m and 1 m/s, respectively. At the beginning, the elbow of upper arm is higher than the obstacle, as shown in Figure 5(a). The main task is to move the end-effector 2.5 m along the opposite direction of y-axis, which should cause a collision with the obstacle without a collision avoidance control objective. At this moment, because the robot is far away from the obstacle, the repulsive force between the upper arm and obstacle has not been activated. As shown in the first curve of Figure 6, the robot configuration moves toward the direction that leads to an increase of manipulability measurement, because singularity avoidance task acts as the secondary control primitive. At the very beginning, the mobile base even moves backward a little to adjust the robot configuration. As the robot moves towards the obstacle, the distance between the upper arm and obstacle decreases. Figure 5(b) shows a moment before collision. Then, the repulsive force is activated. The second and third curves of Figure 6 show variations of the distance between upper arm and obstacle and the corresponding repulsive force, respectively. Because the collision avoidance task has a higher priority than singularity avoidance task, the robot "pushes" the arm away to avoid collision and at the same time moves toward a singular configuration. Figure 5(c) shows a snapshot when the upper arm has been "pushed" away and is moving under the obstacle. At this moment, the upper arm is very close to a singular configuration. As the robot moves away from the obstacle, the distance between them increases and the repulsive force gradually

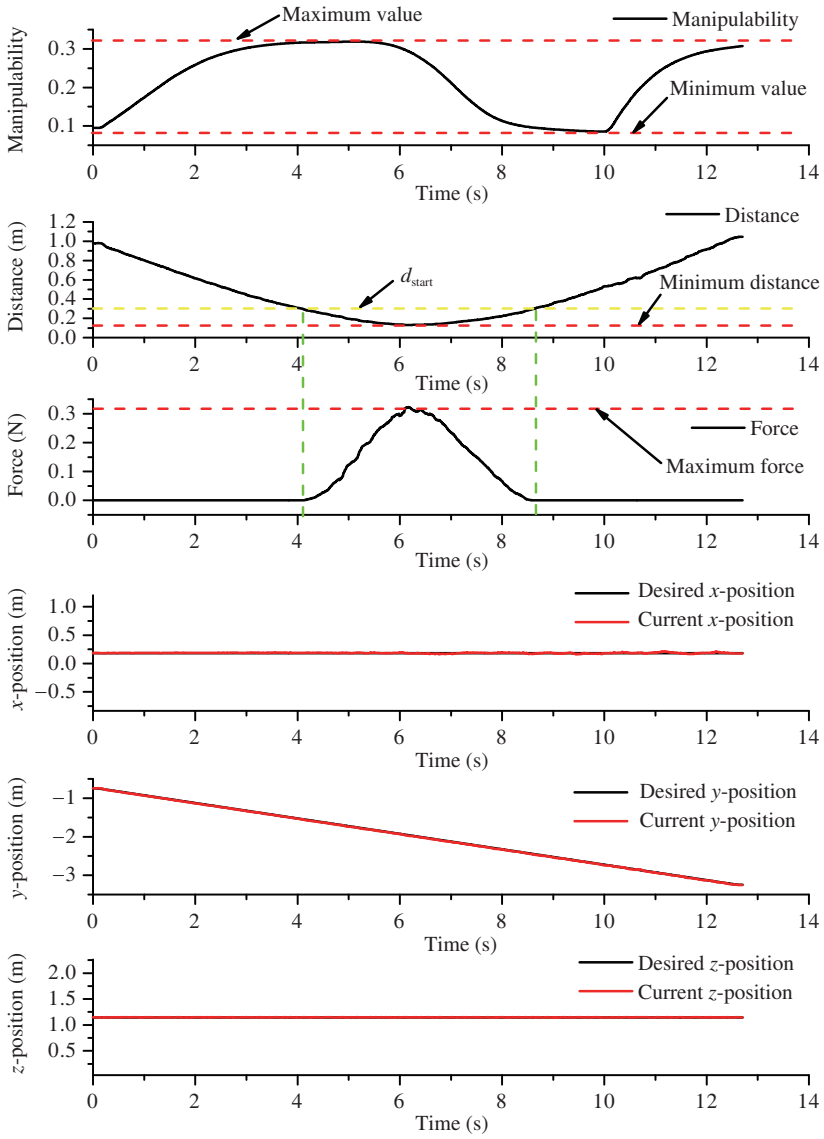

Figure 6 (Color online) Parameters variation of multiple task implementation. The first curve shows the variation of manipulability of the upper arm; the second curve shows the distance between the forearm and the obstacle; the third curve shows the repulsive force between the forearm and the obstacle; and the last three curves show tracking performance of the end-effector.

