Modeling, Control and Simulation of a 6-DOF Reconfigurable Space Manipulator with Lockable Cylindrical Joints

Pooya Merat[1], Farhad Aghili[2], and Chun-Yi Su[1]

[1] Concordia University, Montreal, QC, Canada
[2] Canadian Space Agency, St-Hubert, QC, Canada
{p_merat,cysu}@encs.concordia.ca,
farhad.aghili@asc-csa.gc.ca

Abstract. Reconfigurable manipulators can be very advantageous in dexterity-demanding tasks such as space operations. This paper presents the modeling, control and simulation of a robotic reconfigurable manipulator. The manipulator benefits from passive cylindrical joint design for its links which allows it to change the link parameters. The robot enters the reconfiguration phase in two steps; first, it forms a closed kinematic chain or in other words docks its end-effector to a fixed point in order to increase the constrained DOFs. Second, it releases the built-in locks of its cylindrical joints to enable the reconfiguration of the links. Then, the reconfiguration process is performed by using the proper control method. After achieving the desired configuration, the cylindrical joints are locked again, the end-effector is released and robot enters the operation mode. This paper only focuses on the reconfiguration process of a 6-DOF manipulator with two lockable passive cylindrical joints.

Keywords: Reconfigurable robotic manipulator, constrained multi-body, dynamic modeling.

1 Introduction

The space manipulators should be size efficient in order to be placed on the lunch vehicle. For example Canadarm II has a hinge at the middle of each of its links to be folded on the lunch vehicle, and then it has to be unfolded manually by the astronauts. For specific tasks that no human operator is present it is necessary for such manipulator to be unfolded automatically. Besides it is always desirable to have a multi-task manipulator that is able to change its parameters such as the links lengths or links twist angles, to have a suitable configuration for each particular task. For instance in space operations, large manipulators are suitable for inspection [1], or well-conditioned configurations are ideal for dexterous contact tasks [2].

Many space agencies agree that the reconfigurable space robots will be used in future for space applications. One option could be the modular reconfigurable robot that changes its configuration by making new connections between its robotic

J. Lee et al. (Eds.): ICIRA 2013, Part II, LNAI 8103, pp. 121–131, 2013.

modules [3–5]; however in these modular robots there is still difficulty in implementation of a robust and effective docking mechanism for connecting and releasing the modules [6]. Unlike the modular robots, the new design uses lockable passive cylindrical joints to change the robot configuration [7]. In this kind of manipulator, each cylindrical joint has two passive DOFs; one revolute and one prismatic with the same axes of the link. It is also equipped with an internal normally-passive [1] lock to prevent unwanted movements of these passive DOFs. The reconfiguration phase starts when the robot forms a closed chain by docking its end-effector (EE) to a fixed point. Then the cylindrical joints locks are released and using the proper control method the robot reconfigures its passive DOFs. When the process of reconfiguration is finished, the passive joints are locked again and the EE undocks. An example of the reconfiguration process of this manipulator is presented in Fig. **1**. It is worth mentioning that the reconfiguration process is performed with the same actuators and sensors used for normal operation of the manipulator, in other words the manipulator does not use any extra actuator or sensor during the reconfiguration; this results in a lighter manipulator design which is critical in space applications in order to reduce the lunch cost.

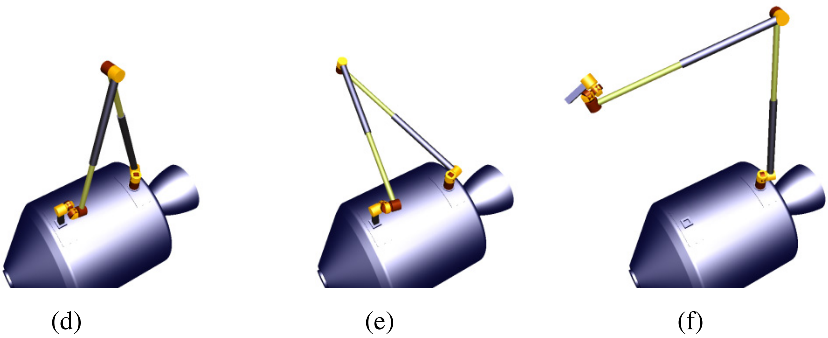

Fig. 1. Unfolding process of a reconfigurable 6-DOF manipulator installed on a satellite (a) The initial folded state; (b) the EE moving towards its docking position; (c) constraining the EE motion in dock and releasing both link locks; (d,e) reconfiguration process; (f) the final configuration.

