Acta Astronautica 146 (2018) 259–272

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Coordinated trajectory planning of dual-arm space robot using constrained particle swarm optimization

Mingming Wang[a,b], Jianjun Luo[a,b,*], Jianping Yuan[a,*], Ulrich Walter[c]

[a] Science and Technology on Aerospace Flight Dynamics Laboratory, Northwestern Polytechnical University, 710072, Xi'an, China
[b] Research Institute of Northwestern Polytechnical University in Shenzhen, 518057, Shenzhen, China
[c] Institute of Astronautics, Technical University of Munich, 85748, Garching, Germany

ARTICLE INFO

Keywords:
Coordinated motion planning
Free-floating
Bézier curve
Particle swarm optimization

ABSTRACT

Application of the multi-arm space robot will be more effective than single arm especially when the target is tumbling. This paper investigates the application of particle swarm optimization (PSO) strategy to coordinated trajectory planning of the dual-arm space robot in free-floating mode. In order to overcome the dynamics singularities issue, the direct kinematics equations in conjunction with constrained PSO are employed for coordinated trajectory planning of dual-arm space robot. The joint trajectories are parametrized with Bézier curve to simplify the calculation. Constrained PSO scheme with adaptive inertia weight is implemented to find the optimal solution of joint trajectories while specific objectives and imposed constraints are satisfied. The proposed method is not sensitive to the singularity issue due to the application of forward kinematic equations. Simulation results are presented for coordinated trajectory planning of two kinematically redundant manipulators mounted on a free-floating spacecraft and demonstrate the effectiveness of the proposed method.

1. Introduction

In light of the space robots currently planned by world wild space agencies, an increase in the number and the capacity of robot applied in space missions will be a foregone conclusion in the coming future to fulfill the increasing demands of satellite maintenance, on-orbit assembly and space debris removal *etc* [1,2]. Space robot exhibits some special characteristics due to the dynamic coupling between the space manipulators and the spacecraft (base). Accordingly, particular trajectory planning techniques have to be developed to cope with the dynamic coupling issue of free-floating space robot.

Many methodologies and strategies of motion planning for single-arm space robot have been proposed in the literature. Torres and Dubowsky [3] derived the concept of Enhanced Disturbance Map (EDM) as a heuristic trajectory planning method; nevertheless, the EDM is hard to attain especially for the space robot with higher DOF. Yamada and Yoshikawa [4] introduced a method of Cyclic Arm Motion (CAM) using the feedback attitude error to regulate the base attitude continuously. Papadopoulos et al. [5] mapped the non-holonomic constraint to a space and employed polynomials to construct smooth and continuous trajectories for planar free-floating space manipulator. Xu et al. [6] presented a point-to-point path planning method using non-holonomic characteristic of free-floating space robot, while the base attitude and the end-effector's pose can be regulated synchronously. Afterwards, Abad et al. [7] designed an optimal control scheme for eliminating or minimizing base attitude disturbance, while the uncertainties in the initial and final boundary conditions were considered. In addition, in Ref. [8], Yoshida et al. employed the concept of Reaction Null-Space (RNS) based reactionless manipulation to remove the time loss and the velocity limit of manipulation both for kinematically non-redundant and redundant space manipulators. Moreover, the RNS-based trajectory planning method was also applied in Refs. [9,10] to capture a tumbling target by using the momentum conservation law. More recently, based on the constrained least-squares approach, a Least-Squares-Based Reaction Control (LSBRC) method [11] was introduced to locally minimize the dynamic disturbance transferred to the spacecraft during trajectory tracking maneuver.

The above mentioned studies focus mainly on the motion planinig with a single-arm robot. Nevertheless, when the orbital target does not possess a grapple, the interception and capture may be very difficult. In such cases, multi-arm robotic system which can increase the probality of grasp and provide dexterous manipulation will be a reasonable alternative. Accordingly, appropriate technical schemes have to be designed to coordinate their motions. A dual-arm robotic system was introduced in Ref. [12], where one of its arms tracks a pre-defined trajectory, while the other arm works for minimizing the base attitude

* Corresponding authors. Science and Technology on Aerospace Flight Dynamics Laboratory, Northwestern Polytechnical University, 710072, Xi'an, China.
E-mail addresses: mwang@nwpu.edu.cn (M. Wang), jjluo@nwpu.edu.cn (J. Luo), jyuan@nwpu.edu.cn (J. Yuan), walter@tum.de (U. Walter).

https://doi.org/10.1016/j.actaastro.2018.03.012
Received 25 December 2016; Received in revised form 21 November 2017; Accepted 6 March 2018
Available online 09 March 2018
0094-5765/

disturbance and optimizing the operational torque of the robotic system. Miyabe et al. [13], used the hybrid position/force control and the vibration suppression control and employed two flexible manipulators to capture a spinning object in space. Alternatively, an equivalent balance arm and its corresponding Dynamic Balance Control (DBC) scheme [14] were designed to reduce the attitude disturbance induced by the mission arm. Xu et al. [15] presented the coordinated motion planning of a dual-arm space robot for capturing a target in space. Shah et al. [16] proposed the strategy for point-to-point reactionless manipulation of the spacecraft mounted with the dual-arm robotic system. More recently, Wang et al. [17] presents a synthesis method of minimizing attitude disturbance, where DBC and RNS were integrated into the framework of task-priority based solution using the redundancy resolution of a space robot. James et al. [18] synthesized the rapidly exploring random trees (RRT) with control-based sampling and timescaling methods to construct reactionless maneuvering of a space robot in pre-capture phase. It is worth noting that in the aforementioned works, the existing coordinated motion planning methods of space robot could be categorized as follows: 1) the generalized Jacobian matrix (GJM) is employed without multi-objective optimization; 2) applying pseudo-inverse of coupling inertia matrix to generate reactionless manipulation; 3) reconstruction the task-level reactionless constraints in terms of end-effector velocities, but with the expense of the tracking error for dependent variables; 4) synthesizing searching and timescaling methods to generate reactionless manipulation. While the coordinated trajectory planning of multi-arm space robot in joint-space with multiple objectives is least explored in the literature.

The main motivation for this paper is to obtain a new coordinated joint trajectory planning method for kinematically redundant manipulators while cope with joint limits and anti-collision constraints with different objectives. The reason for choosing kinematically redundant manipulator is the existence of infinite solutions which can be employed to fulfil additional constraints, such as minimizing base attitude disturbance, or collision avoidance, and so on. Bézier curve for its simplicity and normalization is chosen to represent the shape of joint trajectory and limit the values of joint range, velocity and acceleration. Constrained PSO with adaptive inertia weight and stagnation handling is implemented to search the optimal solution for constructing the shape of the Bézier curve. The original contribution of this paper is the construction of coordinated trajectory planning framework for dual-arm space robot. Moreover, the present work is easily to extend to different kinds of robots, like fixed-base manipulator, planar manipulator, kinematically redundant multi-arm space robot, etc.

The paper is organized as follows: In Sec. 2 we formulate the trajectory planning problem of the space robot as an optimization issue under certain constraints. Kinematics and dynamics of dual-arm space robot are introduced. In Sec. 3, cost functions and constraints employed in coordinated trajectory planning issue are formulated. Moreover, parameterization of joint trajectory using Bézier curve is presented and integrated into the non-linear optimization issue. Sec. 5 shows the simulation results of the proposed coordinated trajectory planning method applied to kinematically redundant dual-arm. Finally, the conclusive remarks are made in Sec. 6 at the end of this paper.

2. Problem formulation

The objective of coordinated trajectory planning for dual-arm space robot is to generate applicable joint motion laws $\theta(t)$ without violating the imposed constraints to complete the desired manipulator tasks. Normally, it can be formulated as a non-convex optimization issues, i.e. minimize a specific objective $\Gamma(\theta)$ subject to a list of inequality constraints $g_i(\theta)$ and equality constraints $h_i(\theta)$:

Table 1
Kinematic and dynamic symbols used in the paper.

symbols	representation
J_i, C_i	joint i and mass center of link i
$\boldsymbol{a}_i$, $\boldsymbol{b}_i \in \mathbb{R}^3$	position vector from J_i to C_i and from C_i to J_{i+1}
$\boldsymbol{r}_{C_i} \in \mathbb{R}^3$	position vector of mass center of link i
$\boldsymbol{r}_b$, $\boldsymbol{r}_e \in \mathbb{R}^3$	position vector of base and end-effector
ω_b, $\omega_e \in \mathbb{R}^3$	angular velocity of base and end-effector
$m_i \in \mathbb{R}$, $\mathbf{I}_i \in \mathbb{R}^{3\times3}$	mass and inertia matrix of link i
$\mathbf{H}_b \in \mathbb{R}^{6\times6}$	inertia matrix of the base
$\mathbf{H}_{bm} \in \mathbb{R}^{6\times n}$	coupling inertia matrix
$\mathbf{H}_m \in \mathbb{R}^{n\times n}$	inertia matrix of the manipulator
$\boldsymbol{c}_b \in \mathbb{R}^6$, $\boldsymbol{c}_m \in \mathbb{R}^n$	velocity dependent non-linear terms
$\boldsymbol{f}_b, \boldsymbol{f}_e \in \mathbb{R}^6$	force and moment exert on base and end-effector
$\tau \in \mathbb{R}^n$	torque exert on manipulator joints

$$\begin{aligned} &\text{minimize} \quad \Gamma(\theta(t)) \\ &\text{subject to: } g_i(\theta(t)) < 0, \quad i = 1,2,\cdots,n_{ieq} \\ &\qquad h_i(\theta(t)) = 0, \quad i = 1,2,\cdots,n_{eq} \end{aligned} \tag{1}$$

2.1. Kinematics and dynamics of space robot

Before discussion dual-arm space robot in detail, some symbols and variables applied in the following sections are listed in Table 1. As shown in Fig. 1, a dual-arm space robotic system is composed of a spacecraft body (base) and two kinematically redundant manipulators both with n DOF, there being $2n + 1$ bodies in total. Many investigations have been conducted in the field of multi-body dynamics. Refer to [19,20], the dynamic equations of a space robotic system using the Lagrangian mechanism can be expressed as follows:

$$\begin{bmatrix} \mathbf{H}_b & \mathbf{H}_{bm}^a & \mathbf{H}_{bm}^b \\ \mathbf{H}_{bm}^{a\mathrm{T}} & \mathbf{H}_m^a & 0_n \\ \mathbf{H}_{bm}^{b\mathrm{T}} & 0_n & \mathbf{H}_m^b \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{x}}_b \\ \ddot{\theta}^a \\ \ddot{\theta}^b \end{bmatrix} + \begin{bmatrix} \boldsymbol{c}_b \\ \boldsymbol{c}_m^a \\ \boldsymbol{c}_m^b \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_b \\ \boldsymbol{\tau}_m^a \\ \boldsymbol{\tau}_m^b \end{bmatrix} + \begin{bmatrix} \mathbf{J}_b^{a\mathrm{T}} & \mathbf{J}_b^{b\mathrm{T}} \\ \mathbf{J}_e^{a\mathrm{T}} & 0_{n\times6} \\ 0_{n\times6} & \mathbf{J}_e^{b\mathrm{T}} \end{bmatrix} \begin{bmatrix} \boldsymbol{f}_e^a \\ \boldsymbol{f}_e^b \end{bmatrix} \tag{2}$$

where $\ddot{\boldsymbol{x}}_b \in \mathbb{R}^6$ is the vector of linear and angular accelerations of the base, $\ddot{\theta}^a \in \mathbb{R}^n$ and $\ddot{\theta}^b \in \mathbb{R}^n$ represent joint accelerations of manipulator a and b. For definiteness and without loss of generality, variables with superscripts a and b denote that they are respectively related to the manipulator a and b. For a free-floating space robotic system, there is no external force and torque applied to the end-effectors ($\boldsymbol{f}_e^a = \boldsymbol{f}_e^b = 0$) and to the base ($\boldsymbol{f}_b = 0$). The motions of the manipulators are governed only by the internal torque on their joints. Hence, according to the momenta conservation law, the linear momentum $\boldsymbol{P}_0$ and angular momentum $\boldsymbol{L}_0$ of the whole robotic system are conserved which can be expressed by:

$$\begin{bmatrix} \boldsymbol{P}_0 \\ \boldsymbol{L}_0 \end{bmatrix} = \mathbf{H}_b \dot{\boldsymbol{x}}_b + \mathbf{H}_{bm}^a \dot{\theta}^a + \mathbf{H}_{bm}^b \dot{\theta}^b \tag{3}$$