vanishes. Again, the upper arm moves towards the direction that causes the increase of manipulability. Figure 5(d) shows a snapshot during this process. Because trajectory tracking of the end-effector is the main task, it should be guaranteed all the time. The last three curves of Figure 6 indicate that the trajectory is well tracked during the whole process. In addition, the non-holonomic constraint of mobile base is satisfied all the time.

4.2 Inner motion experiment

To compare with some previous studies [41, 42] and regulate whole-body configuration of bio-robots, an experiment has been done to analyze the performance of the proposed controller. The main task is to keep the end-effector still. The secondary task is to linearly move the mobile base back and forth and then rotate it back and forth, which is also called inner motion in [16]. In this case, the Cartesian trajectory of the mobile base is also planned. The control command of the mobile base is projected into the null space of that of the end-effector. The trajectory of the end-effector is to keep it at its initial position, that is,

(a) (b)

Figure 7 (Color online) Inner motion experiment. (a) Initial configuration and rotating counterclockwise configuration; (b) rotating clockwise configuration.

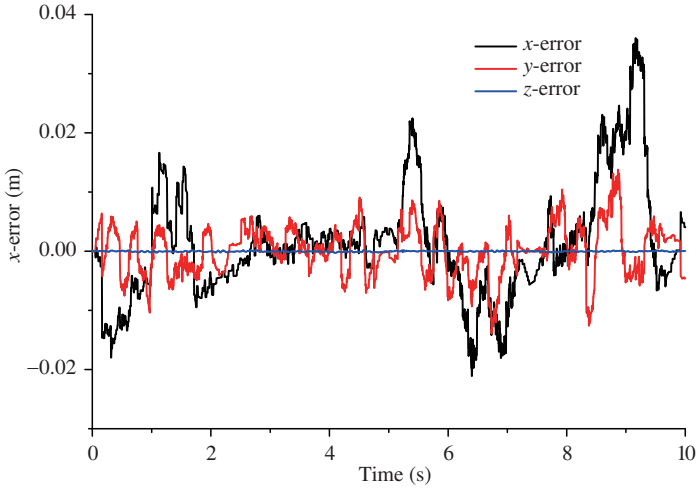

Figure 8 (Color online) Deviations of end-effector coordinates.

the end-effector controller stated in Subsection 3.1 has no feedforward term in this situation. The top and bottom parts of Figure 4 are applied in this experiment. Figure 7(a) shows the initial configuration of the system when the mobile base starts rotating counterclockwise while Figure 7(b) shows a configuration when the mobile base is rotating clockwise. The end-effector position is kept constant during the whole process. Figure 8 shows the position deviations of the end-effector along three coordinate axes when the mobile base is moving. According to Figure 8, deviations exist along three directions. However, the deviation along z-axis is much smaller because there is no movement along z-axis for mobile base.

The experiment results are very similar to that of the study done by Giftthaler et al. [41] with a tracked mobile manipulator. In Giftthaler's opinion, the deviation may be caused by the error of base pose estimator, which may also be one of the reasons for our wheeled mobile robot which is positioned via laser information. In order to test the localization error of the mobile base, we do another experiment. The mobile base alone is commanded to move forward directly without rotation, stop for a while (from the fifth second to the ninth second), and then continue to move forward directly. Figure 9 shows the deviation along the axis that passes through the contact points of two driving wheels and the ground,