[1] The lock consumes energy only during the state change from lock to unlock mode and vice versa [7].

In this paper the reconfiguration process of a 6-DOF reconfigurable manipulator with two cylindrical joints is presented. To enter the reconfiguration process first, the robot docks the EE to a fixed point and loses all its 6 DOFs (Fig. **1**- c). Second, both cylindrical joints are unlocked and 4 DOFs will be added to the system. Then the reconfiguration is performed (Fig. **1**-d,e) and then robot locks the cylindrical links and releases EE from the fixed docking position (Fig. **1**-f). The reconfiguration is done using 6 actuators to achieve 4 desired outputs (two link lengths and two link twist angles). The modeling, control and simulation of the manipulator before and after the reconfiguration process are not discussed in this paper and there are many papers in this regard [8].

2 Modeling

There are multiple ways to model a constrained multi-body system such as coordinate separation [9], augmented Lagrangian formulation [10] and projection-based method [11]. In this paper, the projection-based method is used to model and control the system. For this purpose, first the manipulator should be modeled without the presence of any constraints, which in the case this paper that would be the dynamic modeling of a 10-DOF manipulator. Afterwards, using the projection operator which is calculated from the constraint equation, the control law could be derived and the constrained model could be simulated. All the procedure is performed in MATLAB R2012b.

2.1 Dynamics

The first step for simulation of any mechanical system is to acquire the dynamic model. In this paper the iterative Newton-Euler formulation is used to derive dynamic equations. Unlike the widely used Lagrangian dynamic formulation, this method results in one of the simplest form of analytical equations which leads to dramatic reduction in computation time [12, 13].

When both locks of the 6-DOF manipulator are released, the manipulator has 10 DOFs without considering the constraints on EE i.e. the open-chain system. Using the projection-based method for simulation, it is only needed to calculate the open-chain manipulator dynamics. After forward kinematics calculation, the dynamic equation could be written in the following form:

$$M\ddot{q} + V = \tau \, . \tag{1}$$

where M is symmetric inertia matrix, V is non-linear vector, τ is vector of generalized force and q is generalized joint coordinates which is defined as:

$$q \triangleq [\theta^T \quad \psi^T]^T \, . \tag{2}$$

where θ is vector of active joints generalized coordinates and ψ is the vector of passive ones. After calculation of the dynamic equation, the symbolic parameters are written into *MATLAB m-functions* using the *matlabFunction* included in the *Symbolic Math Toolbox*. To speed up this process, the *MATLAB Parallel Computing Toolbox* was used.

2.2 Constraints

To implement the projection-based model it is necessary to formulate the constraint equation of the manipulator. Let denote the number of DOFs of the manipulator or number of active joints as n, the number of cylindrical joints that are unlocked at the same time as m and the number of constrained DOFs as r. In order to prevent the system from being under-actuated or over-constrained, the following inequalities must be kept satisfied [7]:

$$2m \le r \le \min(\mathrm{n}, 6)\,. \tag{3}$$

In the model studied in this paper, the manipulator has 6 DOFs (n = 6), both cylindrical joints will be reconfigured at the same time (m = 2), and EE is fully constrained by attaching to the satellite during reconfiguration (r = 6). Since the EE has no movements with respect to the satellite, the constraint equations could be extracted from the EE transform. The analytical transform of the EE could be calculated using forward kinematics in terms of q.

$$ {}^{B}_{EE}T = \left[\begin{array}{ccc|c} & {}^{B}_{EE}\mathrm{R} & & {}^{B}P_{EE} \\ \hline 0 & 0 & 0 & 1 \end{array}\right]. \tag{4}$$

where ${}^{B}_{EE}T$ and ${}^{B}_{EE}\mathrm{R}$ are transformation and rotation matrices from EE frame to the base frame and ${}^{B}P_{EE}$ is position of EE in the base frame.