Suppose the initial linear and angular momentum $\boldsymbol{P}_0 = \boldsymbol{L}_0 = 0$, since $\mathbf{H}_b$ is always invertible, the motion of the base can be described by

$$\dot{\boldsymbol{x}}_b = \begin{bmatrix} \dot{\boldsymbol{r}}_b \\ \boldsymbol{\omega}_b \end{bmatrix} = \mathbf{J}_a \dot{\theta} = \begin{bmatrix} -\mathbf{H}_b^{-1}\mathbf{H}_{bm}^a & -\mathbf{H}_b^{-1}\mathbf{H}_{bm}^b \end{bmatrix} \begin{bmatrix} \dot{\theta}^a \\ \dot{\theta}^b \end{bmatrix} \tag{4}$$

By substituting Eq. (4) into the kinematic mapping of the end-effector a, $\dot{\boldsymbol{x}}_e^a = \mathbf{J}_b^a \dot{\boldsymbol{x}}_b + \mathbf{J}_e^a \dot{\theta}^a$, and end-effector b, $\dot{\boldsymbol{x}}_e^b = \mathbf{J}_b^b \dot{\boldsymbol{x}}_b + \mathbf{J}_e^b \dot{\theta}^b$, the motion of the end-effectors can be given as follows

$$\begin{bmatrix} \dot{\boldsymbol{x}}_e^a \\ \dot{\boldsymbol{x}}_e^b \end{bmatrix} = \begin{bmatrix} \mathbf{J}_g^a \\ \mathbf{J}_g^b \end{bmatrix} \dot{\theta} = \begin{bmatrix} \mathbf{J}_e^a - \mathbf{J}_b^a\mathbf{H}_b^{-1}\mathbf{H}_{bm}^a & -\mathbf{J}_b^a\mathbf{H}_b^{-1}\mathbf{H}_{bm}^b \\ -\mathbf{J}_b^b\mathbf{H}_b^{-1}\mathbf{H}_{bm}^a & \mathbf{J}_e^b - \mathbf{J}_b^b\mathbf{H}_b^{-1}\mathbf{H}_{bm}^b \end{bmatrix} \begin{bmatrix} \dot{\theta}^a \\ \dot{\theta}^b \end{bmatrix} = \mathbf{J}_g \dot{\theta} \tag{5}$$

where $\mathbf{J}_g$ is termed Generalized Jacobian Matrix (GJM) which was first derived in Ref. [21] for single space manipulator. From Eqs. (4) and (5),

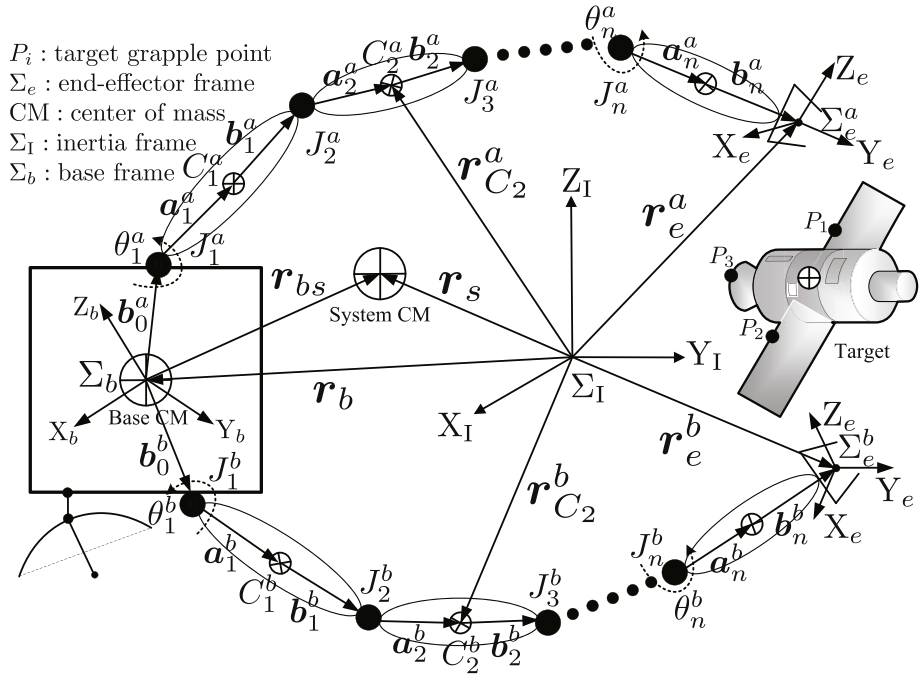

Fig. 1. Schematic diagram of space robot.

one can see that the motion of the base and the end-effectors in free-floating mode highly depends on the joint trajectories due to dynamics coupling effect between the manipulators and the base. Therefore, multiple objectives, such as performing end-effectors' task, minimizing base attitude disturbance or maximizing manipulability can be maintained with appropriate design of coordinated trajectory planning.

2.2. Orientation representation

For a redundant manipulator with dexterous workspace, orientation representation like Euler angles or angle/axis is not sufficient since the existence of orientation singularity [22,23]. The drawback of representation singularity can be overcome by the unit quaternion. Define a quaternion $\boldsymbol{q} = \{\eta, \boldsymbol{\varepsilon}\} \in \mathbb{R}^4 = \cos\frac{\varphi}{2} + \boldsymbol{r}\cdot\sin\frac{\varphi}{2}$, η is the scalar part and $\boldsymbol{\varepsilon}$ is the vector part of the quaternion. $\boldsymbol{r}$ and φ are the unit vector and the rotation angle along the vector $\boldsymbol{r}$, respectively. The quaternion $\boldsymbol{q}$ is constructed by $\eta^2 + \boldsymbol{\varepsilon}^\mathrm{T}\boldsymbol{\varepsilon} = 1$. Note that $\boldsymbol{q}$ and $-\boldsymbol{q}$ represent the same orientation. If $\boldsymbol{q}_1 = \{\eta_1, \boldsymbol{\varepsilon}_1\}$ and $\boldsymbol{q}_2 = \{\eta_2, \boldsymbol{\varepsilon}_2\}$ denote the quaternions corresponding to two rotation frames respectively, the relative quaternion can be calculated as follows where * is the quaternion production operator

$$\{\delta\eta, \delta\boldsymbol{\varepsilon}\} = \boldsymbol{q}_1 * \boldsymbol{q}_2^{-1} = \{\eta_1\eta_2 + \boldsymbol{\varepsilon}_1^\mathrm{T}\boldsymbol{\varepsilon}_2, \eta_2\boldsymbol{\varepsilon}_1 - \eta_1\boldsymbol{\varepsilon}_2 - \tilde{\boldsymbol{\varepsilon}}_1\boldsymbol{\varepsilon}_2\} \tag{6}$$

where $\tilde{\boldsymbol{\varepsilon}}$ is defined as

$$\tilde{\boldsymbol{\varepsilon}} = \begin{bmatrix} 0 & -\varepsilon_z & \varepsilon_y \\ \varepsilon_z & 0 & -\varepsilon_x \\ -\varepsilon_y & \varepsilon_x & 0 \end{bmatrix} \tag{7}$$

Note that when $\delta\boldsymbol{\varepsilon} = 0$ implies that the two rotational frames coincide. The relationship between the angular velocity $\boldsymbol{\omega}$ and the time derivative of the quaternion $\boldsymbol{q}$ is

$$\begin{bmatrix} \dot{\eta} \\ \dot{\boldsymbol{\varepsilon}} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} -\boldsymbol{\varepsilon}^\mathrm{T} \\ \eta\mathbf{E}_3 - \tilde{\boldsymbol{\varepsilon}} \end{bmatrix}\boldsymbol{\omega} = \mathbf{J}_q\boldsymbol{\omega} \tag{8}$$

3. Coordinated trajectory planning

As illustrated in section 2, a coordinated trajectory planning algorithm is a non-convex optimization issue, which should have the following features:

- The specified objectives should be optimized under feasible regions;
- The continuity of the joint position, velocity should be guaranteed;
- Undesirable effects during motion should be minimized.

3.1. Cost functions

Not like manipulator with fixed base, non-holonomic characteristic of free-floating space robot should be taken into account in the coordinated trajectory planning scheme. Accordingly, the final pose of the space robot not only relies on its inverse kinematics, but also depends on the dynamics coupling effect between the robotic arms and the base.

$$\boldsymbol{x}_b^f = \begin{bmatrix} \boldsymbol{r}_b^f \\ \boldsymbol{q}_b^f \end{bmatrix} = \boldsymbol{x}_b^s + \int_{t_s}^{t_f}\begin{bmatrix} \mathbf{E}_3 & 0 \\ 0 & \mathbf{J}_q(\boldsymbol{q}_b) \end{bmatrix}\mathbf{J}_a\dot{\boldsymbol{\theta}}dt \tag{9}$$

$$\boldsymbol{x}_e^f = \begin{bmatrix} \boldsymbol{r}_e^f \\ \boldsymbol{q}_e^f \end{bmatrix} = \boldsymbol{x}_e^s + \int_{t_s}^{t_f}\begin{bmatrix} \mathbf{E}_3 & 0 \\ 0 & \mathbf{J}_q(\boldsymbol{q}_e) \end{bmatrix}\mathbf{J}_g\dot{\boldsymbol{\theta}}dt \tag{10}$$

The relative difference between the final pose and the desired pose both for base and end-effector is given by

$$\delta\boldsymbol{x}_b = \boldsymbol{x}_b^f - \boldsymbol{x}_b^d = \begin{bmatrix} \boldsymbol{r}_b^f - \boldsymbol{r}_b^d \\ \eta_b^d\boldsymbol{\varepsilon}_b^f - \eta_b^f\boldsymbol{\varepsilon}_b^d - \tilde{\boldsymbol{\varepsilon}}_b^f\boldsymbol{\varepsilon}_b^d \end{bmatrix} \tag{11}$$

$$\delta\boldsymbol{x}_e = \boldsymbol{x}_e^f - \boldsymbol{x}_e^d = \begin{bmatrix} \boldsymbol{r}_e^f - \boldsymbol{r}_e^d \\ \eta_e^d\boldsymbol{\varepsilon}_e^f - \eta_e^f\boldsymbol{\varepsilon}_e^d - \tilde{\boldsymbol{\varepsilon}}_e^f\boldsymbol{\varepsilon}_e^d \end{bmatrix} \tag{12}$$

The determination of the joint trajectories requires the final pose of the end-effector achieves its desired pose as much as possible

$$\Gamma(\theta(t)) = \|\delta\boldsymbol{x}_e\|_\mathbf{Q} \tag{13}$$

where $\|\cdot\|$ represents the norm of the vector, $\mathbf{Q}$ is a non-negative definite weight matrix. Except the end-effectors' manipulation task, for kinematically redundant space manipulators, additional cost functions should be defined for other specific reasons [24]. For instance, large orientation disturbance to the base is not expected during the motion of

the space manipulators since the requirements of communication and observation. A cost function that defines the minimum disturbance to the base can be given by

$$\Gamma(\theta(t)) = \|\delta \boldsymbol{x}_b\|_{\mathrm{Q}} \tag{14}$$

One can also define the total joint angle change as a cost function when drastic joint variations are not expected:

$$\Gamma(\theta(t)) = \int_{t_s}^{t_f} \|\theta - \theta^s\|_{\mathrm{Q}} dt \tag{15}$$

Here θ^s is the initial configuration of the space manipulators. In addition, maximum manipulability of space manipulators is normally required at the final stage, one can define the following cost function

$$\Gamma(\theta(t)) = \sqrt{\det(\mathbf{J}^{\mathrm{T}}(\theta^f)\mathbf{J}(\theta^f))} \tag{16}$$

where $\mathbf{J}$ is manipulator's Jacobian matrix, which is derived from the configuration of the space manipulators and can be chosen $\mathbf{J}_e$ or $\mathbf{J}_g$ according to the flying mode. In this paper, $\mathbf{J}_g$ is applied to calculate the manipulability due to the free-floating mode. Additionally, other cost functions can also be defined for the coordinated trajectory planning, such as the end time [25], the distance from collisions [26], or the actuation energy [24], and so forth.