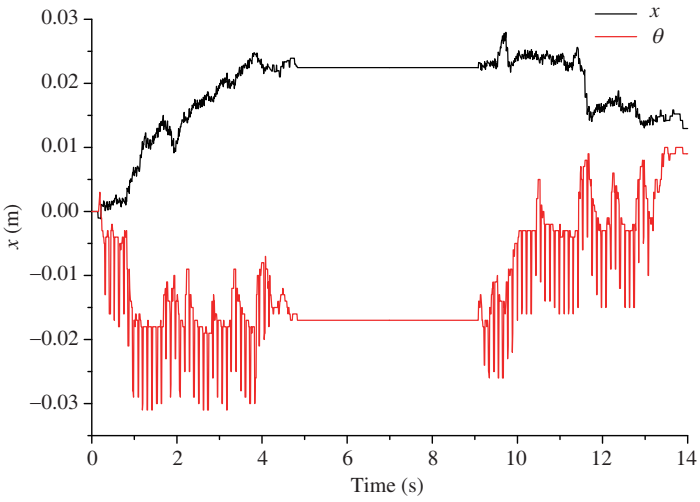

Figure 9 (Color online) Localization error of mobile base.

which is supposed to be zero without localization error. It indicates that localization deviation keeps fluctuating while the robot is moving and becomes stable (not zero) while the robot stops (during the fifth second and the ninth second). The results support that localization error of base is a significant source of the deviation of the end-effector in the above experiment. Alin et al. [42] have done a similar experiment on a manipulator with fixed base based on null-space projection. Deviations show great difference among different projection matrices, which manifests that the null-space projection matrix is another source of deviation. There may be some other reasons for the deviation of the end-effector, such as wheel slippery and model uncertainty.

5 Conclusion and future work

This paper presents a bio-inspired whole-body controller for a velocity controlled mobile manipulator. Formulations of some commonly used control primitives, including operational tasks, postural tasks and physical constraints, are analyzed. Lower priority tasks are carried out without interfering with higher priority tasks by the null-space projection technique. Practical experiments prove that the proposed methodology is an effective way to solve the whole-body control problem of velocity controlled manipulators. Curves of experiments show that deviations with high frequency exist on the end-effector. Moreover, experiments are also carried out to analyze the possible causes of the deviation of the end-effector. Some comparisons are also made with similar research to discuss the possible reasons. Localization error of mobile base and the choice of null-space projection matrix may be another reason for the deviation of the end-effector. Because smooth motion of the end-effector is essential to operational tasks, our future work will try to fix this problem. Besides, trajectory tracking task of the end-effector is arranged to be the main task all the time, but sometimes the tracking task of the end-effector has to be interrupted. As a result, some kind of intelligent judgement of task priories should be figured out to make the robotic system more adaptable and intelligent. In this paper, obstacles in the environment are assumed to be known in advance, but sometimes not all obstacles in the environment are known to robots, especially in dynamic environments. Neural networks have been proved to be effective to handle the problem of unknown environments in bio-robots control. A recent research on robot-environment interaction in [43] proposes an admittance controller for robots to interact with unknown environments. He et al. [44] and Zhang et al. [45, 46] also applied neural network-based methods to deal with interactions between the robot and unknown environments. A similar capability is even more important for mobile manipulators, so we are trying to apply neural network-based methods to deal with interactions between the mobile manipulator and unknown environments. Learning from demonstration [47] is a promising approach for robots to learning skills from humans. Yang et al. [48] recently combined dynamic movement primitive

(DMP) and Gaussian mixture model (GMM) so that the robot can learn from multiple demonstration and show generalization ability as DMP approach. In the future, we are trying to apply learning from demonstration and other bio-inspired approaches to solve mobile manipulation problems of bio-robots.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 61773139), Shenzhen Science and Technology Program (Grant No. KQTD2016112515134654), Shenzhen Special Fund for Future Industrial Development (Grant No. JCYJ20160425150757025), and Open Research Project of the State Key Laboratory of Industrial Control Technology, Zhejiang University, China (Grant No. ICT1900357).