The constraints equation could be written as:

$$\Phi = \begin{bmatrix} {}^{B}P_{EE} \\ {}^{B}O_{EE} \end{bmatrix} - \begin{bmatrix} {}^{B}P_{EE_0} \\ {}^{B}O_{EE_0} \end{bmatrix} = 0\,. \tag{5}$$

O_{EE} is the orientation of the EE in the form of the Euler angles:

$$O_{EE} = f({}^{B}_{EE}R) = \begin{bmatrix} \alpha & \beta & \gamma \end{bmatrix}^T . \tag{6}$$

and ${}^{B}P_{EE_0}$ and ${}^{B}O_{EE_0}$ are position and orientation of EE in the base frame when EE is fixed.

2.3 Projection Operator

Projection-based method is used as a basis for the simulation and control of the closed-chain robot during the reconfiguration process [14]. *Projection operator* is defined as:

$$P \triangleq I - A^{+}A . \tag{7}$$

where P, the projection operator, is a symmetric matrix and is the null-space orthogonal projector of A and A is the constraint Jacobian matrix:

$$A \triangleq \frac{\partial \Phi}{\partial q} . \tag{8}$$

Having the open-chain dynamic equation (**1**) the closed-chain dynamic formulation would be:

$$M_C \ddot{q} = P(\tau - V) + C_C \dot{q} . \tag{9}$$

where M_C and C_C could be calculated from:

$$\begin{aligned} M_C &= M + PM - (PM)^T . \\ C_C &= -MA^{+}\dot{A} . \end{aligned} \tag{10}$$

2.4 Control

The general control process of a reconfigurable manipulator is shown in Fig. 2. There are two control modes; the motion control mode which is not discussed in this paper and the reconfiguration control mode. For the reconfiguration control purpose a projection-based controller is used [7] which is a non-model-based controller. The projection operator and gravity term could be written as:

$$\begin{aligned} P &= \begin{bmatrix} P_{aa} & P_{ap} \\ P_{ap} & P_{pp} \end{bmatrix} . \\ G &= \begin{bmatrix} g_\theta \\ g_\psi \end{bmatrix} . \end{aligned} \tag{11}$$

where $P_{aa} \in \mathbb{R}^{n \times n}$ and $P_{pp} \in \mathbb{R}^{2m \times 2m}$. Also g_θ and g_ψ are the gravity terms of dynamic equation associated with the active and passive joints. Then the control law for the gravity-compensated PD controller would be:

$$\tau_\theta = P_{aa}^{-1} P_{ab} \left(K_d \dot{\psi} + K_p (\psi - \psi_d) + g_\psi \right) + g_\theta . \tag{12}$$

with τ_θ being the active joints torques and K_p and K_d being the PD controller gains [15]. Fig. 3 shows the overall control diagram of the reconfiguration process.

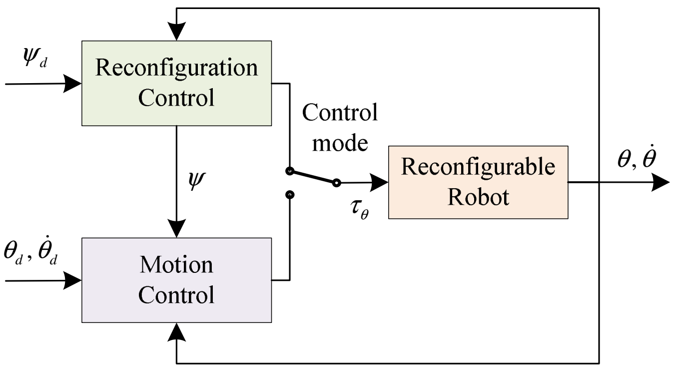

Fig. 2. Control modes of the reconfigurable manipulator [15]

2.5 Passive Joints Estimator

As mentioned before, no sensors or actuators are implemented for the passive joints. Thus, for the control purpose it is necessary to estimate the passive joints states with respect to the states of active joints that are instrumented with sensors.