3.2. Constraints

The solution of trajectory planning problem is to generate appropriate motion laws for each joint yield the following equality and inequality constraints

$$\begin{cases} \boldsymbol{x}_b(t_s) = \boldsymbol{x}_b^s,\ \boldsymbol{x}_e(t_s) = \boldsymbol{x}_e^s,\ \theta(t_s) = \theta^s \\ \dot{\theta}(t_s) = \ddot{\theta}(t_s) = 0,\ \dot{\theta}(t_f) = \ddot{\theta}(t_f) = 0 \\ \theta_{min} \le \theta \le \theta_{max},\ \dot{\theta}_{min} \le \dot{\theta} \le \dot{\theta}_{max},\ \ddot{\theta}_{min} \le \ddot{\theta} \le \ddot{\theta}_{max} \end{cases} \tag{17}$$

θ^s is the start configuration of space robot and $\boldsymbol{x}_b^s$, $\boldsymbol{x}_e^s$ is the initial pose of the base and end-effector, respectively. In Eq. (17), equalities describe the initial and final constraints on joint and end-effector position, velocity and acceleration, while inequalities delineate the capability of each joint.

Another issue that has to be considered in the coordinated trajectory planning is possible collision both with the environment and of the manipulators with itself. To detect collision and to formulate the anti-collision issue within a non-linear programming context, bodies are depicted here as convex polyhedrons and enveloped by tight-fitting bounding volumes to minimize the computational requirements [27,28]. For this purpose, the moving bodies consist of capsules to represent the manipulator links and of boxes to represent the base and target. The anti-collision problem can be formulated straightforwardly as a set of inequalities

$$d_i(\theta(t)) > 0.0 \quad i = 1,2,\ \cdots, n_{coll} \tag{18}$$

where the function d_i represents a minimum distance between two moving bodies, the scalar n_{coll} is the number of body pairs in the given problem.

3.3. Parameterization of the trajectories

The aforementioned optimization issue in Eq. (1) is solved as a non-linear programming problem with imposed equality and inequality constraints. Each joint trajectory $\theta_i(t)$ is parametrized by Bézier curve which is widely used in computer graphics to model smooth curves [29,30]. In this paper, fifth-order Bézier curves are employed to describe the i^{th} joint trajectories in order to allow for smoothness up to the fourth derivative

$$\theta_i(\tau) = \sum_{j=0}^{m} b_{j,m}(\tau) P_{ij} = \sum_{j=0}^{m} \binom{m}{j} (1-\tau)^{m-j} \tau^j P_{ij} \quad \tau \in [0,1] \tag{19}$$

where the polynomials $b_{j,m}(\tau)$ are known as Bernstein basis polynomial of order m and $\binom{m}{j}$ is the binomial coefficient. P_{ij} is the given point to construct the Be′zier curve. Since τ is the normalized time, for the trajectory execution time $T = t_f - t_s$, if we define $t = \tau \cdot T$, the joint trajectory becomes as follows

$$\begin{cases} \theta_i = \sum_{j=0}^{m} \binom{m}{j} \left(1 - \frac{t}{T}\right)^{m-j} \left(\frac{t}{T}\right)^j P_{ij} \\ \dot{\theta}_i = \frac{d\theta_i}{d\tau}\frac{d\tau}{dt} = \frac{1}{T}\dot{\bar{\theta}}_i = \frac{m}{T}\sum_{j=0}^{m-1} b_{j,m-1}\left(\frac{t}{T}\right)(P_{i,j+1} - P_{ij}) \end{cases} \tag{20}$$

The joint velocity and acceleration boundaries in Eq. (17) can be satisfied through determination of the execution time T by the following expression

$$T \ge max\left(\frac{|\dot{\bar{\theta}}_{max}|}{\dot{\theta}_{max}}, \sqrt{\frac{|\ddot{\bar{\theta}}_{max}|}{\ddot{\theta}_{max}}}\right) \tag{21}$$

where $\dot{\bar{\theta}} = \frac{d\theta}{d\tau}$ and $\ddot{\bar{\theta}} = \frac{d^2\theta}{d\tau^2}$. Substituting the equality constraints in Eq. (17)–(20)

$$\begin{cases} P_{i0} = P_{i1} = P_{i2} = \theta_i^s \\ P_{i3} = P_{i4} = P_{i5} \end{cases} \tag{22}$$

Since θ_i^s is already known, the Bézier curve only relies on one parameter P_{i5}. Define $\boldsymbol{p} = [P_{15}, \cdots, P_{2n5}]^{\mathrm{T}}$ as design variables, once $\boldsymbol{p}$ is determined, each joint trajectory is solved accordingly. Consider Eqs. (4) and (5), the pose of base and end-effector (ignoring superscript a and b) can be derived as

$$\boldsymbol{x}_b = \boldsymbol{x}_b^s + \int_0^1 \begin{bmatrix} \mathbf{E}_3 & 0 \\ 0 & \mathbf{J}_q(\boldsymbol{q}_b) \end{bmatrix} \mathbf{J}_a \dot{\bar{\theta}} d\tau \tag{23}$$

$$\boldsymbol{x}_e = \boldsymbol{x}_e^s + \int_0^1 \begin{bmatrix} \mathbf{E}_3 & 0 \\ 0 & \mathbf{J}_q(\boldsymbol{q}_e) \end{bmatrix} \mathbf{J}_g \dot{\bar{\theta}} d\tau \tag{24}$$

Consequently, the trajectory planning issue expressed in Eq. (1) can be transformed to

$$\begin{aligned} &\underset{\boldsymbol{p}}{\text{minimize}} \quad \Gamma(\boldsymbol{p}) \\ &\text{subject to: } g_i(\boldsymbol{p}) < 0 \\ &\qquad h_i(\boldsymbol{p}) = 0 \\ &\boldsymbol{p}_{min} \le \boldsymbol{p} \le \boldsymbol{p}_{max} \end{aligned} \tag{25}$$

4. Particle swarm optimization

PSO is a stochastic search method but with a simpler philosophy. It was inspired by the coordinated motion of swarmed animals like flying birds, swimming fishes, and so forth [31]. The states of each particle adjust in swarm takes into account the effect of stochastic, cognitive and social influence. A schematic diagram of the PSO with 6 particles is shown in Fig. 2.

4.1. Basic PSO algorithm

According to the illustration in section 3.3, the design variables are the position vector $\boldsymbol{p}$. Suppose the position and velocity of the j^{th} particle are presented by $\boldsymbol{p}_j = (p_{j1}, p_{j2}, \cdots, p_{j2n})$ and $\boldsymbol{v}_j = (v_{j1}, v_{j2}, \cdots, v_{j2n})$, respectively. Their update can be expressed as follows as determined by PSO algorithm:

$$\begin{cases} v_{jk} = w v_{jk} + c_1 r_1 \left(p_{b_{jk}} - p_{jk}\right) + c_2 r_2 \left(p_{g_j} - p_{jk}\right) \\ p_{jk} = p_{jk} + v_{jk} \end{cases} \tag{26}$$

where c_1, c_2 are acceleration constants and r_1, r_2 are the uniformly

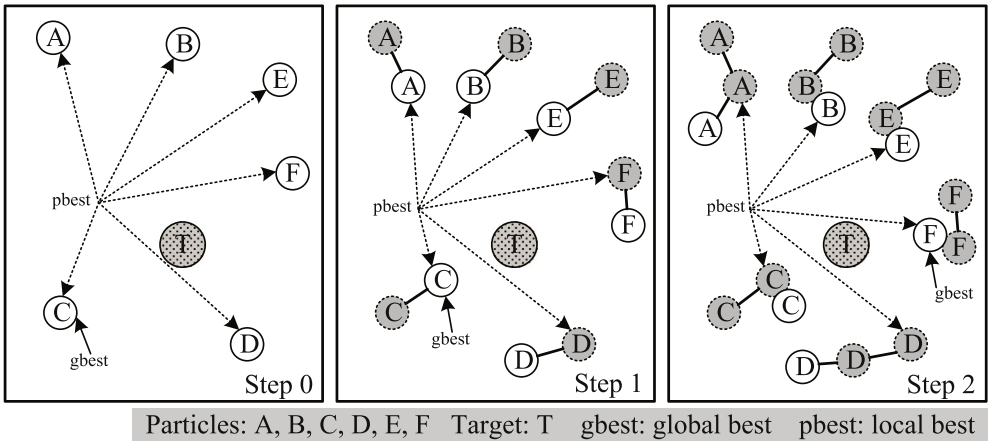

Fig. 2. Schematic diagram of PSO with 6 particles.