References

1 Metta G, Natale L, Nori F, et al. The iCub humanoid robot: an open-systems platform for research in cognitive development. Neural Netw, 2010, 23: 1125–1134
2 Zhou C, Wang X, Li Z, et al. Overview of gait synthesis for the humanoid COMAN. J Bionic Eng, 2017, 14: 15–25
3 Sakagami Y, Watanabe R, Aoyama C, et al. The intelligent ASIMO: system overview and integration. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002. 2478–2483
4 Nishiwaki K, Kagami S, Inoue H. Object manipulation by hand using whole-body motion coordination. In: Proceedings of IEEE International Conference on Mechatronics & Automation, 2005. 1778–1783
5 He W, Zhang S. Control design for nonlinear flexible wings of a robotic aircraft. IEEE Trans Contr Syst Technol, 2017, 25: 351–357
6 Guo W, Cai C, Li M, et al. A parallel actuated pantograph leg for high-speed locomotion. J Bionic Eng, 2017, 14: 202–217
7 Robuffo G P, Fuchs M, Alin A S, et al. On the kinematic modeling and control of a mobile platform equipped with steering wheels and movable legs. In: Proceedings of IEEE International Conference on Robotics and Automation, 2009. 4080–4087
8 Asfour T, Regenstein K, Azad P, et al. ARMAR-III: an integrated humanoid platform for sensory-motor control. In: Proceedings of IEEE/RAS International Conference on Humanoid Robots, 2006. 169–175
9 Ruggiero F, Petit A, Serra D, et al. Nonprehensile manipulation of deformable objects: achievements and perspectives from the robotic dynamic manipulation project. IEEE Robot Automat Mag, 2018, 25: 83–92
10 Ellekilde L P, Christensen H I. Control of mobile manipulation using the dynamical systems approach. In: Proceedings of IEEE International Conference on Robotics and Automation, 2009. 1370–1376
11 Hvilshϕj M, Bϕgh S, Madsen O, et al. The mobile robot "Little Helper": concepts, ideas and working principles. In: Proceedings of IEEE International Conference on Emerging Technologies & Factory Automation, 2009. 1–4
12 Chen J, Kai S X. Cooperative transportation control of multiple mobile manipulators through distributed optimization. Sci China Inf Sci, 2018, 61: 120201
13 Tao B, Zhao X W, Ding H. Mobile-robotic machining for large complex components: a review study. Sci China Technol Sci, 2019, 62: 1388–1400
14 Chen F, Selvaggio M, Caldwell D G. Dexterous grasping by manipulability selection for mobile manipulator with visual guidance. IEEE Trans Ind Inf, 2019, 15: 1202–1210
15 Chen F, Gao B, Selvaggio M, et al. A framework of teleoperated and sterio vision guided mobile manipulation for industrial automation. In: Proceedings of IEEE International Conference on Mechatronics and Automation, 2016. 1641–1648
16 Siciliano B, Sciavicco L, Villani L, et al. Robotics: Modelling, Planning and Control. London: Springer, 2009. 502–506
17 Siciliano B, Slotine J J E. A general framework for managing multiple tasks in highly redundant robotic systems. In: Proceedings of IEEE International Conference on Advanced Robotics, 1991. 1211–1216
18 Nakamura Y, Hanafusa H, Yoshikawa T. Task-priority based control of robot manipulators. Int J Robot Res, 1987, 6: 3–15
19 Khatib O, Sentis L, Park J, et al. Whole-body dynamic behavior and control of human-like robots. Int J Human Robot, 2004, 1: 29–43
20 Sentis L, Khatib O. Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. Int J Human Robot, 2005, 2: 505–518
21 Sentis L, Khatib O. A whole-body control framework for humanoids operating in human environments. In: Proceedings of IEEE International Conference on Robotics and Automation, 2006. 2641–2647
22 Mansard N, Khatib O, Kheddar A. A unified approach to integrate unilateral constraints in the stack of tasks. IEEE Trans Robot, 2009, 25: 670–685
23 Dietrich A, Thomas W, Alin A S. Dynamic whole-body mobile manipulation with a torque controlled humanoid robot via impedance control laws. In: Proceedings of IEEE International Conference on Intelligent Robots and Systems, 2011. 3199–3206
24 Borst C, Thomas W, Schmidt F, et al. Rollin' Justin-mobile platform with variable base. In: Proceedings of IEEE International Conference on Robotics and Automation, 2009. 1597–1598
25 Dietrich A, Bussmann K, Petit F, et al. Whole-body impedance control of wheeled mobile manipulators. Auton Robot, 2016, 40: 505–517
26 Fonseca M D P A, Adorno B V. Whole-body modeling and hierarchical control of a humanoid robot based on dual