Using the formula (13) passive joints states could be estimated:

$$\begin{aligned} \dot{\hat{\psi}} &= Q(\theta,\hat{\psi})\dot{\theta} - A_{\psi}^{T}(\theta,\hat{\psi})K_{\omega}\Phi(\theta,\hat{\psi}) . \\ \hat{\psi} &= \int \dot{\hat{\psi}} dt . \end{aligned} \tag{13}$$

where K_ω is the estimator gain obtained from a positive-definite matrix K and a weight matrix W.

$$K_\omega = W^T K W . \tag{14}$$

It can be proven that using the formula (**13**), the estimation error will exponentially approach to zero as t becomes large [15].

3 Simulation

The simulation of the reconfiguration process of the 6-DOF reconfigurable manipulator is implemented using MATLAB code instead of MATLAB SIMULINK, because of more programming flexibility. The variable time-step simulation method was also used to decrease the simulation computation. The variable time-step is designed to be a function of the maximum speed of the joints during the simulation.

3.1 Simulation Error Compensation

During simulation, the integration will lead to a drift error of q. It is possible that the constraint equation be no longer satisfied or in other words $\Phi(\tilde{q}) \neq 0$, with $\tilde{q}$ being the drifted q. This can lead to instability of the system during simulation. By defining the drift error as:

$$\delta q = q - \tilde{q}\,. \tag{15}$$

and using the first two terms of the Taylor series for the constraint equation $\Phi = 0$ around $\tilde{q}$ we would have:

$$\Phi(q) \equiv \Phi(\tilde{q} + \delta q) \cong \Phi(\tilde{q}) + \frac{\partial \Phi(\tilde{q})}{\partial q}\delta q\,. \tag{16}$$

From (**8**) we know that $A(\tilde{q}) = \dfrac{\partial \Phi(\tilde{q})}{\partial q}$, and it can be concluded that

$$\Phi(\tilde{q}) + A(\tilde{q})\delta q \cong \Phi(q) = 0\,. \tag{17}$$

From (**17**) it can be inferred that if $\tilde{q}$ is close enough to q, by calculating the following two formulas iteratively $\tilde{q}$ will approach to q.

$$\begin{aligned} &\delta q \cong -A^{+}(\tilde{q})\Phi(\tilde{q})\,. \\ &\tilde{q} = q - \delta q\,. \end{aligned} \tag{18}$$

The error compensation of q does not necessarily lead to compensation of the error of $\dot{q}$ and so using the same concept it is possible to compensate the drift error of $\dot{q}$ by using the following formulas:

$$\begin{aligned} &\delta \dot{q} \cong -A^{+}(q)A(q)\tilde{\dot{q}}\,. \\ &\tilde{\dot{q}} = \dot{q} - \delta \dot{q}\,. \end{aligned} \tag{19}$$

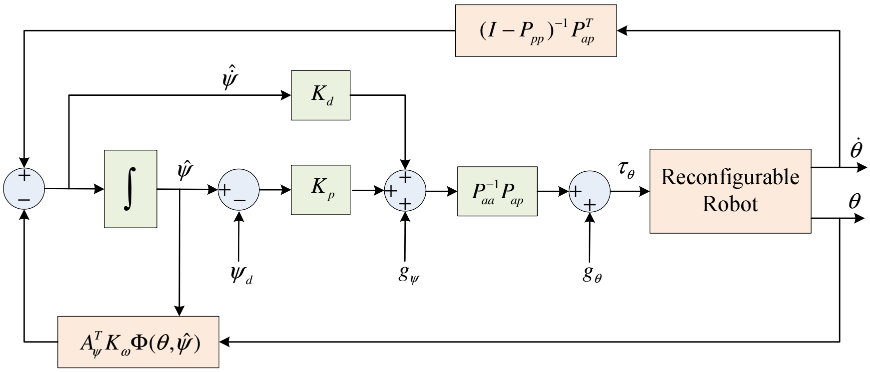

Fig. 3. The reconfiguration control diagram [15]