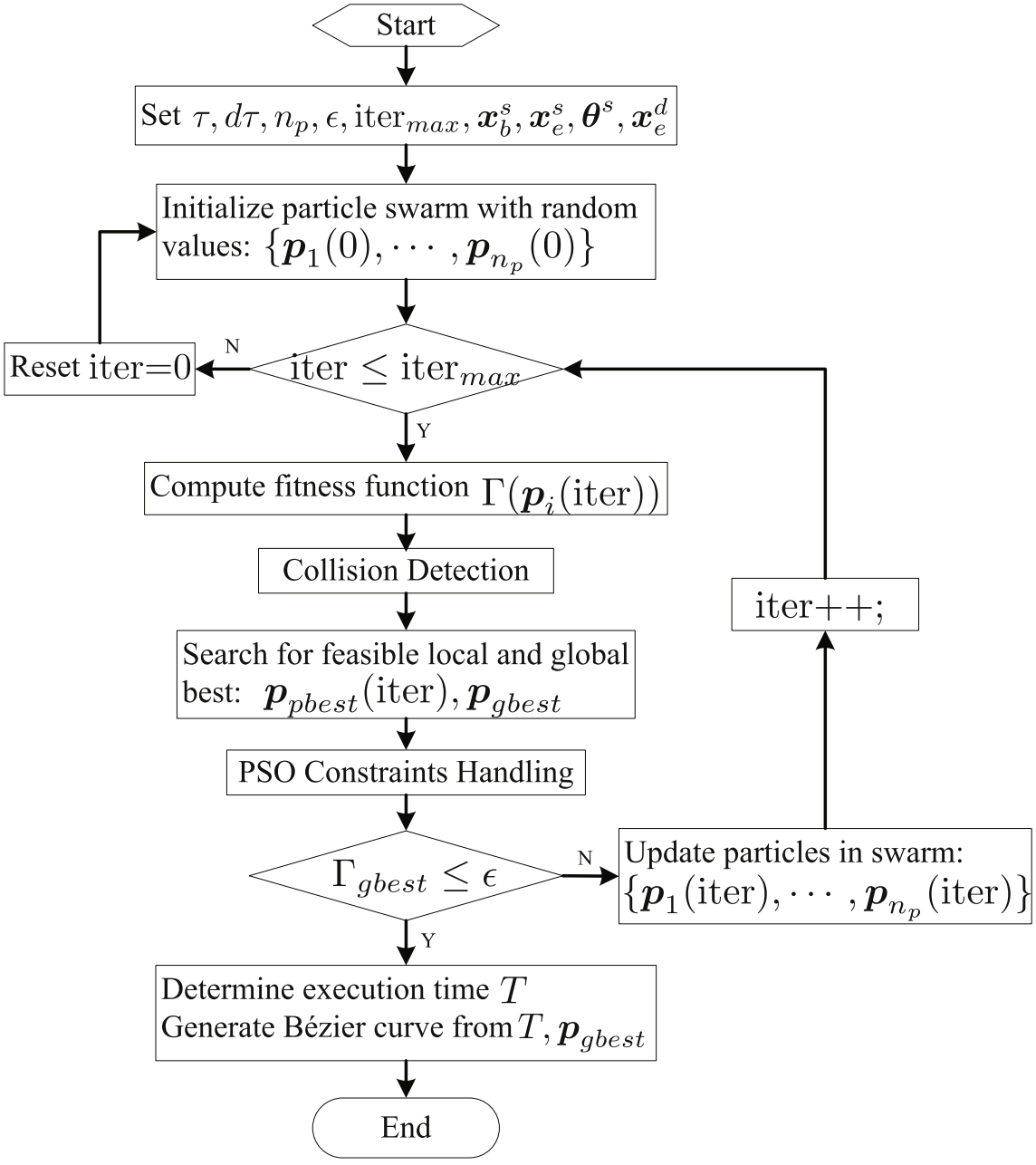

Fig. 3. The flow chart of PSO algorithm for coordinated trajectory planning.

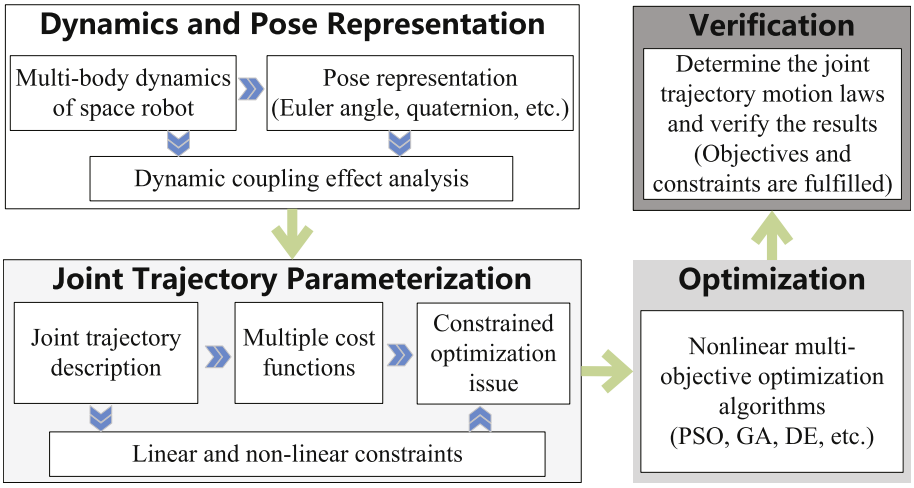

Fig. 4. The framework for space robot coordinated trajectory planning.

Table 2
Kinematic and dynamic parameters of space robot.

	Base	Manipulator a and b						
		Joint1	Joint2	Joint3	Joint4	Joint5	Joint6	Joint7
α, deg	– –	0/0	−90/90	90/-90	90/-90	90/-90	−90/90	90/-90
a, m	– –	0.0	0.0	0.0	0.0	0.0	0.0	0.0
d, m	– –	0.128	0.168	1.450	0.168	1.290	0.168	0.440
θ, deg	– –	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7
		Link1	Link2	Link3	Link4	Link5	Link6	Link7
m, kg	200	3.0	8.0	2.0	6.0	2.0	2.0	4.0
I_{xx}	63.24	0.004	1.382	0.005	0.871	0.005	0.005	0.065
I_{yy}	83.92	0.004	0.026	0.006	0.019	0.006	0.005	0.065
I_{zz}	84.68	0.010	1.382	0.005	0.871	0.005	0.006	0.013
$\boldsymbol{a}_i$,m	0.0 0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	0.0 0.0	0.0	−0.7	0.0	−0.6	0.0	0.0	0.0
	0.0 0.0	0.064	0.084	0.09	0.084	0.09	0.084	0.168
$\boldsymbol{b}_i$,m	2.52 2.52	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	−0.32 0.32	0.084	−0.66	−0.084	−0.46	0.084	−0.09	0.0
	0.0 0.0	0.128	0.0	0.0	0.0	0.0	0.0	0.188

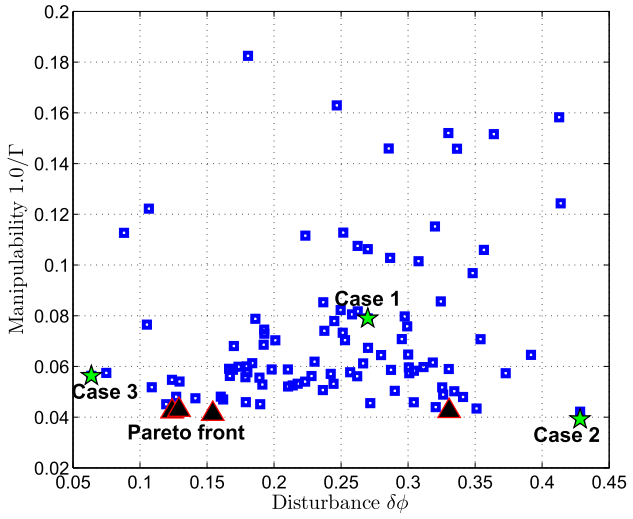

Fig. 5. The description of the Pareto front.

distributed value between [0,1]. The vector $\boldsymbol{p}_{pbest_j} = \left(p_{bj1},\cdots,p_{bj2n}\right)$ stores the local best position of the j^{th} particle so far and $\boldsymbol{p}_{gbest} = \left(p_{g_1},\cdots,p_{g_{2n}}\right)$ represents the global best position of the swarm so far. The role of inertia weight w is considered critical for the PSOs convergence behavior. The inertia weight is employed to control the impact of the previous history of velocities on the current one [32]. Accordingly, the parameter w regulates the trade-off between the global (wide-ranging) and local (nearby) exploration abilities of the swarm. A large inertia weight facilities global exploration (searching new areas), while a small one tends to facilitate local exploration, i.e. fine-tuning the current search area. A suitable value for the inertia weight usually provides balance between global and local exploration abilities and consequently results in a reduction of the number of iterations required to locate the optimum solution. In this paper, a linear decreasing strategy on w is employed:

$$w = w_{min} + \frac{\text{iter}_{max} - \text{iter}}{\text{iter}_{max}}(w_{max} - w_{min}) \tag{27}$$

where iter_{max} is the maximal number of iterations. w_{min} and w_{max} are the lower and upper bound of the inertia weight, respectively. From Eq. (26), the update velocity of the j^{th} particle is consisting of three components: a momentum of its previous velocity, velocity increments according to its local best and global best position. Eventually, the

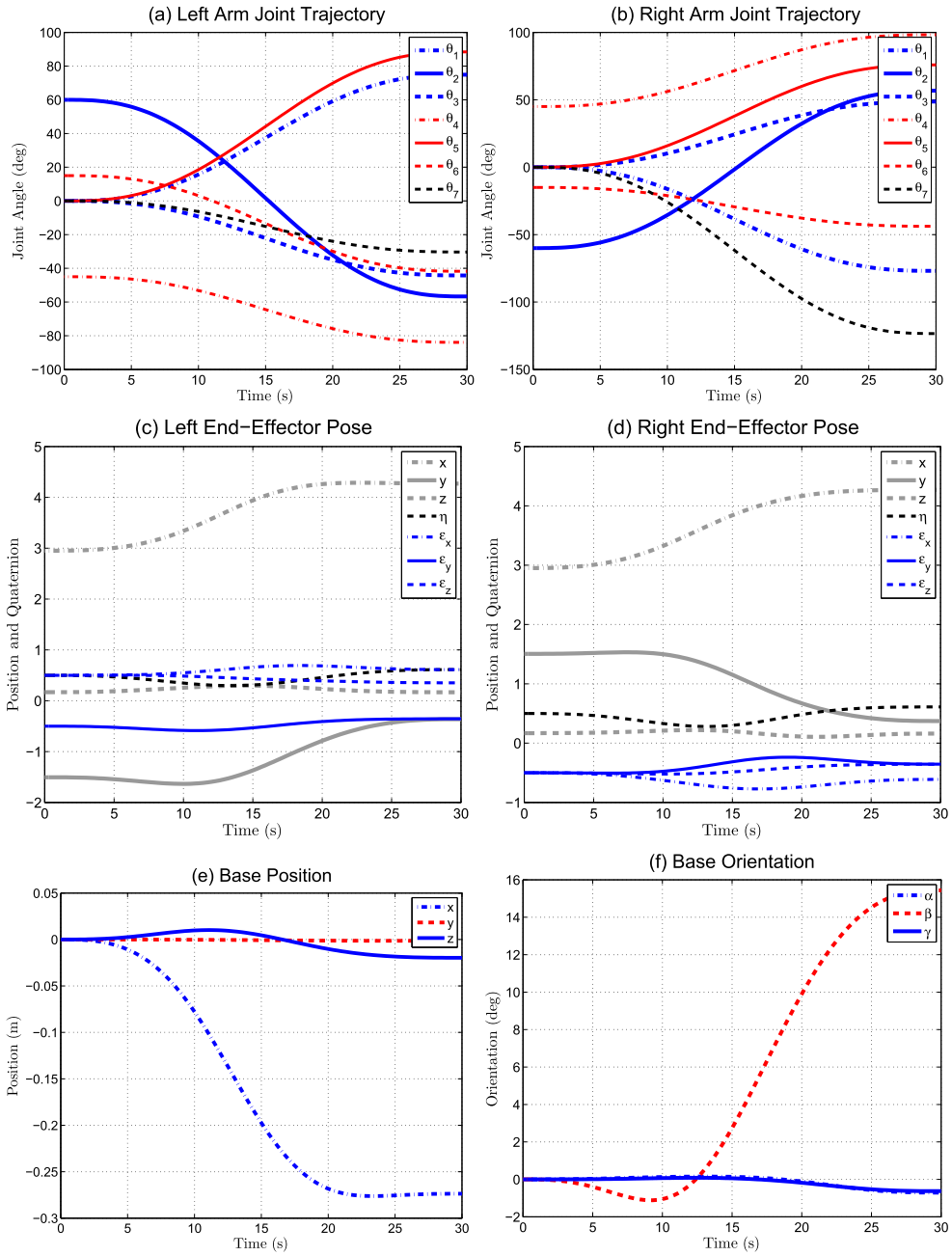

Fig. 6. The joint motion laws and pose change of end-effectors and base for case 1.

position of the particle is renewed with its previous position and new displacement induced by the new velocity.