quaternion algebra. In: Proceedings of XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR), 2016. 505–517
27 Silva F F A, Adorno B V. Whole-body control of a mobile manipulator using feedback linearization and dual quaternion algebra. J Intell Robot Syst, 2018, 91: 249–262
28 Seraji H. An on-line approach to coordinated mobility and manipulation. In: Proceedings of IEEE International Conference on Robotics and Automation, 1993. 28–35
29 Tan J, Xi N. Unified model approach for planning and control of mobile manipulators. In: Proceedings of IEEE International Conference on Robotics and Automation, 2001. 3145–3152
30 Brock O, Khatib O, Viji S. Task-consistent obstacle avoidance and motion behavior for mobile manipulation. In: Proceedings of IEEE Interantional Conference on Robotics and Automation, 2002. 388–393
31 Avanzini G B, Zanchettin A M, Rocco P. Constraint-based model predictive control for holonomic mobile manipulators. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robotics and Systems, 2015. 1473–1479
32 Yamamoto Y, Yun X. Coordinating locomotion and manipulation of a mobile manipulator. IEEE Trans Autom Control, 1994, 39: 1326–1332
33 Padois V, Fourquet J Y, Chiron P. Kinematic and dynamic model-based control of wheeled mobile manipulators: a unified framework for reactive approaches. Robotica, 2007, 25: 157–173
34 Baillieul J. Kinematic programming alternatives for redundant manipulators. In: Proceedings of IEEE International Conference on Robotics and Automation, 1985. 722–728
35 Dietrich A, Ott C, Albu-Schäffer A. An overview of null space projections for redundant, torque-controlled robots. Int J Robot Res, 2015, 34: 1385–1400
36 Kevin M L, Frank C P. Modern Robotics: Mechanics, Planning, and Control. Cambridge: Cambridge University Press, 2017. 413–420
37 Gilbert E G, Johnson D W, Keerthi S S. A fast procedure for computing the distance between complex objects in three-dimensional space. IEEE J Robot Automat, 1988, 4: 193–203
38 Khatib O. Real-time obstacle avoidance for manipulators and mobile robots. Int J Robot Res, 1986, 5: 90–98
39 Alexander D, Thoms W, Holger T, et al. Extensions to reactive self-collision avoidance for torque and position controlled humanoids. In: Proceedings of IEEE International Conference on Robotics and Automation, 2011. 3445–3462
40 Merlet J P. Jacobian, manipulability, condition number, and accuracy of parallel robots. J Mech Des, 2006, 128: 199–206
41 Markus G, Farbod F, Timothy S, et al. Efficient kinematic planning for mobile manipulators with non-holonomic constraints using optimal control. In: Proceedings of IEEE International Conference on Robotics and Automation, 2017. 3411–3417
42 Nakanishi J, Cory R, Mistry M, et al. Operational space control: a theoretical and empirical comparison. Int J Robot Res, 2008, 27: 737–757
43 Yang C, Peng G, Li Y, et al. Neural networks enhanced adaptive admittance control of optimized robot-environment interaction. IEEE Trans Cybern, 2019, 49: 2568–2579
44 He W, Dong Y. Adaptive fuzzy neural network control for a constrained robot using impedance learning. IEEE Trans Neural Netw Learn Syst, 2018, 29: 1174–1186
45 Zhang S, Dong Y, Ouyang Y, et al. Adaptive neural control for robotic manipulators with output constraints and uncertainties. IEEE Trans Neural Netw Learn Syst, 2018, 29: 5554–5564
46 Zhang S, Yang P, Kong L, et al. Neural networks-based fault tolerant control of a robot via fast terminal sliding mode. IEEE Trans Syst Man Cybern Syst, 2019. doi: 10.1109/TSMC.2019.2933050
47 Argall B D, Chernova S, Veloso M, et al. A survey of robot learning from demonstration. Robot Autonom Syst, 2009, 57: 469–483
48 Yang C, Chen C, Wang N, et al. Biologically inspired motion modeling and neural control for robot learning from demonstrations. IEEE Trans Cogn Dev Syst, 2019, 11: 281–291