To validate the constrained manipulator model, the kinetic energy of the open-loop system was monitored without the presence of gravity. The model is excited only with an initial velocity of the joints, and no torque is exerted to the joints. It is expected that the kinetic energy of the system doesn't change during time. The kinetic energy could be calculated from:

$$K = \frac{1}{2}\dot{q}^T M\dot{q}\,. \tag{20}$$

In a test with the simulation time-step of 5ms, an initial angular velocity was exerted which resulted in an initial kinetic energy of $K = 3.2e\text{-}2\ J$. The kinetic energy was monitored over 10 seconds or 2000 iterations. The standard deviation of the monitored kinetic energy was $2.4e\text{-}5\ J$ which is a reasonable error for the simulation.

3.2 Results

The simulation results correspond to the process depicted in Fig. 1 which is the unfolding scenario of a space manipulator from a compact state to a manipulator with longer links. During the reconfiguration process, the length of both links are increased from *1m* (compacted state) to *1.9m* (unfolded state), i.e. an increase of 90% in length. The twist of first link also changes from 0 to π, but the second link twist angle remains the same. Each joint and each cylindrical joint have a mass of 2kg and the masses of the links are linearly distributed among their lengths. The satellite was considered as a fixed object and also to be on-orbit i.e. there is no gravity effect on the manipulator or G = 0. Controller gains for both length and twist angle are set to $K_p = 5,\ K_d = 30$. The simulation results are illustrated in the form of time histories of the passive DOFs (Fig. 4 and Fig. 5), active joints angles (Fig. 6), joint torques (Fig. 7) and joint velocities (Fig. 8). As could be seen the controller is able to reconfigure the manipulator to the desired configuration in about 60 seconds.

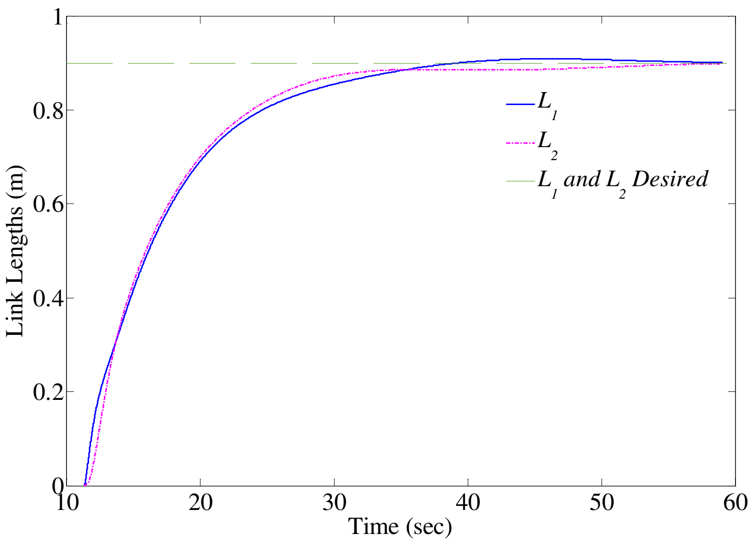

Fig. 4. During the reconfiguration process L_1 and L_2 (length of the first and second cylindrical joints) increase from *1m* to *1.9m*