4.2. Constraints and stagnation handling

In order to choose the local and global best particles in each optimization round, fitness function is employed in PSO to evaluate the quality of each particle in swarm and drive them to the target stepwise. How to choose the fitness function depends on the robot type, required space missions and the optimized objectives. In this paper, the fitness function can be designed as minimizing the end-effector's pose error, minimizing the base attitude disturbance, or maximizing manipulability as illustrated in section 3.1. Due to the existence of the dual-arm's redundancy resolution, a good choice for the fitness function is

$$\Gamma(\boldsymbol{p}) = \|\delta \boldsymbol{x}_e\|_{Q_e} + \sum_i \Gamma_i(\boldsymbol{p}) \tag{28}$$

One issue that encounters in PSO is the practical constraints imposed to the design variables as illustrated in Eqs. (17) and (18). Various constraints handling strategies, such and penalty function and repair algorithms, can be adopted to cope with constraints in PSO [33]. To deal with constraints, only feasible $\boldsymbol{p}_{gbest}$ is considered, while preserving the exploration and exploitation behavior of the swarm. In our method, the global best position $\boldsymbol{p}_{gbest}$ is chosen as the best feasible $\boldsymbol{p}_{pbest_j}$ and the concept of constraint domination [34] is used to update the local best position $\boldsymbol{p}_{pbest_j}$. According to the amount of constraint violation, this constraint handling strategy enables the exploitation of "better unfeasible" optimization solutions to maintain the aforementioned behavior. Refer to Eqs. (17) and (18), the amount of constraint violation can be expressed by

$$\Phi(\boldsymbol{p}_j) = \sum_{i=1}^{2n} \left(\max\left(0,\, \theta^i_{\boldsymbol{p}_j,max} - \theta^i_{max}\right) + \max\left(0,\, \theta^i_{min} - \theta^i_{\boldsymbol{p}_j,min}\right) \right) \tag{29}$$

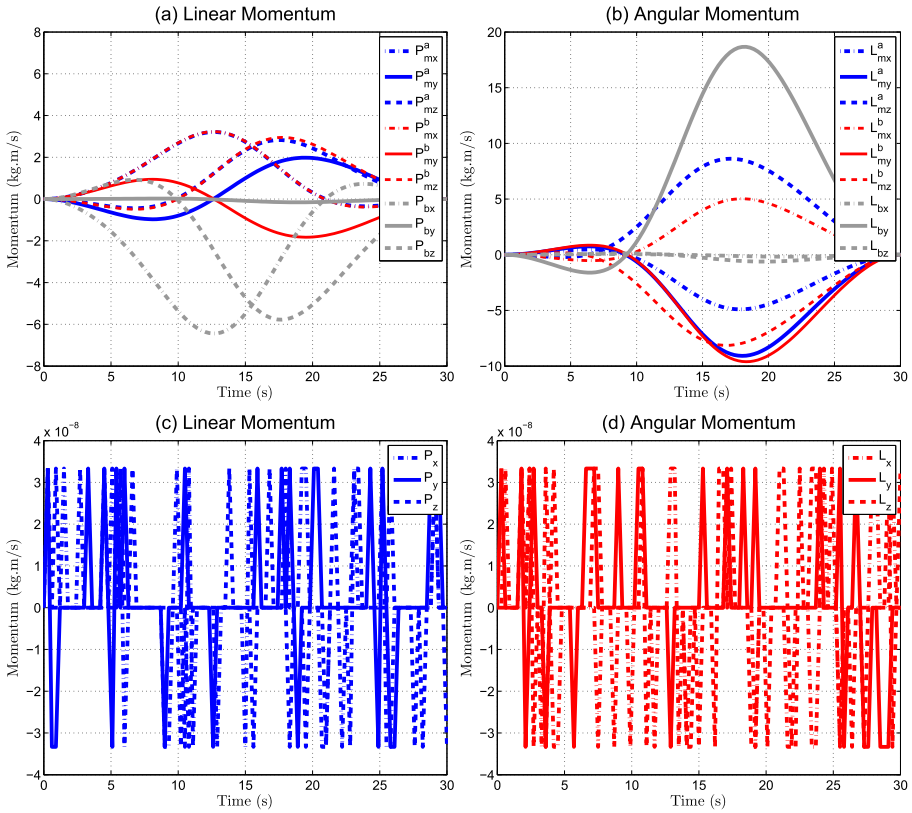

Fig. 7. Linear and angular momentum conservation for case 1.

where θ^i_{min} and θ^i_{max} are the minimal and maximal allowed position of the i^{th} joint, $\theta^i_{\boldsymbol{p}_j,min}$ and $\theta^i_{\boldsymbol{p}_j,max}$ are the encountered minimal and maximal allowed position of the i^{th} joint corresponding to $\boldsymbol{p}_j$, respectively.

Another issue that encounters in PSO is the stagnation of the swarm when no improvements of $\boldsymbol{p}_{pbest_j}$ and $\boldsymbol{p}_{gbest}$ are taking place. After a fixed number of optimization steps, the change rate of the fitness function remains lower than a given threshold near zero. Some strategy have to be employed to handle such a case [35]. The particle swarm is randomly split in two sub-swarms $S_{s,s\in\{1,2\}}$ of equal size and initialized as follows

$$\begin{cases} \boldsymbol{p}_j(\text{iter}+1) = \boldsymbol{p}_{gbest}(\text{iter}) + c_3 \cdot rand(-1,1) & \text{if } \boldsymbol{p}_j \in S_1 \\ w(\text{iter}+1) = w(\text{iter}) + c_4 \cdot rand(0,1) & \text{if } \boldsymbol{p}_j \in S_2 \end{cases} \tag{30}$$

The initialization of the design variables $\boldsymbol{p}$ in sub-swarms will provide additional chance to approach the global optimal solution, which increases the successful rate of finding optimal solution. c_3 and c_4 are two weighting scalars which are employed to regulate the swarm distribution near $\boldsymbol{p}_{gbest}$ and searching velocities near $\boldsymbol{p}_{pbest_j}$, respectively. $rand(a, b)$ is a normal distributed random number between a and b.

Fig. 3 depicts the flow chart of the coordinated trajectory planning algorithm proposed in this paper. Each optimization round begins with $\boldsymbol{p}_1$ and ends with $\boldsymbol{p}_{n_p}$. At the first optimization round, the whole swarm is initialized with uniformly random distributed values. A solution is found if for a $\boldsymbol{p}_j = \boldsymbol{p}^*$, the desired fitness function $\Gamma(\boldsymbol{p}^*) \le \varepsilon$ is satisfied. Otherwise, the particle $\boldsymbol{p}_j$ will be updated according to Eq. (26) for the next optimization round. In summary, the coordinated trajectory planning framework for multi-arm space robot with multi-constraint can be shown in Fig. 4.

5. Simulation results

In this section, we present four case studies to verify the performance of the proposed dual-arm space robot motion strategies. The space robotic system is composed of two 7-DOF kinematically redundant manipulators and a 6-DOF spacecraft base, as shown in Fig. 1. We assumed the initial linear and angular momenta to be $\boldsymbol{P}_0 = 0$ and $\boldsymbol{L}_0 = 0$. The kinematic (Denavit–Hartenberg parameters) and dynamic parameters of the dual-arm space robot are listed in Table 2, where $\boldsymbol{a}_i$, $\boldsymbol{b}_i$, and $\mathbf{I}_i$ are expressed in its own body frames. The flow chart of the proposed algorithm is shown in Fig. 3. During the processing, the PSO algorithm first finds the optimal solution to construct the Bézier curve, after that, the execution time T is determined according to Eq. (21) to fulfil the imposed joint-velocity and acceleration constraints. The joint trajectory can be determined accordingly. In order to verify the effectiveness of the proposed method, four case studies will be investigated by numerical simulations. The involved parameters of the PSO algorithm are listed as follows:

$$\begin{aligned} &n_p = 25;\ \text{iter}_{max} = 2000;\ c_1 = c_2 = 1.496; \\ &c_3 = 1.326;\ c_4 = 1.852;\ w_{max} = 0.7298;\ w_{min} = 0.4; \end{aligned} \tag{31}$$

5.1. Simulation case 1

In the first simulation, the end-effectors of manipulator a and b are commanded to move to grapple points on the target satellite form their initial pose as follows

$$\begin{aligned} \theta^{as} &= \left(0, \frac{\pi}{3}, 0, -\frac{\pi}{4}, 0, \frac{\pi}{12}, 0\right) \rightarrow \boldsymbol{x}_e^{as} \\ &= (2.950, -1.505, 0.168, 0.5, 0.5, -0.5, 0.5) \end{aligned}$$

$$\begin{aligned} \theta^{bs} &= \left(0, -\frac{\pi}{3}, 0, \frac{\pi}{4}, 0, -\frac{\pi}{12}, 0\right) \rightarrow \boldsymbol{x}_e^{bs} \\ &= (2.950, 1.505, 0.168, 0.5, -0.5, -0.5, -0.5) \end{aligned}$$

The initial pose of the base is $\boldsymbol{x}_b^s = (0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0)$. The final pose of the end-effectors can be determined according to the

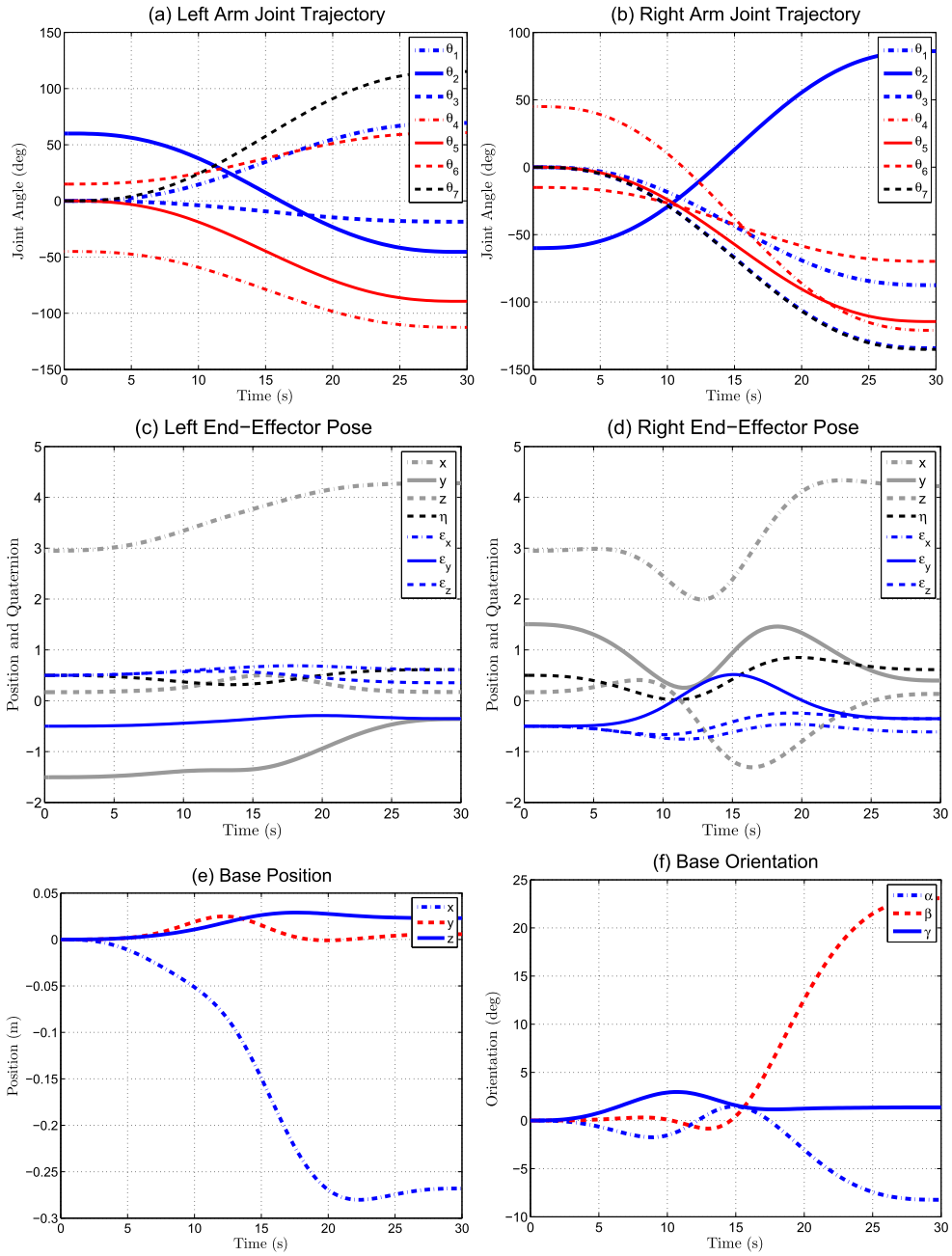

Fig. 8. The joint motion laws and pose change of end-effectors and base for case 2.