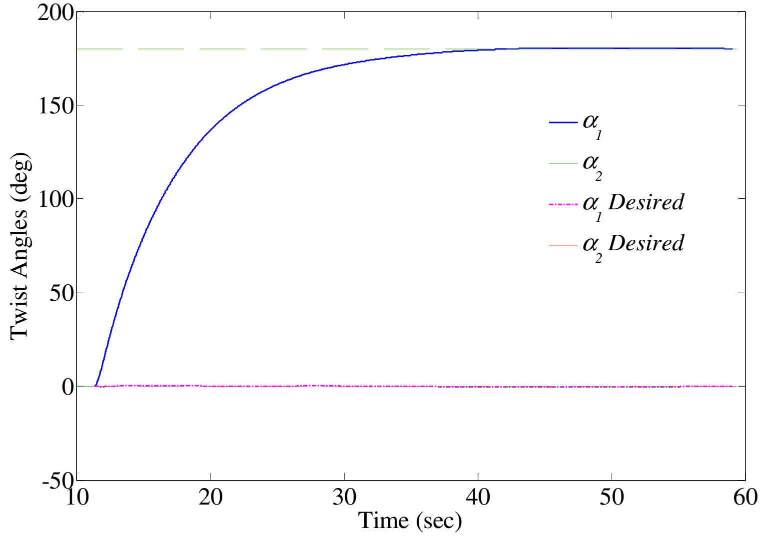

Fig. 5. During the reconfiguration process α_1 (twist angle of the first cylindrical joint) increases from *0* to π and α_2 (twist angle of the second cylindrical joint) remains *0* in this scenario