relative pose of the target's grapple points and the end-effectors, which can be expressed as follows

$$\boldsymbol{x}_e^{af} = (4.271, -0.365, 0.168, 0.612, 0.612, -0.354, 0.354)$$

$$\boldsymbol{x}_e^{bf} = (4.271, 0.365, 0.168, 0.612, -0.612, -0.354, -0.354)$$

Refer to Eq. (22) and considering the limits of joint range in (17), it shows that $\boldsymbol{p} \in [\theta_{min}, \theta_{max}]$. The fitness function in Eq. (28) is employed to evaluate each particle in the swarm without considering other specific objectives. The determination of $\mathbf{Q}_e$ depends on the position and orientation accuracy requirements. In this paper, the admitted position error is set to 0.01 m and attitude error is 2°, the weight matrix $\mathbf{Q}_e$ can be determined as follows:

$$\mathbf{Q}_e = \begin{bmatrix} \frac{1}{0.01}\mathbf{E}_3 & 0_3 \\ 0_3 & \frac{1}{\sin\left(\frac{\pi}{180}\right)}\mathbf{E}_3 \end{bmatrix} \tag{32}$$

The proposed trajectory planning algorithm stops when the convergence criteria are satisfied $\Gamma(\boldsymbol{p}^*) \le \varepsilon = 1.0$ without violating any imposed constraints. PSO algorithm successfully found a solution $\boldsymbol{p}^*$=(1.307, −0.989, −0.772, −1.465, 1.543, −0.729, −0.529, −1.342, 0.992, 0.850, 1.715, 1.325, −0.765, −2.154) that fulfil all the constraints. The execution time T = 30s is determined by Eq. (21). In terms of the searching solution $\boldsymbol{p}^*$ and the execution time T, the joint motion laws and the corresponding pose change of the end-effectors and the base are shown in Fig. 6. It is noteworthy that the orientation representation of the base is Z-Y-X Euler angles $[\alpha, \beta, \gamma]$ for intuitional reasons. One can see that with the designed joint motion, both end-effectors successfully converged to the desired pose. The final errors of

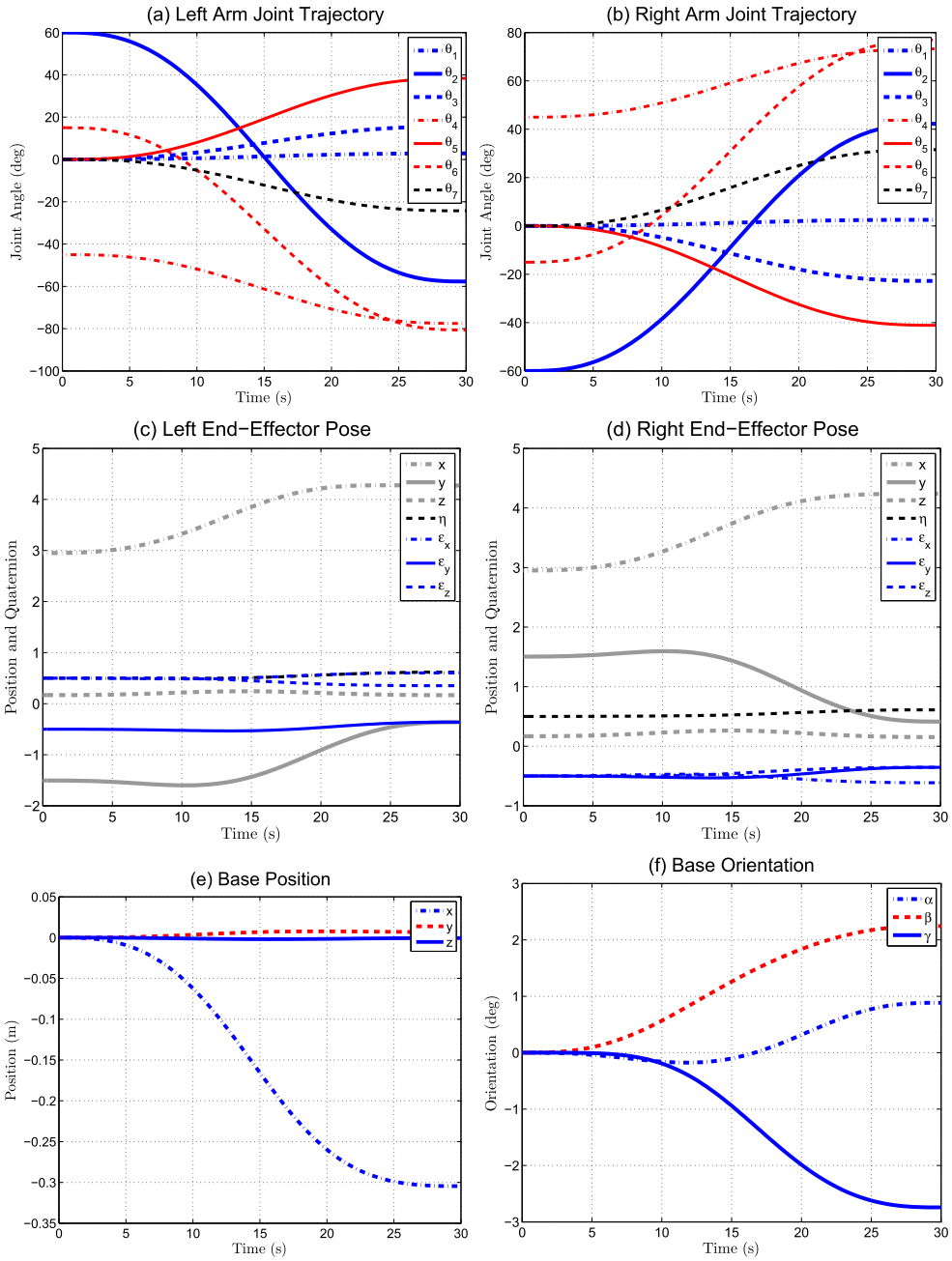

Fig. 9. The joint motion laws and pose change of end-effectors and base for case 3.

the system states are $\delta \boldsymbol{x}_e^a$=(0.047, 2.026, −1.014, 0.550, −0.895, −1.761)× 10^{-3} and $\delta \boldsymbol{x}_e^b$=(-1.760, 7.792, −7.613, −0.870, −3.132, −1.542)× 10^{-3}. As shown in Fig. 5, the manipulability in case 1 is equal to 12.664. The solution of case 1 gives a smaller value of the manipulability than those in case 2 and case 3, since maximizing the end-effectors' manipulability is not considered in case 1. In fact, there are infinite solutions since the existence of the kinematic and non-holonomic redundancy where the chosen $\boldsymbol{p}^*$ is just one of them. This solution can be used as an initial feasible solution for the purpose of subsequent optimization. Additionally, since the coordinated trajectory planning method is designed in free-floating mode, refer to Eq. (3), the linear and angular momentum of the entire space robotic system are conserved. As one can see from Fig. 7, refer to Eq. (3), the linear momentum $\boldsymbol{P}_0 = [P_x, P_y, P_z]^{\mathrm{T}}$ and angular momentum $\boldsymbol{L}_0 = [L_x, L_y, L_z]^{\mathrm{T}}$ obtained from the summation of $\mathbf{H}_{bm}^a \dot{\boldsymbol{\theta}}^a$, $\mathbf{H}_{bm}^b \dot{\boldsymbol{\theta}}^b$ and $\mathbf{H}_b \dot{\boldsymbol{x}}_b$ were conserved during the joint motion from its initial to the final configuration.

5.2. Simulation case 2

In the second and the third simulation, the end-effectors are required to approach the target grapple points as case 1 but with additional objectives. The complete solution for simulation 2 includes not only approaching to the target, but also maximizing the end-effectors' manipulability which makes it a multiple objectives optimization issue. Two objectives are included in this simulation while approaching to the target is the pre-emptive one. 100 solutions as in case 1 were first obtained and the solution for the second objective was selected from this pool. The optimal solution for both end-effectors' approaching mission and maximizing end-effectors' manipulability is $\boldsymbol{p}^*$=(1.211, −0.792, −0.323, −1.964, −1.560, 1.060, 2.010, −1.527, 1.502, −2.343, −2.112, −1.998, −1.219, −2.359), while the total manipulability $\sqrt{\det(\mathbf{J}^{\mathrm{T}}\mathbf{J})}$ is 25.481. The final errors of the system states are $\delta \boldsymbol{x}_e^a$=(0.696, 0.474, 4.245, 0.010, 0.397, 0.282)× 10^{-3} and $\delta \boldsymbol{x}_e^b$=(-5.082,

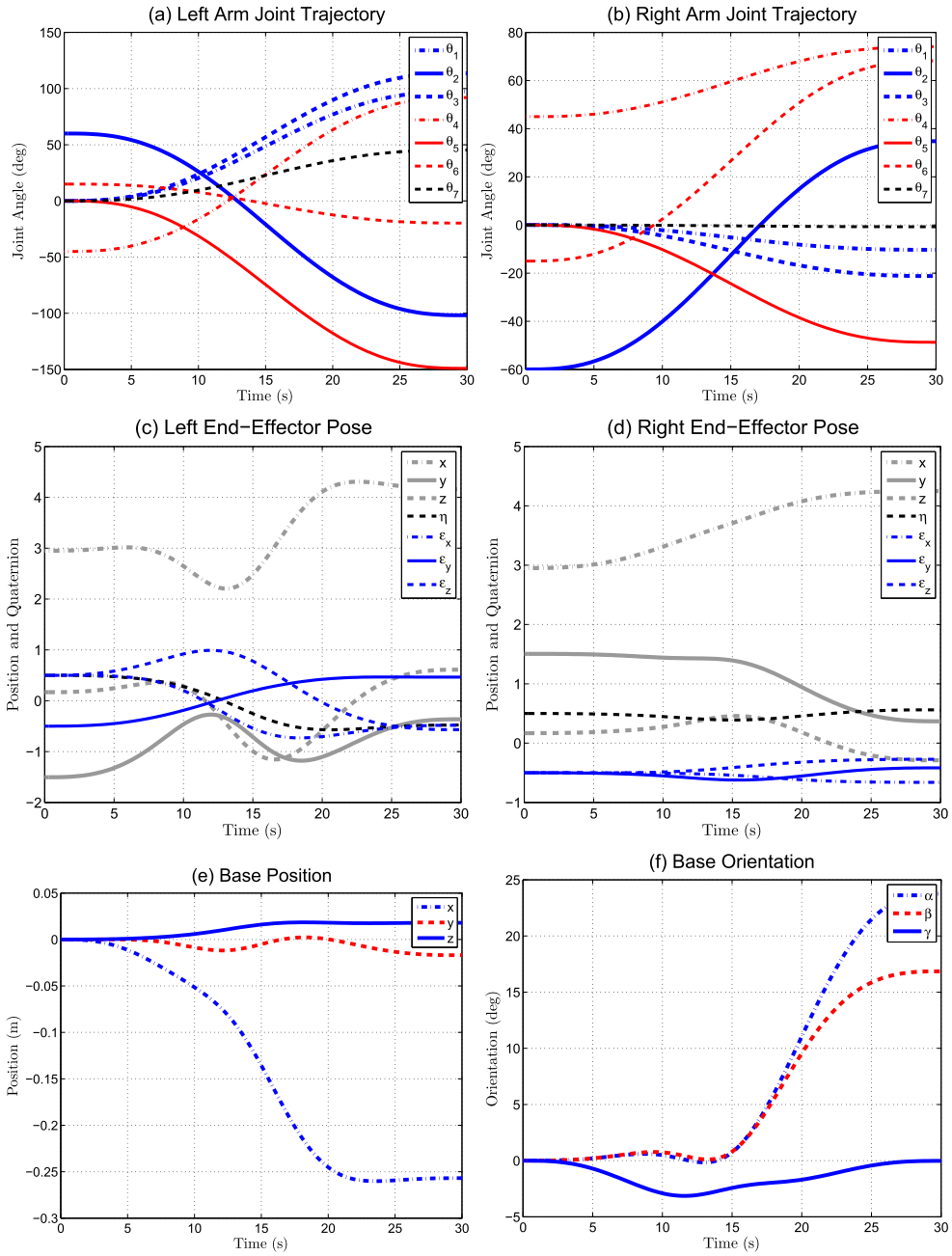

Fig. 10. The joint motion laws and pose change of end-effectors and base for case 4.

3.277, −3.363, −0.488, −0.179, −0.411)× 10^{-3}. Fig. 8 gives out the history of all joint motions and related change of the end-effectors and the base. One can see that with the designed joint trajectories, both end-effectors successfully approaches to the required pose while the objective of maximizing end-effectors' manipulability is obtained. As the space robot is having the degree-of-redundancy as 2, it may not be always possible to minimize base reaction and maximize manipulability synchronously. Trade-off has to be done to modulate multi-objectives which will be illustrated in simulation case 3.

5.3. Simulation case 3

In the third simulation, both minimizing the base orientation disturbance and maximizing the final manipulability are required as additional objectives. As shown in Fig. 5, the relationship between base orientation disturbance and manipulability is conflicting. Consequently, it is not possible to reach an optimal solution with respect to both of the objectives evaluated individually. Attention has to be paid to the concept of Pareto optimality and Pareto front [36]. A linear combination of the two required objectives as expressed in Eq. (28) is employed to evaluate the fitness function. The optimal solution for case 3 is $\boldsymbol{p}^*$=(0.050, −1.007, 0.272, −1.353, 0.670, −1.407, −0.424, 0.044, 0.737, −0.397, 1.278, −0.717, 1.342, 0.551). The corresponding manipulability of both manipulators is 17.765 and the total base orientation disturbance is 3.652°. The final errors of the system states are $\delta\boldsymbol{x}_e^a$=(0.047, 2.026, −1.014, 0.550, −0.895, −1.761)× 10^{-3} and $\delta\boldsymbol{x}_e^b$=(-1.760, 7.792, −7.613, −0.870, −3.132, −1.542)× 10^{-3}. Fig. 9 shows the history of all joint motions and related change of both end-effectors and the base. It is shown that the magnitude of the base orientation disturbance is smaller in case 3 when compared to Figs. 6 and 8, because minimizing the base attitude disturbance is not taken into account in case 1 and case 2. Moreover, due

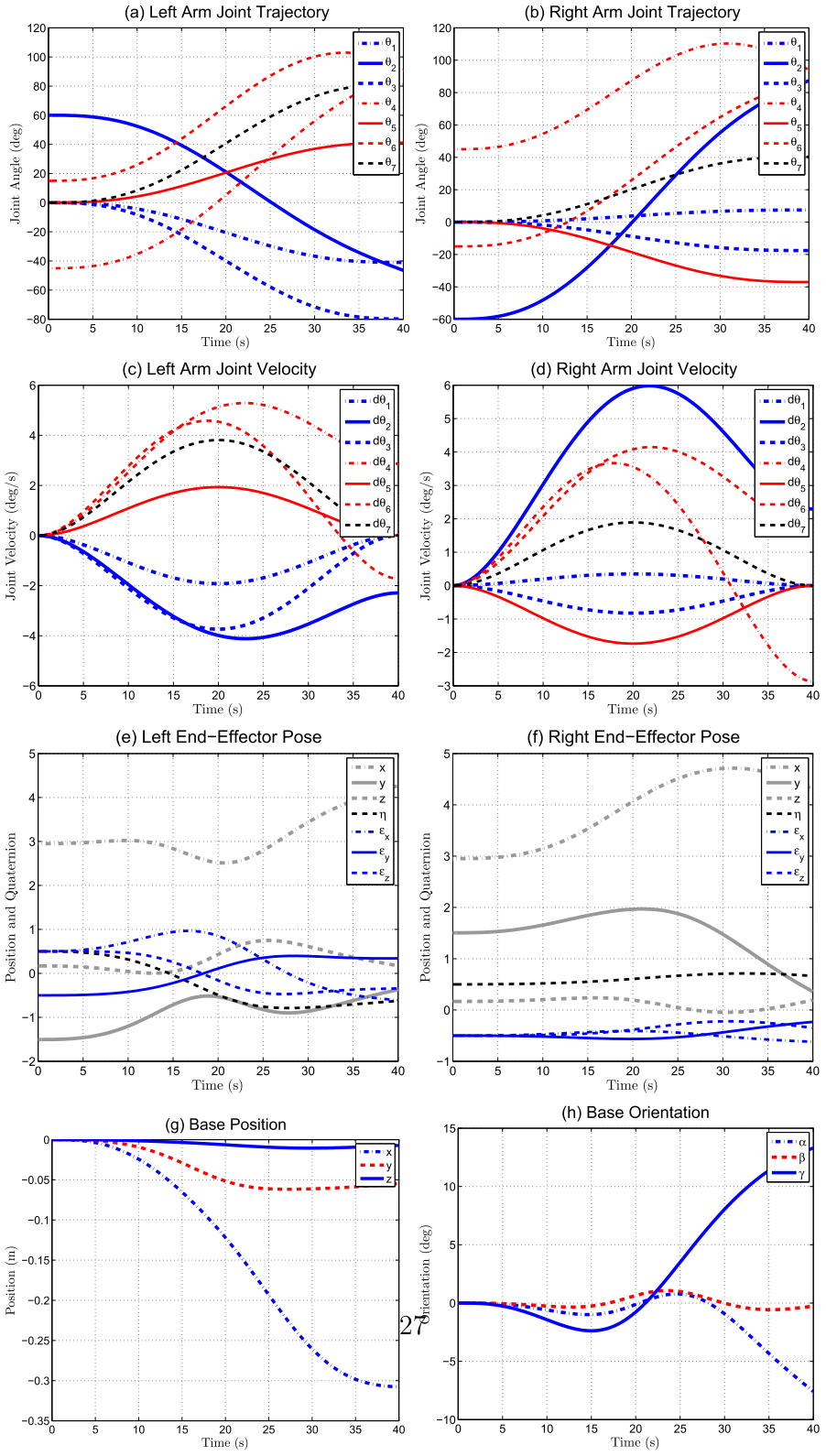

Fig. 11. Simulation results of trajectory planning with nonzero final velocity.

to the trade-off between the manipulability and the base attitude disturbance, as shown in Fig. 5, one can note that the manipulability in case 3 is bigger than that in case 1, but smaller than that in case 2 according to the qualitative analysis. The chosen cost function affects the solution of optimization and eventually influences the final pose of space robotic system.

5.4. Simulation case 4

In order to verify the feasibility and effectiveness of the proposed method, we provided another simulation case with different configurations and final pose of the end-effectors:

$$\theta^{as} = \left(0, \frac{\pi}{3}, 0, -\frac{\pi}{4}, 0, \frac{\pi}{12}, 0\right) \rightarrow \boldsymbol{x}_e^{as}$$
$$= (2.950, -1.505, 0.168, 0.5, 0.5, -0.5, 0.5)$$

$$\theta^{bs} = \left(0, -\frac{\pi}{3}, 0, \frac{\pi}{4}, 0, -\frac{\pi}{12}, 0\right) \rightarrow \boldsymbol{x}_e^{bs}$$
$$= (2.950, 1.505, 0.168, 0.5, -0.5, -0.5, -0.5)$$

$$\boldsymbol{x}_e^{af} = (4.168, -0.365, 0.615, 0.653, 0.561, -0.271, 0.430)$$

$$\boldsymbol{x}_e^{bf} = (4.255, 0.365, -0.291, 0.561, -0.653, -0.430, -0.271)$$

The initial pose of the base is $\boldsymbol{x}_b^s = (0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0)$. The fitness function is Eq. (28) is employed to evaluate each particle in the swarm without considering other specific objectives. The proposed trajectory planning algorithm stops when the convergence criteria are satisfied and finds a solution $\boldsymbol{p}^*$=(1.699, −1.777, 1.983, 1.606, −2.601, −0.344, 0.792, −0.180, 0.608, −0.370, 1.293, −0.850, 1.190, −0.013) that fulfil all the constraints. The execution time T = 30s is determined by Eq. (21). The joint motion laws and the corresponding pose change of the end-effectors and the base are shown in Fig. 10. One can see that with the designed joint motion laws, both end-effectors of the space robot converged to their desired final pose with required accuracy.

When the nonzero final velocity of the end-effectors is required, in order to apply the proposed trajectory planning method, the following assumptions should be added in our manuscript: 1) the final joint velocities should be provided beforehand; 2) the execution time T should be given in order to fulfil the terminal end-effectors velocity constraints. When $\dot{\theta}^f$ and T are given, substitute the initial and final constraints into Eq. (20), one can obtain

$$\begin{cases} P_{i0} = P_{i1} = P_{i2} = \theta_i^s \\ P_{i4} = P_{i5} + \frac{T}{m}\dot{\theta}_i^f,\ P_{i3} = 2P_{i4} - P_{i5} \end{cases}$$

Note that above constraints differ from Eq. (22) due to the introduction of nonzero final velocity of the end-effectors. In fact, $P_{i5} = \theta_i^f$ is a design variable to be determined by the optimization algorithm, since θ_i^s, $\dot{\theta}_i^f$ and T are already known. The Bézier curve only relies on one parameter P_{i5}. Define $\boldsymbol{p} = [P_{15}, \cdots, P_{2n5}]^{\mathrm{T}}$ as design variables, once $\boldsymbol{p}$ is determined, each joint trajectory is solved accordingly. The coordinated trajectory planning can still be transformed to an optimization issue as illustrated in Eq. (25). For sake of demonstration, we provided another simulation case with same initial condition as before, but with different final states of both end-effectors:

$$\boldsymbol{x}_e^{af} = (4.271, -0.365, 0.168, 0.612, 0.612, -0.354, 0.354)$$

$$\boldsymbol{x}_e^{bf} = (4.271, 0.365, 0.168, 0.612, -0.612, -0.354, -0.354)$$

$$\dot{\theta}^{af} = (0.0, 0.04, 0.0, -0.05, 0.0, 0.03, 0.0)$$

$$\dot{\theta}^{bf} = (0.0, -0.04, 0.0, 0.05, 0.0, -0.03, 0.0)$$

The execution time is set to T = 40s to construct the joint trajectories. The fitness in Eq. (28) is employed to evaluate each particle in the swarm without considering other specific objectives. The optimal solution for this simulation case is $\boldsymbol{p}^*$=(-0.716, −0.811, −1.392, 1.592, 0.719, 1.674, 1.419, 0.130, 1.525, −0.307, 1.642, −0.646, 1.539, 0.704). Fig. 11 gives out the history of all joint motions and related change of the end-effector and the base. One can see that with the designed joint trajectories, both end-effectors successfully approach to the required pose with prescribed final joint velocity. Conclusively, the generated joint trajectories for all four cases are smooth and applicable to the control of the space robot, which satisfy the requirements listed in Section 3.2.