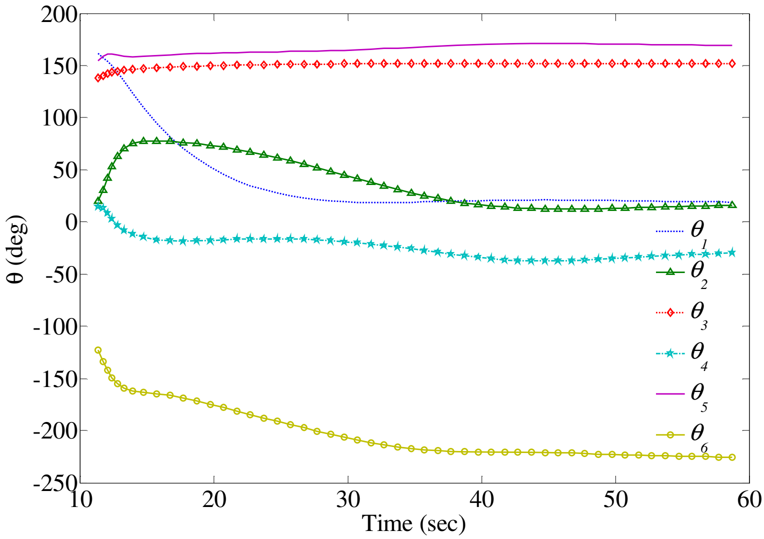

Fig. 6. Joint angles during the reconfiguration process

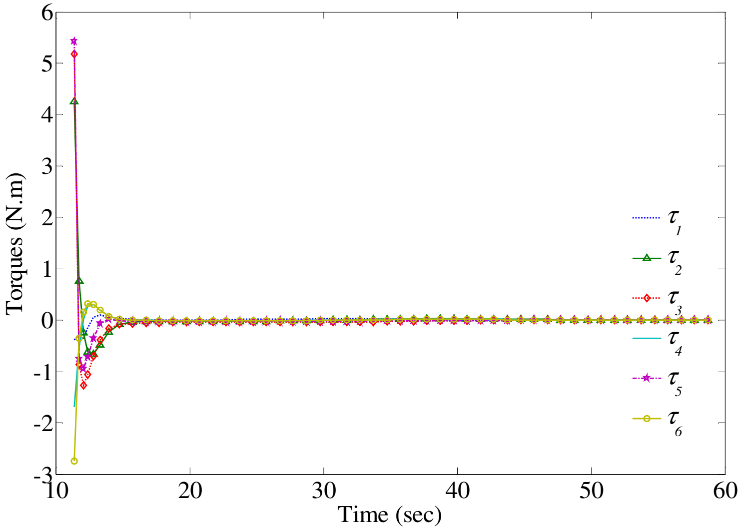

Fig. 7. Active joint torques during the reconfiguration process

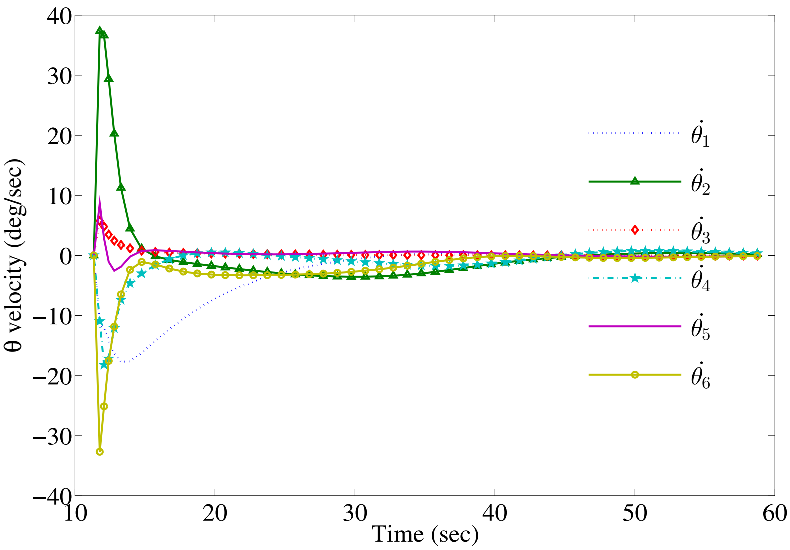

Fig. 8. Joint velocities during the reconfiguration process

4 Conclusion

In this paper the simulation results and the implementation methods of a 6-DOF reconfigurable manipulator are presented. Hence the manipulator controller might lead the manipulator to singular configurations; therefore it is necessary to design a singularity avoidance algorithm to ensure the robustness of the controller. In addition, obstacle avoidance and joints limits avoidance must be also implemented in order to prevent the manipulator from hitting the satellite and clash on its joints limits.

Acknowledgments. This work was supported by NSERC under grant RGPIN/288255-2011.

References

1. Angeles, J.: Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms. Springer (2007)
2. Merlet, J.-P.: Parallel Robots. Springer (2001)
3. Fukuda, T., Kawauchi, Y.: Cellular Robotic System (CEBOT) as one of the Realization of Self-Organizing Intelligent Universal Manipulator. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 662–667. IEEE Comput. Soc. Press (1990)
4. Murata, S., Kurokawa, H., Kokaji, S.: Self-Assembling Machine. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 441–448. IEEE Comput. Soc. Press (1994)
5. Ohkami, Y., Matunaga, S., Hayashi, R.: Operational Aspects of a Super Redundant Space Robot with Reconfiguration and Brachiating Capability. In: Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, SMC 1999, pp. 178–183 (1999)
6. Khoshnevis, B., Will, P.: Highly Compliant and Self-Tightening Docking Modules for Precise and Fast Connection of Self-Reconfigurable Robots. In: IEEE International Conference on Robotics and Automation (Cat. No.03CH37422), pp. 2311–2316 (2003)
7. Aghili, F., Parsa, K.: Design of a Reconfigurable Space Robot with Lockable Telescopic Joints. In: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4608–4614 (2006)
8. Lewis, F.L., Dawson, D.M., Abdallah, C.T.: Manipulator Control: Theory and Practice. CRC Press (2004)
9. McClamroch, N.H., Wang, D.: Feedback Stabilization and Tracking of Constrained Robots. IEEE Transactions on Automatic Control 33, 419–426 (1988)
10. Bayo, E., Garcia De Jalon, J., Serna, M.A.: A Modified Lagrangian Formulation for the Dynamic Analysis of Mechanical Systems. Computer Methods in Applied Mechanics and Engineering 71, 183–195 (1988)
11. Golub, G.H., Van Loan, C.F.: Matrix Computations. JHU Press (2012)
12. Craig, J.J.: Introduction to Robotics. Pearson Prentice Hall (2005)
13. Brady, M.: Robot Motion: Planning and Control. MIT Press (1982)
14. Aghili, F.: A Unified Approach for Inverse and Direct Dynamics of Constrained Multibody Systems Based on Linear Projection Operator: Applications to Control and Simulation. IEEE Transactions on Robotics 21, 834–849 (2005)
15. Aghili, F., Su, C.: Reconfigurable Space Manipulators for In-orbit Servicing and Space Exploration. In: International Conference on Intelligent Robotics and Applications (2012)