6. Conclusions

Due to the dynamics coupling effect between the spacecraft and the manipulators, the determination of the end-effector pose not only relies on the current joint position, but also on the history motion of each joint. Therefore, the given end-effector pose can not be handled only by inverse kinematics algorithm as fixed base manipulator. The main features of this paper can be listed as follows:

(1) The non-holonomic property of the free-floating space robot is considered and two kinematically redundant manipulators are employed for further optimization. Moreover, since the forward kinematics is applied without inversion of Jacobian matrix, the kinematics and dynamics singularities of space robot will not affect the joint motion during the whole trajectory time.
(2) Joint trajectories are parametrized by the Bézier curves for its simplicity and normalization. A constrained PSO algorithm with adaptive inertia weight and stagnation handling strategy is adopted to search for optimal solutions to construct the joint parameterized curve.
(3) A coordinated trajectory planning framework is implemented which can be employed to generate rational motion laws for kinematically redundant space robot with multi-objective and multi-constraint.

The simulation results demonstrates that the proposed method can be used to maximize the manipulability to the end-effectors or perform optimization with multiple objectives according to various cost functions. The coordination control of dual-arm space robotic system with mass change and structure flexibility would be one of our future work.

Acknowledgements

This research was supported by “the National Natural Science Foundation of China (Grant No. 61603304, 61690210, 61690211)”, “the Fundamental Research Funds for the Central Universities” and “Shenzhen Future Industry Special Fund (Grant No. JCYJ20160531174213774)”.

Appendix A. Supplementary data

Supplementary data related to this article can be found at http://dx.doi.org/10.1016/j.actaastro.2018.03.012.

References

[1] M.A. Diftler, J.S. Mehling, M.E. Abdallah, N.A. Radford, L.B. Bridgwater, A.M. Sanders, R.S. Askew, D.M. Linn, J.D. Yamokoski, F.A. Permenter, B.K. Hargrave, R. Platt, R.T. Savely, R.O. Ambrose, Robonaut 2-the first humanoid robot in space, Proceedings of the 2011 IEEE International Conference on Robotics and Automation, 2011, pp. 2178–2183, , http://dx.doi.org/10.1109/ICRA.2011.5979830.
[2] I. Rekleitis, E. Martin, G. Rouleau, R. L'Archevêque, K. Parsa, E. Dupuis, Autonomous capture of a tumbling satellite, J. Field Robot. 24 (4) (2007) 275–296, http://dx.doi.org/10.1002/rob.20194.
[3] M.A. Torres, S. Dubowsky, Minimizing spacecraft attitude disturbances in space manipulator systems, J. Guid. Contr. Dynam. 15 (4) (1992) 1010–1017.
[4] K. Yamada, S. Yoshikawa, Feedback control of space robot attitude by cyclic arm motion, J. Guid. Contr. Dynam. 20 (4) (1997) 715–720.
[5] E. Papadopoulos, I. Tortopidis, K. Nanos, Smooth planning for free-floating space robots using polynomials, Proceedings of the 2005 IEEE International Conference on Robotics and Automation, vol. 1, 2005, pp. 4272–4277, , http://dx.doi.org/10.1109/ROBOT.2005.1570777.
[6] W. Xu, C. Li, X. Wang, B. Liang, Y. Xu, Study on non-holonomic cartesian path planning of a free-floating space robotic system, Adv. Robot. 23 (1–2) (2009) 113–143, http://dx.doi.org/10.1163/156855308X392708.
[7] A.F. Abad, Z. Wei, O. Ma, K. Pham, Optimal control of space robots for capturing a tumbling object with uncertainties, J. Guid. Contr. Dynam. 37 (6) (2014) 2014–2017, http://dx.doi.org/10.2514/1.G000003.
[8] K. Yoshida, K. Hashizume, S. Abiko, Zeros reaction maneuver: flight validation with ets-vii space robot and extension to kinematically redundant arm, Proceedings of

the 2001 IEEE International Conference on Robotics and Automation, vol. 1, 2001, pp. 441–446, , http://dx.doi.org/10.1109/ROBOT.2001.932590.
[9] K. Yoshida, D. Dimitrov, H. Nakanishi, On the capture of tumbling satellite by a space robot, Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 4127–4132, , http://dx.doi.org/10.1109/IROS.2006.281900.
[10] T. Nguyen-Huynh, I. Sharf, Adaptive reactionless motion for space manipulator when capturing an unknown tumbling target, Proceedings of the 2011 IEEE International Conference on Robotics and Automation, vol. 1, 2011, pp. 4202–4207, , http://dx.doi.org/10.1109/ICRA.2011.5980398.
[11] S. Cocuzza, I. Pretto, S. Debei, Least-squares-based reaction control of space manipulators, J. Guid. Contr. Dynam. 35 (3) (2012) 976–986, http://dx.doi.org/10.2514/1.45874.
[12] K. Yoshida, Y. Kurazumeand, Y. Umetani, Dual arm coordination in space free-flying robot, Proceedings of the 1991 IEEE International Conference on Robotics and Automation, vol. 3, 1991, pp. 2516–2521, , http://dx.doi.org/10.1109/ROBOT.1991.132004.
[13] T. Miyabe, A. Konno, M. Uchiyama, Automated object capturing with a two-arm flexible manipulator, Proceedings of the 2003 IEEE International Conference on Robotics and Automation, vol. 2, 2003, pp. 2529–2534, , http://dx.doi.org/10.1109/ROBOT.2003.1241973.
[14] P. Huang, Y. Xu, B. Liang, Dynamic balance control of multi-arm free-floating space robots, Int. J. Adv. Rob. Syst. 2 (2) (2005) 117–124, http://dx.doi.org/10.5772/5797.
[15] W. Xu, Y. Liu, Y. Xu, The coordinated motion planning of a dual-arm space robot for target capturing, Robotica 30 (5) (2012) 755–771, http://dx.doi.org/10.1017/S0263574711001007.
[16] S.V. Shah, I. Sharf, A. Misra, Reactionless path planning strategies for capture of tumbling objects in space using a dual-arm robotic system, Proceedings of AIAA Guidance, Navigation, and Control Conference, 2013, http://dx.doi.org/10.2514/6.2013-4521.
[17] M. Wang, J. Luo, U. Walter, Novel synthesis method for minimizing attitude disturbance of the free-floating space robots, J. Guid. Contr. Dynam. 39 (3) (2016) 695–704, http://dx.doi.org/10.2514/1.G001219.
[18] F. James, S. Shah, A. Singh, K. Krishna, A. Misra, Reactionless maneuvering of a space robot in precapture phase, J. Guid. Contr. Dynam. 39 (10) (2016) 2417–2423, http://dx.doi.org/10.2514/1.G001828.
[19] M. Wang, U. Walter, Joint-space dynamics algorithm of space manipulators with tree structure by using inertia mapping matrix, Proceedings of the 64th International Astronautical Congress, 2013.
[20] S. Xu, H. Wang, D. Zhang, B. Yang, Adaptive reactionless motion control for free-floating space manipulators with uncertain kinematics and dynamics, Proceedings of the 3rd IFAC International Conference on Intelligent Control and Automation Science, vol. 3, 2013, pp. 646–653, , http://dx.doi.org/10.3182/20130902-3-CN-3020.00145.
[21] Y. Umetani, K. Yoshida, Resolved motion rate control of space manipulators with generalized jacobian matrix, IEEE Trans. Robot. Autom. 5 (3) (1989) 303–314, http://dx.doi.org/10.1109/70.34766.
[22] J. Diebel, Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors, (2006).
[23] M. Wang, J. Luo, U. Walter, Trajectory planning of free-floating space robot using particle swarm optimization (pso), Acta Astronaut. 112 (2015) 77–88, http://dx.doi.org/10.1016/j.actaastro.2015.03.008.
[24] R. Lampariello, D. Tuong, C. Castellini, G. Hirzinger, J. Peters, Trajectory planning for optimal robot catching in real-time, Proceedings of the 2011 IEEE International Conference on Robotics and Automation, vol. 1, 2011, pp. 3719–3726, , http://dx.doi.org/10.1109/ICRA.2011.5980114.
[25] F. Aghili, A prediction and motion-planning scheme for visually guided robotic capturing of free-floating tumbling objects with uncertain dynamics, IEEE Trans. Robot. 28 (3) (2012) 634–649, http://dx.doi.org/10.1109/TRO.2011.2179581.
[26] S. Jacobsen, C. Lee, C. Zhu, S. Dubowsky, Planning of safe kinematic trajectories for free flying robots approaching an uncontrolled spinning satellite, Proceedings of the ASME 2002 Design Engineering Technical Conferences and Computer and Information in Engineering Conference, 2002, pp. 1145–1151, , http://dx.doi.org/10.1115/DETC2002/MECH-34336.
[27] C. Ericson, Real-time Collision Detection, first ed., Elsevier, Amsterdam, 2005.
[28] M. Wang, J. Luo, U. Walter, A non-linear model predictive controller with obstacles avoidance for a space robot, Adv. Space Res. 57 (2016) 1737–1746 https://doi.org/10.1016/j.asr.2015.06.012.
[29] G. Farin, Curves and Surfaces for Computer-aided Geometric Design: a Practical Guide, fourth ed., Academic Press, 1997.
[30] J. Faraway, M. Reed, J. Wang, Modelling three-dimensional trajectories by using bezier curves with application to hand motion, Appl. Statist. 56 (5) (2007) 571–585, http://dx.doi.org/10.1111/j.1467-9876.2007.00592.x.
[31] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, Proceedings of the 6th International Symposium on Micro Machine and Human Science, 1995, pp. 39–43, , http://dx.doi.org/10.1109/MHS.1995.494215.
[32] K. Kentzoglanakis, M. Poole, Particle swarm optimization with an oscillating inertia weight, Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation, 2009, pp. 1749–1750, , http://dx.doi.org/10.1145/1569901.1570140.
[33] S. Helwig, Particle Swarms for Constrained Optimization, Ph.D. thesis Erlangen University, Erlangen, Germany, 2010.
[34] L. Li, X. Yu, X. Li, W. Guo, A modified pso algorithm for constrained multi-objective optimization, Proceedings of the 3rd IEEE International Conference on Network and System Security, 2009, pp. 462–467, , http://dx.doi.org/10.1109/NSS.2009.72.
[35] E. Kaigom, T. Jung, J. Rossmann, Optimal motion planning of a space robot with base disturbance minimization, Proceedings of the 11th Symposium on Advanced Space Technologies in Robotics and Automation, vol. 1, 2011, pp. 1–6.
[36] O. Castillo, L. Trujillo, P. Melin, Multiple objective genetic algorithms for path-planning optimization in autonomous mobile robots, Soft Comput. 11 (3) (2007) 269–279, http://dx.doi.org/10.1007/s00500-006-0068-4.